

**AUTOMATED OIL PALM FRUIT GRADING USING  
ARTIFICIAL INTELLIGENCE**

By

**MUHAMMAD HANIF BIN AMARAN**

**FINAL REPORT**

Submitted to the Electrical and Electronics Engineering Programme  
In Partial Fulfillments of the Requirements  
for the Degree  
Bachelor of Engineering (Hons)  
(Electrical & Electronics Engineering)

**MAY 2011**

**Universiti Teknologi PETRONAS  
Bandar Seri Iskandar  
31750 Tronoh  
Perak Darul Ridzuan**

© Copyright 2011

by

**Muhammad Hanif bin Amaran, 2011**

# CERTIFICATION OF APPROVAL

## AUTOMATED OIL PALM FRUIT GRADING USING ARTIFICIAL INTELLIGENCE

by

Muhammad Hanif bin Amaran

Submitted to the  
Electrical & Electronics Engineering Programme  
Universiti Teknologi PETRONAS  
in partial fulfilment of the requirement for the  
Bachelor of Engineering (Hons)  
(Electrical & Electronics Engineering)

Approved:



---

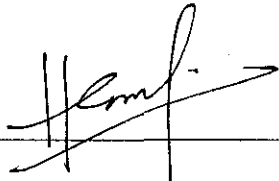
Ms. Zazilah May  
Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS  
TRONOH, PERAK

May 2011

## **CERTIFICATION OF ORIGINALITY**

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

A handwritten signature in black ink, appearing to read 'Hanif', is written over a horizontal line. The signature is stylized with a long horizontal stroke extending to the right.

MUHAMMAD HANIF BIN AMARAN

## **ABSTRACT**

This project deals with the grading of oil palm fruit based on ripeness of oil palm fruit. The current procedure in the palm oil mills is graded manual by human graders. The result from manual grading are very subjective and inconsistent as it varies and depends on techniques and experience of each human graders. Hence, it affects the quality and quantity of the oil that can be extracted. In this project, a new model of automated grading system for oil palm fruit is developed using the RGB color model and artificial fuzzy logic. The purpose of this grading system is to distinguish between the three different classes of oil palm fruit which are underripe, ripe and overripe. The ripeness or color ripening index was based on different color intensity. The grading system uses a computer and a CCD camera to analyze and interpret images correspondent to human eye and mind. The computer program is developed for the image processing part like the segmentation of colors, the calculation of the mean color intensity based on RGB color model and the decision making process using fuzzy logic to train the data and make the classification for the oil palm fruit. The program developed has been able to distinguish the three different classes of oil palm fruit automatically with 86.67% of overall efficiency. This project provides a very good technique to standardize the oil palm fruit grading system over a large area and the research will continue to normalize the system to be able to use under different source of lighting.

## **ACKNOWLEDGEMENTS**

*In the name of Allah, the Most Gracious, the Most Merciful.*

I would like to express the deepest appreciation to my supervisor, Ms Zazilah May, Lecturer of Electrical & Electronic Department, UTP. The supervision and support she gave truly help me throughout completing this project. She has guided and corrected various documents of mine with attention and care.

Special thanks to En Shafie Shaari for his generosity in providing the oil palm fruit. The oil palm fruit was taken from his oil palm plantation which is located at Batu 4, Jalan Bidor-Teluk Intan, Perak. This project would not have been possible without the invaluable data provided.

I am heartily thankful to my lovely mother, father and the whole family whose encouragement, guidance and support from the initial to the final level enable me to successfully finish this project.

Lastly, I offer my regards and blessings to all of those who supported me in any respect during the completion of the project.

## TABLE OF CONTENTS

<b>ABSTRACT</b> .....	i	
<b>ACKNOWLEDGEMENT</b> .....	ii	
<b>LIST OF FIGURES</b> .....	v	
<b>LIST OF TABLES</b> .....	vii	
<b>CHAPTER 1</b>	<b>INTRODUCTION</b> .....	1
1.1	Background of Study.....	1
1.2	Problem Statement.....	2
1.3	Objectives.....	2
1.4	Scope of study.....	2
<b>CHAPTER 2</b>	<b>LITERATURE REVIEWS</b> .....	4
2.1	RGB Color Model.....	4
2.2	Artificial Fuzzy Logic.....	5
	2.2.1 <i>Fuzzy Logic</i> .....	5
	2.2.2 <i>Membership Function</i> .....	5
	2.2.3 <i>Fuzzy Rules</i> .....	6
2.3	Past research on oil palm fruit.....	6
2.4	Past research on other fruit.....	7

<b>CHAPTER 3</b>	<b>METHODOLOGY.....</b>	<b>9</b>
3.1	Process Identification.....	9
3.1.1	<i>Data Acquisition.....</i>	<i>10</i>
3.1.2	<i>Grading system model development.....</i>	<i>11</i>
3.1.3	<i>Fuzzy Logic Development.....</i>	<i>13</i>
3.2	Graphical User Interface.....	16
3.3	Process Duration.....	17
3.4	Tools and Equipment Required.....	17
<b>CHAPTER 4</b>	<b>RESULTS AND DISCUSSION.....</b>	<b>18</b>
4.1	Results.....	18
4.2	Discussion.....	29
<b>CHAPTER 5</b>	<b>CONCLUSION AND RECOMMENDATIONS.....</b>	<b>31</b>
5.1	Conclusion.....	31
5.2	Recommendations.....	31
<b>REFERENCE.....</b>		<b>33</b>
<b>APPENDIX A</b>	Gantt Chart for FYP I and FYP II .....	34
<b>APPENDIX B</b>	Program Code .....	35

## LIST OF FIGURES

Figure 1:	RGB Color Model.....	4
Figure 2:	Project's Methodology.....	9
Figure 3:	Data acquisition setup.....	10
Figure 4:	The original image of the sample.....	11
Figure 5:	The data after remove the background.....	11
Figure 6:	Ripe sample.....	15
Figure 7:	Underripe sample.....	15
Figure 8:	Overripe sample.....	15
Figure 9:	Graphical User Interface.....	16
Figure 10a:	Sample image for ripe category.....	19
Figure 10b:	Red layer of ripe sample.....	19
Figure 10c:	Green layer of ripe sample.....	19
Figure 10d:	Blue layer of ripe sample.....	19
Figure 11a:	Sample image for underripe category.....	20
Figure 11b:	Red layer of underripe sample.....	20
Figure 11c:	Green layer of underripe sample.....	20
Figure 11d:	Blue layer of underripe sample.....	20
Figure 12a:	Sample image for overripe category.....	21
Figure 12b:	Red layer of overripe sample.....	21



