Addis Ababa University

College of Natural Sciences

Automatic barley relevance and quality assessment for brewing industry

Mesfin Adugna Mekonnen

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

2, November 2017

Addis Ababa University

College of Natural Sciences

Mesfin Adugna Mekonnen

Advisor: *Yaregal Assabie (PhD)*

This is to certify that the thesis prepared by *Mesfin Adungna*, titled: *Automatic barley relevance and quality assessment for brewing industry* 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

In beer production malt barley is used as major raw material holding about 90% of the total raw material cost. Homogeneity, a measure of grain uniformity, is important for malting and brewing performance. Uniformity of malt barley might be influenced due to different reasons that result in loss of quality during beer production, which has direct impact on brewing industry and on their customer satisfaction. Currently, malt barley sample quality inspection to keep homogeneity is performed manually by human experts through visual evaluation. 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 assess the quality of malt barley sample by identifying objects found in the image and giving grade level category for malt barley grains using digital image processing techniques, based on the specification standard for malt barely by Ethiopian Standards Agency. The system architecture for assessing malt barley quality sample consists of four components namely preprocessing, segmentation, feature extraction, and classification. Preprocessing convert RGB image to grayscale, filter noise, and apply binarization. A new segmentation that effectively segments malt barley grains from the background is developed from combination of three existing edge detection methods. We also applied ellipse fitting model on contour estimation method to segment overlapping malt barley grains. A total of 17 (1 size, 7 shape, 3 texture, and 6 color) features are identified to model the objects found in malt barley sample.

For the purpose of classification Artificial Neural Network is used, the training data is randomly partitioned into training (70%) and testing (30%). The classifier achieved an overall classification accuracy of 99.4 % for grading malt barley level and 98.7% for classification of object class.

Keywords: Malt barley, Edge segmentation, Contour estimation, Overlapped segmentation, Artificial Neural Network, Digital image processing, Homogeneity, Quality assessment

Dedication

My family, Wife (Edilawit Milkessa), Son (Firankisan Mesfin)

And

In memory of my breast feeder Grandmother

Aberash Moti (Abo)

Acknowledgments

First and foremost, I would like to thank my God, for allowing me to start and giving me strength to accomplish this research work.

I would like to express my deepest gratitude to my advisor, Dr. Yaregal Assabie for his excellent guidance, caring, and patience during the thesis.

I am also grateful to Assela Malt Factory, Dr. Tesfahun Alemu of Qulumsa Agricultural Research Center, Ato Yadessa Abashu of Holeta Agricultural Reseach Center, and Ato Dadhi Dabala of Ganda Shano Farmers Training Center for their insightful comments and support.

Finally, I am extremely grateful to my family and my wife and son for the encouragement, endurance and love they granted me in the long process of this work.

Table of Content

List of Tables

List of figures

Acronyms

A/D	Analogue-to-Digital
ANN	Artificial Neural Network
CCD	Charge Coupled Device
DV	Digital Video
EBC	European Brewery Corporation
GLCM	Gray Level Co-occurrence Matrix
GMM	Gaussian Markov Model
IA	Image Analysis
LGN	Lateral Geniculate Nucleus
LUT	Look-Up-Table
MAL	Minor Axis Length
MRF	Markov Random Field
SRG	Seeded Growing Region
SVM	Support Vector Machine

Chapter One: Introduction

1.1 Background

Beer has a strong bond with human society. This fermented beverage was most likely created by accident thousands of years ago. Despite the massive technological growth that separates ancient brewing from today's high technology breweries, the process in its traditional version remains entirely unchanged [1].

Ethiopian per capita beer consumption is about 4.0 liters, which compares with 11.0 liters in Kenya, 9.5 liters in Uganda, and 55 liters in South Africa [2]. This has started to change over the last decade as the economy has begun to grow. Ethiopia has experienced one of the fastest increases of beer consumption in the recent years, with consumption growing by as much as 90 percent between 2002 and 2011 [3]. The industry has responded to growing demand by expanding their scale of operation. The government invited two of the world's largest breweries Heineken and Diageo to set up operation. There is now growing evidence that, with an increase in income, households are switching from domestically brewed beverages like 'Tella' and 'Areke' to bottled beer. Since traditional beer is sorghum and other grain based and the bottled beers are barley based, this has accelerated the demand for malt barley [4].

Ethiopia is the second largest barley producer in Africa next to Morocco accounting for about 25 percent of the total barley production in the continent [3], also recognized as a center of diversity, as its barley germplasms have global significance because of improved traits, including disease resistance [5, 6, 7]. Barley is cultivated by smallholder farmers in every region since it is able to grow at all elevations, but it performs best at the higher elevations in the northern and central regions of the country [8] and the highlands of Ethiopia are known internationally to harbor valuable barley genetic resources [9]. It is cultivated in two distinct seasons, 'Belg' which relies on the short rainfall period from March to April and 'Meher' which relies on the long rainfall period from June to September [10] and it is the fifth most important cereal crop after teff, wheat, corn, and sorghum. There are two varieties of barley, food barley and malt barley [4, 8]. In Ethiopia, roughly 15% of barley production is in the form of malt barley. Food barley is grown primarily for human consumption, with smallholder farmers consuming upwards of 60% of the food barley they produce [8]. Traditionally barley is used for making local recipes and drinks such as 'dabo (kita)', 'Enjera', 'kolo', 'ganfo', 'kinche', 'beso', 'cuko (shakeka)', 'tela', 'borde', and other types of food. Its straw is a good source of animal feed [11] and it is also used for thatching of roofs, because of its wide range of uses barley is considered as the “king of grains” in much of the country and low farm input supplies such as fertilizer and improved seed [12].

In Ethiopia, the market potential for malt barley which is the major raw material (about 90% of the total raw material cost) for beer production is directly dependent on the market for beer. Its potential can best be assessed by looking at the evolving dynamics of Ethiopia’s growing brewery sector [8]. There are two types of barley [10]. The number of kernel rows in their head can distinguish them. The first type, Hordeum vulgare is the barley variety that is generally cultivated. This type is also called six-row barley. The second type is two-row barley. This type used to be classified as Hordeum distichum but now is also classed as Hordeum vulgare [13]. Mainly, these two species of barley are used in brewing, the two-row barley (with one grain per node) and the six-row barley (with three grains per node). To put it simple, the fewer are the kernels per node, the bigger and richer in starch they are. Conversely, the six row barley has less starch but higher protein content. Therefore, if the brewer wants to increase the extract content, the two row barley is the best option, whereas if enzymatic strength is the aim, the six rows will be the best choice [14]. According to different sources [13, 14, 15], some brewers feel two rowed malt produces a fuller, stronger malt flavor and six-row malt produces a grainier flavor in the finished beer.

Homogeneity, a measure of grain uniformity (the quality of lacking diversity or variation), is important for malting and brewing performance and is of increasing interest to maltsters who want to produce homogeneous malt [16]. Modern malting practice started in Ethiopia in 1974 with the establishment of a small malting plant at St. George brewery in Addis Ababa [17]. At present Assela and Gondar malting factory are operating to satisfy the need of malt for brewing industries [18]. There are different type of barley such as covered barley as show in Figure 1.1, barley growing in a field as show in Figure 1.2, hulled barley (sometimes called dehulled barley), pearl barley (not a whole grain) as show in Figure 1.3, hulless barley as show in Figure 1.4, barley grits, barley flakes, barley flour, and quick pearl barley (not a whole grain). The covered barley is used for malting which means it has a tough, inedible outer hull around the barley kernel.

Figure 1.1 Covered barley used for malting

Figure 1.3 Pearl barley (not a whole grain)

Figure 1.2 Barley growing in a field

Figure 1.4 Hulless barley

Nowadays homogeneity is influenced by so many reasons which may result in low quality beer production [6]. The main factor influencing the homogeneity of malt was the quality of the malt barley and the way it was treated in the field [16]. Malt barley production in Ethiopia is thus based on populations, a mixture of landraces differ in productivity, phenotypic, plasticity, seed color, maturity, height, disease reaction, and other traits [19], also cultivated under extreme marginal conditions of drought, frost, and poor soil fertility. It is the most dependable cereal and is cultivated on highly degraded mountain slopes better than other cereal crops in the highland of Ethiopia [8], but because of its cultivation behavior and influencing factor the uniformity of malt barley grain is lost. This heterogeneity should have to be tested before beer production process. To gain uniformity several chemical methods of testing can be applied, for instance protein electrophoresis, DNA analysis, immunological analysis, isoenzyme analysis and high performance liquid chromatography. Most of those methods are only applied by specialized experts and they are expensive.

Currently image analysis is a well established complement of problem solving mechanism [20]. It makes possible the enhancing of images, as well as the identification and automatic isolation of particles for study, being one accelerated technique that allows the attainment of morphologic information, providing concurrently a reduction of time and work. To speed up the barley crop purchasing process for brewing industry, rapid barley grain evaluation is mandatory before processing [17]. Accordingly, an image analysis technique is favored as number one since it is considerably operate with minimal human intervention and asks low cost.

1.2 Motivation

The possibility of solving observable problem using image processing method and the following points motivate us to work on this research.

- ✓ The restricted working policy of brewing industry to produce quality beer throughout its functioning life time to satisfy consumers need. Unfortunately, the current manual inspection is exposed to error, due to the fact that the manual process is subjected to bias, inconsistencies, and corruption associated with human nature. As a consequence, quality of beer is influenced.
- ✓ The promising standard of malt barley specification published by Ethiopian Standards Agency.
- ✓ The possibility of testing toward homogeneity of malt barley grain before processing using its physical parameters.
- ✓ The difficulty of gaining uniform barley product from farmer, since Ethiopian smallholder farmers cultivate it on highly degraded mountain slopes or poor soil fertility, as a result the difference of growth even if fertilizer was used or not which is directly or indirectly influenced by weather conditions.
- ✓ The chemical testing mechanisms ask for qualified and experienced workers which are expensive and give opportunity of testing the barley grain only by professionals. Farmers have not acquire the chance to check their cultivated barley whether it fit the standard of brewing industry before transporting it to far market, but if computer image analysis technique is applied it will give chance for the farmers to assess the

quality of barley, because image analysis is much easier to apply than chemical methods and it is rapid and low cost technique.

- ✓ The prior research work only consider size of the barley that fit the standard and miss other criteria to evaluate quality, such as broken kernels, dead malt barley kernels, existence of foreign matters , bad appearance in color, and food barley admixture. This work is not satisfactory because it has gap of meeting the standard of malt barley specification of Ethiopian Standards Agency. Hence, malt barley grain quality assessment and grading is still performed manually

1.3 Statement of the problem

Malt barley is a main input for brewing industry specially in giving good flavor for the beer. However, beer production quality is influenced during collection of malt barley from individual farmers holding fragmented pieces of land from which different quality barley is produced [17]. Different growing conditions like variations in temperature, moisture, light intensity and nutrient levels will influence grain condition even within a single field [17]. Several features of the barley itself will lead to non homogeneous malt modification which results in uneven start of germination due to embryo difference, kernel size and shape may influence both water uptake and distribution, enzyme level and distribution, protein level difference in the grain, endosperm structure, and cell wall thickness [16].

Some suppliers deliberately mix poor quality barley (other barley than malt barley such as food barley, broken grains, discolored grains) in the middle or at the bottom of the lot where it is not accessible for sampling as illustrated in Figure 1.5 and 1.6. Regulations are set by the factory in order to minimize such cheating. The penalty invoked if such attempts at cheating are attempted repeatedly that all lots will be rejected [17].

Figure 1.5 Mixed barley

Figure 1.6 Poor quality barley

Currently, quality inspection of malt barley for brewing industry found in Ethiopia is done with the help of chemical method and manual morphological features identification which is time consuming, labor intensive and need specialized experts in the area. Moreover, this manual process is subjected to bias, inconsistencies, and corruption associated with human nature. Even if there was no bias or corruption directly during manual inspection, because of different level of skill and understanding of assessor the judgment is indirectly influenced.

In order to eliminate these shortcomings of manual quality assessment, it is important to employ automated quality assessment system. Automated malt barley quality assessment has many important advantages over the manual technique. It 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 that exists in the manual system.

For this reason, image analysis system is implemented to solve quality assessment problem related to malt barley grain size by C. Ferreira et al. [21] to manage supply chain from the barley crop purchasing to the quality reception control, but the accuracy of this work is not satisfactory because it has gap of meeting the standard of barley malt specification of Ethiopian Standards Agency.

This work only consider size of the barley that fit the standard and miss other criteria to evaluate quality, example it does not check if there is broken kernels, dead malt barley kernels, existence of foreign matters and bad appearance in color. Hence, malt barley grain quality assessment and grading is still performed manually.

Recent development of brewing industry in Ethiopia demands new efforts to automate the complex task of quality assessment and grading for malt barley grain. To develop an image based solution for malt barley grain quality assessment requires deep understanding of the symptoms as well as efficient image analysis techniques, to meet both the accuracy and performance requirements.

1.4 Objective

General objective

The general objective of this research is to develop automatic barley relevance and quality assessor for brewing industry.

Specific objectives

The following specific objectives are identified for achievement of the research:

- Review literature concerning related works on barley grain and other cereals, and on digital image processing in general.
- Collecting samples of malt barley from different location depending on their relevance.
- Observe and understand malt barley morphological characteristics.
- Identify best features in determining quality with intention of Ethiopian Standards Agency standards.
- Design appropriate algorithm for preprocessing, segmentation, feature extraction and classification of malt barley to determine the grade with respect to its quality.
- Use accessible digital image processing tools and techniques to represent the designed algorithms.
- Develop a prototype for evaluation.
- Test and analyze the performance of the prototype.

1.5 Methods

To achieve the stated objectives we employ the following methods.

Literature Review

Explore as much as possible relevant literature to grasp well organized knowledge to understand and indicate direction for solving identified problem.

Sample Collection

Since quality of beer production is influenced during collection of barley from individual farmers holding fragmented pieces of land from which different quality barley is produced.

Samples of barley will be collected from different area of Ethiopia depending on their relevance and from Ethiopian Standards Agency research laboratory.

Prototype Development

For the purpose of evaluating the designed algorithms of this research work prototyping is mandatory. To demonstrate the proposed work to physical world and also for comparing the system output against manual quality assessment practice. The system is tested and analyzed for its performance.

1.6 Scope and Limitations

This work is limited to assessing the relevance and quality of barley samples collected from different locations of Ethiopia to confirm if it will fit Ethiopian Standards Agency specification for malt barley. The research does not identify malt barley variety. This work is limited only in assessing quality and grading of malt barley depending on morphological features and does not consider any kind of chemical assessment result to determine quality.

1.7 Application of Results

This research work has several contributions some of them are stated below

- ✓ This research result introduce automatic image processing to assess quality and grading of malt barley which is free from bias occurred because of human nature.
- ✓ Some suppliers deliberately mix poor quality barley in a lot. This work gives access to brewing industries to check samples in good tempo than manual inspection.
- ✓ Since farmers holding fragmented pieces of land produce different quality of barley, this research result will help them to check their product before transporting it to far market.
- ✓ Satisfaction of consumer will increase because of using image analysis practice, since it bring more reliability in keeping uniformity, which is the ultimate goal of any beer producer.

1.8 Organization of the rest of the Thesis

The rest of this thesis work is organized as follows. In Chapter Two, literature is reviewed in detail. In Chapter Three, digital image processing related works on cereal grain is reviewed. Design of proposed solution is discussed under Chapter Four. In Chapter Five, the experiment used to evaluate the performance of the proposed solution is discussed. In Chapter Six conclusions, future work and the contributions of this research work is presented.

Chapter Two: Literature Review

2.1 Introduction

This chapter presents the review of literatures on the concepts that are starting point for the research including introduction to barley, vision system, digital image representation, and digital image processing by incorporating image preprocessing, segmentation, feature extraction, and classification concepts.

2.2 Barley

Barley, Hordeum vulgare L. is an annual monocotyledonous herb. Belonging to tribe Triticeae, barley is evolutionarily closely related to two other small grain cereal species, wheat and rye, although the genus Hordeum is known to have diverged 12 million years ago [22]. The first signs of the pre agricultural gathering of wild barley are found in the region of fertile crescent in south western Asia 22,000 years ago, and domestication of barley has occurred independently also in central Asia [23]. The early selection by environmental factors and man continued with modern breeding has resulted in hundreds of landraces and cultivars, which are grown from semi arid subtropical to temperate climates, from equatorial to nearly circumpolar latitudes, and from sea level to high altitudes. Characteristically for a grain crop, barley cultivated today has long heads and large grains in comparison to its wild ancestors. These features support high grain yield as well as quality. Today, barley is a significant crop plant globally, and it is mainly exploited as feed or as a raw material for malt production [23].

2.2.1 Barley grain architecture

In botanical terms, the barley grain represents an indehiscent fruit type called a caryopsis [5]. Caryopses develop from spikelets, which are attached to the rachis of the spike by short structures called rachillas. The barley grain has an elongated shape and is divided longitudinally in half by a crease extending over the whole length of the grain as illustrated in Figure 2.1 [22, 23].

Figure 2.1 Appearance of barley grain from ventral (upper image) and dorsal (lower image) side.

The crease marks the ventral side of the grain, and the opposite side is called the dorsal side. The end of the grain where the embryo is located is attached to the rachis [24]. The major parts of the barley grain are the endosperm, the embryo and their covering layers of maternal origin as illustrated in Figure 2.2.

Figure 2.2 Appearance of longitudinal cut surface of barley showing husk, H, embryo, E, and starchy endosperm, SE.

The endosperm consists of the starchy endosperm and a surrounding aleurone layer as illustrated in Figure 2.3. The starchy endosperm forms the largest morphological part of the barley grain comprising to 75% of its weight [23, 24].

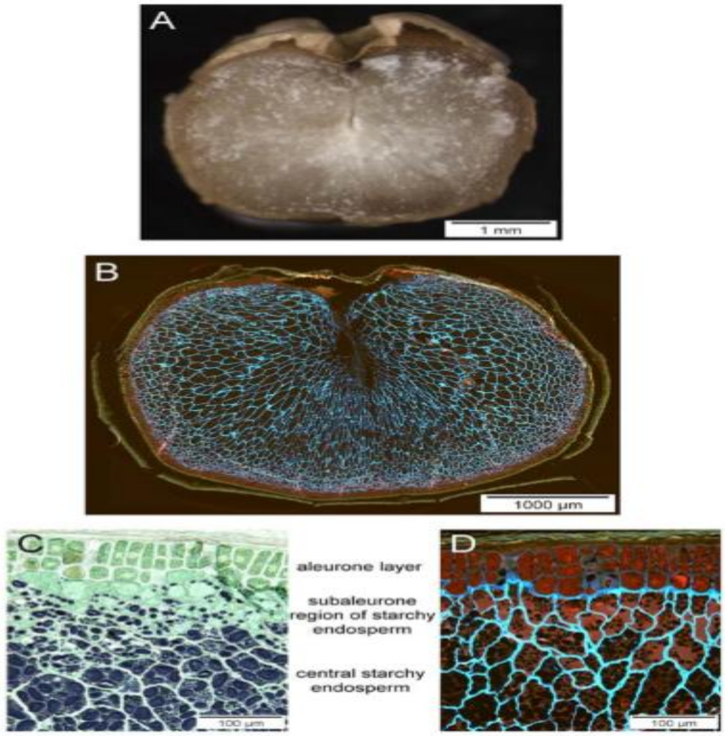


Figure 2.3 Structure of barley endosperm illustrated by a cross-cut surface (A), whole cross section (B) and close-ups (C and D) representing the middle third of the grain. In B and D, protein (red) and glucan in cell walls (blue) are visualized by Acid Fuchsin

The function of the starchy endosperm is to serve as a nutrient storage for the growing embryo during germination. It is comprised of dead cells lacking nuclei and containing starch granules embedded in a matrix of storage proteins [23, 24].

Compared to other grains, barley is the toughest grain in the field. Before exposing the endosperm it recognized as pearl barley, two inedible husks and another softer edible coating called the aleurone are removed. The germ, layered between the two outer husks, is also discarded, leaving a pile of valuable nutrients in the bin that becomes animal feed rather than human food [23, 24].

Barley may be subject to damage by a range of diseases, including powdery mildew, bacterial blight, and viral diseases [22]. These diseases are managed by a combination of strategies including the use of disease resistant varieties, application of fungicides, tillage practices, timing of crop rotation and planting [22]. Additionally, with the genetic modification, scientists are trying to make the plant more resistant to infections [24]. Bacterial blight is one of the more common diseases to strike barley. Bacterial blights first appear as small, water soaked spots on the barley leaves. The small spots grow until they resemble large brown streaks down the leaf's veins. A bad infection of bacterial blight kills leaves. To avoid the problem, use clean seed, practice crop rotation, avoid overhead irrigation if blight becomes a problem, and eliminate crop residue [22, 24].

There are two types of barley. The number of kernel rows in their head can distinguish them as show in Figure 2.4. The first type, Hordeum vulgare is the barley variety that is generally cultivated. This type is also called six rowed barley. The second type is two rowed barley. This type used to be classified as Hordeum distichum but now is also classed as Hordeum vulgare [13, 23].

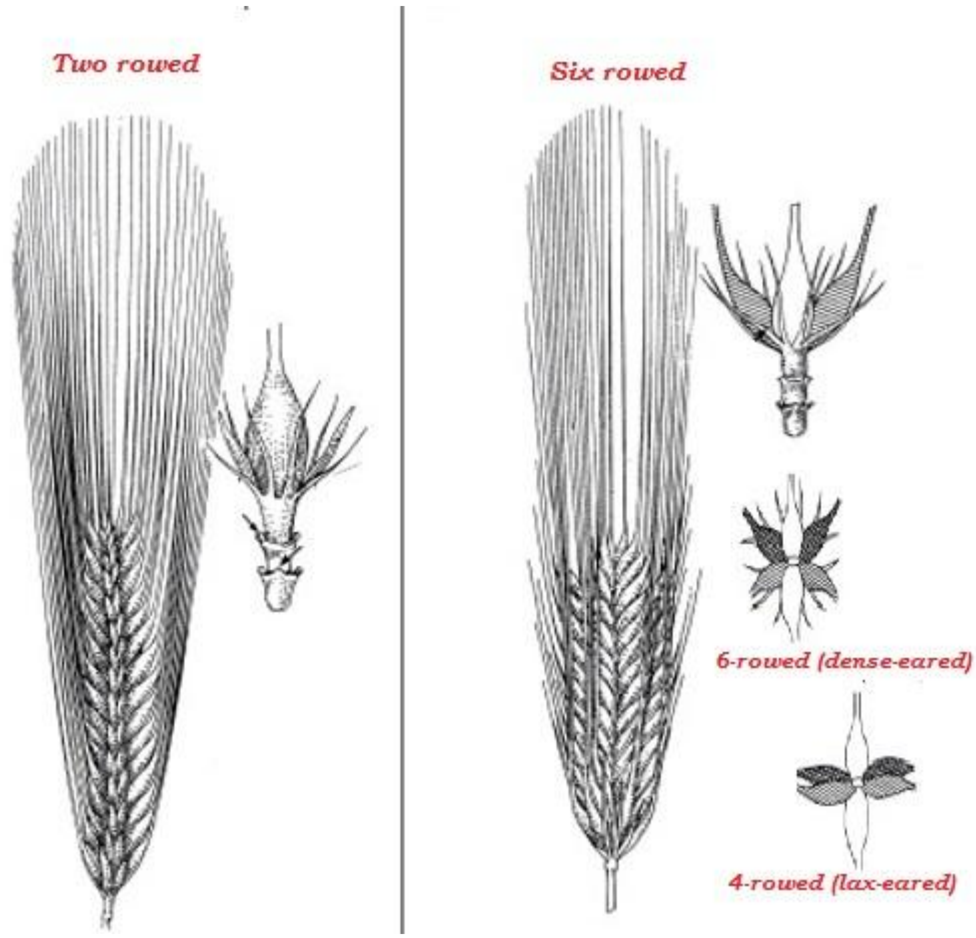


Figure 2.4 Barley (Hordeum): General morphology

There are two main criteria for separation of 2 rowed and many rowed barley. The first one is the number ob fertile spikelets per rachis segment and the second is by checking whether it is hulled or naked Figure 2.5 illustrate more when husks removed [22, 24].

Thus, the two rowed barleys (Hordeum spontaneum and H. distichum L.) are only with one fertile spikelet, the central one, per rachis segment, the 2 outer (equals to lateral) ones are sterile. There are naked and glumed (hulled) forms. Many rowed barleys (also called 6 rowed) (Hordeum vulqare L.) are with 3 fertile spikelets per rachis segment. There are dense eared forms (the “classical” six-rowed forms, var. hexastichum) and lax eared forms (the so called 4 rowed barley, var. tetrastichum also known as “laxeared six-rowed”). Of both there are hulled and naked forms [22, 25]. Table 2.1 clarifies more on the distinction of two and six rowed barley.

