Classification of Fruits using Probabilistic Neural Networks – Improvement using Color Features

Nur Badariah Ahmad Mustafa, Kumutha Arumugam, Syed Khaleel Ahmed, Zainul Abidin Md Sharrif
Signal Processing and Control Systems (*SPaCS*) Research Group,
College of Engineering, Universiti Tenaga Nasional, Malaysia

***Abstract*—This paper presents a novel approach for the development of an intelligent fruit sorting system using techniques from Digital Image Processing and Artificial Neural Networks. The aim is to develop a fast and effective classification method along with a target of 100% efficiency. Five fruits; i.e., apples, bananas, carrots, mangoes and oranges were analysed and seventeen features were extracted based on the fruits' morphological and colour characteristics. A regular digital camera was used to acquire the images, and all manipulations were performed in a MATLAB/SIMULINK environment. The results obtained were a significant improvement over a previous study.**

Keywords: Fruit Classification, Colour Recognition, Morphological Feature Analysis, PNN, RGB, HSI.

I. Introduction

The ability to sort agricultural produce automatically is more efficient compared to the current manual inspection which is slow, labour intensive, tedious and error prone. However, automatic sorting of agri-produce requires an intelligent system that can identify the agri-produce based on its characteristics. Our research focuses on the identification of features for agri-produce and the development of a software package based on the above. These programs were designed to perform basic operations on images of fruits similar to the conventional method practised by labourers to sort the actual agricultural products. Digital Image Processing techniques were utilised for feature extraction from an image and Probabilistic Neural Networks were used for classification.

Shape or other physical dimensions are the easiest measures comprehended by humans to sort fruits and vegetables. Hence, the implementation of an automatic sorting system would require a software development which recognizes agri-produce based on its shapes and sizes. Shapes and sizes of fruits give significant differences between one agriculture produce to another. In the previous study, one way of identifying agricultural produce based on their shapes was investigated and the results obtained were encouraging [1]. However, the significant drawback of the system was misclassification of fruit type when the input images portray almost similar shapes and sizes, such as apples and oranges or bananas and carrots. This issue is overcome by introducing colour recognition into the system. Color plays an important role in object identification [2]. In this research, color feature analysis is added to provide valuable information to further distinguish the difference between various types of fruits and increase the accuracy of the classification system.

There have been a number of studies done on automatic sorting systems for agri-produce. For example, Guo Feng and Cao Qixin proposed a machine vision system that can be used for automatic high-speed fruit sorting; using mainly HSI colour space algorithm and Bayes classifier [3]. Ling Mei Chan et al produced a low-cost visual-based colour classification system to segregate the ripe and unripe fruits [4]. Palm oil grading system was developed by Alfatni et al utilising mainly RGB color space [5]. Following this, Rahim et al presented a system for the automatic recognition of paddy rice color using RGB color extraction [6]. Kyaw et al presented a system for shape based sorting of agricultural produce using Artificial Intelligence [1].

This paper presents an improvement to intelligent sorting of agri-produce by considering both shape and color features. The following sections will discuss further on the techniques used in the research, the methodology and design and finally the results obtained. In sections II and III, the fundamental concepts of Digital Image Processing and Probabilistic Neural Networks utilized in this paper are discussed. In section IV, the methodology used is presented. Section V discusses the experimentation done and analyzes the results obtained. Finally conclusions are drawn and future enhancements listed in Section VI.

II. Digital Image Processing

Digital image processing is the manipulation of computer algorithms to process the digital images. Image processing operations can be roughly divided into three major categories, *Image Compression*, *Image Enhancement and Restoration*, and *Feature Extraction*. Image compression involves reducing the amount of memory needed to store a digital image. Image defects can be corrected using Image Enhancement techniques. Once the image is in good condition, the Extraction operations can be used to obtain useful information from the image [2].

Here, the image processing toolbox of MATLAB is used to analyse the digital images of fruits. Both internal and external features of images are extracted and analysed in order to obtain best precision. External image features describe the boundary information such as shapes and other physical measurements. The features extracted from the properties of pixels inside the object boundary are called internal image features [7]. In this research, color features are the most important internal features being considered.