Figure 12c:	Green layer of overripe sample.....	21
Figure 12d:	Blue layer of overripe sample.....	21
Figure 13:	Graph of the RGB value for 20 samples of fruit from underripe category .....	22
Figure 14:	Graph of the RGB value for 20 samples of fruit from ripe category .....	24
Figure 15:	Graph of the RGB value for 20 samples of fruit from overripe category.....	25
Figure 16:	Grading system plant.....	26
Figure 17:	Fuzzy Interference System, FIS.....	27
Figure 18:	Rule viewer of the fuzzy system.....	28

## LIST OF TABLES

Table 1:	The membership input range of the fuzzy system .....	13
Table 2:	A set of fuzzy rules for oil palm fruit grading system .....	14
Table 3:	The RGB value for 20 samples of fruit from underripe category.....	21
Table 4:	The RGB value for 20 samples of fruit from ripe category.....	23
Table 5:	The RGB value for 20 samples of fruit from overripe category.....	24
Table 6:	The range of RGB intensity of the sample of categories .....	25
Table 7:	Defuzzification Algorithms .....	29

# CHAPTER 1

## INTRODUCTION

### 1.1 Background of Study

Automated grading of agriculture products has been getting special interest of late as the demand for higher quality food products produced within a shorter period of time has increased. Market grade of quality food products are determined by their multiple features: flavor, appearance and texture. While flavor may be measured using chemical compounds to determine the sweetness or acidity, texture properties such as firmness and mouth feel are difficult to measure. In automated fruit grading, appearance (shape, color and size) is generally utilized to classify the fruit's grade.

Color provides helpful information in estimating the maturity and examining the freshness of fruits. Color is one of the most significant criteria related to fruit identification and fruit quality and it is a good indicator for ripeness. The color of an object is determined by wavelength of light reflected from its surface. In biological materials the light varies widely as a function of wavelength. These spectral variations provide a unique key to machine vision and image analysis.

Red, Green and Blue are the primary color components. They are additive, when adding colored lights and subtractive when adding paint pigments. Although the process followed by human brain in perceiving color is a psychological phenomenon that is not yet fully understood, the physical nature of color can be expressed on a formal basis supported by experimental and theoretical results.

## **1.2 Problem Statement**

In Malaysia, the palm oil fruit is currently graded manually by human. Inspections are done based on the color of oil palm fruit and on the plantation period. Even though color can be used to classify the oil palm fruit, it is difficult to differentiate the color from one category to another because the color of the fruit is not uniformly distributed over the whole fruit and the range of the color is quite similar to each other. There is no accurate range of color to differentiate for each class.

Furthermore, manual inspection is very subjective as different human graders classify differently based on experience and expert grader may fail to articulate the grading criteria properly. Dissatisfaction and dispute among estate owners and factories supervisors are frequent due to improper grading. Therefore, to increase the accuracy and quality of oil palm fruit grading in palm mills, an automated grading system based on the outer surface color need to be studied and improved.

## **1.3 Objectives**

The objectives of this project are:

- To study the current grading system of oil palm fruit
- To analyze the color for the each category of oil palm fruit
- To improve and develop a new grading system for oil palm fruit using RGB color model and artificial fuzzy logic method

## **1.4 Scope of Work**

The scope of work of this study is to understand the grading system and the degree of ripeness of oil palm fruit by carrying out the literature review as well as brief research about this topic. Besides, the data gathered from oil palm

plantation is analyzed and then used to develop a new grading system using machines with helps of artificial fuzzy logic.

## CHAPTER 2

### LITERATURE REVIEW

#### 2.1 RGB Color Model

The RGB color model is an additive color model in which red, green, and blue light are added together in various ways to reproduce a broad array of colors. The name of the model comes from the initials of the three additive primary colors, red, green, and blue. It is additive in the sense that the three light beams are added together, and their light spectra add, wavelength for wavelength, to make the final color's spectrum[1][2].

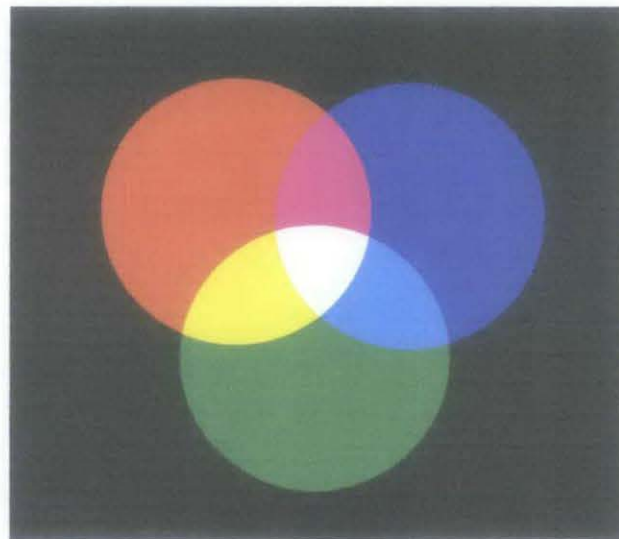


Figure 1: RGB color model

The RGB color model is widely used in the society, covering a lot of aspects such as in television[3], computer, video projector, digital cameras and many more. It is closely related to this project since this project uses the digital camera and the computer to gather and utilize the data. It is also one of the powerful color representations in the society since it has been used in various fields.

## 2.2 Artificial Fuzzy Logic

### 2.2.1 Fuzzy Logic

Fuzzy logic is a form of many-valued logic derived from fuzzy set theory to deal with reasoning that is fluid or approximate rather than fixed and exact. In contrast with "crisp logic", where binary sets have two-valued logic, fuzzy logic variables may have a truth value that ranges in degree between 0 and 1. In other words, fuzzy logic is a superset of conventional (boolean) logic that has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false. Furthermore, when linguistic variables are used, these degrees may be managed by specific functions.

### 2.2.2 Membership function

The membership function is a graphical representation of the magnitude of participation of each input. It associates a weighting with each of the inputs that are processed, define functional overlap between inputs, and ultimately determines an output response. The rules use the input membership values as weighting factors to determine their influence on the fuzzy output sets of the final output conclusion. Once the functions are inferred, scaled, and combined, they are defuzzified into a crisp output which drives the system.

There is a unique membership function associated with each input parameter. The membership functions associate a weighting factor with values of each input and the effective rules. These weighting factors determine the degree of influence or degree of membership (DOM) each active rule has. By computing the logical product of the membership weights for each active rule, a set of fuzzy output response magnitudes are produced. All that remains is to combine and defuzzify these output responses.

Fuzzy sets are sets whose elements have degrees of membership. One of the fuzzy set is the average value of red pixel. The red pixel value is not fixed for each category of oil palm fruit. The value may vary from one fruit to another fruit

since the color of the fruit is not uniformly distributed. The membership function defines the fuzzy set for possible values of the red pixel value. Instead of just determining whether the fruit is red or not red, it can be further define the color like the fruit is 0.8 red or 0.3 red. The membership function choice is the subjective aspect of fuzzy logic, it allows the desired values to be interpreted appropriately.

