UNIVERSITY OF NAIROBI

School of Computing and Informatics

CLASSIFICATION OF SELECTED APPLE FRUIT VARIETIES USING NAIVE BAYES

By

RONALD MISIGO

P52/72666/2014

Supervised By

Dr Evans Miriti

A report submitted in partial fulfilment of requirements for the award of degree of Master of Science in Computational Intelligence, School of Computing and Informatics, University of Nairobi

DECLARATION

This report is my original work and has not been presented for a degree or any other award in any other university.

Student Signature: ……………… **Date: ………………………………**

Name: Ronald Misigo **Registration Number: P52/72666/2014**

Supervisor Signature:...................... **Date: ....**

Name: Dr Evans Miriti

DEDICATION

This report is dedicated to my work colleagues: Miss Aline O'Connor, Miss Anastasia Mbatia, Hafsa Tikolo, Miss Jayne Ngugi, Miss Nkatha Ngichu, Mr. Noel Templer, Mr Mike Kibebe and Mr Victor Okonga.

ACKNOWLEDGEMENT

I acknowledge the dedication and guidance of my Supervisor, Dr. Evans Miriti, who steered me on the right path throughout the project. His valuable insights are greatly appreciated. I am also grateful for the guidance and useful comments from the members of the panel: Dr. Tonny Omwansa, Dr. Christopher Chepken, Dr. Stephen Mburu and Prof. William Okello-Odongo. I also appreciate the contributions by the other Faculty members at School of Computing and Informatics (SCI) and especially recognize the efforts by Dr. Christopher Moturi, the Deputy Director, who ensured that we were enlightened on the requirements of the project. The seminars that he organized also helped a great deal. I am also grateful for the many interactions and useful discussions that I had with fellow graduate students of SCI. I am appreciative of the support by students who were with me on the same panel for being open in discussions. Finally, I would like to thank my employer, Agri Experience, for having been accommodative of my absence to attend academic engagements and also allowing me to use their computing facilities. I finally thank my family for having endured my prolonged working hours both at home and away. This endurance is coming to an end. However, this research was made possible through the contributions of many other individuals all of whom I may not be able to name, though this is inadvertent. Feel acknowledged for your help, inputs and insights.

ABSTRACT

Manual sorting of apple fruit varieties results to high cost, subjectivity, tediousness and inconsistency associated with human beings. A means for distinguishing apple varieties is needed and therefore, some reliable technique is needed to discriminate varieties rapidly and non-destructively.

The main objective of this research was to investigate the applicability and performance of Naive Bayes algorithm in classification of apple fruit varieties. The software methodology involved image acquisition, pre-processing and segmentation, analysis and classification of apple varieties. Apple classification system prototype was built using MATLAB R2015 development platform environment.

The results showed that the averaged values of the estimated accuracy, sensitivity, precision and specificity were 91%, 77%, 100% and 80% respectively. Through previous research works, the literature review identified MLP-Neural (Unay et al., 2006), fuzzy logic (Kavdir et al., 2003), principal components analysis (Bin et al., 2007) and neural networks (Ohali et al., 2011) as other technique which have been used previously to classify apple varieties. Benchmarking the performance of Naive Bayes technique against Principal Components Analysis, Fuzzy Logic and MLP-Neural classification technique showed that the Naive Bayes techniques performance was consistent with that of Principal Components Analysis, Fuzzy Logic and MLP-Neural with 91%, 90%, 89%, and 83% respectively in terms of accuracy. This study indicated that Naive Bayes has good potential for identification of apples varieties nondestructively and accurately.

Keywords: Apple fruit, Sorting and Grading of Agricultural products, Image processing techniques, Naive Bayes Technique, Pattern Recognition, Classification.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 Background

In agricultural science, images are an important source of data and information. Dubey and Jalal, 2014 experimented and found out that it is difficult to process or quantify the photographic data mathematically. Digital image analysis and image processing technology circumvent these problems based on the advances in computers and microelectronics associated with traditional photography. Digital image analysis and image processing technology helps to improve images from microscopic to telescopic visual range and offers a scope for their analysis (Dubey and Jalal, 2014).

Several applications of image processing technology have been developed for the agricultural operations. These applications involve implementation of the camera based hardware systems or color scanners for inputting the images. Researchers have attempted to extend image processing and analysis technology to a broad spectrum of problems in the field of agriculture. The computer based image processing is undergoing rapid evolution with ever changing computing systems. The dedicated imaging systems available in the market, where user can press a few keys and get the results, are not very versatile and more importantly, they have a high price tag on them (Dubey and Jalal, 2014).

Recognizing different kind of vegetables and fruits is a recurrent task in the supermarkets, where the cashier must be able to identify not only the species of a particular fruit or vegetable (i.e., banana, apple, pear) but also identify its variety (i.e., Golden Delicious, Jonagold, Fuji), for the determination of its price. This problem has been solved by using barcodes for packaged products but most of the time consumers want to pick their product, which cannot be packaged, so it must be weighted. Assignment of codes for each kind of fruit and vegetable is a common solution to this problem; but this approach has some problems such as the memorization, which may be a reason for errors in pricing. As an aid to the cashier, a small book with pictures and codes is issued in many supermarkets; the problem with this approach is that flipping over the booklet is time-consuming.

1.1.1 Health Benefits of Apples

Apples (*malus sp., Rosaceae*) are one of the most commonly consumed fruits in the world. In 2011, world apple production was estimated at around 75 millions of tons according to Food and Agriculture Organization stats (15 July 2013). Apples are a rich source of nutrient as well as non-nutrient components and contain high levels of polyphenols and other phytochemicals. Main structural classes of apple constituents include hydroxycinnamic acids, dihydrochalcones, flavonols (quercetin glycosides), catechins and oligomeric procyanidins, as well as triterpenoids in apple peel and anthocyanins in red apples. Several lines of evidence suggest that apples and apple products possess a wide range of biological activities which may contribute to health beneficial effects against cardiovascular disease, asthma and pulmonary dysfunction, diabetes, obesity, and cancer (Hyson, 2011).

Apples have been shown to contain several phytochemicals that are thought to be protective in cancer: these include carotenoids, flavonoids, phenolic acids and lignans (Serra et al., 2012) determined the phytochemical content of 100grams of apples and found that flavonoids and phenolic were more abundant in apples retaining their skins. Hospital-based case-control study carried out in Poland showed a significant beneficial effect of apple consumption (daily number of apple servings) on the risk of colorectal cancer. The dietary interviews focused on food-frequency and quantity, and proved that apples were the most frequent fruit consumed in the study population Out of several types of fruits in the study, which included citrus, berries, and stone fruits, apple was the only specific type of fruit associated with a significant reduction (63%) of colorectal cancer risk (Jedrychowsket et al., 2010)

1.1.2 Importance of Sorting and Grading of Agricultural Products

Quality is one of the important factors in marketing of agricultural products. Apple grading is one of the important processes in post-harvest handling and marketing. Handling (Post harvest) process of fruits is completed in several steps: washing, sorting, grading, packing, transporting and storage. The fruits sorting and grading are considered the most important steps of handling. Product quality and quality evaluation are important aspects of fruit and vegetable production. Sorting and grading are major processing tasks associated with the production of fresh-market fruit types. Considerable effort and time have been invested in the area of automation. Most of the current grading machines operate based on machine vision systems to detect blemishes and defects of products, where one image or more are taken for each individual object and the results of processing will decide the quality of the object (Ebrahimi et al., 2011).Grading and sorting of agricultural products using machine vision in conjunction with pattern recognition techniques, offers many advantages over the conventional optical or mechanical sorting devices. Computer vision technique are cost effective, consistent, has superior speed and high accuracy in sorting (Mahendran et al., 2011).