A. Morphological Features

Morphological features are widely used in automated grading, sorting and detection of objects in industry. It is the fundamental part of a fruit classification system and helps to differentiate based on shape. Morphological features are

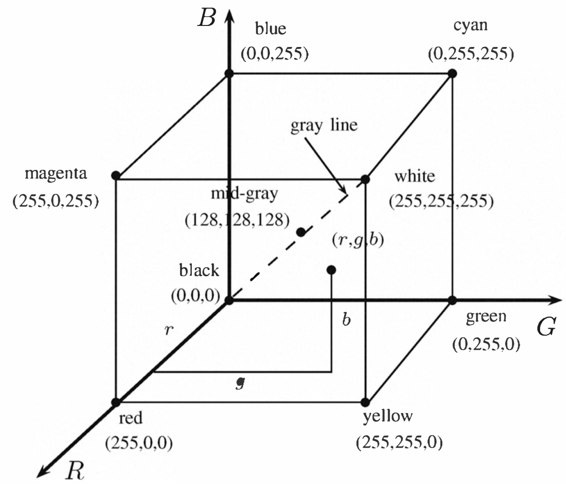

Figure 1. Schematic of the RGB Colour Cube

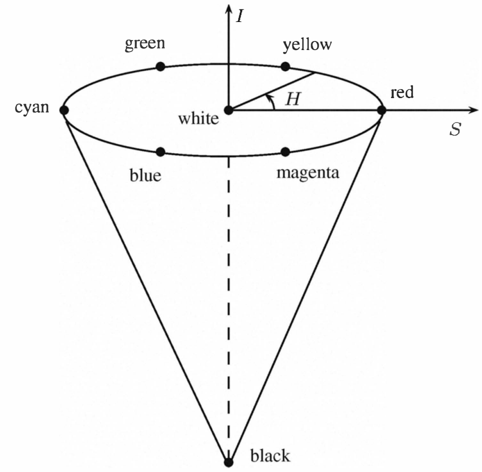

Figure 2. HSI Color Space

physical dimensional measures that characterize the appearance of an object [8]. In this paper, perimeter, major and minor axes lengths and their aspect ratios are measured for feature extraction (see Table I) [1].

B. Color Features

Colour analysis in this project is based on the RGB Colour Space and HSI colour space. In total twelve features were extracted from images by colour analysis as listed in Table I. All colour spaces are three-dimensional orthogonal coordinate systems, meaning that there are three axes that are perpendicular to one another. Basically, colour images provide three times more information compared to a two dimensional grey scale image [2].

1) RGB Color Space: RGB is the most common color feature model in image processing and is based on the primary spectral components of red (R), green (G) and blue (B), which the human eye can perceive. The RGB colour model is based on a Cartesian coordinate system. Each colour (red, green, blue) is assigned to one of the three orthogonal coordinate axes in 3D space as shown in Fig. 1 [2]. Here, every image is separated into its respective red, green and blue planes and the mean and standard deviation of pixel distribution in each plane is calculated. This computation helps to comprehend the most dominant or least dominant primary colour of the image [2], [8], [9].

2) HSI Color Space: HSI is a very different three-dimensional color space from RGB and best suits the colour description in terms that are practical for human interpretation. Human eyes describe a color object by its hue, saturation and intensity (brightness). Hue is the name of a distinct colour of the spectrum such as red, green, yellow, orange, blue, and so on. Saturation (richness of colour) is a measure of the degree to which pure colour is diluted by white light. Brightness is a subjective descriptor and impossible to measure. However, the grey level intensity is a most useful descriptor of monochromatic images and it is easily interpretable and definitely measurable. The Intensity of HSI model decouples the intensity component from the colour carrying information (hue and saturation) in a colour image. As a result HSI model is known as the most ideal tool for developing image processing algorithms [2], [7], [8], [9].

Fig. 2 illustrates a common representation of the HSI model. The cone shape has one central axis representing intensity. Along this axis are all the gray values, with black at the pointed end of the cone and white at its base. The greater the distance along this line from the pointed end, or origin, the brighter or higher the intensity. If this cone is viewed from above, it becomes a circle. Hues are determined by their angular location on this wheel (indicated by H in Fig. 2). Saturation, or the richness of color, is defined as the distance perpendicular to the intensity axis (indicated by S in Fig. 2). Colors near the central axis have low saturation and look pastel. Colors near the surface of the cone have high saturation. Intensity refers to the intensity of light present (indicated by I in Fig. 2) [9].