### 2.2.3 Fuzzy Rules

Fuzzy rule is defined as a conditional statement in the form:

IF x is A  
THEN y is B

Where x and y are language values determined by fuzzy sets on the universe of discourse X and Y, respectively.

Fuzzy logic, with fuzzy rules, has the potential to add human-like subjective reasoning capabilities to machine intelligences, which are usually based on bivalent boolean logic.

## 2.3 Past Research on Oil Palm Fruit

In Malaysia, researches in automated fruit grading have gradually caught the interest of many fruit-producing industries. For the oil palm fruit research, there were several report has been published in the past few years. The research has been done using several techniques and the most recent one are done by using RGB Digital Number[4] and by using Neuro-Fuzzy[5]. There are also other techniques used to evaluate the grading of oil palm fruit[6-7].

The grading system technique using RGB Digital Number is based on the RGB color model. The image taken is converted to digital number. Then the mean of red, green and blue are computed for the whole images. Each category has its



own range of mean value for red, green and blue which has been determined from the sample test. The program will do the ripeness test as well as the decision making process to classify the oil palm fruit and give the end result in terms of oil palm categories. The study reported that it gives the good result but there is still a room for the improvement.

The oil palm fruit can be classified into three different categories which are underripe, ripe and overripe[4]. Investigation on the relationship between the appearance of oil palm fruit and the degree of ripeness has been carried out. According to these standards, the underripe class has a reddish black color[8]. Meanwhile, ripe class appears reddish orange over the upper half, tending to lighten towards base. Finally the fruits belong to overripe class when it exhibits almost entirely reddish orange colouration.

With the help of this research, we are able to classify the oil palm fruits using RGB color model. Even though RGB color model have been popularly employed for palm grading, but there is still a room for the improvement.

### **2.3 Past research on other fruit**

Besides from the oil palm fruit research, there are also other fruit that can be studied in order to develop the classification techniques. The automated grading techniques for apple[9-10], banana[11], orange[12] and other fruit[13-14] also have been discussed for the past few years.

A group of researcher who done the research on the apple grading system using the RGB color space as the medium for their system. While the research on the grading system for bananas using a different color space which is L\*a\*b color space. It is approximately to the human vision and it is believed to have a good representation.

For the other fruit like mango, a research has been done by using artificial intelligent method which is Fuzzy Logic. There are some criteria needs to be set for this method and the criteria are area and length of the fruit.

Based on the past research, most of them were using the RGB color model as the color space and it gives a high percentage of accuracy. So it is suitable to use this color representation on this project.



### 3.1.1 Data Acquisition

The oil palm fruit are collected from the oil palm plantation in Batu 4, Jalan Bidor-Teluk Intan, Perak. For each category, 25 samples were taken and used for this project where 20 samples are allocated for training sample while 5 samples are allocated for testing sample.

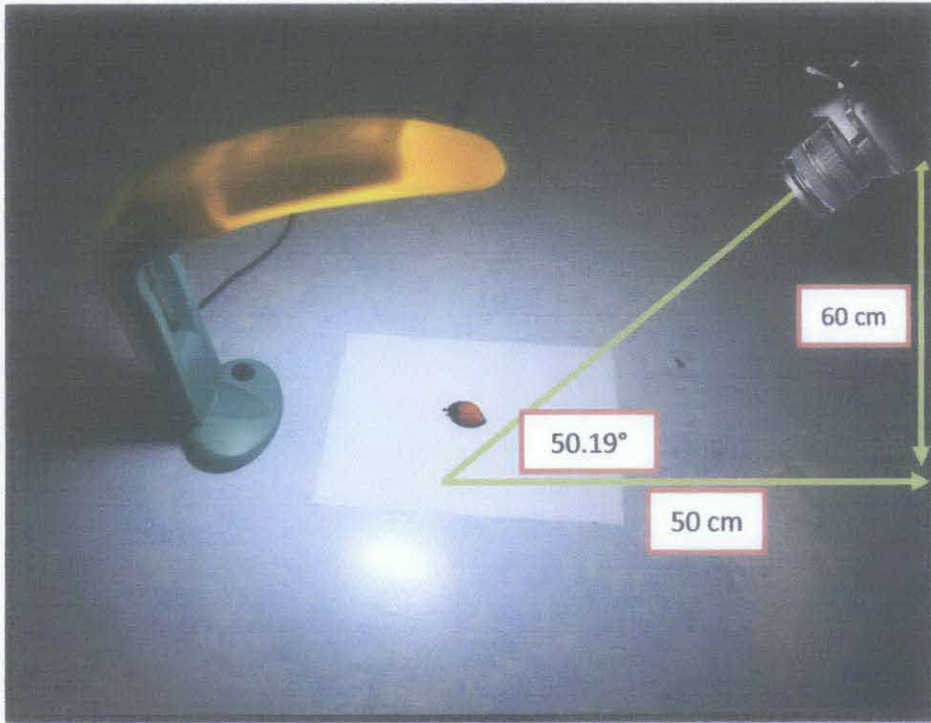


Figure 3: Data acquisition setup

Figure 3 shows the setup for capturing the image data for all samples. The image data is captured using a CCD camera which is located 50 cm away from the sample with the angle of  $50.19^\circ$ . By using setup, the sample image taken can cover a wide surface of the fruit and would be able to process the data for the next step.

The setup is done by a simple measurement using a ruler and a protractor. The location of oil palm fruit is fixed on the center of the A4 paper and the camera is placed 50 cm away from the fruit where it is measured by using a ruler. Then it is done similarly for the height position. The angle is determined by using a protractor. The distance and angle between the oil palm fruit and the camera is fixed to ensure the consistency of the image data.

### 3.1.2 Grading system model development

The grading system model has been developed using data gathered. The intensity of color is analyzed to determine the ripeness of each fruit. However, the background color of the image may mislead the RGB intensity color of each fruit. Because of this, the background of the fruit is removed and converted to black background. The black color represents zero value in terms of number and it rectifies the effect of colored background. Figure 4 shows the sample of the original image while Figure 5 shows the image after the background has been changed to black color.



Figure 4: The original image of the sample



Figure 5: The data after remove the background

After removing the background color of the image data, the image data is ready to be analyzed. In this project, the computer program was written using MATLAB R2009b. The program reads the data from the specific folder which contains the image file of the sample.