1.1.3 Applications of Image Processing Techniques in Agriculture

Digital image processing, as a computer based technique, has been extremely used by scientists to solve problems in agriculture. Fernando et al (2010) built a system to diagnose six different types of surface defects in citrus fruits using a multivariate image analysis strategy. Images were unfolded and projected onto a reference eigen space to arrive at a score matrix used to compute defective maps and 94.2% accuracy was reported. Cho et al. (2013) used hyperspectral fluorescence imaging for detecting cracking defects on cherry tomatoes while Omid et al. (2013) used shape, texture and color features to sort tomato fruits according to their circularity, size, maturity and defects. They achieved 84.4% accuracy for defect detection using a probabilistic neural network (PNN) classifier.

Rocha et al. (2010) presented a unified approach that can combine many features and classifiers. The authors approached the multi-class classification problem as a set of binary classification problem in such a way that one can assemble together diverse features and classifier 4 approaches custom-tailored to parts of the problem. They have achieved classification accuracy up to 99% for some fruits, but they fused three features, namely Border-interior classification (BIC), Color coherence vector (CCV), and Unser features and used top two responses to achieve them. Their method shows poor results for some type of fruit and vegetable such as Fuji Apple. Arivazhagan et al. (2010) combined the color and texture features to classify the fruits and vegetables. They used minimum distance classifier and achieved 86% accuracy over the dataset having 15 different types of fruits and vegetables. Further, Faria et al. (2012) presented a framework for classifier fusion for the automatic recognition of fruits and vegetables in a supermarket environment. They combined low-cost classifiers trained for specific classes of interest to enhance the recognition rate.

Chowdhury et al. (2013) have recognized 10 different vegetables using color histogram and statistical texture features. They have gained the classification accuracy upto 96.55% using neural network as a classifier. Danti et al. (2012) classified 10 types of leafy vegetables using BPNN classifier with a success rate of 96.40%. They first cropped and resized the image and then extracted the mean and range of hue and saturation channel of HSV image to form the feature vector. Suresha et al. (2012) have reached 95% classification accuracy over a dataset of containing 8 types of different vegetables using texture measures in RGB color space. They have used watershed segmentation to extract the region of interest as a pre-processing and decision tree classifier for training and classification purpose.

1.1.4 Different Technique Used for Classification of Apples

The selection of the most appropriate classifier in classification system, is determined by a number of requirements, the most important of which are: enough classification accuracy, simplest possible classification algorithm, computational efficiency, minimum time of execution (boost), easy implementation and possibility of additional settings, versatility, flexibility (Eleonora et al., 2013).

Cetişli et al. (2013) proposed a new prediction model for the early warning of apple scab based on artificial intelligence and time series prediction. The infection period of apple scab was evaluated as the time series prediction model instead of summation of wetness duration. The important hours of duration were determined with the feature selection methods, such as

Pearson's correlation coefficients (PCC), Fisher's linear discriminant analysis (FLDA) and an adaptive neuro fuzzy classifier with linguistic hedges (ANFC_LH). The experimental dataset with selected features was classified by ANFC_LH, and predicted by an adaptive neural network (ANN) model with 2 to 5% error rates compared to the traditional weather station predictions.

1.2 Statement of the Problem

Apple produce dealers have warehouses that stores different varieties of apple fruits. Therefore, different apple varieties can be easily mixed during harvesting and marketing. Most apple produce dealers will sort the apples manually which results to high cost, subjectivity, tediousness and inconsistency associated with manual sorting.

Recognizing different kind of vegetables and fruits is a recurrent task in the supermarkets, where the cashier must be able to identify not only the species of a particular fruit or vegetable (i.e., banana, apple, pear) but also identify its variety for the determination of its price. This problem has been solved by using barcodes for packaged products but most of the time consumers want to pick their product, which cannot be packaged, so it must be weighted. Assignment of codes for each kind of fruit and vegetable is a common solution to this problem; but this approach has some problems such as the memorization, which may be a reason for errors in pricing. As an aid to the cashier, a small book with pictures and codes is issued in many supermarkets; the problem with this approach is that flipping over the booklet is time-consuming.

1.3 Objective of the Study

1.3.1 General Objective

To investigate the applicability and performance of Naive Bayes algorithm in classification of apple varieties

1.3.2 Specific Objectives

1. To develop an apple classification system that uses Naive Bayes algorithm
2. To test the performance of the system with that of a human apple classifier expert
3. To benchmark the performance of Naive Bayes classifier against the performance of other machine learning classification techniques which have been used to classify apples varieties

1.4 Justification

Because of the ever-growing need to supply high quality food products within a short time, automated grading of agricultural products is getting special priority among many farmer associations and suppliers. The impetus for these trends can be attributed to increased awareness by consumers about their better health well-being and a response by producers on the need to provide quality guaranteed products with consistency. Accurate automatic classification of agricultural products is a necessity for agricultural marketing to increase the speed and minimize the miss-classifications. A means for distinguishing apple varieties is needed by apple sellers. Therefore, some reliable technique is needed to discriminate varieties rapidly and non-destructively.

1.5 Research Scope

In this research, only colour patterns and size morphological features will be extracted from the images.

2. LITERATURE REVIEW

Apples, as a widely grown crop, have been appreciated by consumers because of their nutritional and delicious characteristics (Giovanelli et al. 2014). Apples are an important agricultural commodity in the global market of fresh products. The quality for an apple depends on its external characteristics, such as colour, size, and surface texture, and internal parameters, such as sweetness, acidity, firmness, tissue texture, ascorbic acid, and polyphenolic compounds (Wojdylo et al. 2008). These characteristics, especially internal parameters, are similar within a variety. However, each variety has its special characteristics and flavour, which results in different prices and preferences by different people.

Generally, more than one apple varieties are sold by sellers at one time. Therefore, different apple varieties can be easily mixed during transportation, storage and marketing. A means for distinguishing apple varieties is needed by apple sellers. Therefore, some reliable technique is needed to discriminate varieties of apples rapidly and non-destructively.

Recognizing different kind of vegetables and fruits is a recurrent task in the supermarkets, where the cashier must be able to identify not only the species of a particular fruit or vegetable (i.e., banana, apple, pear) but also identify its variety (i.e., Golden Delicious, Jonagold, Fuji), for the determination of its price. This problem has been solved by using barcodes for packaged products but most of the time consumers want to pick their product, which cannot be packaged, so it must be weighted. Assignment of codes for each kind of fruit and vegetable is a common solution to this problem; but this approach has some problems such as the memorization, which may be a reason for errors in pricing. As an aid to the cashier, a small book with pictures and codes is issued in many supermarkets; the problem with this approach is that flipping over the booklet is time-consuming.

Research has been done a lot in the area of agricultural product sorting and grading. These include systems for the apple defect detection (Zou et al., 2010), automated strawberry grading (Xu & Zhao, 2010), banana quality inspection (Mansoory et al., 2010), tomato classification (Laykin et al, 2002) and the defect detection in citrus peel (Blasco et al., 2007). Zou et al. (2010) proposed a three color camera based classification system, that captured the whole surface of apple fruit, for detecting defects in the fruit by segmenting and counting regions of interest (ROIs) corresponding to fruit blemishes. A strawberry grading system developed by Xu & Zhao (2010) divided fruit into four grades using the shape, size and color information obtained from an image processing technique

Lino et al. (2008) proposed a system for grading lemons and tomatoes using color features for ripeness detection. Ripeness levels of the tomatoes were determined by measuring decrements in the luminance, blue and green channels as well as increments in the red channel. Fernando et al (2010) built a system to diagnose six different types of surface defects in citrus fruits using a multivariate image analysis strategy. Images were unfolded and projected onto a reference eigen space to arrive at a score matrix used to compute defective maps and 94.2% accuracy was reported. Haiguang et al. (2012) classified two kinds of wheat diseases based on color, shape and texture features by training a back propagation neural network. The resulting system achieved a classification accuracy of over 90%.