In this paper, RGB images are converted to HSI images and once again the mean and standard deviation of pixel distribution is calculated [2]. Given an image in RGB format, the Hue component of each RGB pixel is given by

$$H = \begin{cases} \theta, & B \leq G \\ 360 - \theta, & B > G \end{cases} \tag{1}$$

where

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

The saturation component is given by

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

Finally, the intensity component is given by

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

III. Probabilistic Neural Networks (PNN)

In recent years, intelligent systems using neural computing have emerged as a practical technology, with successful applications in many fields. Researchers from many scientific disciplines are designing Artificial Neural Networks (ANNs)

Table I
FEATURE LIST

Morphological Features	$F1$	Area	Size
	$F2$	Major-axis	
	$F3$	Minor-axis	
	$F4$	Perimeter^2/Area	Shape
	$F5$	Major-Axis/Minor-axis	
Color Features	$F6$	Mean Red	RGB
	$F7$	Standard Deviation Red	
	$F8$	Mean Green	
	$F9$	Standard Deviation Green	
	$F10$	Mean Blue	
	$F11$	Standard Deviation Blue	
	$F12$	Mean Hue	HSI
	$F13$	Standard Deviation Hue	
	$F14$	Mean Saturation	
	$F15$	Standard Deviation Saturation	
	$F16$	Mean Intensity	
	$F17$	Standard Deviation Intensity	

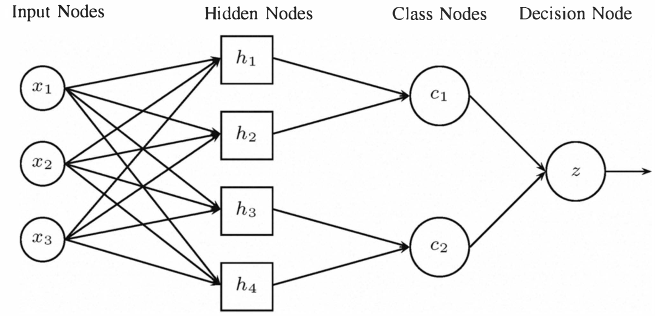

Figure 3. Architecture of a PNN

to solve a variety of problems in pattern recognition, optimization, clustering or categorization, prediction or forecasting and control systems. The majority of these applications are concerned with problems in pattern recognition, and make use of feed-forward network architectures such as the multilayer perceptron and the radial basis function network.

An ANN is a processing device, either an algorithm or actual hardware, whose design was inspired by the design and functioning of human brains and components thereof. Most neural networks have a *training* rule where the *weights* of connections are adjusted on the basis of presented patterns. In other words, neural networks *learn* from examples just like human brains and exhibit some structural capability for generalisation. A simple network has a feed forward structure; signals flow from inputs, forward through any hidden units, eventually reaching the output units as shown in Fig. 3. An artificial neuron receives a number of inputs (x_i), either from original data, or from the output of other neurons in the neural network. Each input comes via a connection that has a strength or weight (w_i). These weights correspond to synaptic efficacy in a biological neuron. Each neuron also has a single threshold value. The weighted sum of the inputs is formed, and the threshold subtracted, to compose the activation of the neuron. The activation signal is passed through an activation function (also known as a transfer function) to produce the output (y_i) of the neuron [7].

In the context of classification problems, a useful interpretation of network outputs was as estimates of probability of class membership, in which case the network was actually learning to estimate a probability density function (PDF). Conventional statistics can, given a known model, inform us what the chances of certain outcomes are. Bayesian statistics turns this situation on its head, by estimating the validity of a model given certain data. More generally, Bayesian statistics can estimate the probability density of model parameters given the available data. To minimize error, the model is then selected whose parameters maximize this PDF. In the context of a classification problem, if we can construct estimates of the PDFs of the possible classes, we can compare the probabilities of the various classes, and select the most-probable. This is effectively what we ask a neural network to do when it learns a classification problem - the network attempts to learn (an approximation to) the PDF. In 1990, Specht proposed a method to formulate this method in the form of a neural network (see Fig. 3). He called this a *Probabilistic Neural Network (PNN)* [10].