The computer program used three separate layers to do the segmentation part based on the RGB layers. The mean color intensity for each layer is calculated by using the following formula:

$$\text{MeanR} = R / \text{No. of pixels}$$

$$\text{MeanG} = G / \text{No. of pixels}$$

$$\text{MeanB} = B / \text{No of pixels}$$

Where:

MeanR = Mean value of Red layer

MeanG = Mean value of Green layer

MeanB = Mean value of Blue layer

R = Red pixel

G = Green pixel

B = Blue pixel

This step was individually performed on each image of the oil palm fruit. The range value (minimum and maximum mean) of RGB intensity for each category (underripe, ripe and overripe) is analyzed and used as a reference of the fuzzy logic system.

### 3.1.3 Fuzzy Logic development

The decision making process or the classification of oil palm fruit is done by using the fuzzy logic system. The fuzzy logic took three inputs or three criteria namely red, green and blue to do the process and gives the best output. Each of these inputs is divided into 3 different ranges which are low, medium and high.

The range of these input are determined based on the contribution of RGB color of the training data from the three different categories. The range is simplified in the Table 1.

Table 1: The membership input range of the fuzzy system

Membership	Low	Medium	High
Red	0-90	75-150	135-200
Green	0-30	22-45	38-80
Blue	0-28	20-50	44-70

The core part of the decision making using fuzzy logic is based on the fuzzy rules which have been defined by the user with the help from the expert grader. A set of rules is condition which can determine the category of the oil palm fruit. There are 17 rules for this system to evaluate a set of fuzzy rules are simplified in the Table 2.

Table 2: A set of fuzzy rules for oil palm fruit grading system

<b>No</b>	<b>Red</b>	<b>Green</b>	<b>Blue</b>	<b>Category</b>
1	Low	Low	Low	Underripe
2	Low	Low	Medium	Underripe
3	Low	Medium	Low	Underripe
4	Low	Medium	Medium	Underripe
5	Low	High	Medium	Ripe
6	Low	High	High	Ripe
7	Medium	Low	Low	Underripe
8	Medium	Low	Medium	Underripe
9	Medium	Medium	Low	Underripe
10	Medium	Medium	Medium	Ripe
11	Medium	Medium	High	Ripe
12	Medium	High	Medium	Ripe
13	Medium	High	High	Ripe
14	High	Medium	Medium	Overripe
15	High	Medium	High	Overripe
16	High	High	Medium	Overripe
17	High	High	high	Overripe

The samples of oil palm fruit from each category, ripe, underripe and overripe are shown in Figure 6-8 respectively. The color are varies from one category to other category and the color is not uniformly distributed over the whole fruits.





Figure 6: Ripe sample



Figure 7: Underripe sample



Figure 8: Overripe sample

## 3.2 Graphical User Interface

The graphical user interface (GUI) is developed by using GUIDE toolbox in the MATLAB. It helps the user to do the classification based on the input image. The user has the option to choose the image and the rest of the classification will be done by the program.

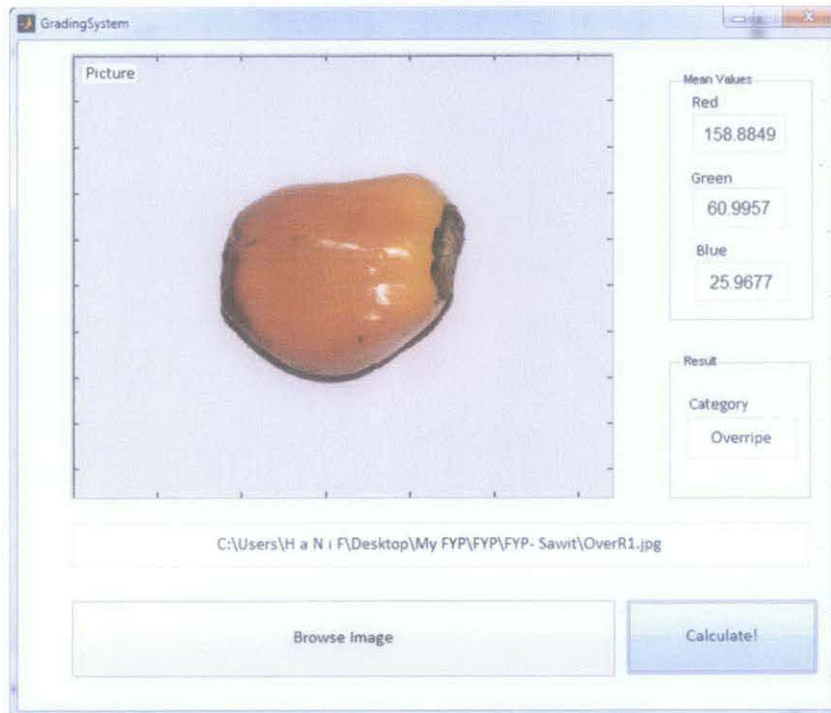


Figure 9: Graphical User Interface

### Button

- Browse Image – to browse the image from the computer
- Calculate – to calculate the mean value and run the fuzzy logic system where it gives the category of the oil palm fruit

### Result

- Mean value of Red, Green and Blue pixel.
- Type of category – Underripe, Ripe and Overripe

### **3.3 Project Duration**

In order to effectively monitor the progress of this project, a Gantt chart consist of one year duration planning had been constructed. (See Appendix B)

### **3.3 Tools and Equipment Required**

- Software
  - MATLAB R2009b
  - Adobe Photoshop CS5
  
- Hardware
  - Olympus E-520
  - Lighting with Diffuser
  - Personal Computer (PC)

## CHAPTER 4

### RESULTS AND DISCUSSION

#### 4.1 Results

The program calculated the mean color intensity for each layers to differentiate between the ripeness of the oil palm fruit. This method is done for 25 images from each category. Out of 25 images, only 20 images represent the training samples for each category (ripe, underripe and overripe) while the other 5 images represent the testing samples.

Figure 10(a-d) shows one sample from ripe category and the three separate layers which are red, green and blue layers respectively. The brighter area shows the higher intensity of pixel for that layer. In this case, red layer appears to have the brighter area on the right side of the fruit which means it has the high intensity of red pixel on that area. It always been observed that for the ripe category, the red pixel has a high intensity for half of the fruit. While for green and blue layer, the ripe category does not have high intensity of green and blue pixels.