Omid et al. (2013) used shape, texture and color features to sort tomato fruits according to their circularity, size, maturity and defects. They achieved 84.4% accuracy for defect detection using a probabilistic neural network (PNN) classifier. Color, texture and shape features have been evaluated for fruit defect detection system, also in conjunctions with PNNs. Dubey & Jalal (2012a cited in Dubey & Jalal 2013) proposed a framework for recognizing and classifying fruits and vegetables. They considered images of 15 different types of fruit and vegetable collected from a supermarket. Their approach was to first segment the image to extract the region of interest and then calculate image features from that segmented region which was further used in training and classification by a multi-class support vector machine. They also proposed an Improved Sum and Difference Histogram (ISADH) texture feature for this kind of problem. From their results, ISADH outperformed the other image color and texture features.

Anderson et al. (2010) looked at a unified approach that can combine many features and classifiers, where all features are simply concatenated and fed independently to each classification algorithm. The fusion approach is validated using a multi-class fruit-and-vegetable categorization task in a semi-controlled environment, such as a distribution centre or the supermarket cashier. The results show that the solution is able to reduce the classification error in up to 15 percentage points with respect to the baseline. Fernando et al. (2010) used an unsupervised method based on a Multivariate Image Analysis strategy which uses Principal Component Analysis (PCA) to generate a reference eigenspace from a matrix obtained by unfolding spatial and color data from defect-free peel samples. In addition, a multiresolution concept is introduced to speed up the process. They tested on 120 samples of mandarins and oranges from four different cultivars: Marisol, Fortune, Clemenules, and Valencia. They reported

91.5% success ratio for individual defect detection, while 94.2% classification ratio for damaged/sound samples. Dubey & Jalal (2012b cited in Dubey & Jalal 2012c) proposed a method to detect and classify the fruit diseases using image processing techniques. First of all, they detected the defected region by k-means clustering based image segmentation technique then extracted the features from that segmented defected region which is used by a multi-class support vector machine for training and classification purpose.

Gabriel et al. (2013) proposed a pattern recognition method to automatically detect stem and calyx ends and damaged blueberries. First, color and geometrical features were extracted. Second, five algorithms were tested to select the best features. Aggelopoulou et al. (2011) developed an image processing based algorithm for early yield estimation in an apple orchard. The algorithm forecasts tree yield by analysing the texture of the tree image at full bloom.

Using Support Vector Machine classifier, they distinguished the blueberries' orientation in 96.8% of the cases. The average performance for mechanically damaged, shrivelled, and fungal decayed blueberries were reported as 86%, 93.3%, and 97% respectively. A synthesis segmentation algorithm is developed for the real-time online diseased strawberry images in greenhouse (Ouyang et al., 2013). The impact of uneven illumination is eliminated through the top-hat transform, and noise interferences are removed by median filtering. They obtained complete strawberry fruit area of the image after applying the methods of grey morphology, logical operation, OTSU and mean shift segmentation. Then, they normalize the extracted eigenvalues, and used eigenvectors of samples for training the support vector machine and BP neural network.

Their Results indicated that support vector machines have higher recognition accuracy than the BP neural network.

Color features and artificial neural network (ANN) classifier have been used by many researchers. Arribas et al. (2011) and Al Ohali (2011) used RGB color space and ANN classifier for sunflower *(Helianthus annuus)* leaves classification and grading of date fruit into three quality categories, respectively. Anami et al. (2011) classified different agriculture and horticulture products by neural network, finding that color and texture features were able to significantly detect normal and affected products.

Fernando et al (2010) built a system to diagnose six different types of surface defects in citrus fruits using a multivariate image analysis strategy. Images were unfolded and projected onto a reference eigen space to arrive at a score matrix used to compute defective maps. 94.2% accuracy was reported. A strawberry grading system developed by Xu & Zhao (2010) divided fruit into four grades using the shape, size and color information obtained from an image processing technique. Mansoory et al. (2010) used Fuzzy C-means segmentation algorithm to identify banana from images.

Haiguang et al. (2012) classified two kinds of wheat diseases based on color, shape and texture features to train a back propagation neural network. The resulting system achieved a classification accuracy of over 90%. Arefi et al. (2011) developed a segmentation algorithm for the guidance of a robot arm to pick the ripe tomato using image processing technique. To reach this aim, they prepared a machine vision system to acquire images from a tomato plant. Their algorithm works in two phases: (1) background subtraction in RGB color space and then extracting the ripe tomato considering a combination of RGB, HSI, and YIQ color spaces and (2) localizing the ripe tomato using morphological features of the image. They achieved accuracy up to 96.36% on 110 tomato images. Haidar et al. (2012) presented a method for classification of date fruits automatically based on pattern recognition and computer vision. They extracted appropriately crafted mixture of 15 different visual features, and then, tried multiple classification methods. Their performance ranged between 89% and 99%.

Cho et al. (2013) used hyperspectral fluorescence imaging for detecting cracking defects on cherry tomatoes. Omid et al. (2013) used shape, texture and color features to sort tomato fruits according to their circularity, size, maturity and defects. They achieved 84.4% accuracy for defect detection using a probabilistic neural network (PNN) classifier. Danti et al. (2012) classified 10 types of leafy vegetables using BPNN classifier with a success rate of 96.40%. They first cropped and resized the image and then extracted the mean and range of hue and saturation channel of HSV image to form the feature vector. Suresha et al. (2012) have achieved 95% classification accuracy over a dataset of containing 8 types of different vegetables using texture measures in RGB color space. They have used watershed segmentation to extract the region of interest as a pre-processing and decision tree classifier for training and classification purpose.

Faria et al. (2012) presented a framework for classifier fusion for the automatic recognition of fruits and vegetables in a supermarket environment. They combined low-cost classifiers trained for specific classes of interest to enhance the recognition rate. Chowdhury et al. (2013) have recognized 10 different vegetables using color histogram and statistical texture features. They have gained the classification accuracy up to 96.55% using neural network as a classifier.

2.1 Classification Technique

2.1.1 Artificial Neural Network

ANN is a type of artificial intelligence that imitates some functions of the person mind. ANN has a normal tendency for storing experiential knowledge. An ANN consists of a sequence of layers; each layer consists of a set of neurones (Pooja et al., 2013). All neurones of every layer are linked by weighted connections to all neurones on the preceding and succeeding layers. It uses nonparametric approach. Performance and accuracy depends upon the network structure and number of inputs (Pooja et al., 2013).

2.1.2 K-Nearest Neighbour Classifier (k-NN)

K-NN is a statistical classifier. K-NN calculates the distance metric for samples and classify based on this distance. It assigns data to the most represented category within its closest k neighbours (Abudulhamid et al., 2012). Mostly Euclidean distance is used for distances calculation between the features values of the test input with training inputs.

2.1.3 Support Vector Machine

A support vector machine builds a hyper plane or set of hyper planes in a high- or infinite dimensional space, used for classification (Pooja et al., 2013). Good separation is achieved by the hyper plane that has the largest distance to the nearest training data point of any class (functional margin), generally larger the margin lowers the generalization error of the classifier. SVM uses Nonparametric with binary classifier approach and can handle more input data very efficiently. Performance and accuracy depends upon the hyper plane selection and kernel parameter.

2.1.4 Naive Bayes Classifier

Bayes classifier is a simple probabilistic classifier. It is based on applying Bayes' theorem (from Bayesian statistics). Bayes theorem is basically strong independence assumptions theorem. It uses Maximum a Posteriori (MAP) Naive Bayes classifier in which probabilities obtained from the estimates of the probability mass function using training data (Sudhoer et al., 2013).

2.1.4.1 Why use Naïve Bayes Classifier

Reviewing other related work on apple classification system, we noted that no scholar had tried to classify apple varieties using naïve bayes technique. Although Naïve Bayes technique uses a small sample size while archiving a high accuracy level. Other classification techniques like MLP-Neural (Unay et al., 2006), fuzzy logic (Kavdir et al., 2003), principal components analysis (Bin et al., 2007) and neural networks (Ohali et al., 2011) have been used to classify apple varieties with an accuracy of 91%, 89% 83% respectively. This research will use naïve bayes technique to classify apple varieties and its results benchmarked with the other techniques already used to classify apple varieties.