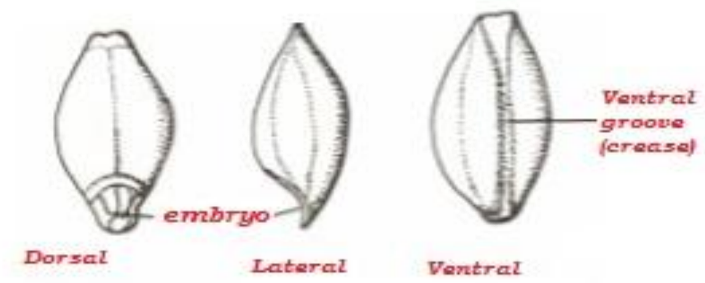


Figure 2.5 Central 6-row or 2-row grain, husks removed

Differences between six-rowed and two-rowed barley

Table 2.1 Two and six rowed barley difference

Part	Many-rowed barley	Two rowed barley
EARS	3 fertile spikelets per rachis segment	1 fertile spikelet per rachis segment= middle spikelet. Both side ones are stunted
SPIKELETS depression in lemma base show in Figure 2.7	Lax-eared forms (4-rowed) small fold. Dense-eared forms (6-rowed) horseshoe shaped	Horseshoe shaped
GRAINS	Straight and twisted grains present (particularly in lax-eared forms). Proportion of twisted: Straight grains theoretically 2:1) maximum width of grain at centre	Only straight grains present. Maximum width of grain somewhat below the centre of the grain
RACHIS SEGMENTS bases of the side florets illustrated in Figure 2.6	Well formed bases of the side florets	Bases of the side florets somewhat stunted

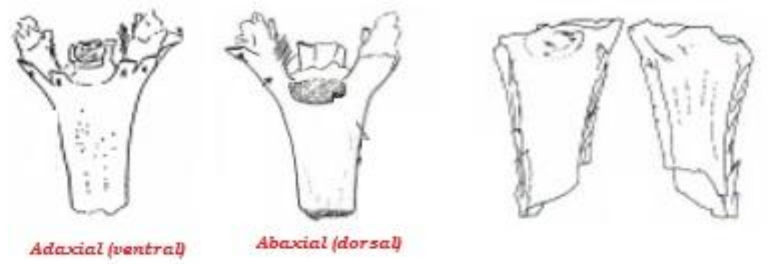


Figure 2.6 Rachis segments

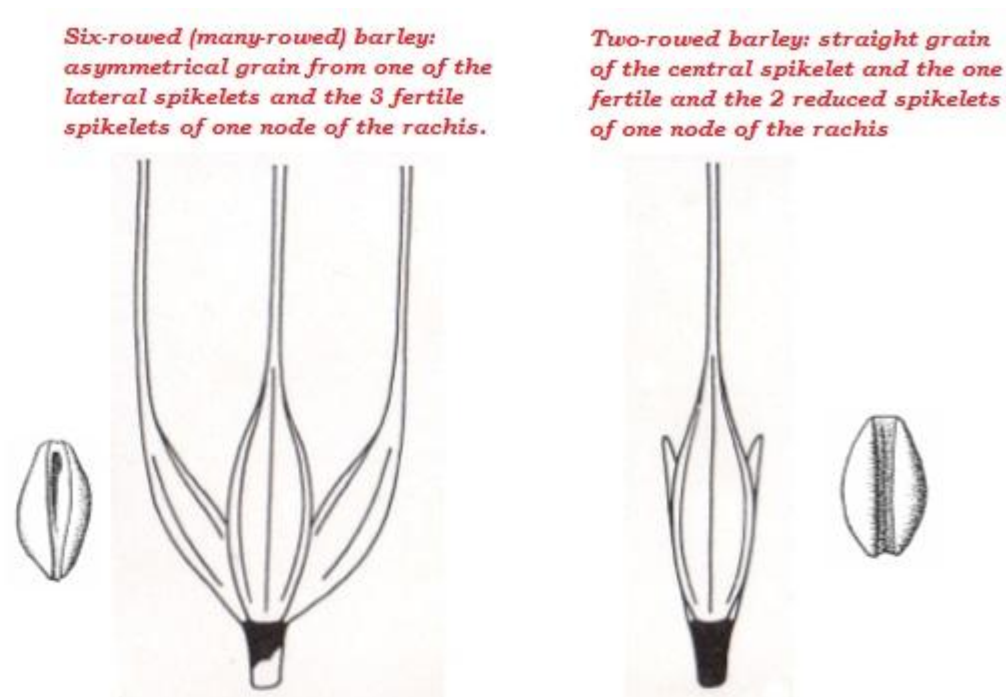


Figure 2.7 Spikelet of 6 and 2 rowed barley

2.2.2 Barley in malt and beer production

Although the cultivation of barley parallels the domestication of animals, studies show that it was the early brewing of beer that really increased barley domestication. The beer making process requires just four ingredients namely barley, water, hops, and yeast. Barley provides sugars and amino acids for yeast growth and the yeast converts the sugars to ethyl alcohol in a process called fermentation. Before barley is used to make beer, it is converted to "malt" to render it a better substrate for brewing [14, 16].

Malting is essentially a process of shortened seed germination. When grain enters the malt house, it is first steeped in water for two to three days. After steeping, the barley is transferred to germination beds for three to four days. There the grain begins to produce enzymes capable of degrading the starch, protein, and the cell walls. The barley grains are then subjected to heat that kills the growing seedling and dries up the remnant grain, but leaves intact the components of the endosperm and as well as the enzymes capable of degrading them. The end product is malt. How the roasting step is carried out plays a large part in determining the color and flavor of the final product, Figure 2.8 illustrate it more [14, 16].

Figure 2.8 Different types of beer depend on the malting process of the barley

Roasting stops the germination process, but if stopped in time leaves needed enzymes active. The malt is the key ingredient used by beer breweries for beer production. The roasted grain then goes through a process called mashing in which the starches are converted to sugars and dissolved in hot water (to make wort) in the first phase of brewing. The mash is then transferred into to a container with a sieve on the bottom, called a lauter tun. Here the liquid fraction of the mash, called wort is separated from the residual solids by filtration [14, 16].

Traditionally, barley is used for beer production because the hulls of the barley malt settle in the lauter tun and participate in filtering out residual solids. The resultant wort contains the soluble amino acids and sugars liberated by enzyme action. Hop plant flowers are added to the wort and boiled. The hop oils add certain flavors to the beer and protect against contamination. The wort is then cooled and transferred to a fermentation vessel, and inoculated with yeast. The yeast use sugars and amino acids from the malt to grow, and as it grows the metabolism of the sugar maltose leads to the production of ethyl alcohol and carbon dioxide as byproducts of the fermentation process. After fermentation, the yeast and other solids are allowed to settle out. This is followed by an aging process, carbonation, and packaging of the final product [14, 16].

2.3 Vision system

A computer vision system processes images acquired from an electronic camera, which is like the human vision system where the brain processes images derived from the eyes [26]. Computer vision is a rich and rewarding topic for study and research for electronic engineers, computer scientists and many others. Increasingly, it has a commercial future. There are now many vision systems in routine industrial use, cameras inspect mechanical parts to check size, food is inspected for quality and images used in astronomy benefit from computer vision techniques [26]. Forensic studies and biometrics (ways to recognize people) using computer vision include automatic face recognition and recognizing people by the texture of their irises. These studies are paralleled by biologists and psychologists who continue to study how our human vision system works and how we see and recognize objects (and people) [26].

2.3.1 The human vision system

Human vision is a sophisticated system that senses and acts on visual stimuli. It has evolved for millions of years, primarily for defense or survival. Intuitively, computer and human vision appear to have the same function. The purpose of both systems is to interpret spatial data, data that is indexed by more than one dimension. Even though computer and human vision are functionally similar, we cannot expect a computer vision system to replicate exactly the function of the human eye. This is partly because we do not understand fully how the vision system of the eye and brain works. Accordingly, we cannot design a system to replicate its function exactly. In fact, some of the properties of the human eye are useful when developing computer vision techniques, whereas others are actually undesirable in a computer vision system. But we shall see computer vision techniques which can to some extent replicate and in some cases even improve upon the human vision system [26, 27].

We might consider this, so put one of the fingers from each of your hands in front of your face and try to estimate the distance between them. This is difficult and for sure we would agree that our measurement would not be very accurate. Now if we put our fingers very close together. We can still tell that they are apart even when the distance between them is tiny. So human vision can distinguish relative distance well, but is poor for absolute distance. Computer vision is the other way around, it is good for estimating absolute difference, but with relatively poor resolution for relative difference. The number of pixels in the image imposes the accuracy of the computer vision system. In human vision, the sensing element is the eye from which images are transmitted via the optic nerve to the brain, for further processing. The optic nerve has insufficient band width to carry all the information sensed by the eye. Accordingly, there must be some preprocessing before the image is transmitted down the optic nerve [26, 27]. The human vision system can be modeled in three parts: eye, the neural system, and processing.

The eye: The function of the eye is to form an image, a cross section of the eye is illustrated in Figure 2.9. Vision requires ability to focus selectively on objects of interest. This is achieved by the ciliary muscles that hold the lens. In old age, it is these muscles which become slack and the eye loses its ability to focus at short distance. The iris or pupil is like an aperture on a camera and controls the amount of light entering the eye. It is a delicate system and needs protection which is provided by the cornea (sclera). This is outside the choroid, which has blood vessels that supply nutrition and is opaque to cut down the amount of light. The retina is on the inside of the eye which is where light falls to form an image. By this system muscles rotate the eye and shape the lens to form an image on the fovea (focal point) where the majority of sensors are situated. The blind spot is where the optic nerve starts, there are no sensors there [26, 27].

Focusing involves shaping the lens rather than positioning it as in a camera. The lens is shaped to refract close images greatly and distant objects little essentially by stretching it. The distance of the focal centre of the lens varies from approximately 14mm to around 17mm depending on the lens shape. This implies that a world scene is translated into an area of about 2mm2. Good vision has high acuity (sharpness) which implies that there must be very many sensors in the area where the image is formed [26, 27].

There are nearly 100 million sensors dispersed around the retina. Light falls on these sensors to stimulate photochemical transmissions which results in nerve impulses that are collected to form the signal transmitted by the eye. There are two types of sensor: first, the rods which are used for black and white (scotopic) vision and secondly, the cones, which are used for color (photopic) vision. There are approximately 10 million cones and nearly all are found within 5^0 of the fovea. The remaining 100 million rods are distributed around the retina, with the majority between 20^0 and 5^0 of the fovea. Acuity is expressed in terms of spatial resolution (sharpness) and brightness/color resolution and is greatest within 1^0 of the fovea [26, 27].

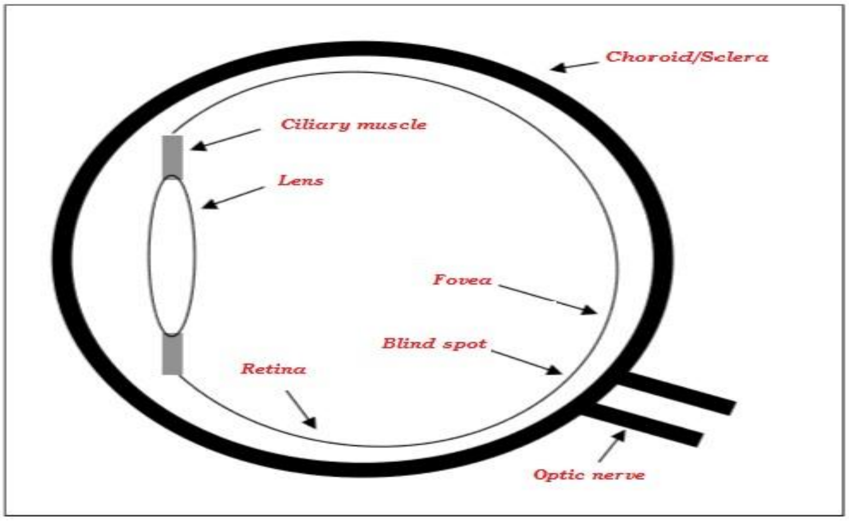


Figure 2.9 Human eye

There is only one type of rod, but there are three types of cone. These are

i. S (short wavelength): these sense light towards the blue end of the visual spectrum
ii. M (medium wavelength): these sense light around green
iii. L (long wavelength): these sense light towards the red region of the spectrum.

The total response of the cones arises from summing the response of these three types of cone which gives a response covering the whole of the visual spectrum. The rods are sensitive to light within the entire visual spectrum, giving the monochrome capability of scotopic vision. Accordingly, when the light level is low images are formed away from the fovea to use the superior sensitivity of the rods, but without the colour vision of the cones [26, 27, 28].

One inherent property of the eye known as Mach bands affects the way we perceive images. These are illustrated in Figure 2.10 and are the bands that appear to be where two stripes of constant shade join. By assigning values to the image brightness levels the cross section of plotted brightness is shown in Figure 2.10(a). This shows that the picture is formed from stripes of constant brightness. Human vision perceives an image for which the cross section is as plotted in Figure 2.10(c). These Mach bands do not really exist, but are introduced by our eye. The bands arise from overshoot in the eyes response at boundaries of regions of different intensity (this aids us to differentiate between objects in our field of view). The real cross section is illustrated in Figure 2.10(b). Note also that a human eye can distinguish only relatively few grey levels. It has a capability to discriminate between 32 levels (equivalent to five bits), whereas the image of Figure 2.10(a) could have many more brightness levels. This is why our perception finds it more difficult to discriminate between the low-intensity bands on the left of Figure 2.10(a) [26, 27, 28].

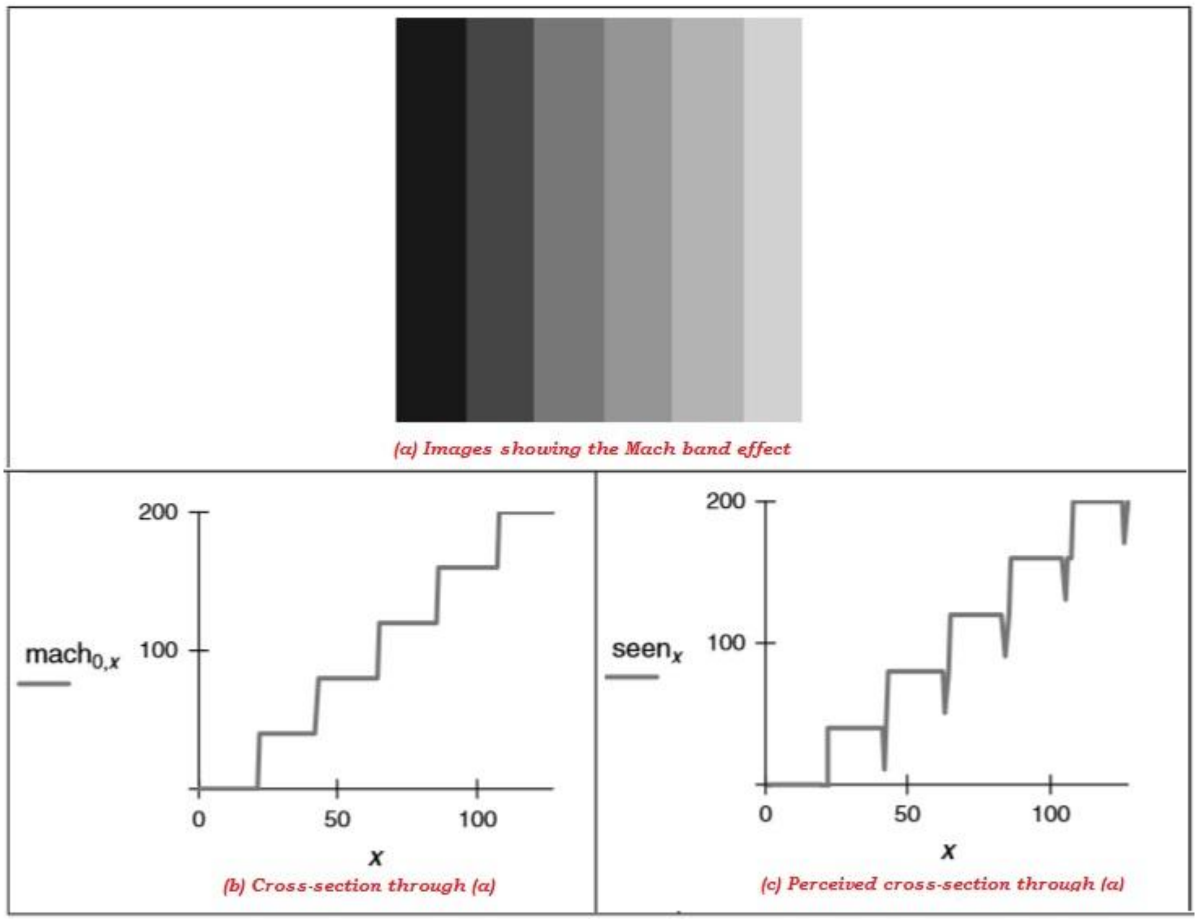


Figure 2.10 Illustrating the Mach band effect.

So we have already identified two properties associated with the eye that it would be difficult to include and would often be unwanted in a computer vision system, Mach bands and sensitivity to unsensed phenomena. These properties are integral to human vision. At present, human vision is far more sophisticated than we can hope to achieve with a computer vision system [26, 27, 28].

The neural system: Neural signals provided by the eye are essentially the transformed response of the wave length dependent receptors, the cones and the rods [26]. One model is to combine these transformed signals by addition as illustrated in Figure 2.11. The response is transformed by a logarithmic function, mirroring the known response of the eye. This is then multiplied by a weighting factor that controls the contribution of a particular sensor. This can be arranged to allow combination of responses from a particular region. The weighting factors can be chosen to afford particular filtering properties.

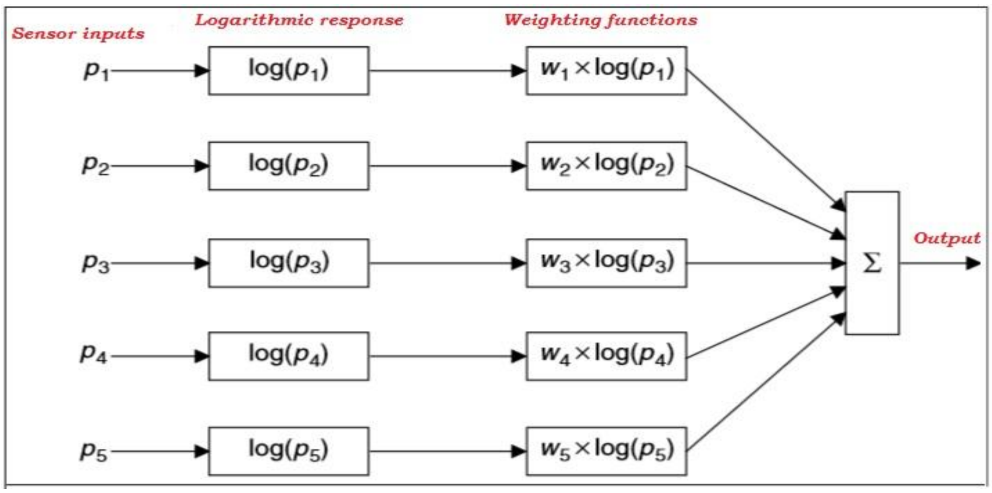


Figure 2.11 Neural processing

For example, in lateral inhibition the weights for the centre sensors are much greater than the weights for those at the extreme. This allows the response of the centre sensors to dominate the combined response given by addition. If the weights in one half are chosen to be negative, while those in the other half are positive, then the output will show detection of contrast (change in brightness) given by the differencing action of the weighting functions [26, 27].

The signals from the cones can be combined in a manner that reflects chrominance (color) and luminance (brightness). This can be achieved by subtraction of logarithmic functions, which is then equivalent to taking the logarithm of their ratio. This allows measures of chrominance to be obtained. In this manner, the signals derived from the sensors are combined before transmission through the optic nerve. This is an experimental model, since there are many ways possible to combine the different signals together [26, 27, 28].

Visual information is then sent back to arrive at the Lateral Geniculate Nucleus (LGN), which is in the thalamus and is the primary processor of visual information. This is a layered structure containing different types of cells with differing functions. The axons from the LGN pass information on to the visual cortex. The function of the LGN is largely unknown although it has been shown to play a part in coding the signals that are transmitted. It is also considered to help the visual system to focus its attention, such as on sources of sound [26, 27, 28].

Processing: The neural signals are then transmitted to two areas of the brain for further processing. These areas are the associative cortex, where links between objects are made and the occipital cortex, where patterns are processed. It is naturally difficult to determine precisely what happens in this region of the brain. To date, there have been no volunteers for detailed study of their brain's function (although progress with new imaging modalities such as positive emission tomography or electrical impedance tomography will doubtless help). For this reason, there are only psychological models to suggest how this region of the brain operates. It is well known that one function of the human vision system is to use edges or boundaries of objects. We can easily read the word in Figure 2.12(a) this is achieved by filling in the missing boundaries in the knowledge that the pattern is likely to represent a printed word. But we can infer more about this image there is a suggestion of illumination, causing shadows to appear in unlit areas. If the light source is bright then the image will be washed out causing the disappearance of the boundaries which are interpolated by our eyes. So there is more than just physical response there is also knowledge including prior knowledge of solid geometry. This situation is illustrated in Figure 2.12(b), which could represent three 'pacmen' about to collide, or a white triangle placed on top of three black circles. Either situation is possible [26, 27].

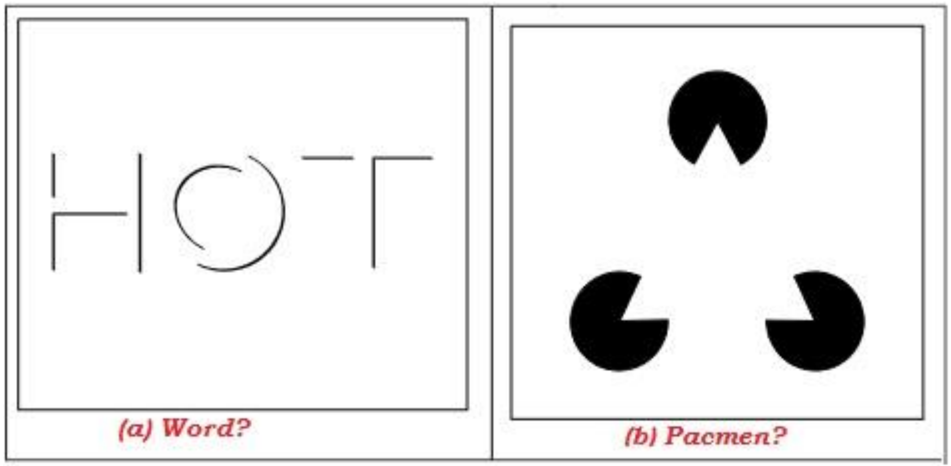


Figure 2.12 How human vision uses edges

It is also possible to mislead human vision, primarily by imposing a scene that it has not been trained to handle. In the famous Zollner illusion as show in Figure 2.13(a), the bars appear to be slanted, whereas in reality they are vertical. The small cross bars mislead our eye into perceiving the vertical bars as slanting. In the Ebbinghaus illusion as show in Figure 2.13(b), the inner circle appears to be larger when surrounded by small circles than it is when surrounded by larger circles [26, 27].

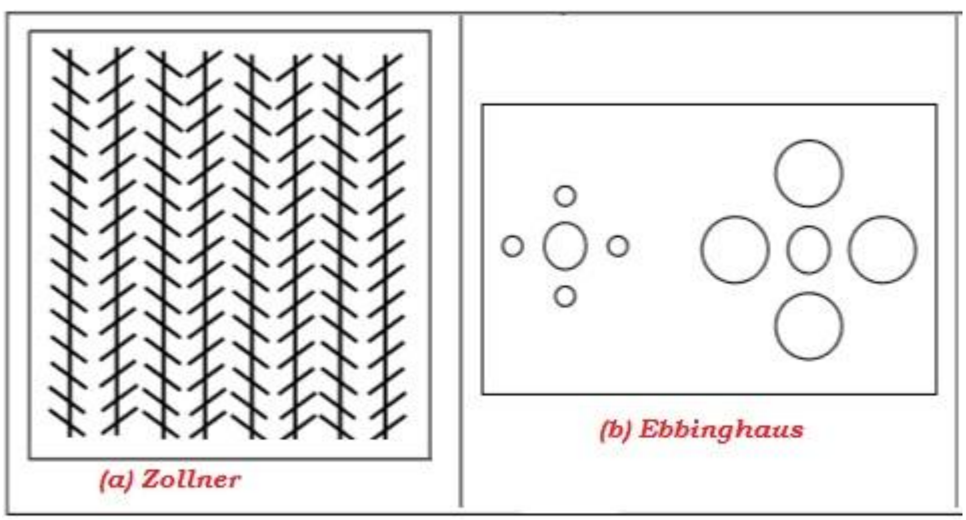


Figure 2.13 Static illusions

2.3.2 Computer vision systems

Machine or computer vision discipline origin is traced back to the 1960s [29]. Following an explosion of interest during the 1970s, it has experienced continued growth both in theory and application. Now a day, machine vision has been widely used for examining, monitoring, and controlling a very broad range of applications. It is the construction of explicit and meaningful descriptions of physical objects from images [27] and it encloses the capturing, processing and analysis of two dimensional images [28].

Applications of these techniques have now expanded to various areas such as medical diagnostic, automatic manufacturing and surveillance, remote sensing, technical diagnostics, autonomous vehicle, robot guidance and in the agricultural and food industry, including the inspection of quality and grading of agricultural products [26, 27, 29].

The progress of computer technology, computer vision hardware is now relatively inexpensive. A basic computer vision system requires a camera, a camera interface and a computer. These days, some personal computers offer the capability for a basic vision system by including a camera and its interface within the system. There are specialized systems for vision offering high performance in more than one aspect. These can be expensive, as any specialist system is [26, 29].

Cameras: A camera is optical instrument for recording or capturing images, which may be stored locally, transmitted to another location or both. The images may be individual still photographs or sequence of images constituting videos or movies. A camera is the basic sensing element. In simple terms, most cameras rely on the property of light to cause electron pairs (the charge carriers in electronics) in a conducting material. When a potential is applied (to attract the charge carriers) this charge can be sensed as current. By Ohm's law, the voltage across a resistance is proportional to the current through it, so the current can be turned in to a voltage by passing it through a resistor. The number of electron pairs is proportional to the amount of incident light. Accordingly, greater charge (and hence greater voltage and current) is caused by an increase in brightness. In this manner cameras can provide as output, a voltage that is proportional to the brightness of the points imaged by the camera [26, 29].