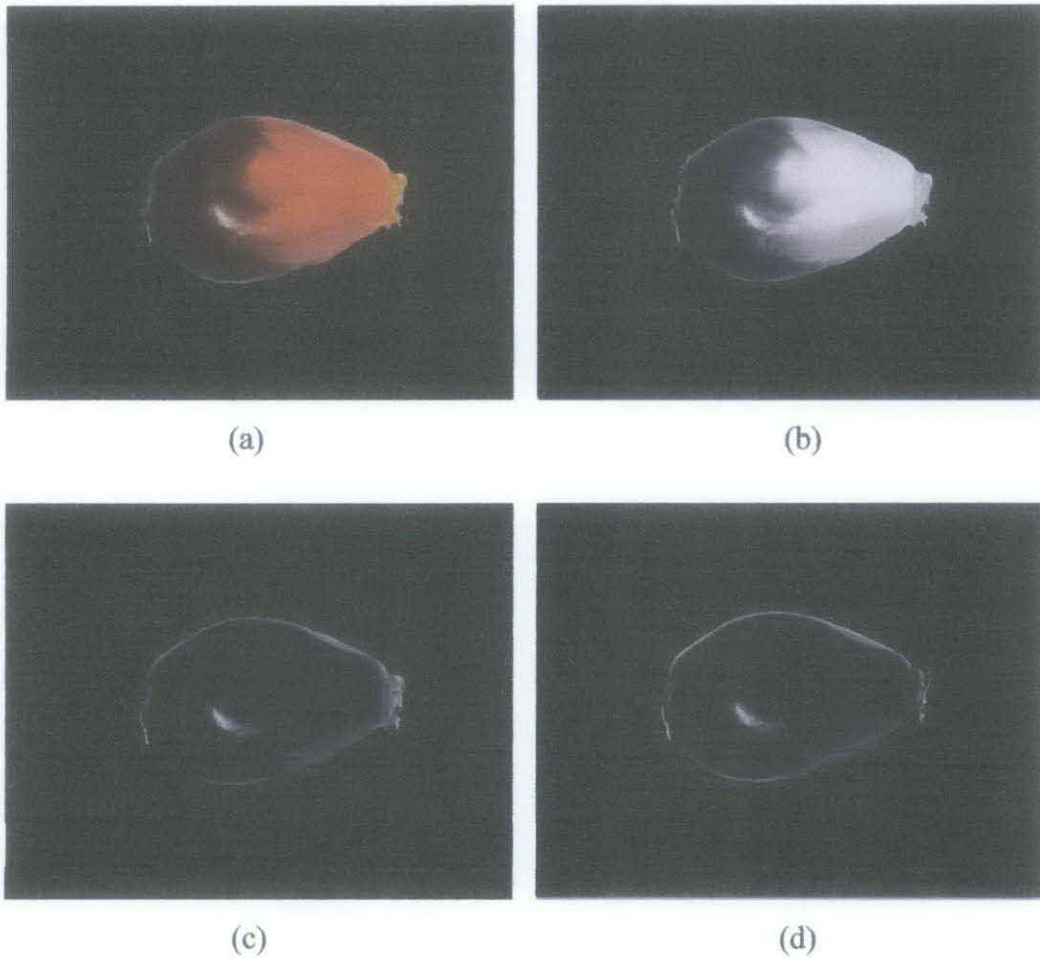


Figure 10: (a) Sample image for ripe category, (b) Red layer of ripe sample, (c) Green layer of ripe sample and (d) Blue layer of ripe sample

Figure 11(a-d) shows one sample from underripe category and the three separate layers which are red, green and blue layers respectively. In this case, red layer appears to have the brighter area on the right side of the fruit but slightly less than the sample from ripe category which means it has red pixel but less than ripe category. While for green and blue layer, the underripe category does not have high intensity of green and blue pixels.

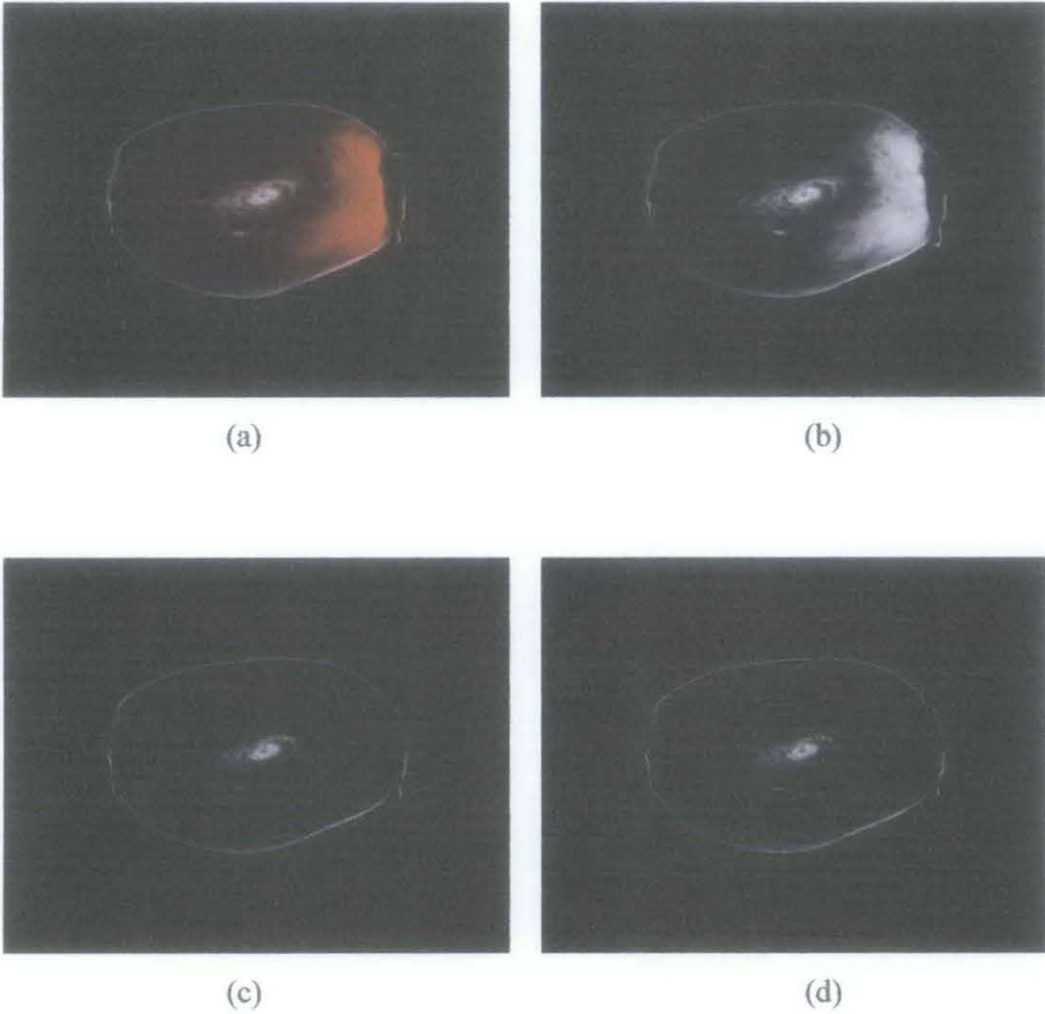


Figure 11: (a) Sample image for underripe category, (b) Red layer of underripe sample, (c) Green layer of underripe sample and (d) Blue layer of underripe sample

Figure 12(a-d) shows one sample from overripe category and the three separate layers which are red, green and blue layers respectively. For this case, red layer appears to have uniformly distributed over the whole fruit which means red pixel is the dominant color for the overripe category. While for green, it appears to have more pixels compared to blue layers.