3. SYSTEM DEVELOPMENT METHODOLOGY

A software development methodology or system development methodology in software engineering is a framework that is used to structure, plan, and control the process of developing a system. This research followed a prototyping software methodology as shown in fig. 2.

It is also argued that Prototyping is especially good for designing good human-computer interfaces. Overmyer (2012) stated that "One of the most productive uses of rapid prototyping to date has been as a tool for iterative user requirements engineering and human-computer interface design". Prototyping enables developers and entrepreneurs have a proof of concept to present before funders or an academic board.

A prototype is a working physical model of a system or a subsystem, a prototype serves as a preliminary version of the system or component from which requirements are extracted and on which subsequent versions are based. The different steps involved in prototyping include:

- Requirement gathering.
- Analysis
- Design.
- Implementation.
- Testing

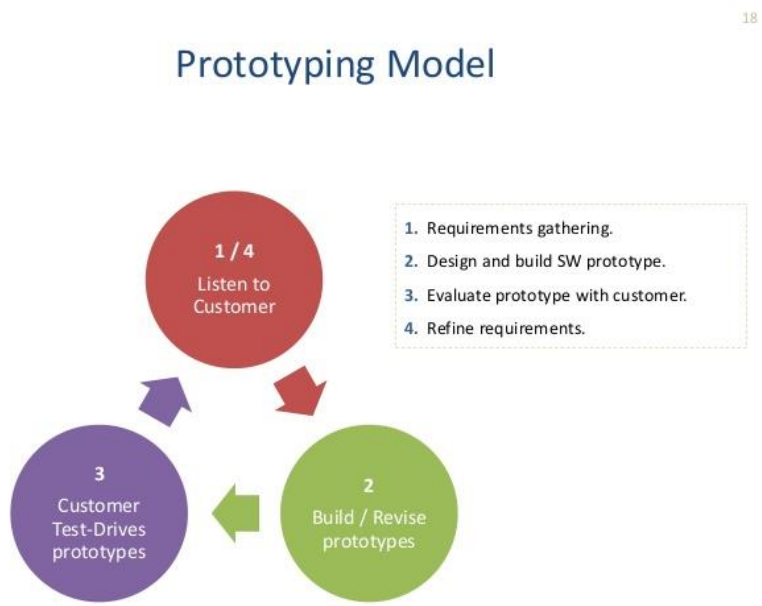


Figure 1. Prototyping Model

3.2 Requirement Gathering

The requirements gathering involve the meeting of the developer and customer to define the overall objectives for the software; identify requirement that are known and outline areas where further definition is mandatory.

The software to be designed will classify apple fruits according to their varieties. The system will have a camera for capturing apple fruits images, a computer storage folder for storing the captured images, image capture box to capture image and avoid light which will distort the picture quality.

3.2.1 Functional Requirement

Below is the functional requirement that the system will be able to perform:

- The system will be able to be trained by using apple varieties training set images until it achieves an accuracy of above 80%
- The system will be able to show the user result of classification when s/he uses it.

3.3 System Analysis

The analysis phase defines the requirements of the system, independent of how these requirements will be accomplished. Apple classification system will be depicted with a use case as shown in Figure 2. A use case shows the behaviour or functionality of a system. It consists of a set of possible sequences of interactions between a system and a user in a particular environment that are related to a particular goal

3.3.1 Apple Classification System Use Case

The apple classification system use-case suggest large-scale partitioning of the problem domain. It also provides structuring of analysis objects (i.e. actors and sub-systems). It clarifies system and object responsibilities. Below is a detailed explanation of each use case of the system:

3.3.1.1. System Training Use Case

The system will be trained by the warehouse operator to achieve an accuracy above 80% to enable other users to use the system which has a higher accuracy.

3.3.1.2 System Testing Use Case

The system will be tested by the warehouse operator after training to ensure that the system has achieved an accuracy above 80%.

3.3.1.3 Classification System Use Case

The system will be used by the warehouse owner, warehouse operator and other apple vendors to classifier apple varieties that may be mixed up while transporting them to the warehouse

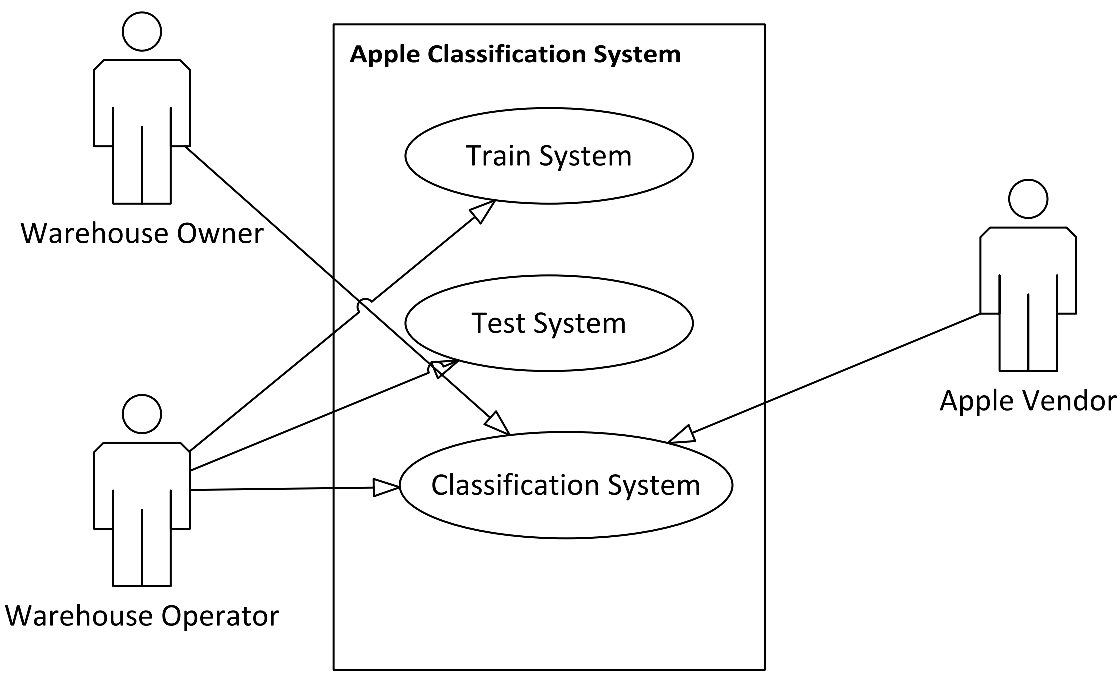


Figure 2:Apple Classification System Use Case

3.4 Design

Design is the process of problem solving and planning for a software solution. It includes low-level component and algorithm implementation issues as well as architectural view.

3.4.1 System Architecture

Below is an overlay of the apple classification system architecture with each component of the system explained in details.