The PNN architecture has four layers [10]

1) Input layer: There is one neuron in the input layer for each predictor variable. In the case of categorical variables, $N-1$ neurons are used where N is the number of categories. The input neurons then feed the values to each of the neurons in the hidden layer.
2) Hidden layer: This layer has one neuron for each case in the training data set. The neuron stores the values of the predictor variables for the case along with the target value. When presented with the x vector of input values from the input layer, a hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies the Radial Basis Function (RBF) kernel function using the sigma value(s). The resulting value is passed to the neurons in the pattern layer.
3) Pattern layer / Summation layer: For PNN networks there is only one pattern neuron for each category of the target variable. The actual target category of each training case is stored with each hidden neuron; the weighted value coming out of a hidden neuron is fed only to the pattern neuron that corresponds to the hidden neuron's category. The pattern neurons add the values for the class they represent. Hence, it is a weighted vote for that category.
4) Decision layer: The decision layer compares the weighted votes for each target category accumulated in the pattern layer and uses the largest vote to predict the target category.

The greatest advantages of PNNs are the fact that the output is probabilistic which makes interpretation of output easy, and increases the training speed. Training a PNN actually consists mostly of copying training cases into the network, and so is as close to instantaneous as can be expected. PNNs are particularly useful for prototyping experiments (for example, when deciding which input parameters to use), as the short training time allows a great number of tests to be conducted in a short period of time. PNN is a kind of supervised neural network, proposed as an alternative to back-propagation neural network. Its training requires only a single pass and decision surfaces are guaranteed to approach the Bayes-optimal decision boundaries, as the number of training samples grows.

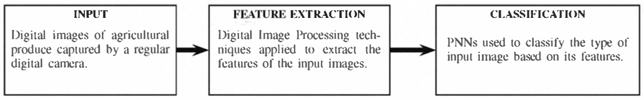

Figure 4. Block Diagram of Fruit Classification System

Figure 5. Images of Banana Taken from 9 Different Angles

The shape of the decision surface can be made as complex as necessary, or as simple as desired, by choosing an appropriate value of the smoothing parameter. Moreover, by using PNN erroneous samples can be tolerated and sparse samples are adequate for network performance [10]. This paper uses PNN as a classification tool to sort fruits into their types based on the features specified in section II.

IV. Methodology

This project involves the development of an automated fruit sorting system. The basic flow of the system is shown in Fig. 4. This system consists of an input device and a feature extraction mechanism that computes numeric information from the processing of images and a classification scheme. The classification scheme classifies the input to its fruit category depending on the features extracted from the digital image. The software development takes part at the Feature Extraction and Classification stage.

A. Input (Images) Data Collection

Data collection is the first step and an important part of this system development. Five fruits are being analysed, namely apples, bananas, carrots, mangoes and oranges. Six samples were used for each fruit for experiments purposes. Moreover, images of each sample were taken from nine different angles (Fig. 5). Thus, in total, this project utilises 270 fruit images. The black background is used to avoid noise variance during the image processing stage. The coin is used as a reference to compute the fruits' morphological features [1].

B. Feature Extraction

Feature extraction is the core of the fruit sorting system where seventeen features were extracted from each of the fruit images. This part can be divided into three parts, namely pre-processing stage, feature calculation and pattern formation as shown in Fig. 6.

The pre-processing involves basic groundwork needed to be done on images to make images most suitable for feature extraction as illustrated in flowchart of Fig. 7. Basic pre-processing includes thresholding, noise removal, boundary detection, etc. This will ensure the precision of features being extracted from fruit images. Following the pre-processing stage, the images will go through the feature calculation to obtain the values of both morphological and colour features as listed in Table I.