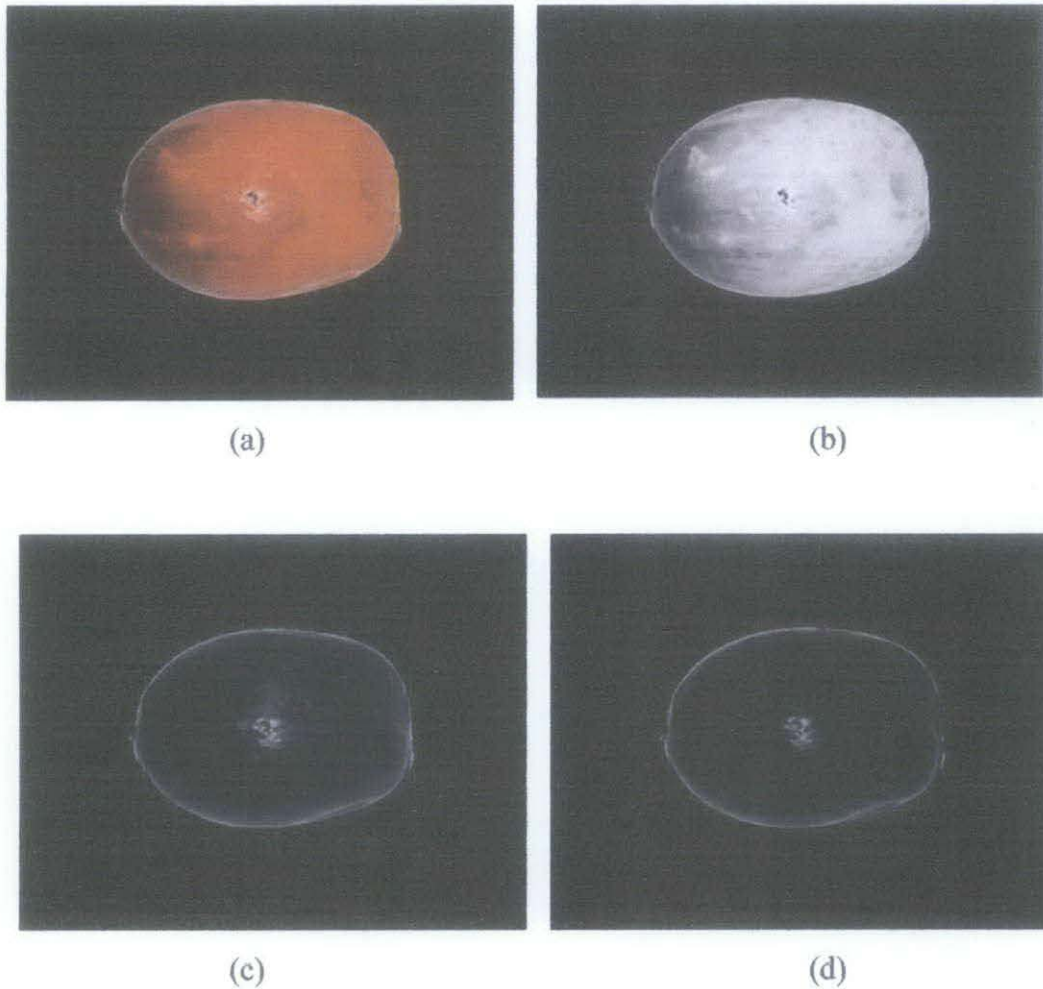


Figure 12: (a) Sample image for overripe category, (b) Red layer of overripe sample, (c) Green layer of overripe sample and (d) Blue layer of overripe sample

The mean RGB value for each category of the oil palm fruit sample maturity is tabulated in Table 3-5. Table 3 shows the RGB value for 20 samples of fruit from underripe category.

Table 3: The RGB value for 20 samples of fruit from underripe category

Samples	Red	Green	Blue
1	66.07297	27.5404	25.23904
2	81.95456	33.22463	25.19915
3	61.53014	22.44869	19.53323
4	66.34909	25.47982	19.93603
5	52.53006	26.51204	24.19147

6	69.67237	19.85053	15.26797
7	69.53163	28.70314	23.23422
8	65.17297	28.7404	26.33904
9	71.55456	32.82463	23.29915
10	63.53014	24.34869	20.98323
11	67.64909	23.97982	18.63603
12	54.53006	25.45204	23.23147
13	68.76163	26.99314	25.18422
14	62.77297	27.4404	23.83904
15	75.55456	31.52463	26.79915
16	66.53014	28.74869	23.88323
17	66.94909	24.12982	19.89603
18	55.58006	24.95204	23.94747
19	67.27337	21.65053	19.15397
20	68.82363	25.11984	26.88422
Mean	66.11615	26.48319	22.73386

Figure 13 shows the graph of the RGB value for 20 samples of fruit from underripe category. The range of color for each layer can be clearly seen from this graph.

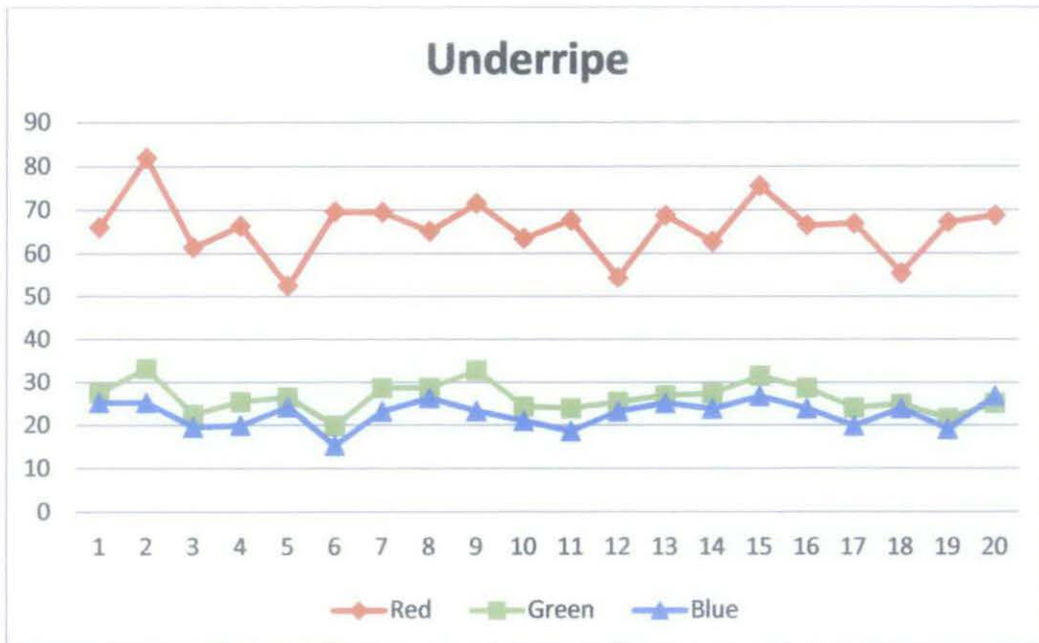


Figure 13: Graph of the RGB value for 20 samples of fruit from underripe category



Table 4 shows the RGB value for 20 samples of fruit from ripe category.