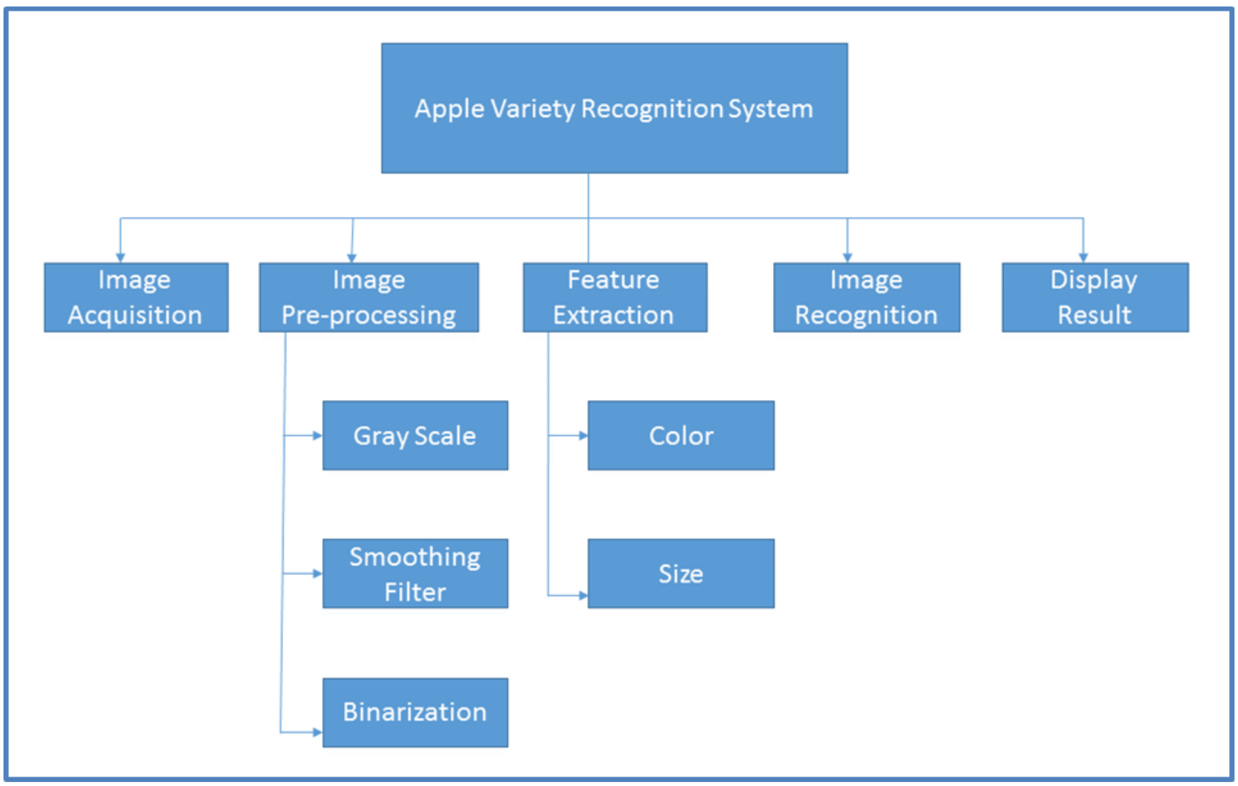


Figure 3: System Architecture

3.4.1.1 Image Acquisition

It is a preparation process to obtain apple varieties images. The 150 RGB colour images of apple varieties were captured using a phone camera with pixel resolution of 2048x1024 on a white background. These images were cropped into smaller image and stored in JPG format.

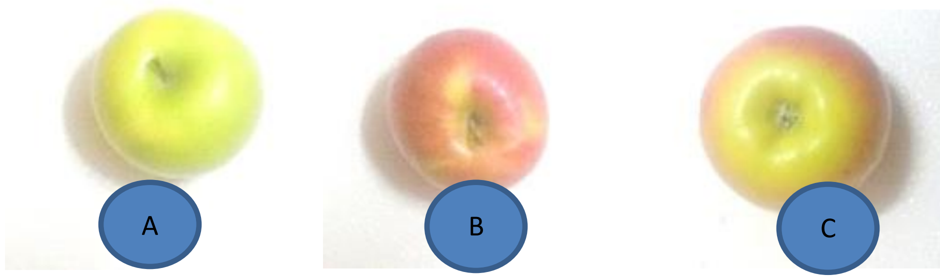


Figure 4: Images of Golden Delicious (A) Honey Crisp (B) Pink lady (C)

3.4.1.2 Image Segmentation and Pre-processing

The raw data was subjected to several preliminary processing steps to make it functional in the descriptive stages of classification and grading. In order to get apple features accurately, apple fruits images were pre-processed through different pre-processing methods. These methods were converting RGB to gray scale images and filtering the images to remove noise as described below:

3.4.1.3 Converting RGB to Grey Scale Image

The segmentation and pre-processing task are the initial stage before the image is used for the next process. The main objective of this process is to obtain the binary image with Otsu method. The Otsu method is based on selecting the lowest point between two classes of the histogram by considering the between-class variance.

A. Filtering

Averaging filter was implemented in this process to remove noise. The average filter computes the mean (average) of the grey-scale values within a rectangular filter window surrounding each pixel. This has the effect of smoothing the image (eliminating noise). The filtered pixel was calculated using the equation:

$$r = \frac{(a1 + a2 + \cdots + a9)}{9} \qquad (6)$$

3.4.1.4 Image Pre-processing

Image pre-processing involves removing low frequency background noise, normalizing the intensity of individual particles images, removing reflection and masking portion of images. It is the technique for enhancing data images prior to computational processing. Pre-processing required for shadow removal, image correction. Shadow removal is very important because shadow may disturb segmentation and feature extraction.

3.4.1.5 Feature Extraction

It is the process of generating the features to be used in selection and classification. Color pattern and size feature vectors are used for feature extraction.

A. Colour Pattern

A colour feature is one of the most widely used visual features. Color image processing is categorized into three principle areas:

- Colour transformation
- Spatial processing of individual color planes
- Color vector processing

We will use RGB color model for the representation of color. An RGB image, sometimes referred as a true color image is stored as an m-by-n-by-3 data array that defines red, green and blue color components for individual pixels. The mean value of R, the mean value of G, the mean value of B, the standard deviation values of RGB, are computed for all images in the database and saved which can be used for comparison of query image with database images.

3.4.1.6 Morphology Processing

The expression morphology denotes the study of structure. In medical image processing us use mathematical morphology by means of identify and extract significate image descriptors by using properties of the shape in the image. Morphological operations are the logical transformation established on comparison of the pixel neighbourhood with a specified pattern that is known as a structural element (Heijmans et al., 1994).

The standard morphological operations are dilation and erosion. Dilation permit objects to extend, hence possibly filling in small holes also connecting disjoint objects (Heralick et al., 2010). Erosion contract objects by turn away their borders. The composition of the main operations is dilation and erosion, it can product more complex gradation (Serral et al., 1994). Opening and closing are the widely utilitarian of these for morphological filtering. An opening process is definite as erosion followed by dilation by using the similar structuring element for together. A Closing process is definite as dilation followed by erosion (reverse opening) by using the similar structuring element for together (Raid et al., 2014).

3.4.1.7 Display Result

After image segmentation Naive Bayes technique will be trained using a training data set and the training validated using a validation dataset. After the training the system to achieve a 80% accuracy level the system will be feed with a fresh group of apple images to classify and display there results.

3.5 Implementation

3.5.1 Software Configuration

To implement the system, MATLAB R2015a environment was used to code the system. MATLAB is a high-performance language for technical computing. It is a programming environment for algorithm development, data analysis, visualization and numerical computation, using MATLAB, one can solve technical computing problems faster than with traditional programming languages, such as C, C++and FORTRAN. The name MATLAB stands for matrix laboratory. MATLAB was originally written to provide easy access to matrix software developed by the LINPACK and EISPACK projects. Today, MATLAB engines incorporate the LAPACK and BLAS libraries, embedding the state of the art in software for matrix computation. MATLAB has evolved over a period of years with input from many users. In university environments, it is the standard instructional tool for introductory and advanced courses in mathematics, engineering, and science. One can use MATLAB in a wide range of applications, including signal and image processing, communications, control design, test and measurement, financial modelling and analysis and computational biology. For a million engineers and scientists in industry and academia, MATLAB is the language of technical computing.

3.5.2 Hardware configuration

Digital camera captures image directly and stores in its memory device. Image acquisition in image processing can be defined as the action of retrieving an image from some source, usually a hardware-based source, so it can be passed through the system processes afterward. Performing image acquisition in image processing is always the first step in the implementation, without an image, no processing is possible. We used a Sony Camera model DSC-H70 with an exposure time 1/30 sec to capture the apple varieties images

3.6 Testing

A total of 150 apples varieties of honey crisp, pink lady and golden delicious were bought from the market and we used a Sony camera to capture their images in JPG format and stored them in a memory card.