There are three main types of camera: vidicons, charge coupled devices (CCDs) and, more recently, CMOS cameras (complementary metal oxide silicon) now the dominant technology for logic circuit implementation. Vidicons are the older (analogue) technology which, although cheap (mainly by virtue of longevity in production), are being replaced by the newer CCD and CMOS digital technologies. The digital technologies now dominate much of the camera market because they are lightweight and cheap [26, 29].

Vidicons operate in a manner similar to a television in reverse. The image is formed on a screen and then sensed by an electron beam that is scanned across the screen. This produces an output which is continuous the output voltage is proportional to the brightness of points in the scanned line and is a continuous signal a voltage which varies continuously with time. In contrast, CCDs and CMOS cameras use an array of sensors, these are regions where charge is collected which are proportional to the light incident on that region. This is then available in discrete or sampled form as opposed to the continuous sensing of a vidicon. This is similar to human vision with its array of cones and rods, but digital cameras use a rectangular regularly spaced lattice, whereas human vision uses a hexagonal lattice with irregular spacing. Two main types of semiconductor pixel sensor are illustrated in Figure 2.14. In the passive sensor, the charge generated by incident light is presented to a bus through a pass transistor [26, 29].

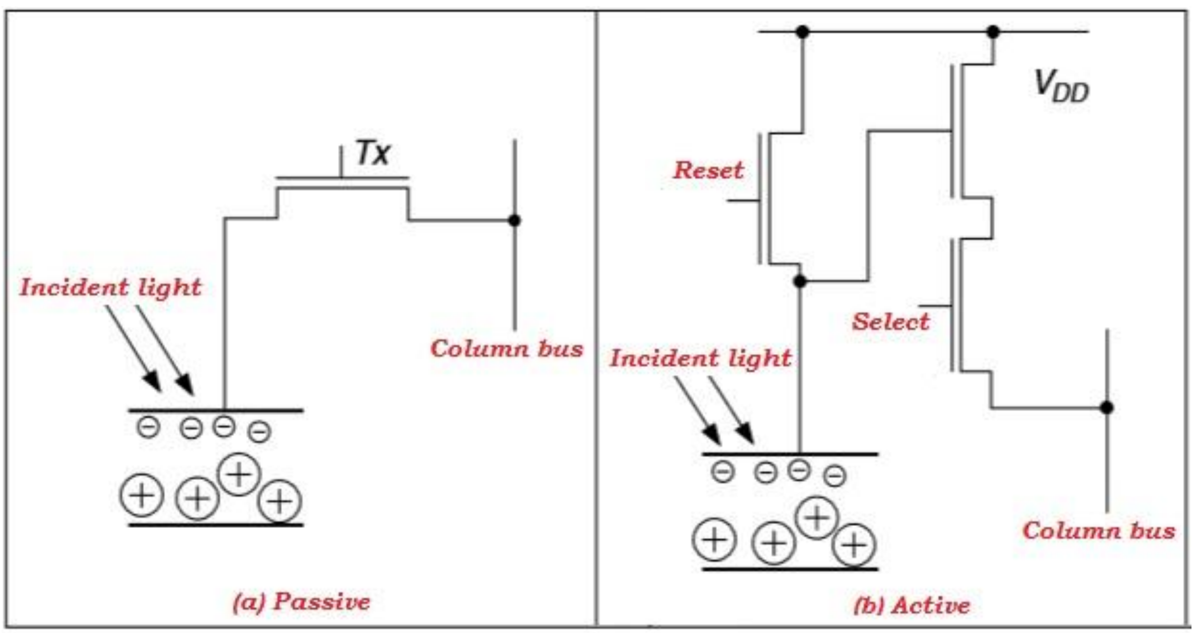


Figure 2.14 Pixel sensors

When the signal Tx is activated, the pass transistor is enabled and the sensor provides a capacitance to the bus, one that is proportional to the incident light. An active pixel includes an amplifier circuit that can compensate for limited fill factor of the photodiode. The select signal again controls presentation of the sensor's information to the bus. A further reset signal allows the charge site to be cleared when the image is rescanned [26, 29].

Finally, there are specialist cameras which include high resolution devices which can give pictures with a great number of points, low light level cameras, which can operate in very dark conditions (this is where vidicon technology is still found) and infrared cameras, which sense heat to provide thermal images. Increasingly, hyperspectral cameras are available which have more sensing bands [26, 29].

Computer interfaces: This technology is in a rapid state of change remaining to the emergence of digital cameras. Essentially, the image sensor converts light into a signal which is expressed either as a continuous signal or in sampled (digital) form. Modern digital systems convert the sensor information into digital information with on chip circuitry and then provide the digital information according to a specified standard. Video implies delivering the moving image as a sequence of frames and these can be in analogue or discrete form, of which one format is Digital Video (DV). An interface that converts an analogue signal into a set of digital numbers is called a frame grabber, since it grabs frames of data from a video sequence. Note that cameras that provide digital information do not need this particular interface (it is inside the camera). However, an analogue camera signal is continuous and is transformed into digital (discrete) format using an analogue-to-digital (A/D) converter. Flash converters are usually used remaining to the high speed required for conversion. The output of the A/D converter is often fed to look-up tables (LUTs), which implement designated conversion of the input data, but in hardware rather than in software and this is very fast. The outputs of the A/D converter are then stored [26, 28, 29].

In digital camera systems this processing is usually performed on the camera chip and the camera eventually supplies digital information often in coded form. IEEE 1394 (or firewire) is a way of connecting devices external to a computer and is often used for digital video cameras as it supports high speed digital communication and can provide power. This is similar to universal serial bus (USB), which can be used for still cameras [26, 29].

2.4 Digital image representation

An image is a single picture which represents something. Pictures are the most common and convenient means of conveying or transmitting information. A picture is worth a thousand words. Pictures in brief transmit information about positions, sizes and inter-relationships between objects. They represent spatial information that we can recognize as objects. Human beings are good at deriving information from such images, because of our innate visual and mental abilities. About 75% of the information received by human is in pictorial form. In analysis of pictures that employ an overhead perspective including the radiation not visible to human eye are considered [29, 30].

A digital image is a representation of a two dimensional image as a finite set of digital values, called picture elements or pixels. Pixel values typically represent gray levels, colors, heights, opacities etc. A digital image a[m,n] described in a two dimensional discrete space is derived from an analog image a(x,y) in a two dimensional continuous space through a sampling process that is frequently referred to as digitization. The effect of digitization is shown in Figure 2.15. The pixel at coordinates [m=10, n=3] has some integer brightness value. Digitization implies that a digital image is an approximation of a real scene [29, 30, 31].

The two dimensional continuous image a(x,y) is divided into N rows and M columns. The intersection of a row and a column is termed as a pixel. The value assigned to the integer coordinates [m,n] with {m=0,1,2,…,M–1} and {n=0,1,2,…,N–1} is a[m,n]. In fact, in most cases a(x,y) which we might consider to be the physical signal has two dimension sensor which is actually a function of many variables including depth (z), color (λ), and time (t) [26, 29, 31].

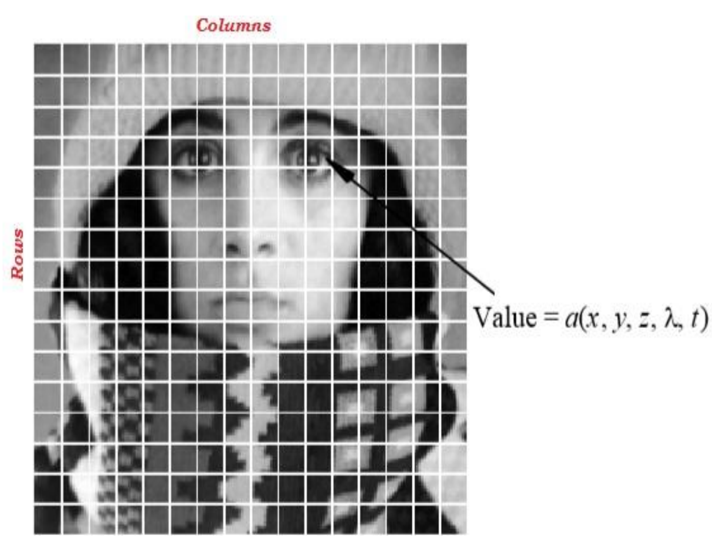


Figure 2.15 Digitization of a continuous image.

The value assigned to every pixel is the average brightness in the pixel rounded to the nearest integer value. The process of representing the amplitude of the two dimensional signal at a given coordinate as an integer value with L different gray levels is usually referred to as amplitude quantization or simply quantization [26, 27, 31].

An image may be continuous with respect to the x and y coordinates, and also in amplitude. Converting from a continuous image a(x,y) to its digital representation b[m,n] requires the process of sampling. In the ideal sampling system a(x,y) is multiplied by an ideal two dimensional impulse train as show in Equation (1) [26, 27, 31].

$$b_{ideal}[m,n] = a(x,y) \cdot \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} \sigma(x - mX_\circ, y - nY_\circ) \tag{1}$$

$$= \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} a(mX_\circ, nY_\circ)\sigma(x - mX_\circ, y - nY_\circ)$$

where X_o and Y_o are the sampling distances or intervals and δ (•,•) is the ideal impulse function. At some point, of course the impulse function δ (*x*, *y*) is converted to the discrete impulse function δ [m,n]. Square sampling implies that $X_o = Y_o$. Sampling with an impulse function corresponds to sampling with an infinite small point. This, however, does not correspond to the usual situation as illustrated in Figure 2.15. To take the effects of a finite sampling aperture p(*x*, *y*) into account, we can modify the sampling model as in Equation (2) [29,31].

$$b[m,n] = \left(a(x,y) \otimes p(x,y)\right) \cdot \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} \sigma(x - mX_\circ, y - nY_\circ) \tag{2}$$

Generally, digitizing the coordinate values is called sampling. Digitizing the amplitude values is called quantization [26, 27, 32].

2.4.1 Binary image

A binary image is a digital image that has only two possible values for each pixel or each pixel assumes one of only two discrete values. Essentially, these two values correspond to on and off. Looking at an image in this way makes it easier to distinguish structural features. For example, in a binary image, it is easy to distinguish objects from the background. The two colors used for a binary image are black and white. The color used for the object(s) in the image is the foreground color while the rest of the image is the background color [26, 31, 32].

Binary image is used for morphological image processing. Morphological image processing is a collection of non linear operations related to the shape or morphology of features in an image. Morphological operations rely only on the relative ordering of pixel values, not on their numerical values, and they are especially suited to the processing of binary images. Morphological techniques investigate an image with a small shape or template called a structuring element. This structuring element used for fundamental operations like erosion and dilation. Binary images also arise in digital image processing as masks or as the result of certain operations such as segmentation and thresholding [26, 31, 32].

2.4.2 Grayscale image

A pixel with 8 bits depth has 256 possible values known as grayscale image. In this image intensity is the only attribute involved. The intensity varies from black to white (gray levels in between). This means grayscale is a range of shades of gray without other visible color. The darkest possible shade is black, which is the total absence of transmitted or reflected light. The lightest possible shade is white, the total transmission or reflection of light at all visible wavelengths [26, 31, 32].

Grayscale image is practical in digital image processing because it only contain one image plane or it mean that it only contain the gray scale intensity values when compared to RGB which holds three components. This allows data to be reduced during processing. This data reduction allows the algorithm to run in a reasonable amount of time than using RGB which asks to extract three components individual and finally combining those together to get assumed result [26, 31, 32].

2.4.3 Color image

The appearance of an object is basically resulted from the nature of the light reflected from the object, its optical characteristics and the human perception. The colors are actually electromagnetic waves described by their wave length. The visible spectrum or the portion of the electromagnetic spectrum that can be detected by the human eye ranges from 390nm (violet) to 750nm (red) [26, 31, 32].

There are four main attributes that characterize the light: intensity, radiance, luminance, and brightness. In the case of achromatic light, the intensity is the only attribute involved. This is the case where the called gray scale is used intensity varies from black to white (gray levels in between). On the other hand, in the case of chromatic light, the other three attributes are used to measure the quality of the light source. The radiance refers to the amount of emitted energy by the light source and it is measured in watts (W). The luminance measures the amount of radiation perceived by an observer and it is measured in lumens (lm). The brightness is associated to the light intensity. Although the brightness has an accurate interpretation in monochromatic images, it is a very subjective property in the case of chromatic images [26, 31, 32].

Because of the absorption characteristics of the human eye, the colors are considered to be formed from different combinations of the primary colors red, green, and blue. These three colors can be added to create the secondary colors magenta (red + blue), cyan (green + blue), and yellow (green + red). The white color can be formed if the three primary colors are mixed or if a secondary color is mixed with its opposite primary color (all in the right intensities) [26, 31, 32].

In color image analysis three attributes are used to differentiate one color from another brightness, hue and saturation. The hue attribute brings the information concerning the main wavelength in the color. It is responsible for verifying the color in the complete spectrum from red to violet and magenta. The saturation describes the level of mixture between the hue and the white light. It determines the "purity" of the color. High values of saturation result in more gray scale pixels and small values result in pixels with high "purity". For instance, the red color is highly saturated and the pink color is unsaturated. A fully saturated color does not contain white light. Finally, the chromaticity is a description that combines hue and saturation. Hence, it is possible to describe an image according to brightness and chromaticity [31, 32].

The color depth measures the amount of color information available to display or print each pixel of a digital image. A high color depth leads to more available colors and consequently to a more accurate color representation. For example, a pixel with one bit depth has only two possible colors black and white (known as binary image). A pixel with 8 bits depth has 256 possible values (known as grayscale image) and a pixel with 24 bits depth has more than 16 million of possible values (known as RGB or true color image). Usually, the color depths vary between 1 and 64 bits per pixel in digital images [26, 31, 32].

The color models are used to specify colors as points in a coordinate system, creating a specific standard. The most common color spaces are RGB, CMYK, and HSV.

RGB Color Model

The RGB (Red, Green, and Blue) color space is one of the most used color spaces, specially for 8 bit digital images. This model is usually used for representing colors in electronic devices as TV and computer monitors, scanners, and digital cameras. The RGB is an additive model where the red, green, and blue colors are combined on different quantities to reproduce other colors. The pixels of an image represented in the RGB model have usually 8 bits depth, resulting in 256 possible intensities, the range of [0, 255] for each color. A color in the RGB model can be described indicating the amount of red, green, and blue using Equation (3), (4) and (5).

$$R = \frac{R}{R + G + B} \tag{3}$$

$$G = \frac{G}{R + G + B} \tag{4}$$

$$B = \frac{B}{R + G + B} \tag{5}$$

Each color can vary between the minimum value (totally dark) and the maximum value (totally intense). When all the colors have the minimum value, the resulting color is black. On other hand, when all the colors have the maximum value, the resulting color is white. This model is known as the RGB color cube as illustrated in Figure 2.16, because the model is based on the Cartesian coordinate system its color subspace of interest is a cube [32, 33].

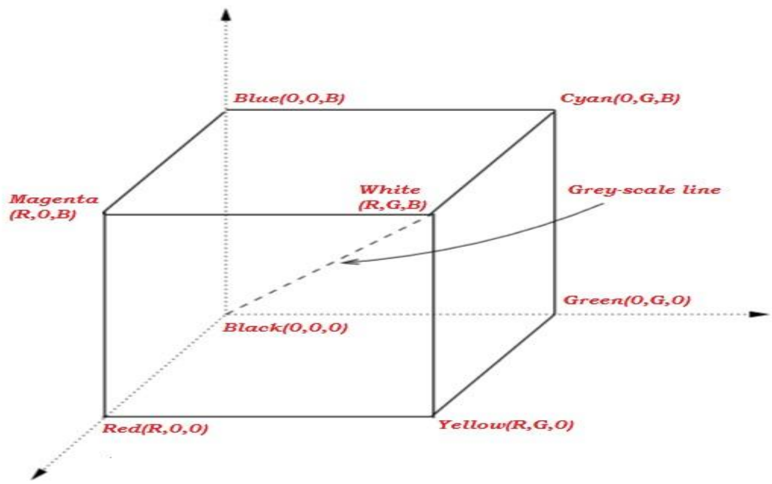


Figure 2.16 The RGB color model

The primary and secondary colors are at the corners of the cube. The black color is at the origin and the white color is at its opposite corner. The diagonal between the black and the white colors is the gray scale.

CMYK Color Mode

The CMYK model is composed by the cyan, magenta, yellow, and black colors. The basis of this model is the light absorption, as the visible colors come from the non-absorbed light. This space is usually used by printers and photocopiers to reproduce the majority of the colors in the visible spectrum. The system used is called quadrichromie, the subtractive color system, in opposition of the additive system RGB. Cyan is the opposite color of red; it plays as a filter that absorbs the red color. The same occurs with magenta and green, and with yellow and blue. Actually, the original subtractive model is CMY. Although equal amounts of cyan, magenta, and yellow produce the black color in theory, this combination in practice (printing on a paper) does not produce a true black. In order to overcome this problem, the fourth color (black) is added to the model (CMYK). It usually occurs that some visible colors on the screen of a computer monitor are not printed properly on a paper. This happens because the CMYK used in the printers is based on a mixture of inks on the paper, and the CMYK used in the computer monitors is a variation of the RGB space. Consequently, the CMYK color spectrum happens to be smaller than the RGB color spectrum [32, 33].

HSV Color Model

The HSV color system is composed by three components; hue, saturation, and value. This model is also known as HSB (Hue, Saturation and Brightness). These three parameters are used to define the color space. HSV color space is shown in Figure 2.17 expresses hue as a number from 0 to 360 degrees representing hues of red (starts at 0), yellow (starts at 60), green (starts at 120), cyan (starts at 180), blue (starts at 240), and magenta (starts at 300).

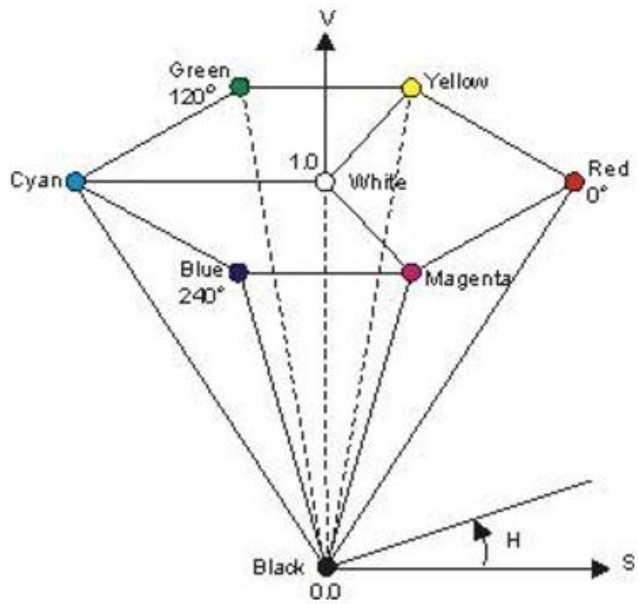


Figure 2.17 The HSV color space

Saturation indicates the range of grey in the color space. It ranges from 0 to 100%. Sometimes the value is calculated from 0 to 1. A faded color is due to a lower saturation level, which means the color contains more grey. Value is the brightness of the color and varies with color saturation. It ranges from 0 to 100%. When the value is '0' the color space will be totally black. With the increase in the value, the color space brightness up and shows various colors [29, 31, 32].

The HSV model is a nonlinear transformation of the RGB system. Hence, it is possible to transform directly a color from the HSV system to the RGB system, and contrariwise. There are two other color systems related to HSV, the HSL (Luminosity) system and the HSI (intensity) system [32, 33].

This color system is very interesting, because it allows the separation of the three components of a specific color (hue, saturation, and intensity). It is broadly used in artificial vision systems, as it is a powerful tool for the development of digital image processing algorithms based on the human color perception model. Indeed, the HSV model is well suited to characterize colors in practical terms for human interpretation, differently from the RGB and CMYK models [33].

Since HSV color model is preferred, given an image in RGB color format using Equations (6), (7) and (8) conversion is possible.

The Hue component is give by

$$H = \begin{cases} \theta & if\ B \leq G \\ 360 - \theta & if\ B > G \end{cases} \tag{6}$$

with $\theta = cos^{-1}\left\{\frac{\frac{1}{2}[(R-G)+(R-B)]}{[(R-G)^2+(R-B)(G-B)]^{1/2}}\right\}$

The Saturation component is give by

$$S = 1 - \frac{3}{(R+G+B)}[\min(R, G, B)] \tag{7}$$

Finally, the intensity component is give by

$$I = \frac{1}{3}(R + G + B) \tag{8}$$

2.5 Digital image processing

Digital image processing is the use of computers to manipulate digital images. Digital image processing involves changing the nature of an image in order to either improve its pictorial information for human interpretation or render it more suitable for autonomous machine perception for storage, transmission and representation [29, 32].

Digital image processing or analysis involves a series of steps, which can be broadly divided into three levels these are low, mid, and high level processes. Low level processes involve image acquisition and 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 (partitioning an image into regions or objects), description of those objects to reduce them to a form suitable for computer processing and classification (recognition) of individual objects. A mid level process is characterized by the fact that its inputs generally are images, but its outputs are attributes extracted from those images such as edges, contours, and the identity of individual objects. Finally, higher level processing involves "making sense" of an ensemble of recognized objects, as in image analysis and at the far end of the range, performing the cognitive functions normally associated with vision [29, 32]. These steps of digital image processing are discussed as follows.

2.5.1 Image preprocessing

Preprocessing does not increase the image information content. It is useful on a variety of situations where it helps to suppress information that is not relevant to the specific image processing or analysis task, example background subtraction. The aim of preprocessing is to improve image data so that it suppresses undesired distortions and it enhances image features that are relevant for further processing [31,32]. Contrast enhancement for contour detection, restoration aim to suppress degradation using knowledge about its nature that may occur do to relative motion of camera and object or wrong lens focus etc.. Binarization and noise removal are some of preprocessing type [29, 32].

Image binarization is a process of separation of pixel values of an input gray image into two pixel values (0 and 1). To separate the pixel values of the input grayscale image into background and foreground, each and every pixel should be compared with the threshold value and transformed to its respective class. Thresholding is a process of finding an appropriate threshold value for binarization. Thresholding value is a certain intensity value [29, 32].

Noise reduction or smoothing is one of the most important processes in image processing. In general the results of the noise removal have a strong influence on the quality of the image processing technique. There are different techniques for noise removal, some of them are median filter, conservative smoothing, crimmins speckel removal, Gaussian filter, mean filter, Laplacian of Gaussian filter, frequency filters and unsharp filter. In median filtering, the neighboring pixels are ranked according to brightness (intensity) and the median value becomes the new value for the central pixel. In the median filtering operation, the pixel values in the neighborhood window are ranked according to intensity, and the middle value (the median) becomes the output value for the pixel under evaluation [29, 32].

2.5.2 Image segmentation

Image segmentation is an important topic in the field of digital image processing. Image segmentation is the division of an image into regions or categories, which correspond to different objects or parts of objects. Segmentation of an image can also be defined by a set of regions that are connected and non-overlapping, so that each pixel in a segment in the image acquires a unique region label that indicates the region it belongs to. Every pixel in an image is allocated to one of a number of these categories. For the segmentation we need the images, but the images are either in form of black and white or color. Color images are due to the grey level. As the grey level contrast changes the color of color image also changes [31].

A good segmentation is typically one in which pixels in the same category have similar grey scale of multivariate values and form a connected region or neighbouring pixels which are in different categories have dissimilar values [34]. There are different techniques of image segmentation. Some of them are region based, edge based, threshold, feature based clustering, model based, and using external information about object is discussed as follows:

Region based, in this technique pixels that are related to an object are grouped for segmentation. The thresholding technique is bound with region based segmentation. The area that is detected for segmentation should be closed. Region based segmentation is also termed as "Similarity Based Segmentation". There won't be any gap due to missing edge pixels in this region based segmentation. The boundaries are identified for segmentation. In each and every step at least one pixel is related to the region and is taken into consideration. After identifying the change in the color and texture, the edge flow is converted into a vector. From this the edges are detected for further segmentation [32, 34, 35].

Edge based, segmentation can also be done by using edge detection techniques. There are various techniques of this segmentation. In this technique the boundary is identified to segment. Edges are detected to identify the discontinuities in the image. Edges on the region are traced by identifying the pixel value and it is compared with the neighboring pixels. This classification use both fixed and adaptive feature of Support Vector Machine (SVM). In this edge based segmentation, there is no need for the detected edges to be closed. There are various edge detectors that are used to segment the image. In that Canny edge detector has some step by step procedure for segmentation. To reduce the effect of noise, the surface of the image is smoothened by using Gaussian convolution. Then Sobel operator is applied to the image to detect the edge strength and edge directions. The edge directions are taken into considerations for non-maximal suppression, this means the pixels that are not related to the edges are detected and then, they are minimized. The final step is removing the broken edges that mean the threshold value of an image is calculated and then the pixel value is compared with the threshold that is obtained. If the pixel value is high than the threshold then, it is considered as an edge or else it is rejected. Edges are the discontinuities in the sense of intensity, which gives a layout of an object. All objects in the image are traced when the intensities are calculated accurately. The edges are detected by calculating the minimum and maximum of first derivative in gradient edge detector. Zero crossing is found in second derivative to identify the edges in Laplacian edge detector. Sobel edge detector uses convolution kernel to detect the edges. Magnitude of the spatial gradient is calculated for edges in Robert's edge detector. Canny edge detector also uses high spatial gradient but it takes more computation than Sobel and Robert's edge detector [34, 35].