Table 4: The RGB value for 20 samples of fruit from ripe category

Samples	Red	Green	Blue
1	122.70217	53.03518	35.747
2	104.33397	44.62691	35.33436
3	106.88838	46.1473	32.39508
4	123.3934	52.41404	29.52857
5	101.14357	43.31923	40.45395
6	123.9201	58.35737	48.32685
7	110.85539	41.26137	33.81142
8	97.878099	45.15866	39.99622
9	102.52091	40.96407	27.82305
10	109.90977	44.32078	27.34902
11	106.83126	44.94613	39.92829
12	115.02606	45.49092	32.29389
13	93.816257	40.63406	33.80871
14	82.410545	41.64921	33.38578
15	127.05724	59.01403	35.73546
16	83.18044	39.26493	28.47809
17	94.994498	47.02371	40.30538
18	99.321155	40.70074	33.28778
19	119.39637	48.76313	31.67773
20	117.78262	44.36361	29.81631
Mean	107.16811	46.80127	34.47415

Figure 14 shows the graph of the RGB value for 20 samples of fruit from ripe category. The range of color for each layer can be clearly seen from this graph.

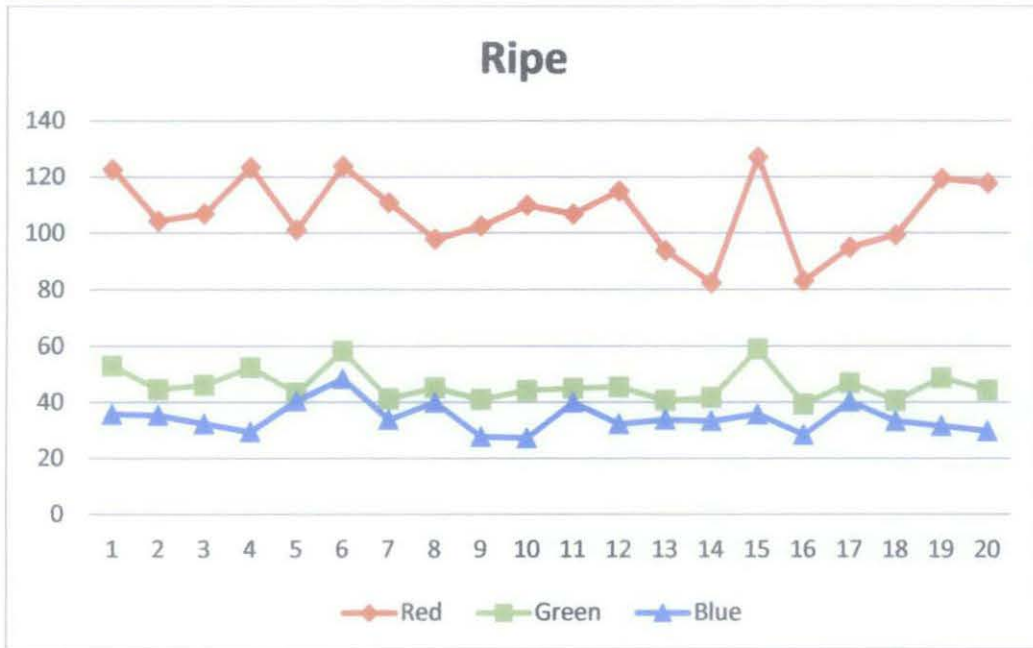


Figure 14: Graph of the RGB value for 20 samples of fruit from ripe category

Table 5 shows the RGB value for 20 samples of fruit from overripe category.

Table 5: The RGB value for 20 samples of fruit from overripe category

Samples	Red	Green	Blue
1	179.20884	70.39504	29.83546
2	161.5677	60.85167	37.40677
3	148.34364	49.7975	35.44774
4	168.20036	51.73259	29.33385
5	174.67484	64.68428	37.08261
6	169.95362	74.54857	49.2994
7	166.25002	55.09309	30.23311
8	158.55347	57.64539	33.61754
9	166.42453	60.41864	38.48266
10	156.82073	67.13824	42.4685
11	153.83068	61.34125	45.33802
12	150.66194	51.16196	45.33802
13	162.5677	59.85167	38.40677
14	149.34364	51.7975	36.44774
15	164.20036	48.73259	28.33385
16	176.47484	63.18428	36.58261
17	168.95362	72.54857	50.2994

18	167.25002	56.19309	32.43311
19	159.55347	56.54539	34.81754
20	166.88453	61.51864	37.98266
Mean	163.48593	59.75899	37.57706

Figure 15 shows the graph of RGB value for 20 samples of fruit from overripe category. The range of color for each layer can be clearly seen from this graph.

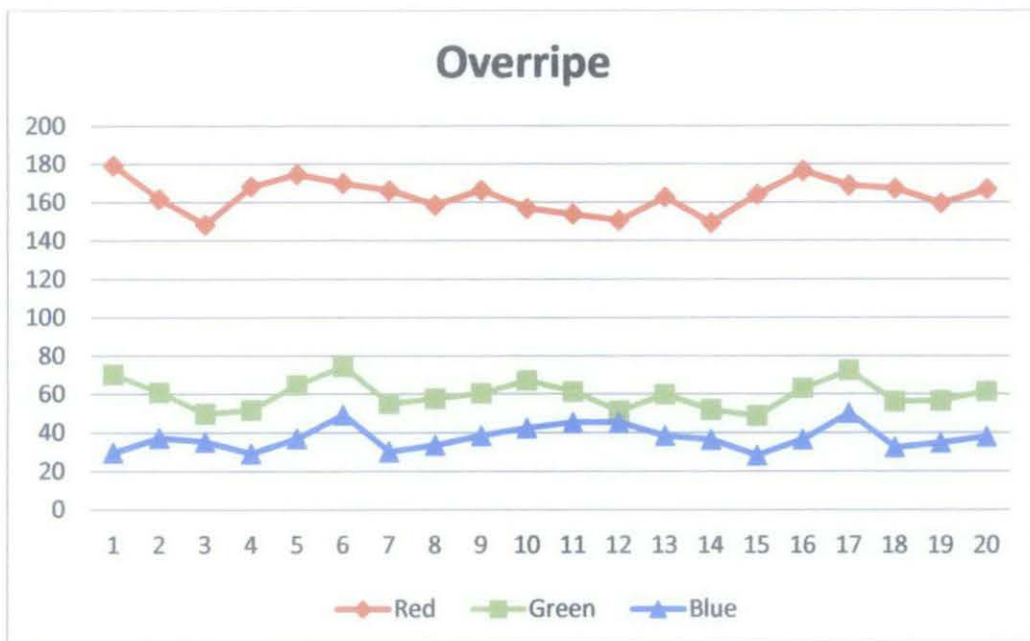


Figure 15: Graph of samples from overripe category

After obtaining the RGB intensity of the samples for each category, the values are compared with each other in order to calculate the suitable range of RGB intensity of the oil palm fruit for each category as shown in the Table 6.