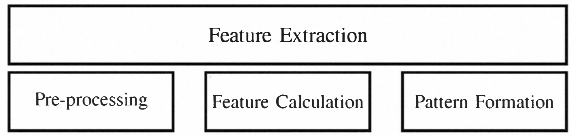

Figure 6. Subdivision of Feature Extraction

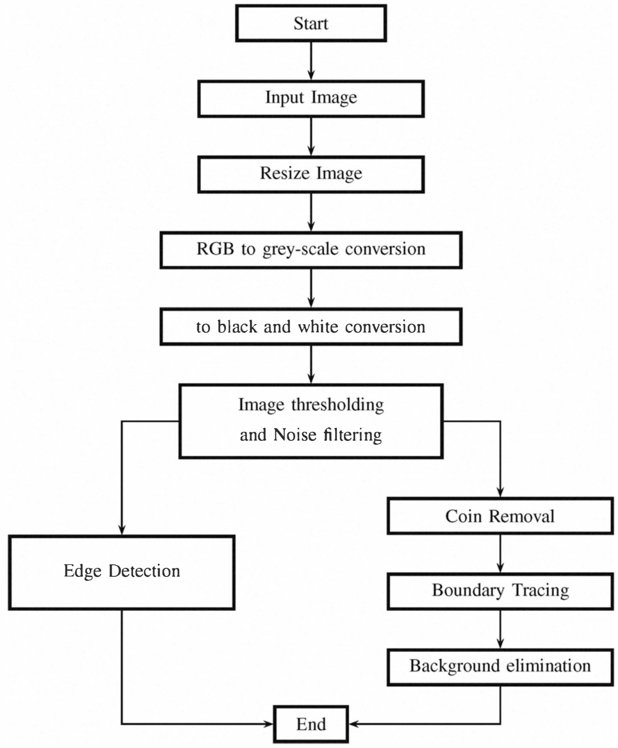

Figure 7. Flow Chart of Pre-processing Stage

Morphological feature analysis plays a fundamental role in the classification stage. The morphological analysis begins with the identification of the fruit boundary. Based on the identified fruit boundary, five morphological (dimensional and dimensionless) features are extracted from each fruit images. The determination of fruit boundary allows us to calculate the fruit area based on the pixel count. Therefore, we need a reference to translate this pixel count to exact area (unit: cm). This reference is an object whose area is known. In this paper, we choose a (Malaysian) 50-cent coin as a reference and obtain all the parameter values [1].

After morphological feature extraction, color features are analysed and extracted from each fruit image. Twelve color features are extracted in total from each image. Color feature analysis must be done only within the fruit region. Thus in the pre-processing stage after thresholding and noise removal, the coin and the entire background is removed as well. This leaves only the fruit in the image (as shown in Fig. 9). Fig. 8 and Fig. 9 show the extraction stages. Then Hue (H), Saturation

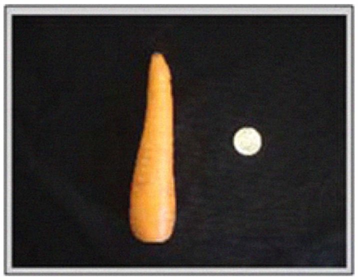

Figure 8. Original Image

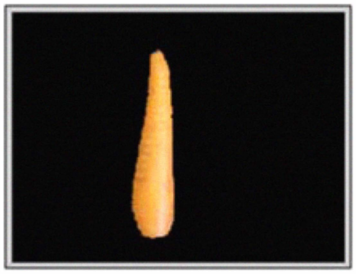

Figure 9. Fruit without Background

(S) and intensity (I) of the image can be obtained by executing the HSI algorithm on the fruit image (see Fig. 10, Fig. 11 and Fig. 12).

For the RGB analysis the red, green and blue planes of each image are separated in order to calculate the average pixel distribution of each plane. This process yields three images representing the RGB planes of the fruit image and the values obtained show the most dominant colour in each image. Following this, the HSI analysis is conducted. After the conversion of RGB input images to their respective HSI images, once again the average pixel distribution is computed (refer section II (B)). This process also yields three images of hue, saturation and intensity and the computed values. The flowchart of the color analysis is shown in Fig. 13.