We captured an image of 50 samples of golden delicious apple images, 50 samples of pink lady apple image and 50 samples of honey crisp which were then transferred from the camera memory card to the system storage folder in the laptop. These images were used for training, validation and testing the naïve bayes algorithm. We used 60 images for training, 30 images for validation and the remaining 60 were used for testing purposes. A total of three iterations of training and testing the system were done until an accuracy of above 90% was achieved.

3.6.1 System Evaluation

The identification performance of a classification framework is always evaluated by four different metrics: sensitivity, specificity, accuracy and precision (Yousef and Moghadam, 2013). According to Yousef and Moghadam 2015, sensitivity measures the proportion of correct apple varieties which were correctly identified and classified to the total number of apple varieties. Specificity measures the proportion of incorrect apple varieties which were correctly rejected to the total number of apple varieties. Accuracy measures the proportion of correctly and incorrect apple varieties that are identified correctly to the total number of apple varieties. To evaluate the performance of the system statistical analysis of experimented results was done.

They are calculated by the formulas given below:

$$Sensitivity = \frac{TP}{TP + FN} 100\% \qquad (2)$$

$$Specificity = \frac{TN}{TN + FP} 100\% \qquad (3)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} 100\% \qquad (4)$$

$$Precision = \frac{TP}{TP + FP} 100\% \qquad (5)$$

Where: TP = True Positive, FP = False Negative, TN = True Negative, FN = False Negative

According to Yousef and Moghadam 2015, Accuracy alone is not a dependable factor, since Accuracy is derived from Sensitivity and Specificity. I.e. Accuracy= (Sensitivity + Specificity)/2. In the case where, the number of correctly identified apple variety and incorrectly identified apple variety are equal, if any one of the factors, Sensitivity or Specificity is high then Accuracy will bias towards that highest value. (I.e., if Sensitivity is high, Accuracy will bias towards Sensitivity, or, if Specificity if high, Accuracy will bias towards Specificity. If both are high, Accuracy will also high and if both are low, then Accuracy will be low.).

4. RESULTS AND DISCUSSION

System Results

While testing the system we recorded true positive, true negative, false positive and false negative for each apple variety for the validation and testing set with the result shown in Table 1 & 2.

Table 1 Result for Validation Data Set

Apples Name	T	TP	FP	TN	FN
Pink Lady	10	6	0	4	1
Honey Crisp	10	8	0	4	1
Golden Delicious	10	9	0	0	1

Table 2 Result for Testing Data Set

Apples Name	T	TP	FP	TN	FN
Pink Lady	20	15	0	6	2
Honey Crisp	20	17	0	5	2
Golden Delicious	20	18	0	0	2

Where T = Number of Validation data in Table 1 and Testing data in Table 2, TP = True Positive, FP = False Negative, TN = True Negative, FN = False Negative

4.3 System Evaluation Result

Accuracy, sensitivity, specificity and precision were used to evaluate the system. Below are the results from the system. Sensitivity is the ability of a test to correctly classify an individual as ′correct apple varieties′. The ability of a test to correctly classify an apple as not the correct variety is called the test′s specificity. Accuracy compares how close the result is to the true value (Kwetishe Danjuma and Adenike Osofisan, 2010).

Table 3: Evaluation Result of Training Data Set

Apple Name	**Sensitivity (%)**	**Precision (%)**	**Specificity (%)**	**Accuracy (%)**
Pink Lady	75	100	80	91
Honey Crisp	67	100	80	92
Golden Delicious	90	100	0	90
Average	77	100	80	91

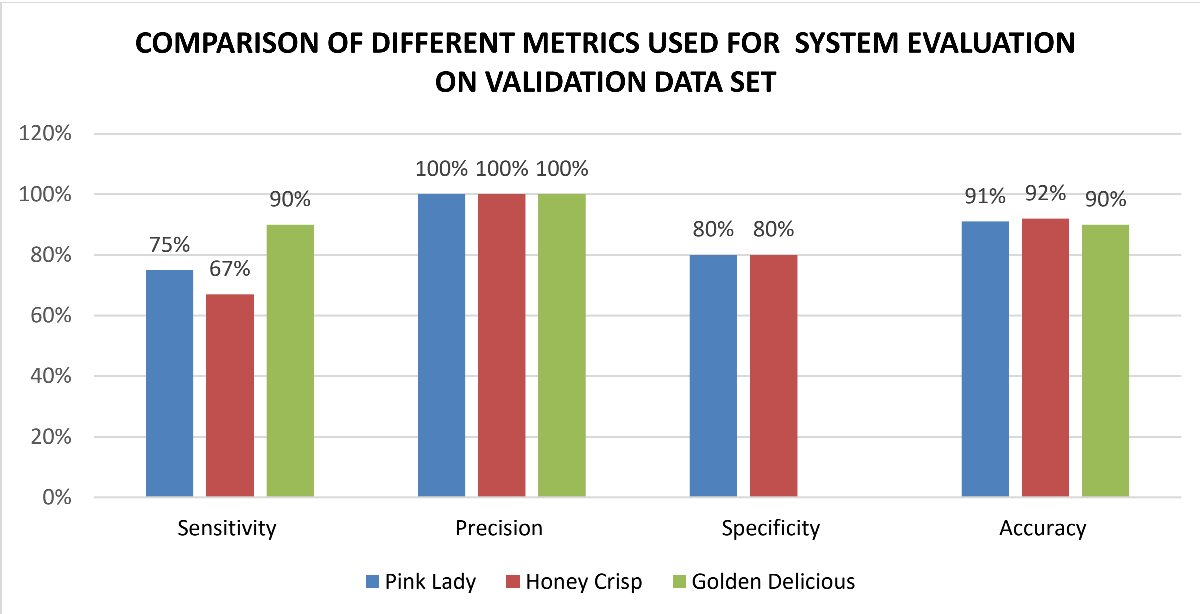


Figure 5: Graph Showing Comparison of Different Metrics Used to Evaluate the System Using Validation Data Set

Table 4: Evaluation Result of Testing Data Set

Apple Name	**Sensitivity (%)**	**Precision (%)**	**Specificity (%)**	**Accuracy (%)**
Pink Lady	88	100	75	91
Honey Crisp	89	100	72	92
Golden Delicious	90	100	0	90
Average	89	100	74	91