Sobel edge detector performs a two dimensional spatial gradient measurement on an image and emphasizes regions of high spatial frequency that correspond to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an input grayscale image. This operator consists of a pair of 3×3 convolution kernels as shown in Table 2.2. One kernel is simply the other rotated by 90° [34, 35].

Table 2.2 Sobel convolution kernels

-1	0	1
-2	0	2
-1	0	1

Gx

1	2	1
0	0	0
-1	-2	-1

Gy

These kernels are designed to respond highly to edges running vertically and horizontally relative to the pixel grid, one kernel for each of the two perpendicular orientations. The kernels can be applied separately to the input image, to produce separate measurements of the gradient component in each orientation of Gx and Gy. These can then be combined together to find the absolute magnitude of the gradient at each point and the orientation of that gradient using Equation (11). The gradient magnitude is given by Equation (9) [34, 35].

$$|G| = \sqrt{Gx^2 + Gy^2} \tag{9}$$

an approximate magnitude is computed using Equation (10).

$$|G| = |Gx| + |Gy| \tag{10}$$

$$\theta = \arctan\left(\frac{Gy}{Gx}\right) \tag{11}$$

Orientation 0 is taken to mean that the direction of maximum contrast from black to white runs from left to right on the image and other angles are measured anti-clockwise [34, 35].

The Roberts edge detection is simple and quick to compute. It is a two dimensional spatial gradient measurement on an image. It thus highlights regions of high spatial frequency which often correspond to edges. In its most common usage the input to the operator is a grayscale image. Pixel values at each point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point. This edge detection consists of a pair of 2×2 convolution kernels as shown in Table 2.3 [34, 35].

Table 2.3 Roberts convolution kernels

1	0
0	-1

0	1
-1	0

These kernels are designed to respond maximally to edges running at 45° to the pixel grid, one kernel for each of the two perpendicular orientations [34, 35].

Canny edge detector takes a gray scale image as input and produces an image showing the positions of tracked intensity discontinuities as output. Canny operator works in a multi stage process. First of all the image is smoothed by Gaussian convolution. Then a simple two dimensional first derivative operator is applied to the smoothed image to highlight regions of the image with high first spatial derivatives. Edges give rise to ridges in the gradient magnitude image. The algorithm then tracks along the top of these ridges and sets to zero all pixels that are not actually on the ridge top so as to give a thin line in the output, a process known as non-maximal suppression. The tracking process exhibits hysteresis controlled by two thresholds *T1* and *T2*, with *T1* > *T2*. Tracking can only begin at a point on a ridge higher than *T1*. Tracking then continues in both directions out from that point until the height of the ridge falls below *T2*. This hysteresis helps to ensure that noisy edges are not broken up into multiple edge fragments. The effect of the canny operator is determined by three parameters the width of the Gaussian kernel used in the smoothing phase, and the upper and lower thresholds used by the tracker. Increasing the width of the Gaussian kernel reduces the detector's sensitivity to noise, at the expense of losing some of the finer detail in the image. The localization error in the detected edges also increases slightly as the Gaussian width is increased [34, 35].

Usually, the upper tracking threshold can be set quite high and the lower threshold quite low for good results. Setting the lower threshold too high will cause noisy edges to break up. Setting the upper threshold too low increases the number of fake and undesirable edge fragments appearing in the output [34, 35].

One problem with the basic canny edge detector is to do with Y junctions that mean places where three ridges meet in the gradient magnitude image. Such junctions can occur where an edge is partially occluded by another object. The tracker will treat two of the ridges as a single line segment, and the third one as a line that approaches, but doesn't quite connect to, that line segment [34, 35].

Threshold, this technique is the easiest way of segmentation. It is done through that threshold values which are obtained from the histogram of those edges of the original image. The threshold values are obtained from the edge detected image. So, if the edge detections are accurate then the threshold too. It is used to discriminate foreground from background. In this method, a grey scale image is converted into binary image. The binary image contains the whole necessary data regarding location and shape of the objects. Conversion to binary image is useful because it reduces the complexity of data. Threshold methods can be global thresholding, variable thresholding or multiple thresholding. In the global threholding, the intensity value of the input image should have two peak values which correspond to the signals from background and objects. It tells the degree of intensity separation between two peaks in an image. Appropriate threshold for global thresholding is show in Equation (12) [34, 35].

$$g(x,y) = \begin{cases} 1, & \text{if } f(x, y) > T \\ 0, & \text{if } f(x, y) \leq T \end{cases} \tag{12}$$

In variable thresholding, we separate out the foreground image objects from the background based on the difference in pixel intensities of each region. Variable thresholding, if T can change over the image. Local or regional thresholding, if T depends on a neighborhood of (x, y). Adaptive thresholding, if T is a function of (x, y). Multiple thresholding can be defined as that segments a grey level image into several distinct regions. It defines more than one threshold for the given image and divides the image into certain brightness regions and it corresponds to the background and several objects as show in Equation (13) [34, 35].

$$g(x,y) = \begin{cases} a, & \text{if } f(x, y) > T2 \\ b, & \text{if } T1 < f(x, y) \leq T2 \\ c, & \text{if } f(x, y) \leq T \end{cases} \tag{13}$$

Segmentation through thresholding has fewer computations compared to other techniques. Threshold method is the mostly used technique in image segmentation. The drawback of this segmentation technique is that it is not suitable for complex images [34, 35].

Feature based clustering, segmentation is also done through clustering. They followed a different procedure, where most of them apply the technique directly to the image but here the image is converted into histogram and then clustering is done on it. Pixels of the color image are clustered for segmentation using an unsupervised technique Fuzzy clustering. This is applied for ordinary images. If it is a noisy image, it results to fragmentation [34, 35].

A basic clustering algorithm of K-means is used for segmentation in textured images. It clusters the related pixels to segment the image. Segmentation is done through feature clustering and it will be changed according to the color components. Segmentation is also purely depending on the characteristics of the image. Features are taken into account for segmentation. Difference in the intensity and color values are used for segmentation [34, 35].

For segmentation of color image Fuzzy clustering technique is used, which iteratively generates color clusters using Fuzzy membership function in color space regarding to image space. The technique is successful in identifying the color region. Real time clustering based segmentation. A virtual attention region is captured accurately for segmentation. Image is segmented coarsely by multithresholding. It is then refined by Fuzzy C-Means clustering. The advantage is applied to any multispectral images [34, 35].

Segmentation approach for region growing is K-Means clustering. A clustering technique for image segmentation is done with cylindrical decision elements of the color space. The surface is obtained through histogram and is detected as a cluster by thresholding. Seeded Growing Region (SRG) is used for segmentation. It has a drawback of pixel sorting for labeling. So, to overcome this boundary oriented parallel pixel labeling technique is obtained to SRG [34, 35].

Model based, Markov Random Field (MRF) based segmentation is known as model based segmentation. An inbuilt region smoothness constraint is presented in MRF which is used for color segmentation. Components of the color pixel tuples are considered as independent random variables for further processing. MRF is combined with edge detection for identifying the edges accurately. MRF has spatial region smoothness constraint and there are correlations among the color components. Expectation-Maximization (EM) algorithm values the parameter is based on unsupervised operation. Multiresolution based segmented technique named as “Narrow Band”. The initial segmentation is performed at coarse resolution and then at finer resolution. The process moves on in an iterative fashion. The resolution based segmentation is done only to the part of the image. So, it is fast. The segmentation may also be done by using Gaussian Markov Random Field (GMRF) where the spatial dependencies between pixels are considered for the process Gaussian Markov Model (GMM) based segmentation is used for region growing. The extension of GMM that detects the region as well as edge cues within the GMM framework. The feature space is also detected by using this technique [34, 35].

Using external information about the objects in the image, a prior knowledge about the shape of the interest can significantly improves the segmentation results. Considering the information about the shape of objects in the image even in the form of a very basic shape template can help in image segmentation problems. This method is useful for segmenting overlapping objects in the image and it follows three steps. The first step is to extract the seed points or regions corresponding to each overlapping object. The seed points usually refer to the geometric central points or a point in the interior of the object, which conceptually provide a basic cue to separate overlapping objects. The goal is to recognize the presence and the number of the individual objects in an image as identified by seed points. The detected seed points within the process of segmentation of the object with blocked are considered priori information by which the performance of the ultimate segmentation results can be improved [36].

The second step is determining the contour evidence. The contour evidence is the visible parts of the objects boundaries that can be used to inference the blocked parts of overlapped objects. The contour evidence extraction aims to group of edge points that belonged to each object using seed points or seed regions [36].

The last step in segmentation of overlapping objects is dedicated to contour estimation, where, by means of the visual information produced from the previous two steps, the missing parts of the object contours are estimated [36].

2.5.3 Feature extraction

Feature extraction is the process to represent raw image in a reduced form to facilitate decision making such as pattern detection, classification or recognition. Finding and extracting reliable and discriminative features is always a crucial step to complete the task of image recognition and computer vision [26, 29, 37].

Feature is defined as a function of the basic measurement variables or attributes that specifies some quantifiable property of an object and is useful for classification and pattern recognition. Obtaining a good data representation is a very domain specific task and it is related to the available measurements. Various features currently employed can be classified into low level features and high level features. However, there is no clear guideline along which features should be classified as low or high level ones. Generally, quantitatively and qualitatively more complex processing is needed to derive high level features from an image than low level features. Low level features are the fundamental features that can be extracted directly from an image without any object description, while high level features extraction concerns finding shapes and objects in computer images and it is based on low level features. The Low level features can be categorized as follows, general features, global features and domain specific features. General features are application independent features such as size, color, texture, and shape. These features are morphology feature that expressed by geometric property of images [26, 29, 37].

Shape features are extracted using area, major axis length, roundness, aspect ratio, elongation, perimeter and compactness. Size feature is extracted using minor axis length. While texture features are extracted using homogeneity, correlation, energy, entropy and contrast. These features, according to the abstraction level, can be further divided into pixel level features, which are the features calculated at each pixel example color, location, etc. and local features, which are features computed over a subdivision of the image bands that are resulted from image segmentation or edge detection. The global features are features that are calculated over the entire image or just regular sub area of an image. The domain specific features are application dependent features such as human faces, fingerprints, character recognition and conceptual features. Obviously, this classification is not that sharp, there exist overlaps between them. Also, it must be considered that for some applications such as computer vision applications, the feature used should be both expressive and meaningful (associated with significant scene elements) and detectable (the location algorithm must exist). It’s worth be noted that, the lower the abstraction level of the features employed, the easier to locate them in the image, yet the more difficult to use them for understanding the meaning of that image, and vice versa [26, 29, 37].

Under feature extraction, feature selection is primarily performed to select relevant and informative features of the objects in the image. It also offers general data reduction to limit storage requirements and increase algorithm speed, feature set reduction to save resources in the next round of data collection or during utilization, performance improvement to gain in predictive accuracy, and data understanding to gain knowledge about the process that generated the data or simply visualize the data [26].

2.5.4 Classification

Object classification step categorizes detected objects into predefined classes by using suitable method that compares the image patterns with the target patterns. There are various image classification approaches. The most common classification approaches are on the basis of characteristic used, training sample used, assumption of parameter on data, pixel information used, number of outputs for each spatial element, spatial information, and multiple classifiers approach.

On the basis of characteristic used, there is shape based, this methods make use of the objects in two dimension spatial information. Common features used in shape based classification schemes are the points (centroid or set of points), primitive geometric shapes (rectangle or ellipse), skeleton, silhouette and contour. Another is motion based, this methods use temporal tracked features of objects for the classification [32, 38].

On the basis of training sample used, there is supervised classification, the process of using samples of known informational classes (training sets) to classify pixels of unknown identity. Examples of this type are minimum distance to means algorithm, parallelepiped algorithm and maximum likelihood algorithm. Another is unsupervised classification, this type of classification is a method which examines a large number of unknown pixels and divides it into number of classes based on natural groupings present in the image values. Computer determines spectrally separable class and then defines their information value. No extensive prior knowledge is required. Example of this type is Kmeans clustering algorithm [32, 38].

On the basis of assumption of parameter on data, there is parametric classifier, the parameters like mean vector and covariance matrix are used. There is an assumption of Gaussian distribution. The parameters like mean vector and covariance matrix are frequently generated from training samples. Examples of this type are Maximum likelihood and linear discriminant analysis. Another is non parametric classifier, there is no assumption about the data. Non parametric classifiers do not make use of statistical parameters to calculate class separation. Examples of this type are ANN (Artificial Neural Network), SVM, decision tree classifier, and expert system. Artificial Neural Network is well suited for complex, non linear and multi class problems [32, 38].

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 must be configured for the specific purpose before it is used. 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 [39, 40].

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 be 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's typically is 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 [39, 40].

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 [39, 40].

A multilayer neural network for learning by backpropagation 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. 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 [39, 40].

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, a multilayer neural network 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. Multilayer neural network classifier learns the class knowledge directly from the training data set. 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 [39, 40].

On the basis of pixel information used, there is per pixel classifier, conventional classifier generates a signature by using the combination of the spectra of all training set pixels from a given feature. The contributions of all materials present in the training set pixels are present in the resulting signature. It can be parametric or non parametric the accuracy may not meet up because of the impact of the mixed pixel problem. Examples of this type are maximum likelihood, SVM and minimum distance. Another is sub pixel classifiers, the spectral value of each pixel is assumed to be a linear or non linear combination of defined pure materials called end members, providing proportional membership of each pixel to each end member. Sub pixel classifier has the capability to handle the mixed pixel problem, suitable for medium and common spatial resolution images. Examples of this type are spectral mixture analysis, sub pixel classifier and Fuzzy set classifiers. Also there is per field classifier, the per field classifier is intended to handle the problem of environmental heterogeneity and also improves the classification accuracy. Generally used by Geographic Information System based classification approaches. There is also object oriented classifiers, pixels of the image are united into objects and then classification is performed on the basis of objects. It involves two stages, image segmentation and image classification, image segmentation unites pixels into objects and a classification is then implemented on the basis of objects. Example of this type is eCognition [32, 38].

On the basis of number of outputs for each spatial element, there is hard classification, also known as crisp classification. In this classification each pixel is required or forced to show membership to a single class. Examples of this type are maximum likelihood, minimum distance, ANN, decision tree, and SVM. Another is soft classification, also known as fuzzy classification. In this classification each pixel may exhibit numerous and partial class membership. It produces more accurate result [32, 38].

On the basis of spatial information, there are spectral classifiers, this image classification uses pure spectral information. Examples of this type are maximum likelihood, minimum distance, ANN. Another is contextual classifiers, this image classification uses the spatially neighbouring pixel information. Example of this type is frequency-based contextual classifier. Also there are spectral contextual classifiers, this classification uses both spectral and spatial information initial classification images are generated using parametric or non parametric classifiers and then contextual classifiers are implemented in the classified images. Examples are combination of parametric or non parametric and contextual algorithms [32, 38].

On the basis of multiple classifiers approach, different classifiers have their own advantages and disadvantages. In this approach different classifiers are combined. Some of the methods for combining multiple classifiers are voting rules, Bayesian formalism, evidential reasoning, and multiple neural networks [32, 38].

Chapter Three: Related Work

3.1 Introduction

Digital image processing is applicable in different application area to solve special problems. Quality analysis is one of them to assess quality and grading of different cereal grains and other agricultural products. Grain quality is a term that refers to the quality of each kernel. Quality of grains is an important requirement for today's market, to protect the users from substandard products. Identifying quality malt barley grain that meets brewing industry standards nowadays is difficult, as a result keeping beer uniformity during production is challenging.

Accordingly, to tackle this problem we brought some related works into consideration. In this chapter we review classification and quality assessment of grains.

3.2 Classification of grain

Zayas *et al.* [41] illustrate the use of image analysis to discriminate between wheat and non-wheat components in a grain sample. They presented two methods, multivariate discriminate and a structural prototype method for pattern recognition. The main concern in this method is the misclassification of irregularly shaped stones as wheat. The limitation in the proposed method is the requirement to manually orient the kernels.

Lai *et al.* [42] suggested some pattern recognition techniques for identifying and classifying cereal grains. This method yielded 100% accurate prediction for the samples used in the study. The pattern obtained is selected out of a great number of possible ones. These are obtained by subjective judgment and by using trial and error approach. The grains considered here were Corn, Wheat, Soyabean and sorghum.

Visen *et al.* [43] proposed algorithms to acquire and process color images of bulk grain samples of five grain types, namely oats, barley, rye, wheat, and durum wheat. The developed algorithms were used to extract over 150 color and textural features. A back propagation neural network based classifier was developed to identify the unknown grain types. The color and textural features were presented to the neural network for training purposes. The trained network was then used to identify the unknown grain types. Classification accuracies of over 98% were obtained for all grain types.

Anami *et al.* [44] have developed a Neural network approach to classify single grain kernel of different grains like wheat, maize, groundnut, redgram, greengram and blackgram based on color, area covered, height and width. The minimum and maximum classification accuracies are 80% and 90% respectively.

Mebatsion *et al.* [45] proposed a method for the classification of cereal grains, namely; barley, rye, oats and wheat (Canada Western Amber Durum (CWAD) and Canada Western Red Spring (CWRS)). This was performed using morphological and color features. The combined model defined by morphological and color features achieved a classification accuracy of 98.5% for barley, 99.97% for CWRS, 99.93% for oat, and 100% for rye and CWAD.

3.3 Quality assessment of grains

Daniel Hailemichael [46] developed automatic maize quality assessment system using image processing techniques and with help of ANN classifier based on the standard for maize set by the Quality and Standards Authority of Ethiopia. This work has a total of 24 features (14 color, 8 shape and 2 size) have been identified to model maize sample constituents. A feed forward artificial neural network classifier with back propagation learning algorithm is used. 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.

Chhabra and Reel [47] developed morphology based feature extraction and recognition for enhanced wheat quality evaluation. This work is done on the basis of simple mathematical calculations and with a number of wheat grains parameters calculated. Classification of wheat grains using morphological parameters. The quality of the wheat can be judge by its length, thickness, width, area, etc. The grain types used in the study were hard wheat and tender wheat. This work used neural network for assessment of wheat grain. The contours of whole and broken grains have been extracted, precisely normalized and then used as input data for the neural network. Based on 5 basic geometric features, 12 digital morphological features are defined which are used for wheat recognition. By measuring different parameters and forming a matrix of the input data which is given as an input to the neural network for training purposes the output is obtained. Output data obtained which is greater than 0.5 is termed as wheat grain otherwise as a broken grain. The accuracy achieved is 98% which is quite high and can be made more accurate if more number of hidden neurons is used as the authors suggested.

Ivanov *et al.* [48] developed quality assessment of corn grain sample using color image analysis. This work articulates that grain quality can be assessed based on different grain features like appearance, shape, color, smell, flavour, moisture content, infections, presence of impurities, etc. The main indexes for the quality of grain samples are related to the color characteristics and the shape of the grain sample elements. Corn grain samples of the 433 variety were used in the investigation. The extraction of a specific zone in the color image is a typical classification task for association of RGB pixels to one of the predetermined classes. Evaluation of the color features of each of the pixels from the object region, followed by the association of the pixels to one of the models defined and detection of regions with neighboring pixels, whose number exceeds some threshold value. The color zone models were used to determine if one of the typical combinations of color zones exists in the image of the investigated object. The classifiers used for object shape recognition were trained and validated using K-fold cross-validation procedure. Two different approaches for fusing the results from object color and object shape analyses were investigated. The training and testing errors of the developed procedures were evaluated. The developed methods and tools for description and analysis of the color characteristics and the shape of the grain sample elements gave the acceptable accuracy when the non grain impurities were excluded from the training sets. When the non-grain impurities were included in the test samples, the classification accuracy sufficiently decreased. The authors suggested that the selection of a proper classifier for recognition of the object color and shape classes had a significant influence over the classification accuracy.

C. Ferreira *et al.* [21] developed image analysis methods to evaluate barley malt grain size. The main objective of this work is to develop image processing and analysis field methodology that allow to determine the barley malt grains weight distribution throughout commonly used size ranges within the supply chain management from the barley crop purchasing to the quality reception control. This work was developed in three distinct stages, the first one is related to the sampling and grain size determination by the standard EBC

(European Brewery Corporation) method, the second to the development of the image analysis (IA) proposed methodology and the third to the proposed IA method results evaluation. The grain size ranges studied in this work were below 2.2 mm, 2.2 to 2.5 mm, and 2.5 to 2.8 mm and above 2.8 mm in minor axis. Also the sum of the 2.5 to 2.8 mm and above 2.8 mm fractions (leading to an above 0.5 mm fraction) was studied being one of the most important parameters in the supply chain management. The obtained results were thereafter plotted against the traditional standard EBC method results and their correlation studied. The analysis of sampling size also carried out revealed that for a subset down to around 200 grains (5 images) the results were still rather similar to the whole set analysis (550 grains from around 16 images). This work was found that the accuracy of the proposed IA method weight distribution was quite sensitive to the image resolution. Indeed, a relatively small error of 2.5% in the determination of the grains minor axial axis could propagate into a maximum error of 15 to 20% in the determination of the estimated weight percentage of each barley malt size fraction. The acquisition of a higher resolution camera such as 3200x2600 pixels, however, could drop down these values to an error below 10%. One must keep in mind, though, that the maximum propagation errors do not necessarily imply average final percentage differences of the same magnitude such as it was proven the fact that during this work the final absolute percentage differences did not surpass 5% in any case.

3.4 Summary

There are several research works done on classification and quality assessment for different grains. Some use image analysis method to discriminate grain and non grain components in a sample, other use pattern recognition techniques for identifying and classifying cereal grains, while other use back propagation neural network to identify unknown grains, and also assess quality (evaluate quality) using different features.

Despite the fact that there are several related works done in the area of image analysis to check quality of grains specially for cereal grains, to the best of our knowledge there is no earlier work that was fully implemented to occupy the specification of Ethiopian Standards Agency for assessing malt barley quality using observable measurement. Assessing malt barley quality is a critical task to ensure the quality of beer. Due to the complex problem in identifying quality of malt barley, the identification of quality process is carried out manually by quality assurance staff. To overcome the expense and inconsistencies of the process, an automated solution for assessing malt barley grain quality is desirable. Other researcher works on other cereals grain could not be directly applied to solve this problem since other cereals grain morphological characteristics are different. Cereals differ in detail in the way the characters of each species are expressed, despite the fact that they share a general commonality in growth habit and flowering parts. A part from this, each works use segmentation and features that model the problem they faced. The features identified in others work cannot be representative for malt barley. Example maize quality assessment has its own features (color features, shape features, size features, and texture features) that can model maize. Using maize features it is impossible to model malt barley, this holds true for other research works we cannot use their features used to solve their specific problem to solve malt barely quality assessment and grading problem. Hence, malt barley grain quality assessment is still performed manually.

There is attempt of image analysis methods to evaluate malt barley grain size, but it is not accurate since it misses some criteria to evaluate quality as per Ethiopian Standards Agency. Besides, differently from this attempt, we use other additional criteria to assess quality of barley and give grade as per Ethiopian Standards Agency standards to keep uniformity of malt barley grain for brewing industry.

Chapter Four: Design for barley relevance and quality assessment for brewing industry

4.1 Introduction

One way of meeting the requirement of standard is through conducting a quality assessment and finally giving grade. Accordingly this research dealt with gaining results that specify quality of malt barley using Ethiopian Standards Agency specification by implementing image processing method.

Under this chapter the system architecture is described with its entire component which contains image preprocessing, segmentation, feature extraction and classification that lead to quality determination.

4.2 System architecture

In malt barley quality assessment image preprocessing, segmentation, feature extraction and classification techniques are mandatory for gaining excellent result for the determination of quality.

The system architecture for assessing malt barley quality sample consists of four components namely preprocessing, segmentation, feature extraction and classification. The system architecture for the proposed system is diagrammatically showed in Figure 4.1. The first part of the proposed architecture is the preprocessing component. This part does the activity of preprocessing on the input malt barley image. Noise filtering from input image, binarization, and converting of RGB to grayscale image is done here. In addition to this, the component does the preliminary task of making the input image ready for segmentation component.

The segmentation component of the system architecture is responsible for carrying out the separation of malt barley sample constituents from each other. This component contains two sub components namely edge segmentation and overlapped malt barley segmentation. The edge segmentation component use combination of three edge detectors Canny, Sobel, and Roberts to get more sharp and clear edges. The overlapped malt barley segmentation component use output of edge segmentation as input then seed point extraction is performed by combining morphological filtering and intensity based region growing. This step followed by contour evidence extraction process and finally contours estimation process deals with ellipse fitting techniques to estimate contours of missing parts of malt barley.

The feature extraction component of the system architecture is responsible for extracting the seventeen features. This component contains the size feature extraction, texture features extraction, shape features extraction, and the color features extraction sub components.

The classification component of the proposed architecture is the last component. It classifies objects found in the image to classes using extracted features and finally gives grade for pure malt barley grains depending on size feature.