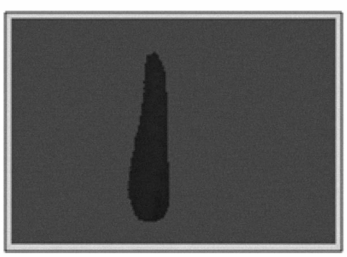

Figure 10. Hue Image

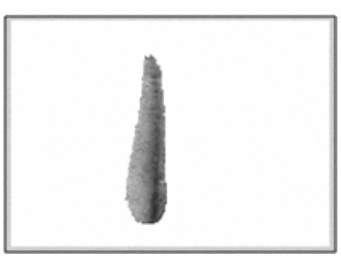

Figure 11. Saturation Image

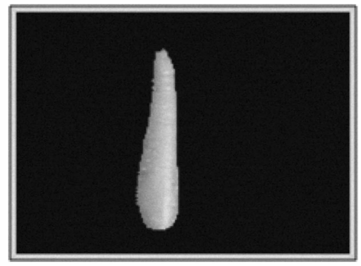

Figure 12. Intensity Image

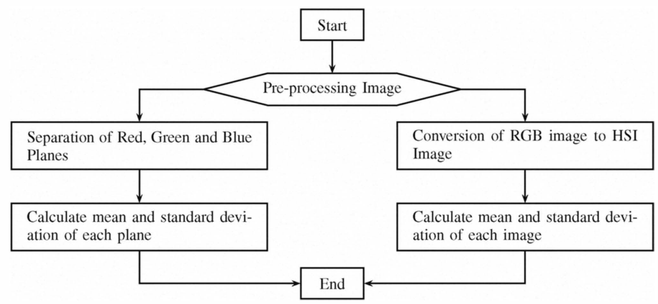

Figure 13. Flow Chart of Color Analysis

The final stage of image feature extraction is the pattern formation of the output values; i.e., putting together the numerical values of features extracted. This stage involves a large amount of data, where the data is collected from each fruit and manual manipulation can be very tedious. Thus, systematic data collection is crucial in this stage. An automatic data collection system was developed in this stage in order to complete all matrix manipulations automatically and simplify the process of experimentation. Firstly, all the features were given systematic names and saved in a database. Then, the features of each fruit image are grouped together in a 1×17 row matrix. For classification purposes each fruit type is denoted with a number at the first column making the matrix a 1×18 row matrix. Class C denotes the fruit type where C is an integer between 1 and 5. The values of C for each fruit are: apple = 1, banana = 2, carrot = 3, mango = 4, orange = 5) (see Table II).

Finally all the training and testing matrices needed were formed based on the pre-defined specifications. This will be discussed further in section V. These matrices are used as an input for the PNN classification. But before that, for better performance, each column which refers to one specific feature of the fruits is normalized using

$$x_{norm} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{5}$$

This ensures that x_{norm} lies in the closed interval $[0, 1]$. Fig. 14 summarizes the flow of the automatic data collection system.

Table II
LAYOUT OF GROUPED FEATURES FOR EACH FRUIT

Class	Morphological Features	Color Features
C	F_1 F_2 F_3 F_4 F_5	F_6 F_7 F_8 F_9 F_{10} F_{11} F_{12} F_{13} F_{14} F_{15} F_{16} F_{17}

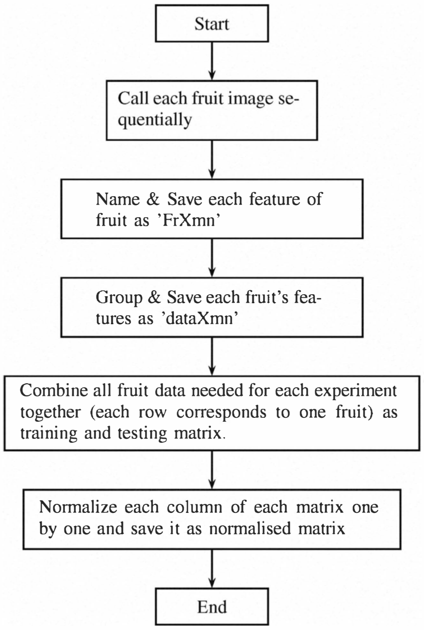

Figure 14. Flow Chart of Automatic Data Collection

V. Results and Analysis

In this section, the viability of the classification system is shown. To test the classification system, seven experiments using PNNs were conducted. The overall results have improved significantly when color features are included.

Each of the seven experiments is divided into two parts. In the first part, the fruit is classified using morphological features only (length, width, area, length over width, and perimeter squared over area). In the second part, the fruit is classified based on both the morphological and color features. The results between these two parts are compared. The improvement in the overall results can be seen from Table III.