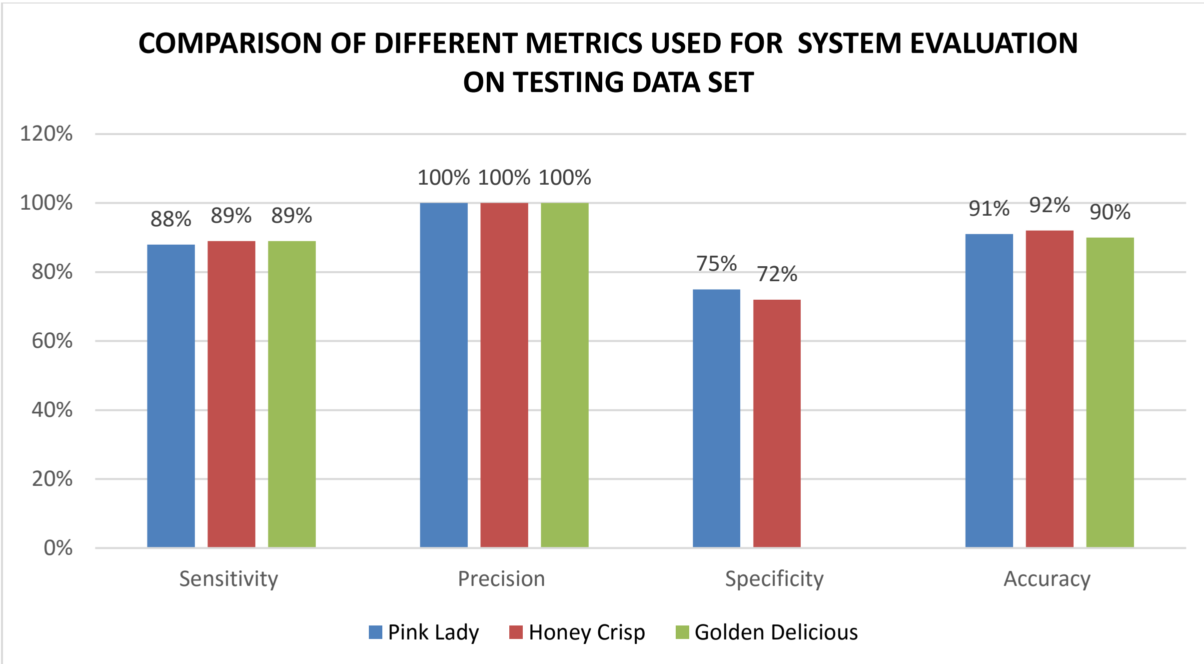


Figure 6: Graph Showing Comparison of Different Metrics Used to Evaluate the System Using Testing Data Set

4.4 Discussion

Comparison of the apple variety accuracy for the validation and testing data set showed that the highest accuracy in Naive Bayes was observed in Honey Crisp (92%), Pink lady (91%) and the last was Golden delicious (90%).The precision for the validation and testing data was 100% .The sensitivity rates of the validation data set were 75% for pink lady, 67% for honey crisp and 90% for golden delicious. For the testing data set the sensitivity were 88% for pink lady, 89% for honey crisp and 90% for golden delicious. The average sensitivity for the system was 77% for the validation data set and 89% for the testing data set.

The specificity rates of the validation data set were 75% for pink lady, 72% for honey crisp and 0% for golden delicious. For the testing data set the specificity were 80% for pink lady, 80% for honey crisp and 0% for golden delicious. Golden delicious had a specificity of 0% because during validation and testing we did not find its true negative and false positive values. This can be attributed to its unique color (green) which was easily distinguishable from the other two apple varieties whose color where almost similar. The average specificity for the system was 80% for the validation data set and 74% for the testing data set.

4.4.1 Benchmarking Naïve Bayes Technique with Existing Methods

The performance of Naive Bayes classifier was benchmarked against other classification techniques which have been used to classify apple varieties from other literatures as shown in Table 5. They were benchmarked on the basis of features extracted, classifiers used, and accuracy achieved.

Table 5: Classification Methods which have Been Used to Classify Apple Varieties

Reference	**Features**	**Training**	**Evaluation Criteria**	**Accuracy**
Unay et al. (2006)	Color and Texture	MLP-Neural	Accuracy	83%
Kavdir et al. (2003)	Color and Size	Fuzzy Logic	Accuracy	89%
Bin et al. (2007)	Color and Size	Principal Component Analysis	Accuracy	90%
Ohali et al. (2011)	Flabbiness, Size and Shape	Neural Networks	Accuracy	80%

Benchmarking the performance of the Naïve Bayes with this other techniques: MLP-Neural (Unay et al., 2006), fuzzy logic (Kavdir et al., 2003), principal components analysis (Bin et al., 2007) and neural networks (Ohali et al., 2011) showed that Naive Bayes had an accuracy of 91%, followed by principal components analysis at 90%, fuzzy logic with 89% and lastly MLP-Neural with 83%. Naive Bayes technique performance was consistent with the rest of the classification technique. Although Naïve Bayes technique used a small sample size, increasing the sample size will increase its performance.

5. CONCLUSION

Apple classification system prototype using image processing technique and Naive Bayes algorithm was built using MATLAB R2015a development platform environment. The results related to the three apple varieties: Honey crisp, golden delicious and pink lady showed that the averaged values of the estimated accuracy, sensitivity, precision and specificity were 91%, 77%, 100% and 80% respectively. Through previous research works, the literature review identified MLP-Neural (Unay et al., 2006), fuzzy logic (Kavdir et al., 2003), principal components analysis (Bin et al., 2007) and neural networks (Ohali et al., 2011) as other technique which have been used previously to classify apple varieties. Benchmarking Naive Bayes technique against principal components analysis, fuzzy logic and MLP-Neural classification technique showed that the Naive Bayes techniques performance was consistent with that of principal components analysis, fuzzy logic and MLP-Neural with 91%, 90%, 89%, and 83% respectively in terms of accuracy. Though these systems cannot match the precision and accuracy of the human eye and hand, but the speed and the cost at which they work can be easily be overcome.

5.1 Achievement

The objectives of this project were:

1. To develop a classification system that uses Naive Bayes algorithm
2. To test the performance of the system with that of a human apple classifier expert
3. To benchmark the performance of Naive Bayes classifier against the performance of other machine learning classification techniques which have been used to classify apples varieties

All the objectives stated above were achieved. The system was developed and it used image processing in order to enhance the image and applied Naive Bayes algorithm to classify the apple varieties; golden delicious, pink lady and honey crisp. The performance of the system was compared to a human expert and also benchmarked against other machine learning classification techniques used by other scholars to classify apple varieties.

5.2 Limitation

There are some constraints that happened that affected the system when it was running. First, the image produced from the thresholding process gained a lot of noise which affected the Naive Bayes training and testing output. The qualities of the image samples were blurring and sometimes they were not able to be processed by the system.

5.3 Recommendation and Further Work

Some assumptions had to be considered during the research process. The phone camera must have a good pixel to make the pictures clear to enable the system extract the morphological feature easily.

In the view of this disadvantages stated, further research should be carried out to enhance the current research. This system should be improved on the thresholding method so that there is less noise or free noise on the images. There are still a lot of techniques beside the Otsu method that can be implemented to improve the efficiency of the system.

We recommend other scholars to perform an experiment in the same environment and using the same sample size using naïve bayes, principal components analysis, fuzzy logic and MLP-Neural classification technique to ascertain the best technique which can be used to classify apple varieties accurately.