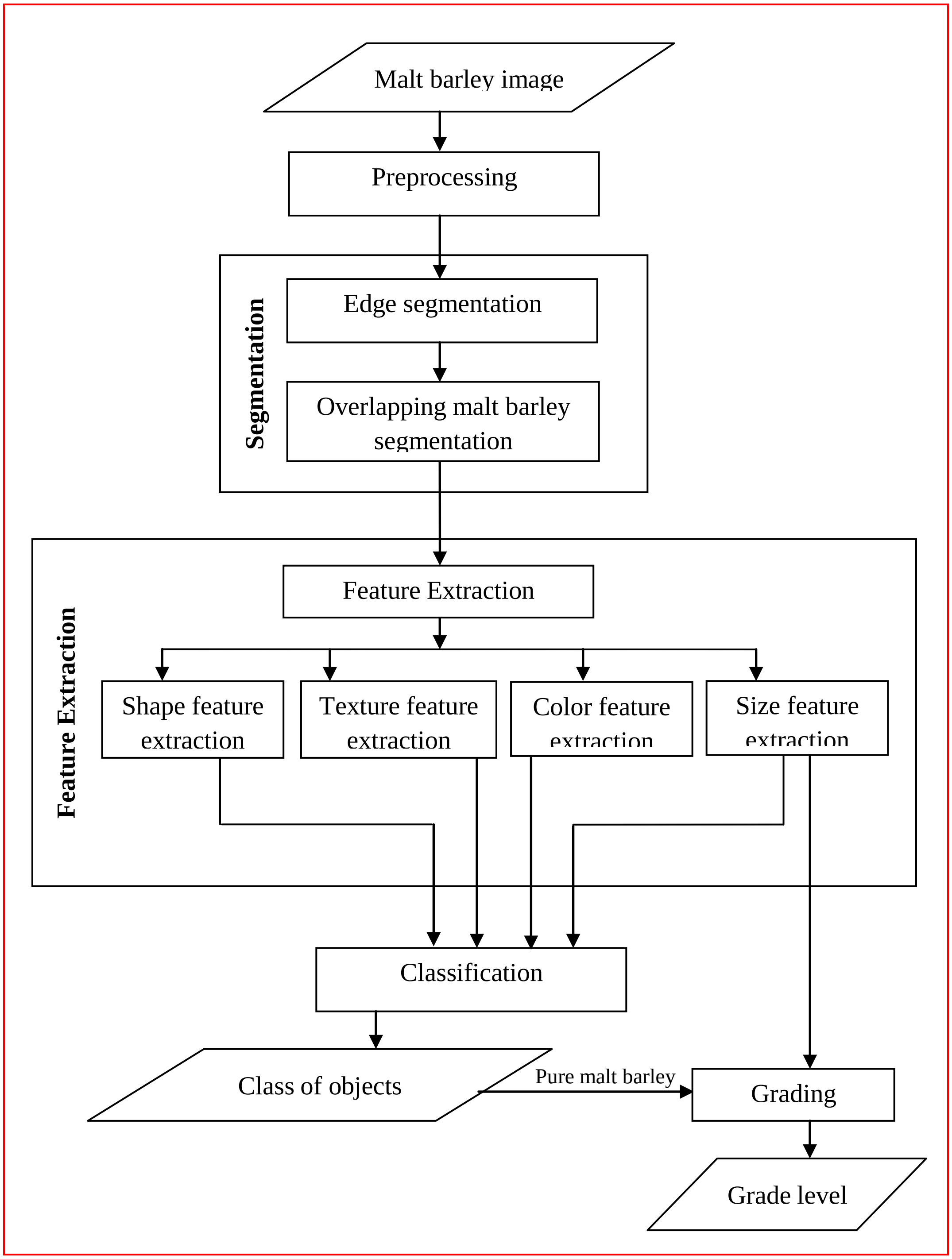
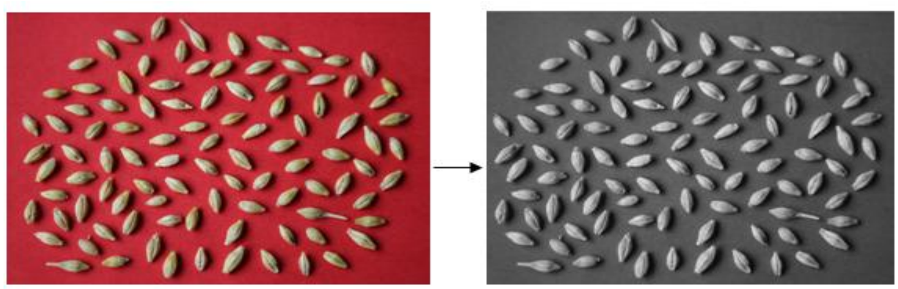


Figure 4.1 Architecture of proposed system

4.3 Preprocessing

The preprocessing component of the architecture is used for improvement of the image data that suppresses unwilling distortions or enhances some image features important for further processing. It is operations with images at the lowest level of abstraction.

In preprocessing the first step is converting RGB to grayscale as show in Figure 4.2. This is done because grayscale images only contain one image plane or it mean that it only contain the gray scale intensity values when compared to RGB which holds three components. This allows data to be reduced during process. This data reduction allows the algorithm to run in a reasonable amount of time than using RGB which asks to extract three components individual and finally combining those together to get assumed result.



(a) (b)

Figure 4.2 (a) RGB image (b) The convert grayscale image

Noise filtering is used to filter the unnecessary information from an image in preprocessing stage. Filters change a pixel’s value taking into account the values of neighbouring pixels. There are various filters, but in this research we consider the median filtering techniques since it offer advantages such as, no reduction in contrast across steps, it does not shift boundaries, and it is less sensitive to outliers as show in Figure 4.3.

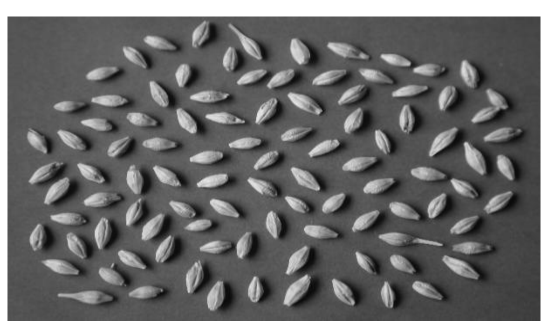


Figure 4.3 Grayscale image filtered by median filtering techniques

Image binarization is a process of separation of pixel values of an input gray image into two pixel values (0 and 1) or conversion of image to only black and white as show in Figure 4.4. This is done for each pixel in the image by using appropriate threshold value for binarization. Here white is used for the objects in the image as color of foreground while the rest of the image is background color viewed as black in this image.

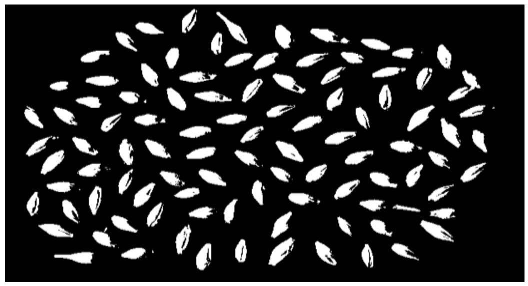


Figure 4.4 Binarized image

4.4 Segmentation

This component part of the architecture is used to partition a digital image into multiple segments. The goal of segmentation is to simplify and change the representation of an image into something that is more meaningful and easier to analyze. Segmentation can also be regarded as a process of grouping together pixels that have similar attributes.

4.4.1 Edge segmentation

Edge is the boundary between two regions with relationally distinct gray level properties. Edge detection is the process of localizing pixel intensity transitions or it is a process of finding sharp discontinuities in an image. Different operators used for edge detection, but in this study, we used Sobel, Roberts, and Canny in combination for better edge detection since it gives more sharp and clear edges as compared to separate result of each edge detector.

Sobel edge detector performs a two dimensional spatial gradient measurement on an image and emphasizes regions of high spatial frequency that correspond to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an input grayscale image. This operator consists of a pair of 3×3 convolution kernels and the result of Sobel edge detector is illustrated in Figure 4.5.

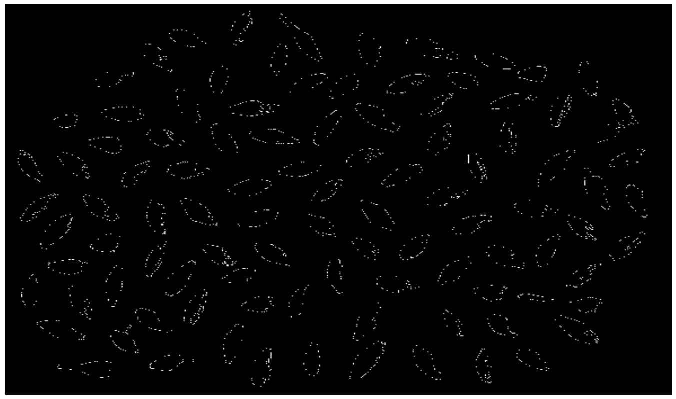


Figure 4.5 Sobel edge detector result

The Roberts edge detection is simple and quick to compute. It is a two dimensional spatial gradient measurement on an image. It thus highlights regions of high spatial frequency which often correspond to edges. In its most common usage the input to the operator is a grayscale image and edge as output as illustrated in Figure 4.6. Pixel values at each point in the output represent the estimated absolute magnitude of the spatial gradient of the input image at that point. This edge detection consists of a pair of 2×2 convolution kernels.

Figure 4.6 Roberts edge detector result

Canny edge detector takes a gray scale image as input and produces an image showing the positions of tracked intensity discontinuities as output. The effect of the canny operator is determined by three parameters the width of the Gaussian kernel used in the smoothing phase, and the upper and lower thresholds used by the tracker. Increasing the width of the Gaussian kernel reduces the detector's sensitivity to noise, at the expense of losing some of the finer detail in the image. The localization error in the detected edges also increases slightly as the Gaussian width is increased.

Usually, the upper tracking threshold can be set quite high and the lower threshold quite low for good results. Setting the lower threshold too high will cause noisy edges to break up. Setting the upper threshold too low increases the number of fake and undesirable edge fragments appearing in the output as show in Figure 4.7.

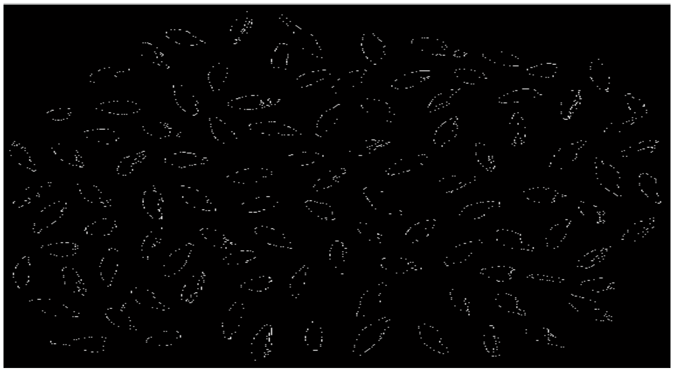


Figure 4.7 Canny edge detector result

Most of the major edges are detected and lots of details have been picked out well, but this has too much detail for subsequent processing. In this study, we used combination of these three edge detectors named Sobel, Canny, and Roberts that merges different qualities of individual detectors. This combination method gives better sharp and clear edges than individual detection method as show in Figure 4.8.

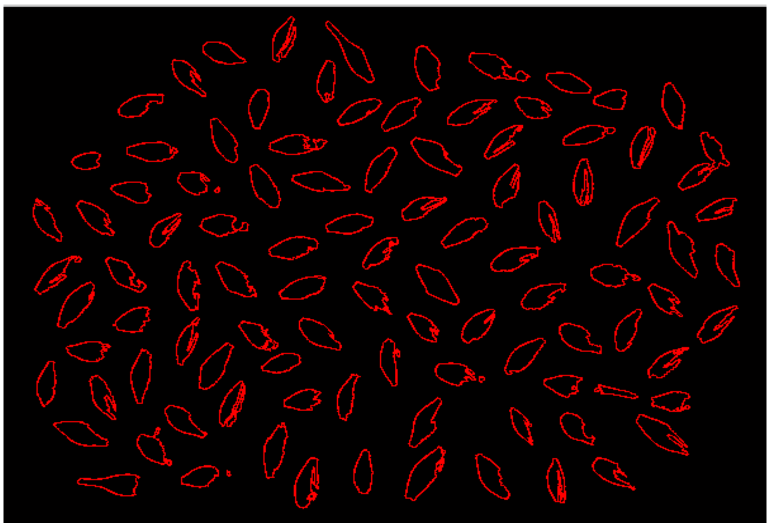


Figure 4.8 Combination of Sobel, Canny and Roberts result

4.4.2 Overlapping malt barley segmentation

In image processing segmentation of overlapping objects is considered to be the challenging task that tries to address the segmentation of multiple objects with partial views. The presence of multiple blocked objects is the main difficulty for segmenting of overlapping objects. This method is useful for segmenting overlapping objects in the image and it follows three process steps as show in Figure 4.9.

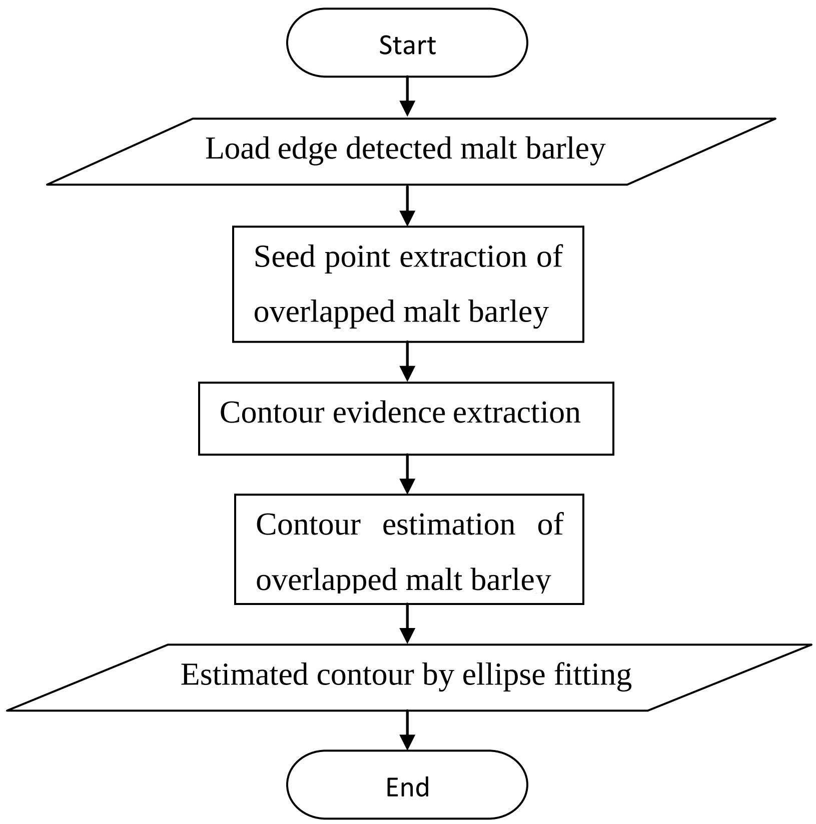


Figure 4.9 Diagrammatic depiction for overlapped malt barley image segmentation

Seed point extraction of overlapped malt barley use edge detected malt barley image as an input which is gained from edge segmentation stage and then proceeds to seed point detection of overlapped malt barley as illustrated in Figure 4.10. Seed point extraction is performed by combining morphological filtering and intensity based region growing.

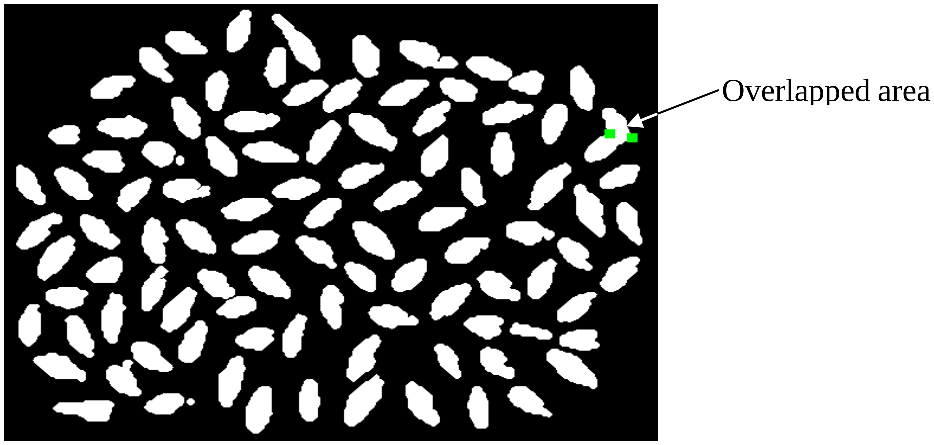


Figure 4.10 Seed point of overlapped area

The obtained seed points are then introduced to the contour evidence extraction process. The contour evidence extraction is done by two different methods and the result is show in Figure 4.11. First is edge to marker association method. This method relies on the relationship between the visual contours of overlapped objects and the object seed points. By combining the distance and divergence metrics the method assigns the edge pixel obtained from image gradient to seed points. Second is concave point based method. This method applied to overlapping object for segmentation by performing the task of contour evidence extraction by combining a contour splitting method with ellipse fitting.

Figure 4.11 Extracted contour evidence

The contour estimation process deals with ellipse fitting techniques which are basically used in the task of contour estimation to guess the missing parts of partially observed object. This method generally depend exclusively on available object boundary points where the ellipse is fitted using a parametric model. In contour estimation, active contour is useful for knowing the set of overlapping objects with shape prior. Active shape model is used which is extended to address the segmentation of multiple overlapping objects simultaneously. Active shape model in its level set form is an implementation of level set based active contour image segmentation where a shape prior knowledge is combined as shape driven functional.

The active contour with shape prior in its original form is an extension of the two phase in which the region and short boundary enforcing functional are combined with a functional governing the shape prior constraint as show in Equation (14).

$$F = F_{region} + F_{boundary} + F_{shape} \tag{14}$$

by means of the visual information produced from seed point extraction of overlapped malt barley and contour evidence extraction process steps, the missing parts of the object contours are estimated as show in Figure 4.12.

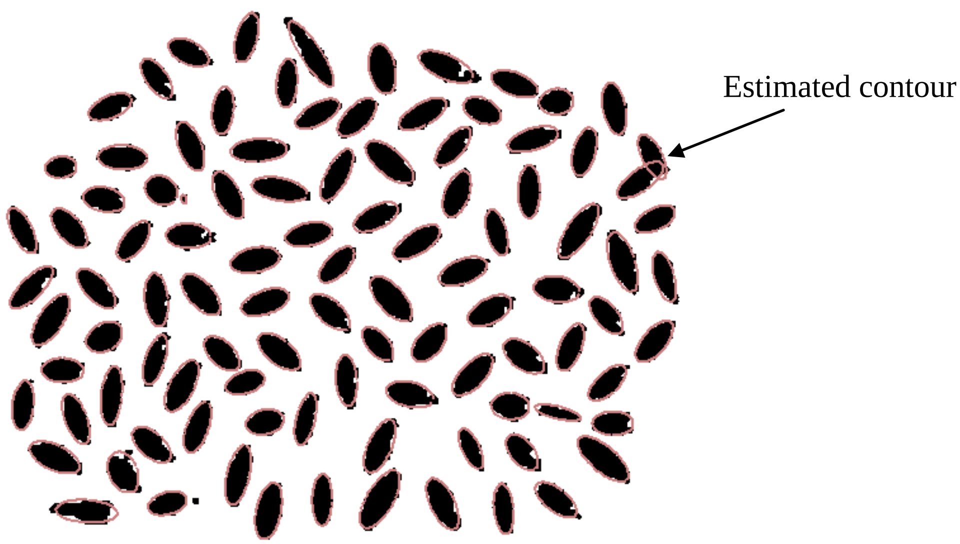


Figure 4.12 Estimated contours by ellipse fitting

4.5 Feature Extraction

This component part of the architecture extracts basic features needed in order to perform classification of targets. Features are those items which uniquely describe a target. Accordingly size, shape, texture, and color are used for malt barley quality assessing purpose. Segmentation techniques are used to isolate the desired object from the image so that measurements can be made on it independently. Quantitative measurements of object features allow classification and description of the image. Low level features are used for extracting features for assessing quality of malt barley.

In this study after we preprocess and achieve the desired level of segmentation, we apply some feature extraction techniques to the segmented object to obtain features, which is followed by application of classification. It is essential to focus malt barley feature extraction phase as it has an observable impact on the efficiency of the classification. Feature selection of a feature extraction method is the single most important factor in achieving high classification and grading performance. Feature selection is obtaining raw data information that is most suitable for classification purposes, while minimizing the within class pattern variability and enhancing between class pattern variability. Thus, selection of a suitable feature extraction technique for malt barley image needs to be done with maximum care. Accordingly size, shape, texture and color feature are extracted. These four features are extracted because the discrimination power obtained from one, combination of two or three were not enough to illustrate the actual difference between objects found in the sample image of malt barley grain as show in Table 4.1.

Table 4.1 Object types that might found in sample image

No	Object type
1	Pure malt barley
2	Food barley in mixed form
3	Foreign matter
4	Discolored barley
5	Broken malt barley grains
6	Wrinkled malt barley grains

4.5.1 Size feature extraction

Size features for this study extracted based on output result of sieve size as show in Figure 4.13 gained from Holeta Agricultural Research Center. Morphological features used for extracting malt barley grain size are calculated from the basic feature.

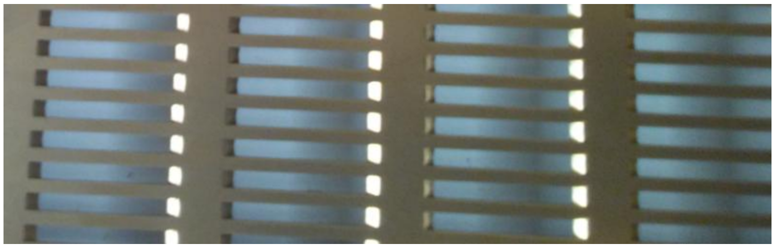


Figure 4.13 Size sieve

The feature used for size extraction is minor axis length. It is the distance between the end points of the longest line that could be drawn through the malt barley grain while maintaining perpendicularity with the major axis. Since malt barley grain looks like ellipse, the minor axes of an ellipse are diameters (lines through the center) of the ellipse with the shortest diameter as illustrated in Figure 4.14.

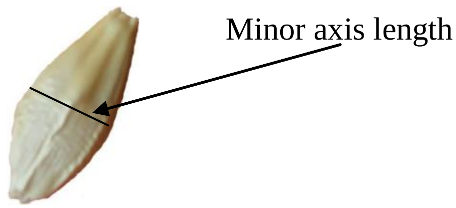


Figure 4.14 Minor axis length of malt barley grain

Sieve size depends on the minor axis length of malt barley grain to determine size for grading purpose.

4.5.2 Shape feature extraction

The shape of an object refers to its physical structure and profile. Shape features are mostly used for finding and matching shapes, recognizing objects or making measurement of shapes. In this study we used shape feature of malt barley grain to measure its shape for identifying pure malt barley grain from foreign matter and mixed food barley. Major axis length, area, aspect ratio, elongation, perimeter, roundness and compactness are characteristics used for shape feature extraction technique.

i. Major Axis Length. It is the distance between the end points of the longest line that could be drawn through the malt barley grain. The major axis end points are found by computing the pixel distance between every combination of border pixels in the malt barley grain boundary and finding the pair with the maximum length. The major axes of an ellipse are diameters (lines through the center) of the ellipse with the longest diameter as illustrated in Figure 4.15.

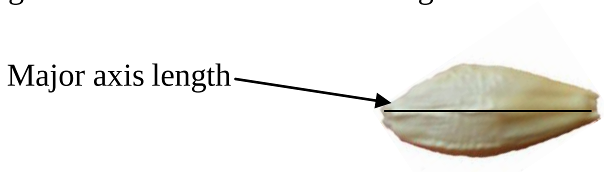


Figure 4.15 Major axis length of malt barley grain

ii. Area. The number of pixels inside the region covered by a malt barley grain depicted by white color, including the boundary region pixels as show in Figure 4.16.

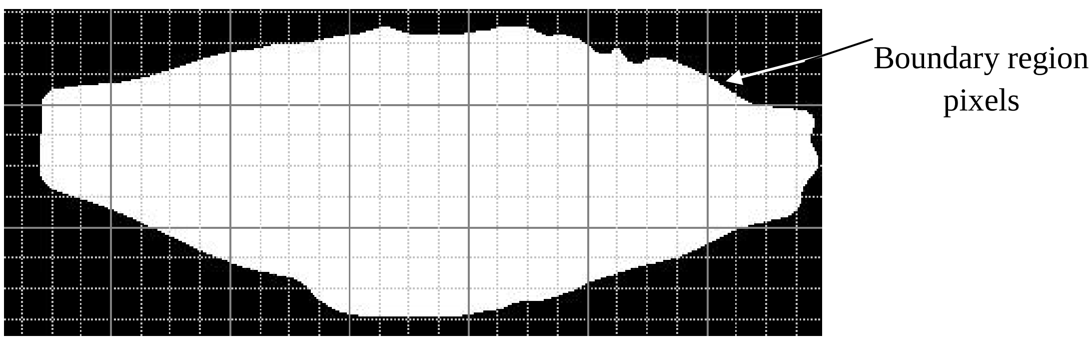


Figure 4.16 Area of malt barley grain

iii. Aspect ratio. It is the ratio of the length of the major axis to the length of the minor axis.

iv. Elongation. It is the inverse of aspect ratio which is calculated as a ratio of minor axis length to major axis length.

v. Perimeter. The length of the outside boundary of the region covered by a malt barley grain.

vi. Roundness. Measures the degree of roundness or circularity.

vii. Compactness. It is the ratio of perimeter square to the area of the malt barley grain.