Table III gives the specification of each experiment and the results obtained. The number of specimen refers to the number of fruits taken from each class. Therefore, for experiment 1, 1 fruit was selected from each class, and 9 images of each taken to produce a total of 45 images. Training and testing data refers to the number of images (of the same specimen) or number of specimen (all its images) used for training and testing the PNN respectively. Therefore, for experiment 2, 6 out of the 9 images of each fruit were used for training. Since 3 specimen of each fruit are taken, there are 15 specimens in total producing $15 \times 9 = 125$ images. 6 images of each fruit, leading to $15 \times 6 = 90$ images in total, are used for training. The remaining 3 images of each fruit, totalling $15 \times 3 = 45$, are used for testing or validation. On the other hand, in experiment 4, 2 specimen of each fruit producing 18 images for a total of $18 \times 5 = 90$ images, are used for training. The remaining fruit and its $9 \times 5 = 45$ images are used for testing the PNN.

The above experiments with different choices and amount of data allow us to test and determine the efficiency of the classification under all conditions. Ignoring experiment 1, it can be seen that the best results are obtained for experiment 6 where a larger chunk of the data is used for training.

Table III
Classification Accuracy obtained for Each Experiment

Expt.	Specification	Classification Accuracy (%) Morphological	Morphological + color
1	Number of Specimen: 1 Training: 7 out of 9 images Testing: remaining 2 images	70	100
2	Number of Specimen: 3 Training: 6 out of 9 images Testing: remaining 3 images	71	87
3	Number of Specimen: 3 Training: 7 out of 9 images Testing: remaining 2 images	63	84
4	Number of Specimen: 3 Training: 2 specimens Testing: 1 specimen	68	82
5	Number of Specimen: 6 Training: 6 out of 9 images Testing: remaining 3 images	66	83
6	Number of Specimen: 6 Training: 7 out of 9 images Testing: remaining 2 images	70	90
7	Number of Specimen: 6 Training: 4 specimens Testing: 2 specimens	66	79

VI. Conclusion

In this paper a scheme for fruit classification has been presented and its efficacy tested by experiments using five fruits. The introduction of color features has significantly improved the performance of the system. The classification efficiency has improved to between $79 - 90\%$. To improve the performance of this system, texture analysis can be added as well. Texture analysis will provide significant differences between one object's textures from another. The current system requires the images taken to be done under proper and consistent lighting. This is because lighting affects the true color of the fruits when images are being captured. A system that is robust to inconsistencies would be an advantage.

References

[1] M. M. Kyaw, S. K. Ahmed, and Z. A. M. Sharrif, "Shape-based sorting of agricultural produce using artificial intelligence in a matlab / simulink environment," in *5th International Colloquium on Signal Processing and it's Applications (CSPA 2009)*, March 2009.

[2] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Prentice Hall, 2nd ed., 2002.

[3] G. Feng and C. Qixin, "Study on color image processing based intelligent fruit sorting system," in *Proceedings of the 5th World Congress on Intelligent Control and Automation*, pp. 706 – 709, June 2004.

[4] L. M. Chan, R. Tan, and G. Thio, "Design of visual-based color classification system," *Journal for the Advancement of Science and Arts*, vol. 2, pp. 30 – 33, 2007.

[5] M. S. M.Alfatni, H. Z. M. S. Abdul Rashid Mohammad Shariff, and O. M. Eshanta, "Oil palm fruit bunch grading using red, green and blue digital number," *Journal of Applied Sciences*, vol. 8, no. 8, pp. 1444 – 1452, 2008.

[6] A. A. Rahim, N. E. Abdullah, H. Hashim, and M. M. Kamal, "Intelligent paddy rice colour recognition suitable for harvesting," in *ISITS*, pp. 56 – 64, 2007.

[7] D. S. Jayas, J. Paliwal, and N. S. Visen, "Multi-layer neural networks for image analysis of agricultural products," *J. Agric. Eng. Res.*, vol. 77, no. 2, pp. 119 – 128, 2000.

[8] J. Paliwal, N. S. Visen, D. S. Jayas, and N. D. G. White, "Cereal grain and dockage identification using machine vision," *Biosystems Engineering*, vol. 85, no. 1, pp. 51 – 57, 2003.

[9] H. Luijten, "Basics of color based computer vision implemented in matlab," traineeship report, Technische Universiteit Eindhoven, June 2005.

[10] M. Negnevitsky, *Artificial Intelligence: A Guide to Intelligent Systems*. Addison Wesley.