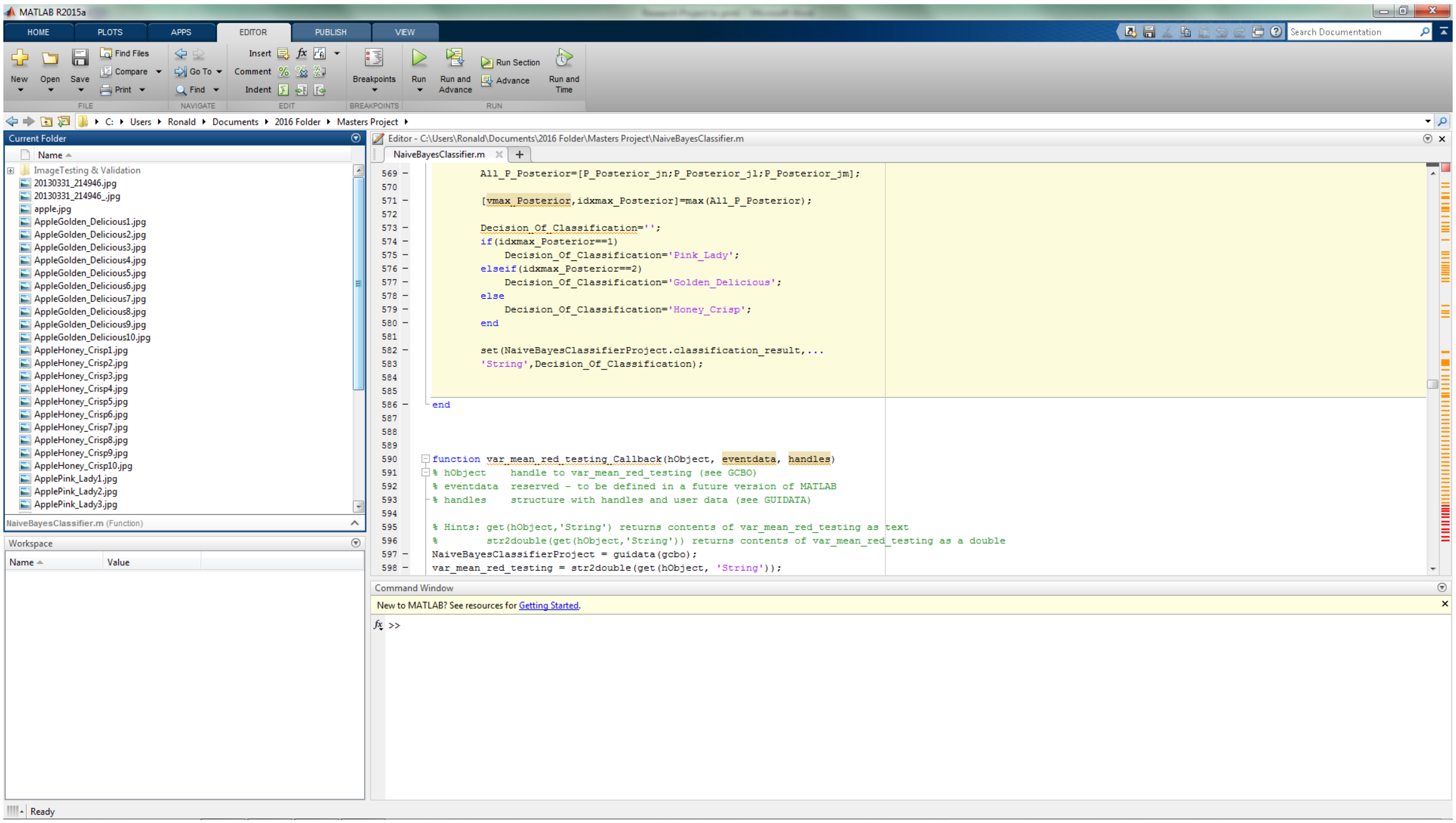
REFERENCES

1. Arribas JI, Sánchez-Ferrero GV, Ruiz-Ruiz G and Gómez-Gil J. (2011). Leaf classification in sunflower crops by computer vision and neural networks, Computer Electron Agr 78: 9-18
2. Al Ohali Y. (2011). Computer vision based date fruit grading system: design and implementation. KSU_CIS 23: 29-36.
3. Anami BS, Pujari JD and Yakkundimath R. (2011). Identification and classification of normal and affected agriculture/horticulture produce based on combined color and texture feature extraction. IJCAES I (III): 356-360.
4. Dubey, S. R. (2012). Automatic Recognition of Fruits and Vegetables and Detection of Fruit Diseases. Master's theses, GLA University Mathura, India.
5. Dubey, S. R., & Jalal, A. S. (2012a). Robust Approach for Fruit and Vegetable Classification. Procedia Engineering, 38, 3449 – 3453.
6. Dubey, S. R., & Jalal, A. S. (2012b). Detection and Classification of Apple Fruit Diseases using Complete Local Binary Patterns. In Proceedings of the 3rd International Conference on Computer and Communication Technology (pp. 346-351), MNNIT Allahabad, India.
7. Dubey, S. R., & Jalal, A. S. (2012c). Adapted Approach for Fruit Disease Identification using Images. International Journal of Computer Vision and Image Processing.
8. Dubey, S. R., & Jalal, A. S. (2013). Species and Variety Detection of Fruits and Vegetables from Images. International Journal of Applied Pattern Recognition, 1(1), 108 – 126.

9. Dubey, S. R., Dixit, P., Singh, N., & Gupta, J. P. (2013). Infected fruit part detection using K-means clustering segmentation technique. International Journal of Artificial Intelligence and Interactive Multimedia, 2(2).
10. Ebrahimi E, Mollazade K, Arefi A. (2011). Detection of Greening in Potatoes using Image Processing Techniques. Journal of American Science. 7(3).
11. Fernando, L. -G., Gabriela, A. G., Blasco, J., Aleixos, N. and Valiente, J. M. (2010). Automatic detection of skin defects in citrus fruits using a multivariate image analysis approach. Computers and Electronics in Agriculture. 71(2), 189-197.
12. Gabriel, A. L. V. and Aguilera, J. M. (2013). Automatic detection of orientation and diseases in blueberries using image analysis to improve their postharvest storage quality. Food Control, 33(1), 166–173.
13. Ouyang, C., Li, D., Wang, J., Wang, S., and Han, Y. (2013). The Research of the Strawberry Disease Identification Based on Image Processing and Pattern Recognition. Computer and Computing Technologies in Agriculture VI, 392, 69-77.

APPENDICES

Screenshot of a working station



MATLAB code for apple varieties classification system

```
function varargout = NaiveBayesClassifier(varargin)


gui_Singleton = 1;

gui_State = struct('gui_Name',       mfilename, ...

                   'gui_Singleton',  gui_Singleton, ...

                   'gui_OpeningFcn', @NaiveBayesClassifier_OpeningFcn, ...

                   'gui_OutputFcn',  @NaiveBayesClassifier_OutputFcn, ...

                   'gui_LayoutFcn',  [] , ...

                   'gui_Callback',   []);

if nargin && ischar(varargin{1})

    gui_State.gui_Callback = str2func(varargin{1});

    clc;


    % set window potition (get_size_screen/gsl_)

    gsl_ = get(0,'ScreenSize');


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 NaiveBayesClassifier is made visible.
function NaiveBayesClassifier_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

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

% Set the figure icon by matlabfreecode.wordpress.com
warning('off','MATLAB:HandleGraphics:ObsoletedProperty:JavaFrame');
jframe=get(handles.NaiveBayesClassifier,'javaframe');
jIcon=javax.swing.ImageIcon('apple.jpg');
jframe.setFigureIcon(jIcon);
```

```
% --- Outputs from this function are returned to the command line.
function varargout = NaiveBayesClassifier_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 NaiveBayesClassifier.
function trainingdata_Callback(hObject, eventdata, handles)
% hObject    handle to NaiveBayesClassifier (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
NaiveBayesClassifierProject=guidata(gcbo);
GetImageTraining=get(NaiveBayesClassifierProject.ImageTraining,'Userdata');

% determine the training data path
path_data_train=strrep(cd,...
    'Matlab_Code_To_Classification_Citrus','CitrusImage\Data Training');
```

```
% data training of Citrus nipis (jn)
lots_of_data_train_jn=10;

% data training of Citrus lemon (jl)
lots_of_data_train_jl=10;

% data training of Citrus orange (jm)
lots_of_data_train_jm=10;

lots_of_feature=4;
lots_of_class=3;

% initialization of matrix dataset
dataset=zeros(lots_of_data_train_jn,lots_of_feature);

for i=1:(lots_of_data_train_jn+lots_of_data_train_jl+lots_of_data_train_jm)

    if(i<=lots_of_data_train_jn)
        % membaca setiap file citra jeruk nipis
        filename=strcat(path_data_train,'\','ApplePink_Lady',...
        num2str(i),'.jpg');
```

```
    class{i}='Pink_Lady';
elseif(i<=(lots_of_data_train_jn+lots_of_data_train_jl))
    % membaca setiap file citra jeruk lemon
    filename=strcat(path_data_train,'\','AppleGolden_Delicious',...
        num2str(i-lots_of_data_train_jn),'.jpg');
    class{i}='Golden_Delicious';
else
    % membaca setiap file citra jeruk manis
    filename=strcat(path_data_train,'\','AppleHoney_Crisp',...
        num2str(i-(lots_of_data_train_jn+lots_of_data_train_jl)),'.jpg');
    class{i}='Honey_Crisp';
end
```