4.5.3 Texture feature extraction

Texture feature is a basic characteristic of image that measures properties, such as smoothness, roughness and regularity of pixel structure. There are different texture feature extraction methods, but in this study we used homogeneity, correlation and contrast value to get useful information of texture of malt barley grain. This features extracted because farmers holding fragmented pieces of land produce different quality of malt barley in the country. As a result beer quality decrease, brewing industry nowadays need uniform malt barley to come up with similar production level of beer for their customer. Furthermore, uniformity in production of beer is the ultimate goal of any beer producer. As a consequence the following textural features are used to test uniformity of malt barley grain.

i. Homogeneity. It measures the sameness feature of the intensity variation within the image and calculated using Equation (15).

$$Homogeneity = \sum_{x,y} \frac{p(x,y)}{1+|x-y|} \tag{15}$$

ii. Contrast. It measures the strength difference between intensity in an image and is calculated using Equation (16).

$$Contrast = \sum_{x,y} |x,y|^2 p(x,y) \tag{16}$$

iii. Correlation. The connection or link between intensity variation within the image and is calculated as Equation (17).

$$Correlation = \frac{\sum x,y\,[(xy)p(x,y)] - \mu_x \mu_y}{\sigma_x \sigma_y} \tag{17}$$

Where p denotes the number of occurrences of gray levels within a given image, which shows the value of the element within co-occurrence matrix, while x and y show the intensity couple from the neighboring intensity. This neighboring couples in co-occurrence matrix act as row and column matrix and μx, μy are means and σx, σy are the corresponding standard deviations.

In this research we used statistical method of examining texture that considers the spatial relationship of pixels in the gray-level co-occurrence matrix (GLCM), also known as the gray-level spatial dependence matrix for texture feature extraction. The GLCM functions characterize the texture of an image by calculating how often pairs of pixel with specific values and in a specified spatial relationship occur in an image, creating a GLCM, and then extracting statistical measures from this matrix. Accordingly, contrast measures the local variations in the gray-level co-occurrence matrix, and correlation measures the joint probability occurrence of the specified pixel pairs, while homogeneity measures the closeness of the distribution of elements in the GLCM to the GLCM diagonal.

Graycomatrix creates the GLCM by calculating how often a pixel with gray-level (grayscale intensity) value *i* occurs horizontally adjacent to a pixel with the value *j*. It is possible to specify other pixel spatial relationships using the 'Offsets' parameter. Each element (*i*,*j*) in GLCM specifies the number of times that the pixel with value *i* occurred horizontally adjacent to a pixel with value *j*. By default, graycomatrix calculates the GLCM based on horizontal proximity of the pixels [0 1]. That is the pixel next to the pixel of interest on the same row. The GLCM is implemented using MATLAB code for the system is presented in Annex A.

4.5.4 Color feature extraction

Color is a commonly used important feature for digital image representation. This is very important as it is invariant with respect to scaling, translation and rotation of an image. Similarity measurements of colors are done by extracting color features.

The color depth measures the amount of color information available to display or print each pixel of a digital image. A high color depth leads to more available colors and consequently to a more accurate color representation. For example, a pixel with one bit depth has only two possible colors black and white (known as binary image). A pixel with 8 bits depth has 256 possible values (known as grayscale image) and a pixel with 24 bits depth has more than 16 million of possible values (known as RGB or true color image).

In true color image analysis three attributes are used to differentiate one color from another brightness, hue, and saturation. The hue attribute brings the information concerning the main wavelength in the color. It is responsible for verifying the color in the complete spectrum from red to violet and magenta. The saturation describes the level of mixture between the hue and the white light. It determines the "purity" of the color. High values of saturation result in more gray scale pixels and small values result in pixels with high "purity". The chromaticity is a description that combines hue and saturation. Hence, it is possible to describe an image according to brightness and chromaticity.

In this study we convert RGB color system to HSV, because it allows the separation of the three components of a specific color (hue, saturation, and intensity). It is broadly used in artificial vision systems, as it is a powerful tool for the development of digital image processing algorithms based on the human color perception model. In fact, the HSV model is well suited to characterize colors in practical terms for human interpretation, differently from the RGB and CMYK models. In this research we used color feature to identify amount of discolored malt barley grain.

4.6 Classification

Object classification step categorizes detected objects into predefined classes as show in Table 4.1 by using suitable method that compares the image patterns with the target patterns. There are different types of classification techniques, but in this study we used Support Vector Machine, K Nearest Neighbors and Artificial Neural Network. The performances of each classifier compared and ANN is selected as permanent classifier for this study. Despite the fact that it perform better than SVM and KNN, it also holds several reasons for choosing artificial neural network over other methods for the purpose of this research work. Although there are other methods like mathematical functions, rule-based algorithm or statistical methods available for classification, we chose ANNs over others.

The classification of objects found in malt barley sample image cannot be easy to classify using unique mathematical functions. This is due to the variation in size, shape, color, and texture of the objects under consideration in sample image. 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 back propagation network could be easily modified to accommodate more features.

To add empirical experience to the above claims, we trained SVM, KNN, 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 Support Vector Machine and K Nearest Neighbors classifier.

The neural network architecture in this work is a 3 layered feed forward 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 17 neurons corresponding to each 17 selected features as inputs and the output layer consists of 6 neurons corresponding to each 6 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 12 and 25 neurons for object class determination and grade level categorization respectively.

This number of neurons in the hidden layer is selected empirically based on the performance it exhibited over smaller and larger number of neurons. The network is designed to use back propagation algorithm training.

To measure the performance of the network during training phase, we preferred to use cross-entropy error function over mean square error. Mean squared error is appropriate to regression (line/curve fitting) where the goal is to minimize the mean squared error between the training set (points) and the fitted curve. Cross entropy is appropriate to classification where the goal is to minimize the number of miss classified training samples by imposing an exponentially increasing error the closer an output comes to being "1" when it should be "0", and vice versa. Compared to mean square error, cross entropy function is proven to accelerate the back propagation algorithm and it provide good overall network performance. The architectural design of this ANN is depicted in Figure 4.17 for classifying objects in a sample image as a class and Figure 4.18 for giving grade level category of pure malt barley.

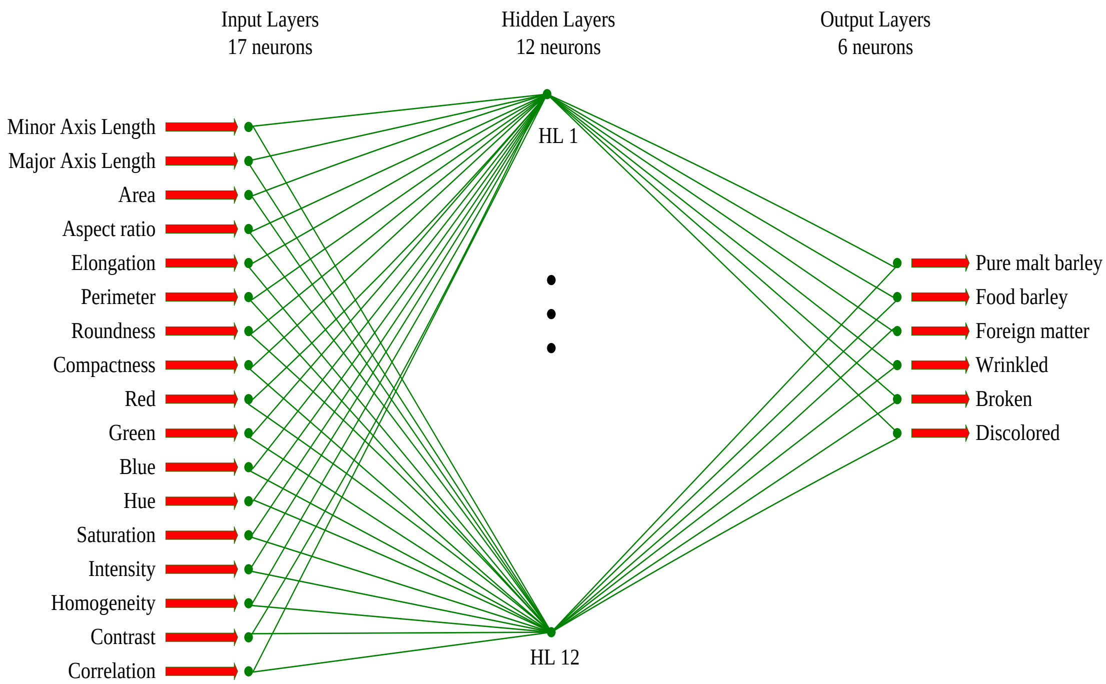


Figure 4.17 Design of ANN used for the classification objects found in malt barley sample

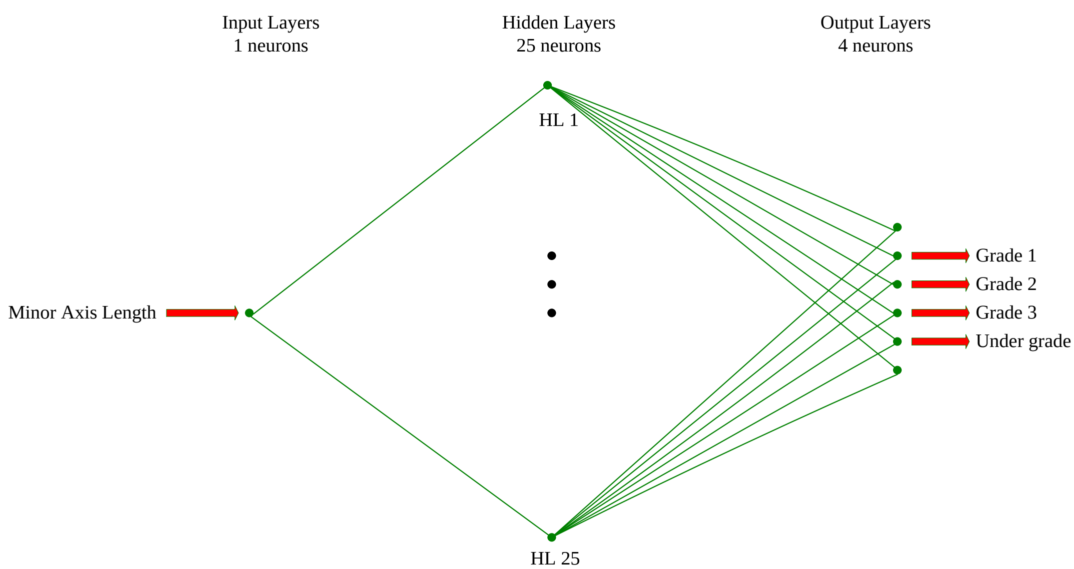


Figure 4.18 Design of ANN used for the grading pure malt barley sample

4.7 Grading

Grading is necessary in the full development of quality assessment to define the relationship between grades and prices for each grade level category. In this study we grade pure malt barley to identify its grade level. The practices used to categorize malt barley into different grade level contain other constituents found within the malt barley grain sample. Example if there is foreign matter in a sample, it is considered as some grade that it fit on seize size such as, it may remain on 2.8mm, 2.5mm, or 2.2mm seize size in manual work, but in our study we first classify objects in to identified class then only those pure malt barley grain is graded.

4.8 Summary

In this chapter, the design of automatic barley quality assessment for brewing industry is discussed in detail. The components that constitute the system, the relationship among these components and the responsibility of each component are presented. The proposed system architecture consists of four components namely, preprocessing, segmentation, feature extraction, classification and grade level. The preprocessing component care out RGB to grayscale conversion, image filtering and binarization then pass output image for segmentation component. The segmentation component introduced new edge segmentation method by combining three existing edge detection method then followed by ellipse fitting method to segment overlapped malt barley grain by estimating contour using prior knowledge about the object. The feature extraction component performs feature extraction on the output of the segmentation component. This component extracts a total of 17 (1 size, 7 shape, 3 texture and 6 color) features that are identified for the purpose of modeling the different characteristics of malt barley sample constituents. The classification component, classifies objects found in the image as show in Table 4.1 based on the extracted features from feature extraction component and then inform the grade level of malt barley grain as per Ethiopian Standards Agency specification.

Chapter Five: Experiment

5.1 Introduction

In this chapter, we discuss the experiments carried out to test the effectiveness of our proposed system. Accordingly, detail implementation procedure, dataset preparation, the type of classifier to be used depending on performance of classifying and the results achieved in the classification and grading process will be discussed.

5.2 Sample collection

Since there is no readymade dataset for this type of research, we have prepared our own dataset for training and performance evaluation. To do this, different malt barley grains have been collected from Ethiopian Standards Agency laboratory, Assela Malt Factory, Qulumsa Agricultural Research Center, Holeta Agricultural Reseach Center and Ganda Shano Farmers Training Center (FTC). Eleven malt barley variety was collected from different area of Ethiopia as show in Table 5.1 and one variety of food barley called dimtu used in this study to get some property of food barley to differentiate it from pure malt barley to solve existence of admixture of food barley in to malt barley lot during purchasing by brewing industry.

Table 5.1 Malt and food barley varieties

No.	Variety
1	Holker
2	Bekoji-1
3	EH-1847
4	Bahati
5	Sabini
6	Grace
7	Traveller
8	IBON 174/03
9	Fanaka

10	HB 1963
11	HB 1964
12	Dimtu

A total of 468 images are captured for training and testing the proposed system. The data were partitioned randomly into training, validation and test sets. Image acquisition is done using a Canon EOS 5D mark III and SONY DSC-H300 as show in Figure 5.1. All sample images stored as JPEG file format.

Figure 5.1 Cameras used during sample image collection

5.3 Implementation

In this study three classification techniques used, for each classification techniques 70% of training data is used and the rest of the data used for validation and testing each consisting of 15% of the input data.

The number of training dataset is high because it should be ensure that the training dataset should include all possible patterns used for defining the problem. The validation dataset provides an unbiased evaluation of a model fit on the training dataset. Validation datasets used for regularization by early stopping. Stop training when the error on the validation dataset increases, as this is a sign of over fitting to the training datasets. A test dataset is independent of training dataset, but that follows the same probability distribution as the training dataset.

To collect data of correct measurement an equipment called seize size analyzer were used as show in Figure 5.2 and the result of seize size used for capturing training data that feed into each classification techniques is illustrated in Figure 5.3.

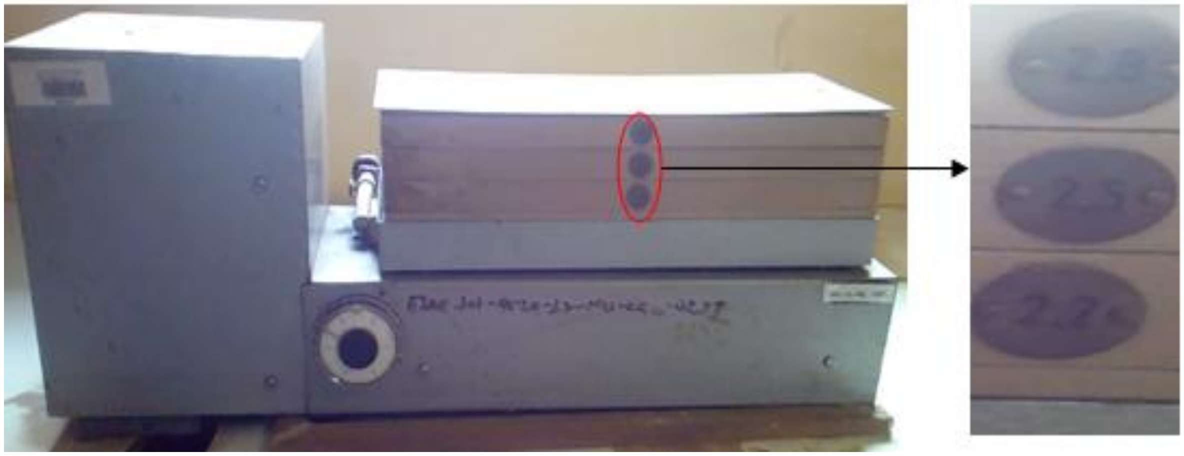


Figure 5.2 Seize size analyzer

Seize size analyzer has three seize size corresponding 2.8mm, 2.5mm and 2.2mm that looks like picture show in Figure 4.13 at in side for each size. Malt barley or other object found in the sample below 2.2 captured on flat metal as illustrated in Figure 5.3.

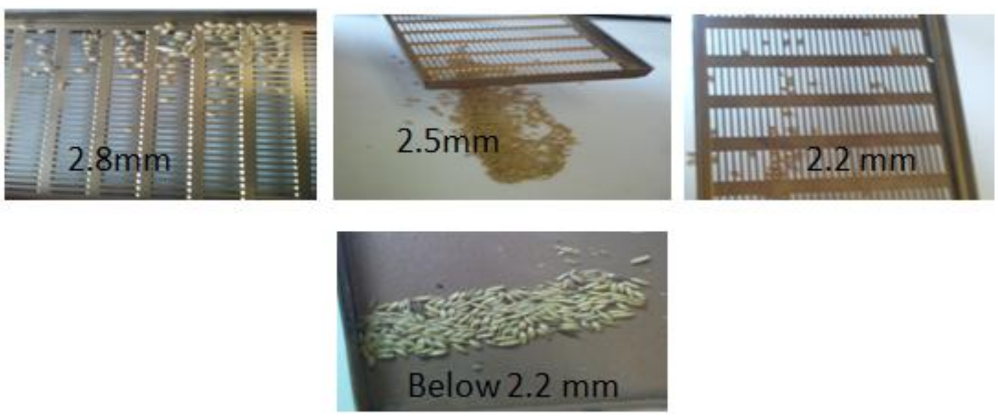


Figure 5.3 Seize analyzer results

After collecting seizes size result we captured each category by taking 20 malt barley grains with total of 80 grains as show in Figure 5.4. The image background used during image capturing for training data is black. This is selected because it gives superior distinguishing capacity of malt barley grain found as foreground. In fact the mean RGB value of malt barley is calculated and then the opposite of mean RGB value is used as background as Figure 2.16 illustrated.

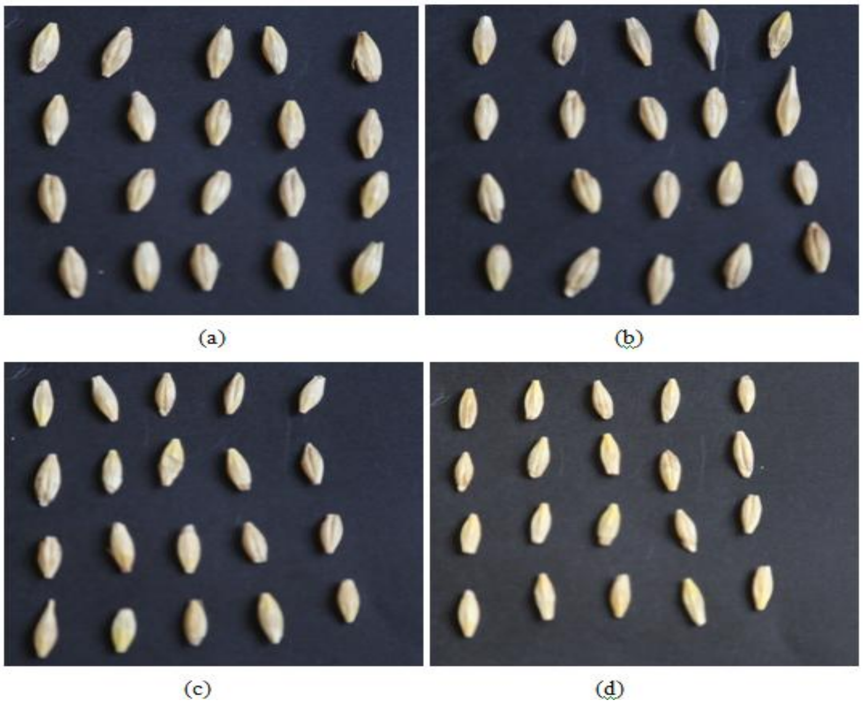


Figure 5.4 Image captured from seize size (a) 2.8mm (b) 2.5mm (c) 2.2mm (d) below 2.2mm

Each captured image minor axis length is used for classification of grade since the seize size constructed based on shortest diameter of malt barley grain.

Accordingly Minor Axis Length (MAL) for each 20 grains found in the image are extracted and sample of 15 grains MAL showed in Figure 5.6 for each category.

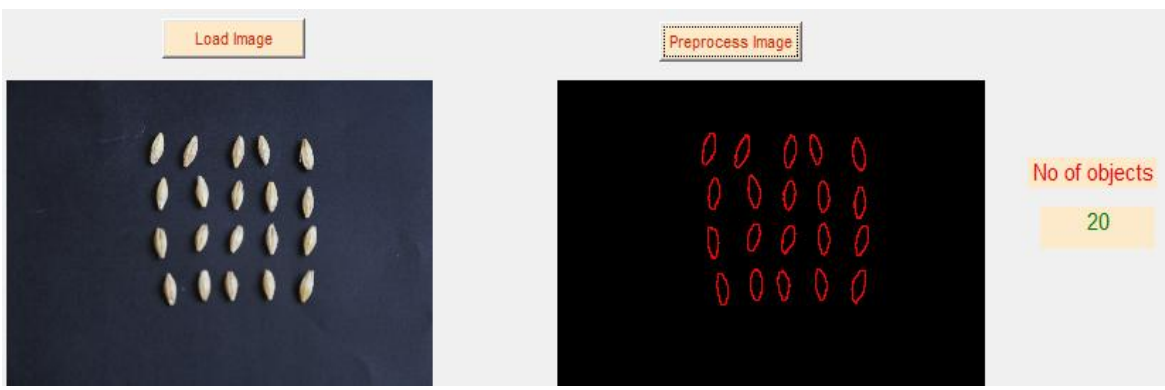


Figure 5.5 Preprocessed image

The input sample image implemented using MATLAB code of the system is presented in Annex A. After preprocessing and segmentation sample image showed in Figure 5.5 MAL is extracted.

	MAL of 2.8mm		MAL of 2.5mm		MAL of 2.2mm		MAL of below 2.2mm
1	151.0339	1	136.8158	1	121.9972	1	103.1320
2	144.8484	2	138.1795	2	119.7101	2	124.0772
3	140.4755	3	134.6766	3	136.9467	3	109.9079
4	141.6226	4	145.5645	4	125.5174	4	122.7332
5	145.4923	5	135.3673	5	117.8371	5	113.5088
6	147.8080	6	143.5987	6	117.9693	6	122.6023
7	148.4621	7	154.0329	7	129.0941	7	117.3605
8	155.0914	8	138.4719	8	141.9155	8	116.3796
9	147.5623	9	116.8888	9	109.6380	9	105.5908
10	150.7656	10	147.5905	10	134.0573	10	122.5360
11	146.5973	11	141.6110	11	133.9060	11	115.2165
12	143.4563	12	149.6801	12	119.9864	12	112.0955
13	130.2587	13	119.5508	13	100.3675	13	120.1442
14	145.7061	14	141.9528	14	123.0998	14	108.2297
15	147.7783	15	148.9587	15	124.9960	15	106.4281

Figure 5.6 MAL of each category for grading

There is challenge during extracting minor axis length of sample image, it is because of grain setup during capturing the image. Example one grain can be viewed from four different angles while capturing. These differences make MAL size that should be found in one category to be appeared in other category and it influenced classification accuracy during evaluation. To overcome this problem the mean value of MAL is used to gain representative for each category as show in Table 5.2.

Table 5.2 Mean value of MAL for all categories

MAL mean value			
2.8mm	2.5mm	2.2mm	Below 2.2mm
147	136	124	113

These mean value of MAL used for classifying grade level. In addition to MAL other features are used for class determination such as major axis length, area, roundness, aspect ratio, elongation, perimeter and compactness for shape. Homogeneity, coloration and contrast for texture and 6 features of color used to classify sample image into six classes. Those classes mentioned in Table 4.1 as objects found in malt barley image. Over all graphical user interface of the proposed system for the prototype looks like Figure 5.7.

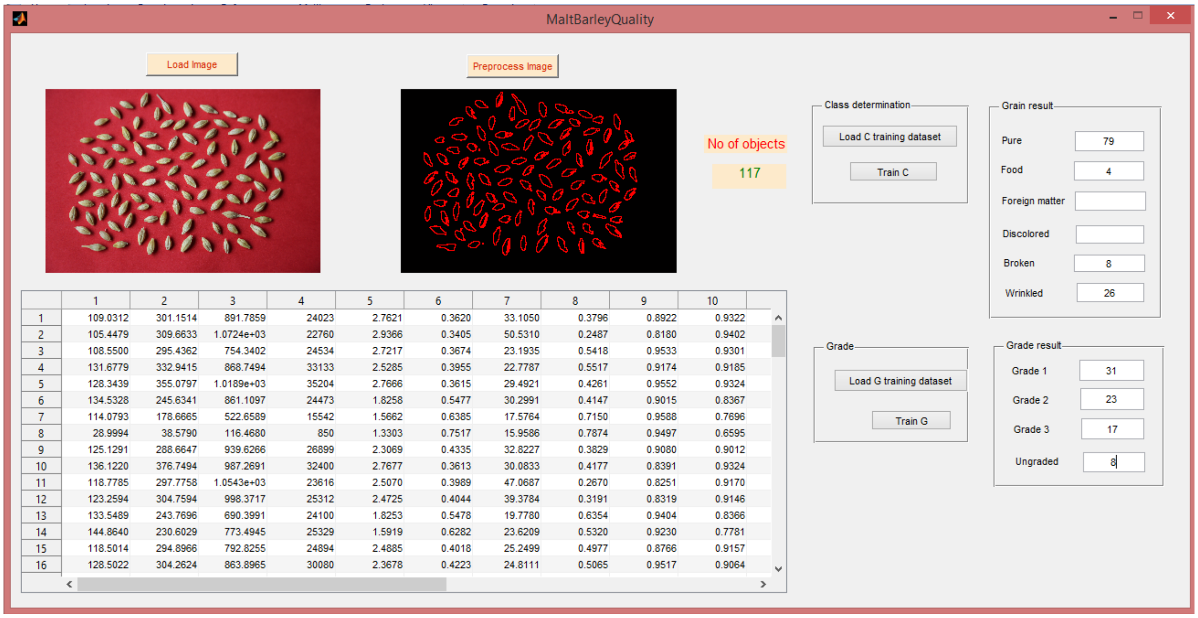


Figure 5.7 Graphical user interface

5.4 Test result

Tests are conducted on Artificial Neural Network, Support Vector Machine and K Nearest Neighbors classifiers to determine the best performing classifier based on the criterion of classification accuracy. These three classifiers performance check with grading and class determination.

5.4.1 Artificial Neural Network test results

This classifier trained with 70% of the data. 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.

A. Object class classification result

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 6 classes for class determination the corresponding number of bits in the binary code is also set to six. These target classes with respect to binary code is show in Table 5.3.

Table 5.3 Binary code for class determination

Target class description	Binary code
Pure malt barley	000001
Food barley	000010
Foreign matter	000100
Discolored	001000
Broken grains	010000
Wrinkled grains	100000

After partitioning the target class using binary code the neural network is trained. During training, cross-entropy was used as the error function. The neural network training process is halted at the 67th iteration (epoch) at which the validation error started to rise and the training error was dropping. This training process is shown Figure 5.8.

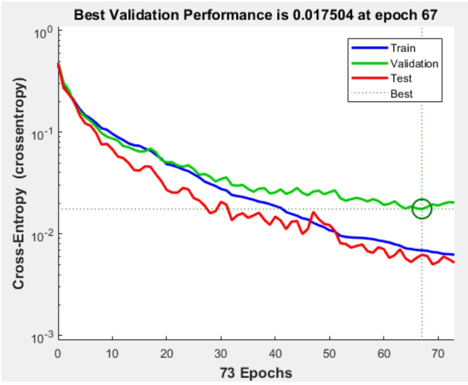


Figure 5.8 Performance of ANN for class determination

Accordingly, classification accuracies of 99.5%, 93.5%, and 100% have been achieved for training, validation and testing respectively. Moreover, an overall classification accuracy of 98.7% is achieved. The confusion matrix showing the overall classification results (including training, validation and testing) is shown in Table 5.4.

Table 5.4 Confusion matrix of object class

Output Class	Pure malt barley	56	0	0	0	0	0	100%
	Food barley	2	106	1	0	0	0	97.2%
	Foreign matter	0	0	33	0	0	0	100%
	Discolored	0	0	0	0	0	0	0.0%
	Broken grains	0	0	0	0	5	0	100%
	Wrinkled grains	0	0	3	0	1	104	99.0%
		96.6%	100%	100%	0.0%	83.3%	100%	98.7%
		Pure malt barley	Food barley	Foreign matter	Discolored	Broken grains	Wrinkled grains	
	Target Class							

The view of pattern recognition neural network for object class determination is show in Figure 5.9. This image illustrate that the input layer has 17 input neurons that correspond to each features extracted before, the hidden layer has 12 neurons which is empirically found that let the accuracy percentage higher than other numbers of neurons for classification. The output layer has 6 neurons that correspond to each object types that may found in malt barley image sample, but the actual output status show that there are only 5 outputs this is because discolored barley sample is not found during the work.

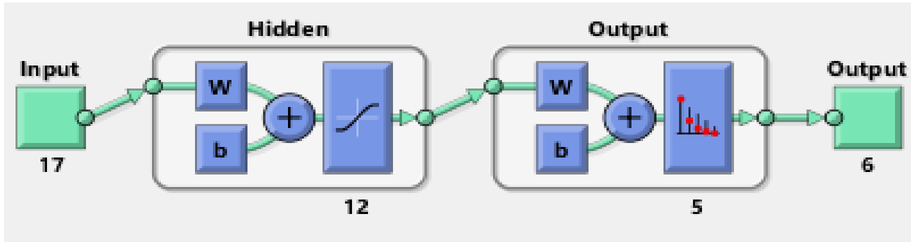


Figure 5.9 View of pattern recognition neural network for class

B. Grade level category result

The labeling holds true for grade level category as show in Table 5.5.

Table 5.5 Binary code for grade level

Target grade description	Binary code
Grade 1	0001
Grade 2	0010
Grade 3	0100
Under grade	1000

Performance of training is show in Figure 5.10 and the grading over all classification accuracies is 99.4%. The confusion matrix showing the overall classification results (including training, validation and testing) is shown in Figure 5.11 and the rewritten all confusion matrix of Figure 5.11 is show in Table 5.6 in simple form.

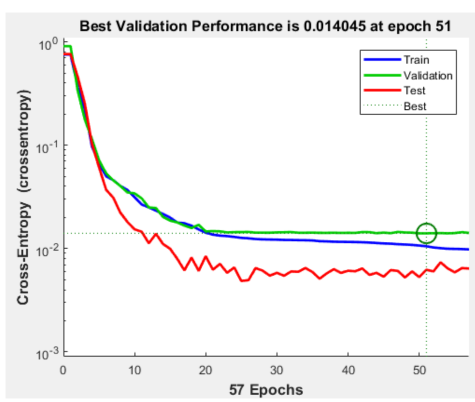


Figure 5.10 Performance of ANN for grade category

Figure 5.11 Confusion matrix of grade level

The indicator numbers used for output class and target class like 1, 2, 3, and 4 shows Grade 1, Grade 2, Grade 3, and Under grade respectively.

Table 5.6 Rewritten all confusion matrix of Figure 5.11 for simplicity

Output Class	Grade 1	42	0	0	0	100%
	Grade 2	0	36	0	0	100%
	Grade 3	1	0	37	0	97.4%
	Under grade	0	0	0	44	100%
		97.7%	100%	100%	100%	99.4%
		Grade 1	Grade 2	Grade 3	Under grade	
	Target Class					

The view of pattern recognition neural network for grade level categorization is show in Figure 5.12. This image illustrate that the input layer has 1 input neurons that correspond to size feature used for grading, the hidden layer has 25 neurons which is empirically found that let the accuracy percentage higher than other numbers of neurons for grade level categorization. The output layer has 4 neurons that correspond to each grade level.

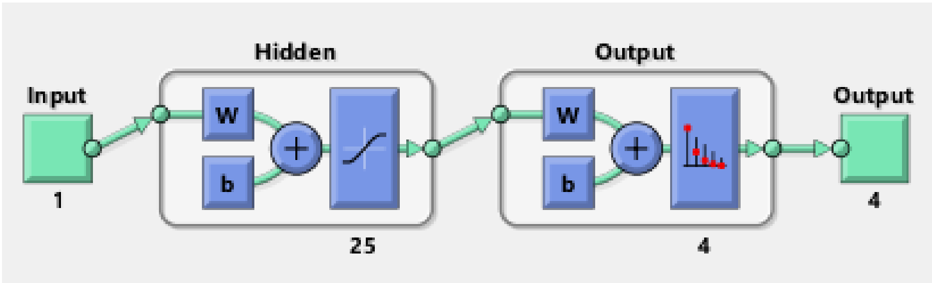


Figure 5.12 view of pattern recognition neural network for grade

5.4.2 Support Vector Machine test results

Similarly this classifier use 70% of the data for training and the rest 30% for testing. Under this classifier the number of observation is illustrated using Figure 5.13 for object class and Figure 5.14 for grade level.

A. Object class classification result

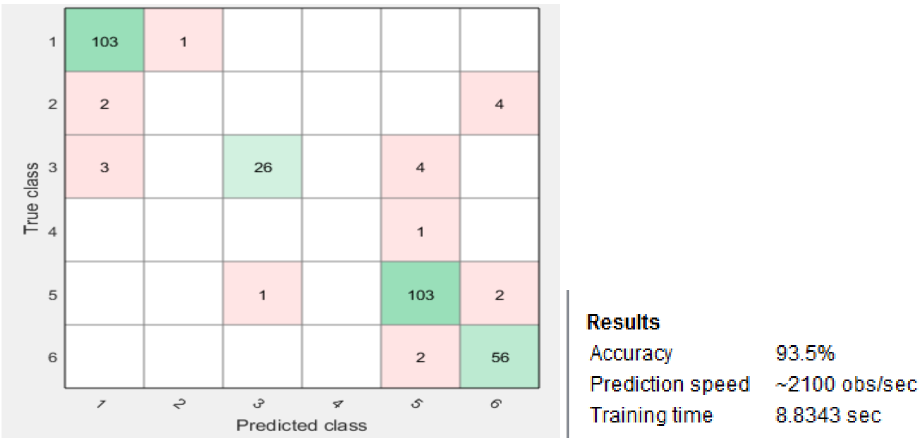


Figure 5.13 Confusion matrix of object class

The indicator numbers used for true class and predicted class like 1, 2, 3, 4, 5, and 6 shows pure malt barley, food barley, foreign matter, discolored, broken grains, and wrinkled grains respectively.

B. **Grade level category result**

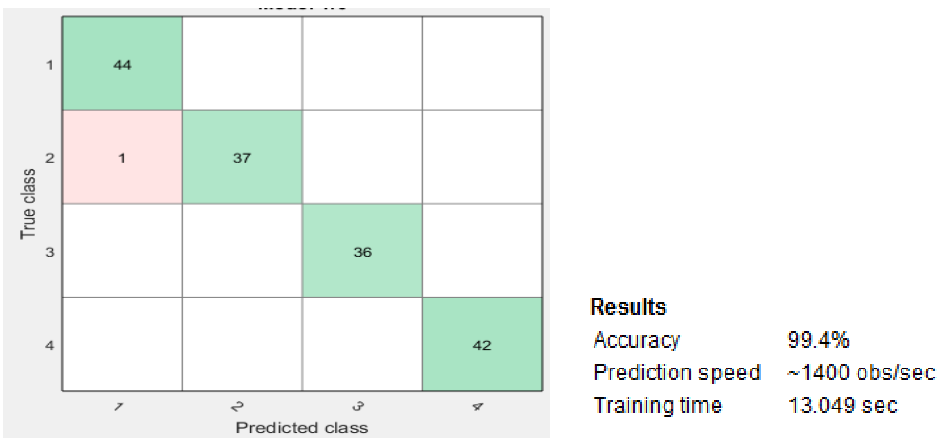


Figure 5.14 Confusion matrix of grade level

The indicator numbers used for true class and predicted class like 1, 2, 3, and 4 shows Grade 1, Grade 2, Grade 3, and Under grade respectively.

5.4.3 K Nearest Neighbors classifier test results

Accordingly this classifier use 70% of the data for training and the rest 30% for testing. Under this classifier the number of observation is illustrated using Figure 5.15 for object class and Figure 5.16 for grade level.

A. **Object class classification result**

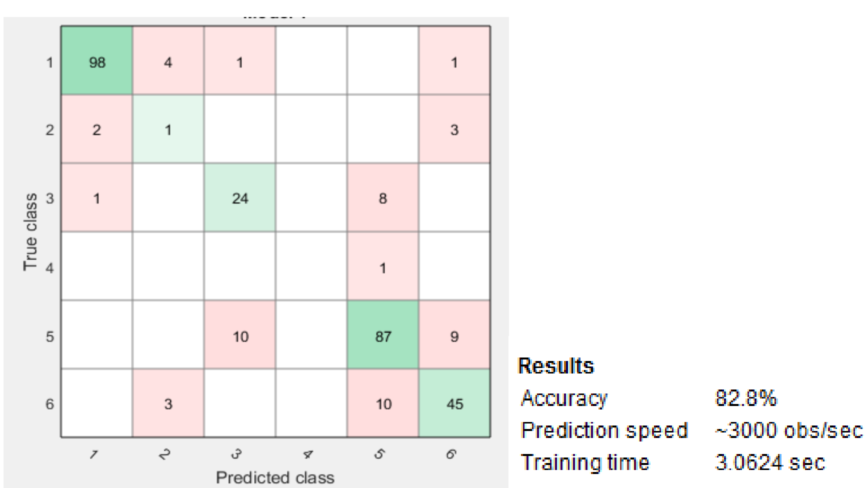


Figure 5.15 Confusion matrix of object class

The indicator numbers used for true class and predicted class like 1, 2, 3, 4, 5, and 6 shows pure malt barley, food barley, foreign matter, discolored, broken grains, and wrinkled grains respectively.

B. **Grade level category result**

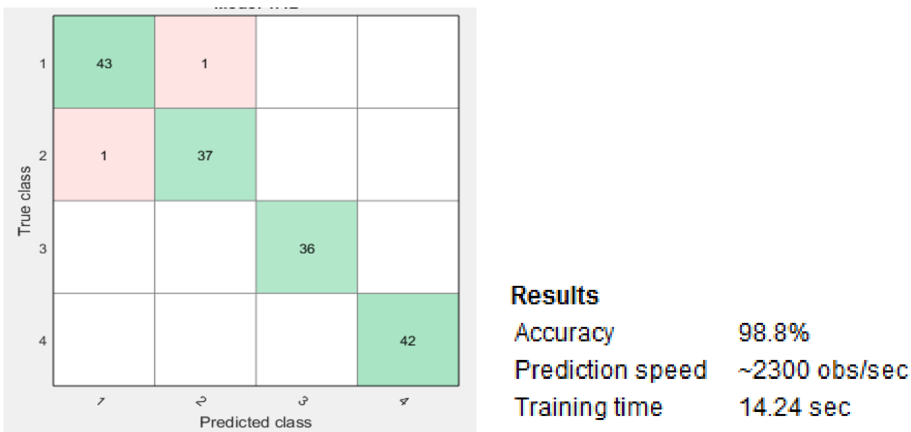


Figure 5.16 Confusion matrix of grade level

The indicator numbers used for true class and predicted class like 1, 2, 3, and 4 shows Grade 1, Grade 2, Grade 3, and Under grade respectively.

The performance of each classifier in this study evaluated based on the counts of test records which are correctly and incorrectly predicted by the classifiers. The performance of each individual classifier evaluated using statistical measures such as accuracy and error rate. Correctly and incorrectly classified counts of the each classifier are tabulated in a table known as a confusion matrix. The statistical measures which are used to evaluate the performance of the model are formulated as follows

$$Accuracy(recognition\,rate) = \frac{Number\ of\ correctly\,predicted}{Total\ number\ of\ samples} = \frac{TP + TN}{N + P} \quad (18)$$

$$Error\,rate\ (misclassification\,rate) = \frac{Number\ of\ wrongly\,predicted}{Total\ number\ of\ samples} = \frac{FP + FN}{N + P} \quad (19)$$

Where:

P: positives which refer to the total number of positive tuples.

N: negatives which refer the total number of negative tuples.

TP: True positives which refer positive tuples that were correctly labeled by classifier.

TN: True negatives which refer negative tuples that were correctly labeled by classifier.

FP: False positives which refer the negative tuples that were mislabeled as positive.

FN: False negatives which refer the positive tuples that were mislabeled as negative.

Accordingly depending on the result of confusion matrix of each classifier for class determination and grade level categorization the ANN classifier performs better than the rest two classifiers and selected as a classifying technique for this study. The comparison between the classifiers is show in Table 5.7 and plotted in Figure 5.17 for visualization.

Table 5.7 Classifier performance

Classifier name	Class determination performance	Grade categorization performance	Overall performance
ANN	98.7%	99.4%	99.05%
SVM	93.5%	99.4%	96.5%
KNN	82.8%	98.8%	90.8%

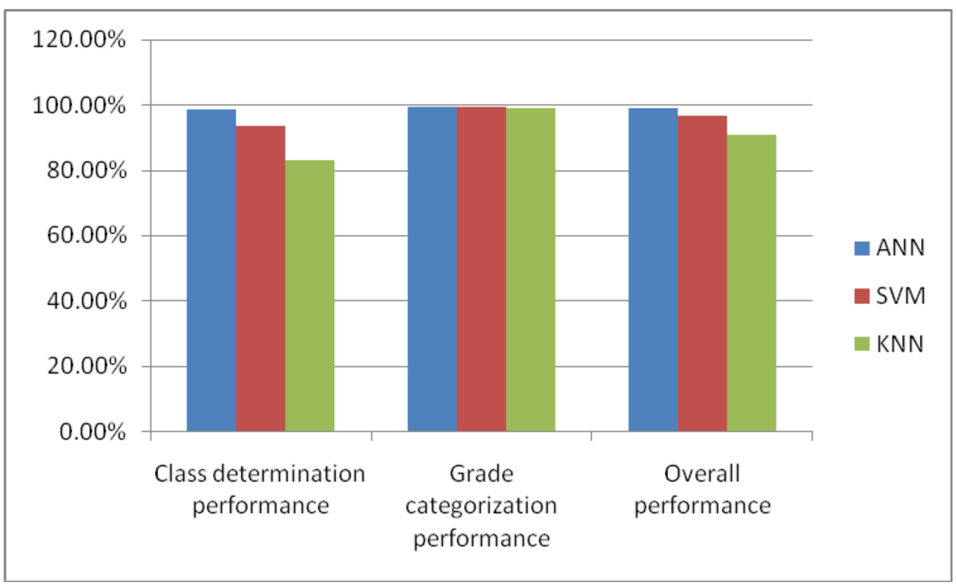


Figure 5.17 Classifier performance

5.5 Discussion

For the purpose of grading one variety of malt barley is selected and used. This is because the sieve size illustrated in Figure 4.13 works for all without change in sieve. Accordingly the selected malt barley variety MAL is extracted and then used as benchmark for grading other malt barley variety.

Grade level categorization only grade malt barely grains having size of 2.8mm, 2.5mm, 2.2mm, and below 2.2mm those identified in sample image, but grade level categorization does not acquired capability of separating other objects found in the sample image. It considers other objects in the sample image to one of the size it fit too. Hence, during implementation the classification of class determination done first. This is because of the fact that grade level classification does not identify other objects, so class of objects is separated as per Table 5.3 and then only those that identified as pure malt barley are used for grading.

For preprocessing purpose black background is used since it holds true as it is opposite to mean RGB value of malt barley. This helps for extracting foreground objects easily, but this does not hold true for capturing images of foreign matter, since some foreign matter color relatively similar to the background color. We then used other color for capturing foreign matter as show in Figure 5.18.

Figure 5.18 Foreign matter

Apart from background color selection, the most challenges of this work are the lack of proper laboratory settings for image acquisition. In addition to this, the image acquisition environment and other imaging factors may affect the result.

While segmenting if there is overlapping or touched grains the overlapped or touched grains is counted as one as show in Figure 5.19.

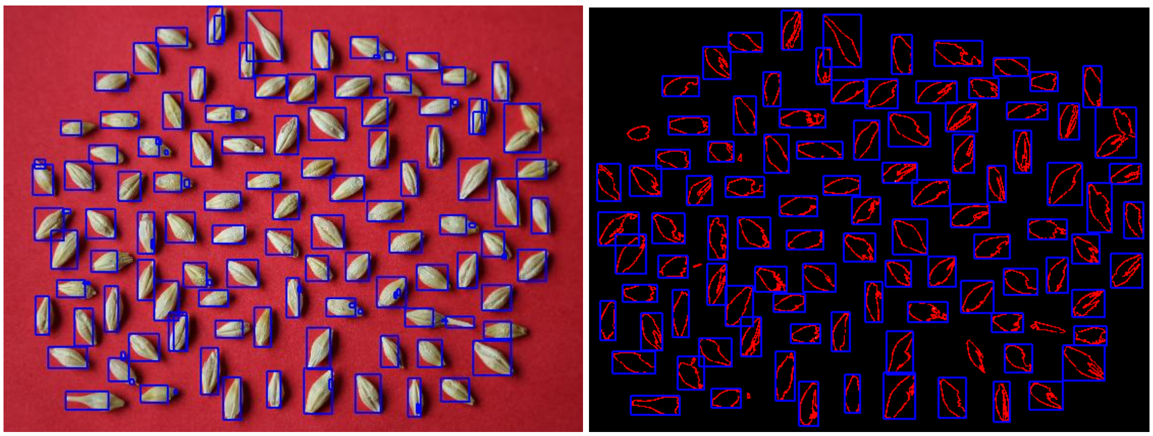


Figure 5.19 Error in segmentation

This error in segmentation is solved by contour estimation using ellipse fitting method as show in Figure 4.12, but this does not hold true if the sample is too much dense.

During classifier training for class determination the accuracy of each class is identified. Example for pure malt barley 100%, food barley 97.2%, foreign matter 100%, broken 100%, and wrinkled grains 100%, but the loss of discolored grains is 100% this is because the sample image does not contain sample of discolored grain. Most of the time malt barley does not change its color because it has hard inedible cover as discussed in Section 1.1.

Selecting the best number of neurons for hidden layer is difficult since there is no appropriate reference to choose number of neurons depending on theory. There are some empirically derived rules of thumb (a broadly accurate guide or principle, based on experience or practice rather than theory.) of selecting neurons number. The most commonly relied on is the optimal size neuron of the hidden layer is usually between the size of the input and size of the output layers. Other say the number of neurons in hidden layer is the mean of the neurons in the input and output layers. Despite the fact that this seems true for object class classification which has 12 neurons that is between input and output layer neurons, but it does not hold true since for grade level categorization 25 neurons used.

To know the discriminative power of extracted feature we train the neural network by including and excluding some features. By using only size, shape, color, and texture features the accuracy percentage of 63.2, 88.9, 92.3, and 72.6 is gained respectively. The network also trained with absence of size feature, this means using shape, color, and texture features which register 96.6% of accuracy. With absence of size and shape features or by including only color and texture accuracy of 65.8% is gained. All features together registered accuracy of 98.7. Accordingly, we can conclude that color features have the most discriminative power and shape feature has the second most discriminative power whereas texture feature has third most discriminative power. However, we found out that size features have the least discriminating power for object class determination of malt barley sample image.

The comparison of discriminative power of size, shape, color, and texture features is presented in Figure 5.20 as column chart. In this chart, the discriminative power of size, shape, color, and texture features is compared by using the observed difference in the classification accuracy during the experiment.

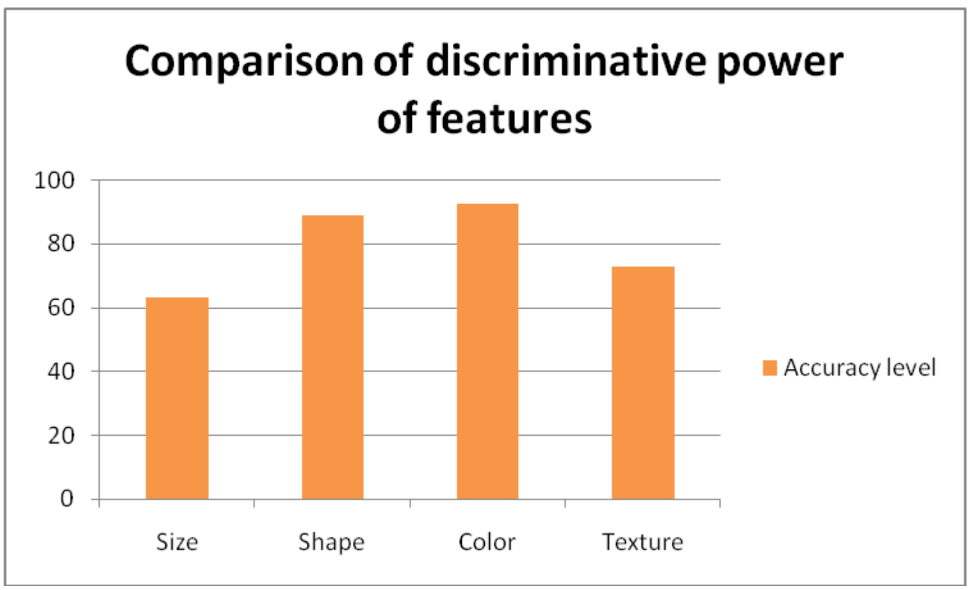


Figure 5.20 comparison of discriminative power of size, shape, color, and texture features

The ANN average epoch performance for object class determination is 34.5 and 21.6 for grade level categorization. Average time for grade level categorization is 5.44012 seconds whereas 6.78575 seconds for object class determination.

The work of C. Ferreira and others says the final absolute percentage differences did not surpass 5% in any case, this mean that in the work they achieved around 95% of accuracy, but in our case we achieve more registering 99.4% for grade level classification.

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 Assela Malt Factory. The expert has taken 7 minutes to identify and count food barley, foreign matter, broken grains, wrinkled, discolored and pure malt barley grain from a mixture. However our proposed system completed the job within 41 seconds.

5.6 Summary

In this chapter, we have proved that our proposed edge segmentation using three edge detection method that help for estimating contour of overlapping area of malt barley algorithm and the models we used to represent features of malt barley sample fulfilled their intended purposes. This is shown by high level of classification accuracy we achieved both in class determination and grade level categorization. Finally, the best classifier for the study is select based on accuracy performances of three classifiers.

Chapter Six: Conclusion, Recommendation and Future work

6.1 Conclusion

Beer has a strong bond with human society. This fermented beverage was most likely created by accident thousands of years ago. Despite the massive technological growth that separates ancient brewing from today's high technology breweries, the process in its traditional version remains entirely unchanged. Ethiopian per capita beer consumption is about 4.0 liters which is small compared to other African countries, but this has started to be change over the last decade as Ethiopian economy has begun to grow. Now, Ethiopia has experienced one of the fastest increases of beer consumption in the recent years. The industry has responded to growing demand by expanding their scale of operation. The government invited two of the world's largest breweries Heineken and Diageo to set up operation. There is now growing evidence that, with an increase in income, households are switching from domestically brewed beverages like 'Tella' and 'Areke' to bottled beer. Since traditional beer is sorghum and other grain based and the bottled beers are barley based, this has accelerated the demand for malt barley

Ethiopia is the second largest barley producer in Africa next to Morocco. Barley is cultivated by smallholder farmers in every region since it is able to grow at all elevations, but it performs best at the higher elevations in the northern and central regions of the country. There are two varieties of barley produced, food barley and malt barley. In beer production malt barley is used as major raw material (about 90% of the total raw material cost). Homogeneity, a measure of grain uniformity, is important for malting and brewing performance and is of increasing interest to maltsters who want to produce homogeneous malt.

Now a day's homogeneity is influenced by so many reasons which may result in low quality beer production. The main factor influencing the homogeneity of malt was the quality of the malt barley and the way it was treated in the field. Malt barley production in Ethiopia is thus based on populations, a mixture of landraces differ in productivity, phenotypic, plasticity, seed color, maturity, height, disease reaction and other traits, also it cultivated under extreme marginal conditions of drought, frost and poor soil fertility. As a result, the so called measures of grain uniformity know as homogeneity is lost. As consequence, uneven start of germination due to embryo difference, kernel size and shape may influence both water uptake and distribution, enzyme level and distribution, protein level difference in the grain, endosperm structure and cell wall thickness have influence during malting.

Currently, there is no automated technique that can assess malt barley quality to keep homogeneity by using specification criteria of Ethiopian Standards Agency. C. Ferreira and others developed image analysis methods to evaluate barley malt grain size which is a good work, but it is not accurate since it misses some criteria to evaluate quality as per Ethiopian Standards Agency. Hence, malt barley grain quality assessment is still performed manually. This 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. In order to eliminate these shortcomings of manual quality assessment, it is important to employ automated quality assessment system. Automated malt barley quality assessment has many important advantages over the manual technique. It 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 that exists in the manual system.

Therefore, in this research work, automatic malt barley quality assessment system is developed to determine class of objects found in the image and give grade level category for malt barley sample. For this, a new combination method used for gaining sharp and clear edge by combining three existing edge detection method and introduced overlapped malt barley segmentation algorithm using contour estimation method. A total of 17 features are identified to model the constituents of malt barley sample.

For classification purpose, three classifiers are used and compared with each other depending on their performance and finally artificial neural network is selected for permanent use in the research work. Results show that the overall success rate for the classification of malt barley grade level is 99.4%, 99.4%, and 98.8% for ANN, SVM, and KNN respectively and success rate for class determination is 98.7%, 93.5%, and 82.8% for ANN, SVM, and KNN respectively. Overall success rate of classification by the classifier is 99.05%, 96.5%, and 90.8% for ANN, SVM, and KNN respectively.

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 Assela Malt Factory. The expert has taken 7 minutes to identify and count food barley, foreign matter, broken grains, wrinkled, discolored and pure malt barley grain from a mixture. However our proposed system completed the job within 41 seconds.

Moreover, these results show that the proposed edge segmentation and overlapped malt barley contour estimation for segmentation algorithm and system architecture are effective in assessing the quality of malt barley sample to determine class of objects found in the image and in giving grade level category according to the specification of Ethiopian Standards Agency. Hence, it is feasible to assess the quality of malt barley sample using digital image processing and Artificial Neural Network. 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 malt barely grain quality assessment.

- ✓ The touching and overlapping of grains influence the segmentation and feature extraction stage. We introduced overlapping malt barley segmentation method using contour estimation and by applying prior knowledge of malt barley that help for implementing it with ellipse fitting.
- ✓ We identified new method to detect edge of malt barley with sharp and clear edge by combining existing edge detection method. The method could potentially be extended for other grains as well.
- ✓ We identified a total of 17 features that are used to successfully classify objects found in malt barley sample image and give grade level category for malt barley grains.
- ✓ We come up with a classifier (using Artificial Neural Network) that can successfully determine the class of objects found in malt barley sample image and give grade level category for malt barley grains.

- ✓ We proposed system architecture for the automatic quality assessment of malt barley for brewing industry.

6.3 Future work

Though this study has been able to assess the quality of malt barley successfully, few works still remain unsolved. The following are the possible future works, also can be considered as recommendation for those who engage themselves in image processing area.

- ✓ An acquisition of sample image is tedious work since there is no pre facilitated equipment to acquire image. It is good if material like box covering all direction with little space for camera view to capture the objects on the surface is advisable. In the future we construct such equipment to minimize light variations and surface reflectance problem.
- ✓ The proposed overlapped malt barley image segmentation use prior knowledge about the object which is ellipse structure of malt barley. This segmentation for estimating contour of overlapped and touched grain works well, but failed if dense malt barley sample introduced, this is one potential work in the future.

References

[1] E. Pires and T. Brányik, *Biochemistry of Beer Fermentation*, Springer International Publishing, New York, 2015.

[2] Food and Agriculture Organization of United Nation, "Food Balance Sheets", 2011, retrieved from http://faostat3.fao.org/download/FB/FBS/E, last accessed on November 28, 2016.

[3] Food and Agriculture Organization of United Nation, "Food Balance Sheets", 2014, retrieved from http://faostat3.fao.org/download/FB/FBS/E, last accessed on December 2, 2016.

[4] Shahidur Rashid, Gashaw T. Abate, Solomon Lemma, James Warner, Leulsegged Kasa, and Nicholas Minot, "The Barley Value Chain in Ethiopia", *International Food Policy Research Institute (IFPRI),* Washington DC., 2015.

[5] Vavilov N.R., *The origin, variation, immunity and breeding of cultivated plants Chronica Botanica*, Interscience publisher, New York, 1951.

[6] Qualset C. O., "Sampling germplasm in a center of diversity: an example of disease resistance in Ethiopian barley", Cambridge University Press, 1975.

[7] Bonman, J.M., Bockelman, H.E., Jackson, L.F. and Steffenson, B.J., "Disease and Insect Resistance in Cultivated Barley Accession from the USDA National Small Grains Collection", 2005, retrieved from https://naldc.nal.usda.gov/download/3292/PDF, last accessed on November 25, 2016.

[8] Tura Kaso and Gashaw Guben, "Review of Barley Value Chain Management in Ethiopia", *Journal of Biology, Agriculture and Healthcare*, Vol. 5, No. 10, 2015, pp. 84-95.

[9] Fetien Abay, Asmund Bjørnstad and Melinda Smale, " Measuring on Farm Diversity and Determinants of Barley Diversity in Tigray: Northern Ethiopia", *International Food Policy Research Institute*, Vol. 1,No. 2, 2009, pp. 44-66.

[10] Firdissa Eticha, Woldeyesus Sinebo and Heinrich Grausgruber, "On-farm Diversity and Characterization of Barley (Hordeum vulgare L.) Landraces in the Highlands of West Shewa, Ethiopia", *Ethnobotany Journal* , 2010, pp. 26-34.

[11] Bedada Begna, Mesay Yami and Eshetu Lemma, "Characterization of a barley-based faraming system in the Arsi highlands, Ethiopia", *Journal of Agricultural Extension and Rural Development*, Vol. 8, No. 10, 2014, pp. 309-320.

[12] The state of food insecurity in the world, "*Eradicating world hunger taking stock ten years after the World Food Summit*", 2006, retreived from ftp://ftp.fao.org/docrep/fao/009/a0750e/a0750e00.pdf, last accessed November 29, 2016.

[13] Daniel Lapedes, *Encyclopedia of Food, Agriculture & Nutrition*, McGrawHill, New York, 1977.

[14] Wunderlich S., "Overview of manufacturing beer: ingredients, processes, and quality criteria.", *Beer in health and disease prevention*, London, 2009, pp. 3–16.

[15] Gina Egbert, "Barley", *in College Seminar 235 Food for Thought: The Science, Culture, & Politics of Food*, Hamilton college, Spring 2008.

[16] R Muller, "Barley quality and grain size homogeneity for malting", *Brewing Research International, Project report No. 320*, Lyttel Hall, Coopers Hill Road, Redhill, Surrey RY1 4HY Vol.2, 2003.

[17] Tadesse Kasshun, "Malting barley marketing and malt production from barley in Ethiopia", *Barley research and development in Ethiopia, In Proceedings of the 2nd National Barley Research and Development Review Workshop*, 2011.

[18] New malt factory to begin production, retrieved from https://newbusinessethiopia.com/ethiopia-to-end-malt-import-boosting-local-production, last accessed October 24, 2017.

[19] Kemelew Muhe and Alemayehu Assefa, " Diversity and Agronomic Potential of Barley (Hordeum vulgare L.) Landraces in Variable Production System, Ethiopia", *World Journal of Agricultural Sciences*, Vol.7, No.5, 2011, pp.599-603.

[20] Russ C.R. *The Image Processing Handbook*, CRC Press, Boca Raton, 1995.

[21] Eugénio C.Ferreira, António L.Amaral, Orlando Rocha, Cristina Gonçalves and António Augusto Ferreira, " Development of Image Analysis Methods to Evaluate Barley / Malt Grain Size", *Journal of industrial crops and products*, 2009, pp. 366-371.

[22] Von Bothmer R and Komatsuda T., "Barley Origin and Related Species", Chichester, UK: WileyBlackwell, 2011, pp.14–62.

[23] Piperno DR, Weiss E, Holst I, Nadel D., "Processing of wild cereal grains in the Upper Palaeolithic revealed by starch grain analysis", *Nature 430*, 2004, pp. 670–673.

[24] Evers T and Millar S., "Cereal grain structure and development: Some implications for quality", *Journal of Cereal Science*, No.36, 2002, pp. 261–284.

[25] Stefanie Jacomet and collaborators Archaeobotany Lab IPAS, Basel University, "*Identification of cereal remains from archaeological sites*", 2nd edition 2006.

[26] Mark S. Nixon and Alberto S. Aguado, *Feature Extraction and Image Processing*, 2nd edition, 2008.

[27] Ballard DA. and Brown CM., *Computer vision*, Eaglewoood Cliffs, Prentice-Hall, 1982.

[28] Timmermans AJM., "Computer vision system for on-line sorting of pot plants based on learning techniques", Acta Hort. 1998, pp. 91–98.

[29] Baxes GA., *Digital image processing principles and applications*, New York, Wiley, 1994.

[30] Minakshi Kumar, "Digital Image Processing", *Satellite Remote Sensing and GIS Applications in Agricultural Meteorology*, pp. 81-102

[31] Ian T. Young, Jan J. Gerbrands and Lucas J. van Vliet, *Fundamentals of Image Processing*, 2007.

[32] R. Gonzalez and R. Woods, *Digital Image Processing*, 3rd edition, AddisonWesley, 2008.

[33] K.N. Plataniotis and A.N. Venetsanopoulos, *Color Image Processing and Applications*, 2000.

[34] Amanpreet Kaur, "A Review Paper on Image Segmentation and its Various Techniques in Image Processing", *International Journal of Science and Research*, Volume 3, 2014, pp. 12-14.

[35] R.Yogamangalam and B.Karthikeyan, "Segmentation Techniques Comparison in Image Processing", *International Journal of Engineering and Technology*, Vol. 5, No. 1, 2013 pp. 307-313.

[36] Sahar Zafari, "Segmentation of partially overlapping convex objects using branch and bound algorithm", *The 13th Asian Conference on Computer Vision (ACCV' 16)*, Nov. 2016.

[37] Heba Ahmed Elnem, Nourhan Mohamed Zayed and Mahmoud Abdelmoneim Fakhreldein, *Emerging Perspectives in Intelligent Pattern Recognition, Analysis, and Image Processin,* 2016.

[38] Pooja Kamavisdar , Sonam Saluja and Sonu Agrawal, "A Survey on Image Classification Approaches and Techniques", *International Journal of Advanced Research in Computer and Communication Engineering*, Vol. 2, 2013, pp. 1005-1009.

[39] S. Arun Balaj and K. Baskaran, "Design And Development Of Artificial Neural Networking (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.

[40] Ms. Sonali and B. Maind, "Basic of Artificial Neural Network", *International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC),* Vol. 2, Issue. 1, 2014, pp. 96-100

[41] Zayas I., Pomeranz Y., and F. S. Lai, "Discriminate between wheat and non-wheat components in a grain sample", *Cereal Chemistry*, vol. 66, no.3, 1989.

[42] Lai FS, Zayas I, Pomeranz Y., "Application of pattern recognition techniques in the analysis of cereal grains", *Cereal Chemistry*. Vol. 63, No. 2, 1986, pp. 168–174.

[43] N. S. Visenl, J. Paliwall, D. S. Jayas1, and N. D. G.White, "Image analysis of bulk grain samples using neural networks", *Can. BioSyst. Eng*, vol. 46, 2004.

[44] B. S. Anami, D. G. Savakar, Aziz Makandar, and P.H. Unki, " A neural network model for classification of bulk grain samples based on color and texture", *Proceedings of International Conference on Cognition and Recognition*, 2005.

[45] H.K. Mebatsion, J. Paliwal, D.S. Jayas, "Automatic classification of non-touching cereal grains in digital images using limited morphological and color features", *Computers and Electronics in Agriculture*, 2013.

[46] Daniel Hailemichael Lemessa, "Development of Automatic Maize Quality Assessment System Using Image Processing Techniques", Unpublished Masters Thesis, Department of Computer Science, Addis Ababa University, 2015.

[47] Manish Chhabra and Parminder Singh Reel, " Morphology Based Feature Extraction and Recognition for Enhanced Wheat Quality Evaluation", *Communications in Computer and Information Science (168)*, 2011, pp. 41–50

[48] Miroljub Ivanov Mladenov, Stanislav Miroslavov Penchev, Martin Plamenov Dejanov, Metin Sebahatin Mustafa, " Quality assessment of corn grain sample using color image analysis", *Sensing and Instrumentation for Food Quality and Safety*, Vol. 1, No. 1, 2011.

Annex A

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

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

% Last Modified by GUIDE v2.5 29-Oct-2017 07:21:30

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @MaltBarleyQuality_OpeningFcn, ...
                   'gui_OutputFcn',  @MaltBarleyQuality_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

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

% Choose default command line output for MaltBarleyQuality
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

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


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

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


% --- Executes on button press in Loadimage.
function Loadimage_Callback(hObject, eventdata, handles)
% hObject    handle to Loadimage (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
handles.output=hObject;
[a b]=uigetfile({'*.*'});
img=imread([b a]);
handles.rgbimg=img;
imshow(img,'Parent',handles.axes1);
guidata(hObject,handles);


% --- Executes on button press in Preprocessimage.
function Preprocessimage_Callback(hObject, eventdata, handles)
% hObject    handle to Preprocessimage (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
handles.output=hObject;

tic
```

```
I = handles.rgbimg;
imgGray=rgb2gray(I);
%imgGray=imclearborder(imgGr);
gra = medfilt2(imgGray,[11 11]);
L=graythresh(gra);%find appropriate gray thresh value
BW1=im2bw(gra,L);%convert to binary
bw2=imfill(BW1,'holes');
filterim=bwareaopen(bw2,350);%opens area greater than 350
[~,thresh]=edge(filterim,'sobel');
b=edge(filterim,'canny',thresh);
bi=im2double(b);
bwf= edge(bi,'roberts');
%figure, imshow(bw), title(' s c r');
bwcon=bwconncomp(filterim,8);% Find connected components in binary image
num=bwcon.NumObjects;
set(handles.text2,'string',num);
s = regionprops(bwf,{...
    'Centroid',...
    'Area',...
    'MajorAxisLength',...
    'MinorAxisLength',...
    'Orientation'});
% figure;
% imshow(bwf);
imshow(bwf,'Parent',handles.axes2);
t = linspace(0,2*pi,70);
hold on
for k = 1:length(s)
    a = s(k).MajorAxisLength/2;
    b = s(k).MinorAxisLength/2;
    Xc = s(k).Centroid(1);
    Yc = s(k).Centroid(2);
    phi = deg2rad(-s(k).Orientation);
    x = Xc + a*cos(t)*cos(phi) - b*sin(t)*sin(phi);
    y = Yc + a*cos(t)*sin(phi) + b*sin(t)*cos(phi);
    plot(x,y,'r','Linewidth',1)
end
 f = getframe(handles.axes2);              %# Capture the current window
imwrite(f.cdata,'demo.png');

hold off
%--------------------------------------------------------------------------
-
segmentationP
im = imread('demo.png');
Im =imresize(im, [470 680]);
cform=makecform('srgb2lab');
J=applycform(Im,cform);
% figure, imshow(J),title('Color form');
K=J(:,:,1);
% figure, imshow(K),title('equalise brightness');
gra=medfilt2(J(:,:,1),[3 3]);
L=graythresh(gra(:,:,1)); %find appropriate gray thresh value
im=im2bw(J(:,:,1),L); %convert to binary
% figure, imshow(im),title('Black and White');
bw2=imfill(im,'holes');
% figure, imshow(bw2),title('hole filled');
```

```
imggry=~im2bw(bw2);
% figure, imshow(imggry),title('change');
param = readparam();
 stats = mia_particles_segmentation(imggry,param);
%________________________________________________________________________
_
MinorAxis= zeros(num,1);
MajorAxis= zeros(num,1);
Area= zeros(num,1);
SolidityB= zeros(num,1);
EccentricityB=zeros(num,1);
Perimeter= zeros(num,1);
Elangation=zeros(num,1);
AspectRatio=zeros(num,1);
Roundness=zeros(num,1);
Compactness=zeros(num,1);
Red=zeros(num,1);
Green=zeros(num,1);
Blue=zeros(num,1);
RGBimg= im2double(I);
Hue= zeros(num,1);
Satur= zeros(num,1);
Inten= zeros(num,1);
Homogeneity=zeros(num,1);
Correlation=zeros(num,1);
Contrast=zeros(num,1);
%waitBar
%hWaitBar = waitbar(0,'Preprocessing');
Mbarley=regionprops(bwcon,'all');
for i=1:num
    Area(i)=Mbarley(i).Area;
    Perimeter(i)=Mbarley(i).Perimeter;
    MinorAxis(i)=Mbarley(i).MinorAxisLength;
    MajorAxis(i)=Mbarley(i).MajorAxisLength;
    Elangation(i)=(MinorAxis(i)/MajorAxis(i));
    AspectRatio(i)=(MajorAxis(i)/MinorAxis(i));
    Roundness(i)=4*pi*Area(i)/((Perimeter(i))^2);
    Compactness(i)=(Perimeter(i))^2/Area(i);
    SolidityB(i)=Mbarley(i).Solidity;
    EccentricityB(i)=Mbarley(i).Eccentricity;
    SBarley  = false(size(RGBimg));
    SBarley(bwcon.PixelIdxList{i}) = true;
    R = SBarley(:, :, 1);
    G = SBarley(:, :, 2);
    B = SBarley(:, :, 3);
    % conversion
    num = 0.5*((R - G) + (R - B));
    den = sqrt((R - G).^2 + (R - B).*(G - B));
    theta = acos(num./(den + eps));
    if B<=G
        H=theta;
    else
        H=360-theta;
    end
    num = min(min(R, G), B);
    den = R + G + B;
    den(den == 0) = eps;
```

```
    S = 1 - 3.* num./den;
    H(S == 0) = 0;
    I = (R + G + B)/3;
    Red(i)= mean2(R);
    Green(i) = mean2(G);
    Blue(i) = mean2(B);
    Hue(i) = mean2(H);
    Satur(i) = mean2(S);
    Inten(i)= mean2(I);
    SMBarleygrain=false(size(imgGray));
    SMBarleygrain(bwcon.PixelIdxList{i}) = true;
    glcm = graycomatrix(SMBarleygrain);
    MBTexture=graycoprops(glcm,{'Homogeneity','Correlation','Contrast' });
    Homogeneity(i)=MBTexture.Homogeneity;
    Correlation(i)=MBTexture.Correlation;
    Contrast(i)=MBTexture.Contrast;
      %waitbar(i/num);
end
Featuredata_texture=[Red Green Blue Hue Satur Inten Homogeneity
Correlation Contrast];
texturee = mapstd(Featuredata_texture); %processes matrices by
transforming the mean and standard deviation of each row to ymean and ystd
% %display(texturee);
%_______________________________________________________________________
_
Extracted_Features=[ MinorAxis MajorAxis Perimeter Area AspectRatio
Elangation Compactness  Roundness SolidityB EccentricityB texturee];
%save tempmorpholoy.mat Extracted_Features
%_______________
xlswrite('barleyfeatures.xls',Extracted_Features,1);
diintable= handles.uitable1;
set(diintable,'Data',Extracted_Features);
%msgbox('Preprocessing Completed','Preprocessing');
%handles.Ex_Features=Extracted_Features;
%delete(hWaitBar);
set(handles.pushbutton3,'Enable','on');
toc
guidata(hObject,handles);



function Grade1_Callback(hObject, eventdata, handles)
% hObject    handle to Grade1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Grade1 as text
%        str2double(get(hObject,'String')) returns contents of Grade1 as a
double


% --- Executes during object creation, after setting all properties.
function Grade1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Grade1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
```

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



function Grade2_Callback(hObject, eventdata, handles)
% hObject    handle to Grade2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Grade2 as text
%        str2double(get(hObject,'String')) returns contents of Grade2 as a
double


% --- Executes during object creation, after setting all properties.
function Grade2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Grade2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

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



function Grade3_Callback(hObject, eventdata, handles)
% hObject    handle to Grade3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of Grade3 as text
%        str2double(get(hObject,'String')) returns contents of Grade3 as a
double


% --- Executes during object creation, after setting all properties.
function Grade3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Grade3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end



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

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


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

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



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

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


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

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

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

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


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

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



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

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


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

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

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

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


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

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



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

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


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

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



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

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


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

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



% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton6 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
  handles.output=hObject;
 feat = handles.G_Features;
 [Gradee x]= size(feat);
 %disp(features);
 Grade1=0;
 Grade2=0;
 Grade3 =0;
 Undergrade = 0;
for i=1:Gradee
    if(feat(i,1)==1)
        Grade1 = Grade1 + 1;
    elseif(feat(i,1)==2)
        Grade2 =  Grade2 + 1;
    elseif(feat(i,1)==3)
        Grade3 = Grade3 + 1;
    elseif(feat(i,1)==4)
        Undergrade = Undergrade + 1;
    end
end
set(handles.Grade1, 'String', Grade1);
set(handles.Grade2, 'String', Grade2);
set(handles.Grade3, 'String', Grade3);
set(handles.Undergraded, 'String', Undergrade);
 guidata(hObject,handles);


% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
```

```matlab
% handles    structure with handles and user data (see GUIDATA)
handles.output=hObject;
Input = xlsread('Book1.csv', 1, 'A1:A160');
Target = xlsread('Book1.csv', 1, 'C1:F160');
input=Input';
target=Target';
x=input;
t=target;
%setdemorandstream(491218382)
net = patternnet(25);
view(net)
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
%Train ANN
[net,tr] = train(net,x,t);
nntraintool
% Test the Network
outputs = net(input);
errors = gsubtract(target,outputs);
performance= perform(net,target,outputs);
disp(outputs);
%plotperform(tr);
figure, plotperform(tr)
figure, plottrainstate(tr)
figure, plotconfusion(target,outputs)
figure, ploterrhist(errors)
save('Grade_network.mat', 'net')
inputs = xlsread('Book1.csv', 1, 'A1:A160');
targets = xlsread('Book1.csv', 1, 'A1:A160');
groupG= xlsread('Book1.csv', 1, 'B1:B160');
% function
Grade=knnclassify(inputs,targets,groupG);
% disp('Result:');
%disp(Grade);
handles.G_Features=Grade;
set(handles.pushbutton6,'Enable','on');
guidata(hObject,handles);



% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
handles.output=hObject;
Input = xlsread('allfeatures.csv', 1, 'A1:Q117');
Target = xlsread('allfeatures.csv', 1, 'S1:X117');
input=Input';
target=Target';
x=input;
t=target;
%setdemorandstream(491218382)
net = patternnet(12);
net.divideParam.trainRatio = 70/100;
```

```
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
view(net)
%train ANN
[net,tr] = train(net,x,t);
nntraintool
% Test the Network
outputs = net(input);
errors = gsubtract(target,outputs);
performance= perform(net,target,outputs);
disp(outputs);
%plotperform(tr);
figure, plotperform(tr)
figure, plottrainstate(tr)
figure, plotconfusion(target,outputs)
figure, ploterrhist(errors)
save('Class_network.mat', 'net')
inputs = xlsread('allfeatures.csv', 1, 'A1:Q117');
targets = xlsread('allfeatures.csv', 1, 'A1:Q117');
groupC= xlsread('allfeatures.csv', 1, 'R1:R117');
% function
class=knnclassify(inputs,targets,groupC);
% disp('Result:');
 %disp(groupC);
handles.Ex_Features=class;
set(handles.pushbutton4,'Enable','on');
guidata(hObject,handles);



% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
handles.output=hObject;
 features = handles.Ex_Features;
 [Class y]= size(features);
 %disp(features);
 puremaltB=0;
 foodB=0;
 Foreign =0;
 Discolored = 0;
 Broken = 0;
 wrinkled =0;
for i=1:Class
    if(features(i,1)==1)
        puremaltB = puremaltB + 1;
    elseif(features(i,1)==2)
        foodB =  foodB+ 1;
    elseif(features(i,1)==3)
        Foreign = Foreign + 1;
    elseif(features(i,1)==4)
        Discolored = Discolored + 1;
    elseif(features(i,1)==5)
        Broken = Broken + 1;
```

```
    elseif(features(i,1)==6)
        wrinkled = wrinkled + 1;
    end
end
set(handles.pure, 'String', puremaltB);
set(handles.Food, 'String', foodB);
set(handles.Foreign, 'String', Foreign);
set(handles.Discolored, 'String', Discolored);
set(handles.Broken, 'String', Broken);
set(handles.Wrinkled, 'String', wrinkled);
set(handles.pushbutton5,'Enable','on');
 guidata(hObject,handles);
```

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: _______________________________________