Table 6: The range of RGB intensity of the sample of categories

Category	Red		Green		Blue	
	Min	Max	Min	Max	Min	Max
Ripe	82.42054	127.05724	39.26493	59.01403	27.34902	48.32685
Underripe	52.53006	81.95456	19.85053	33.22463	15.26797	26.88422
Overripe	148.34364	179.20884	48.73253	74.54856	28.33385	50.2994

## The Fuzzy System

The fuzzy system uses three inputs which are the average value of red, green and blue layers of the fruit. The fuzzy logic is depends on the set of rules that has been determined. The possibilities of each rule are calculated and the best answer is comes from output value of the system.

A simple plant is constructed in order to integrate between the graphical user interface and Fuzzy Logic Controller. Figure 16 shows the grading system plant which has three inputs namely meanR, meanG and meanB connected to the Fuzzy Logic Controller. The result from the Fuzzy Logic can be observed in the scope.

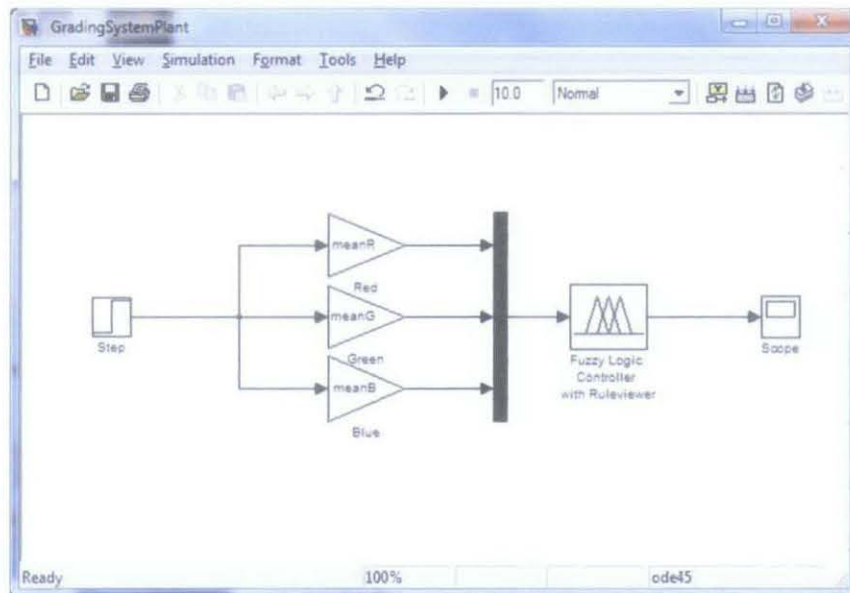


Figure 16: Grading system plant

Figure 17 shows the fuzzy inference system editor. It requires three inputs which are red, green and blue. Each input has its own range and it is based on the range value of each category.

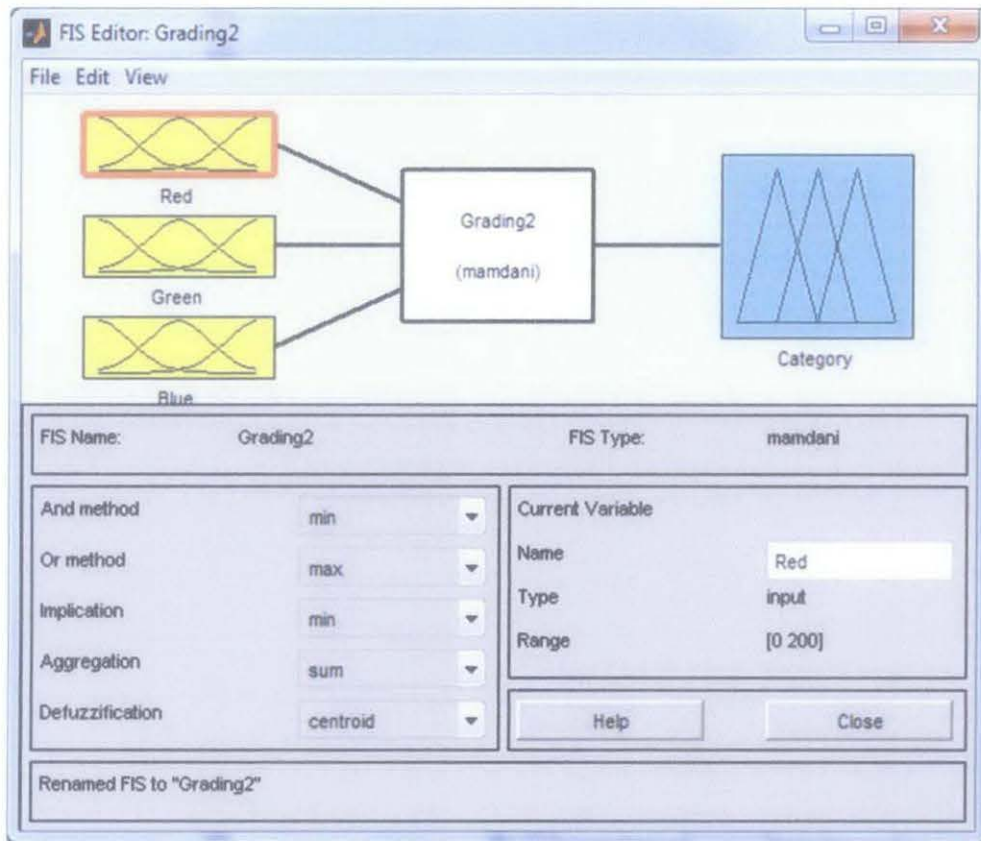


Figure 17: Fuzzy Inference System, FIS

Figure 18 shows the example of the fuzzy logic process. The system takes three inputs where red = 113, green = 48.7 and blue = 33.1. The inputs are fed into each rules and the contribution for each input is calculated.

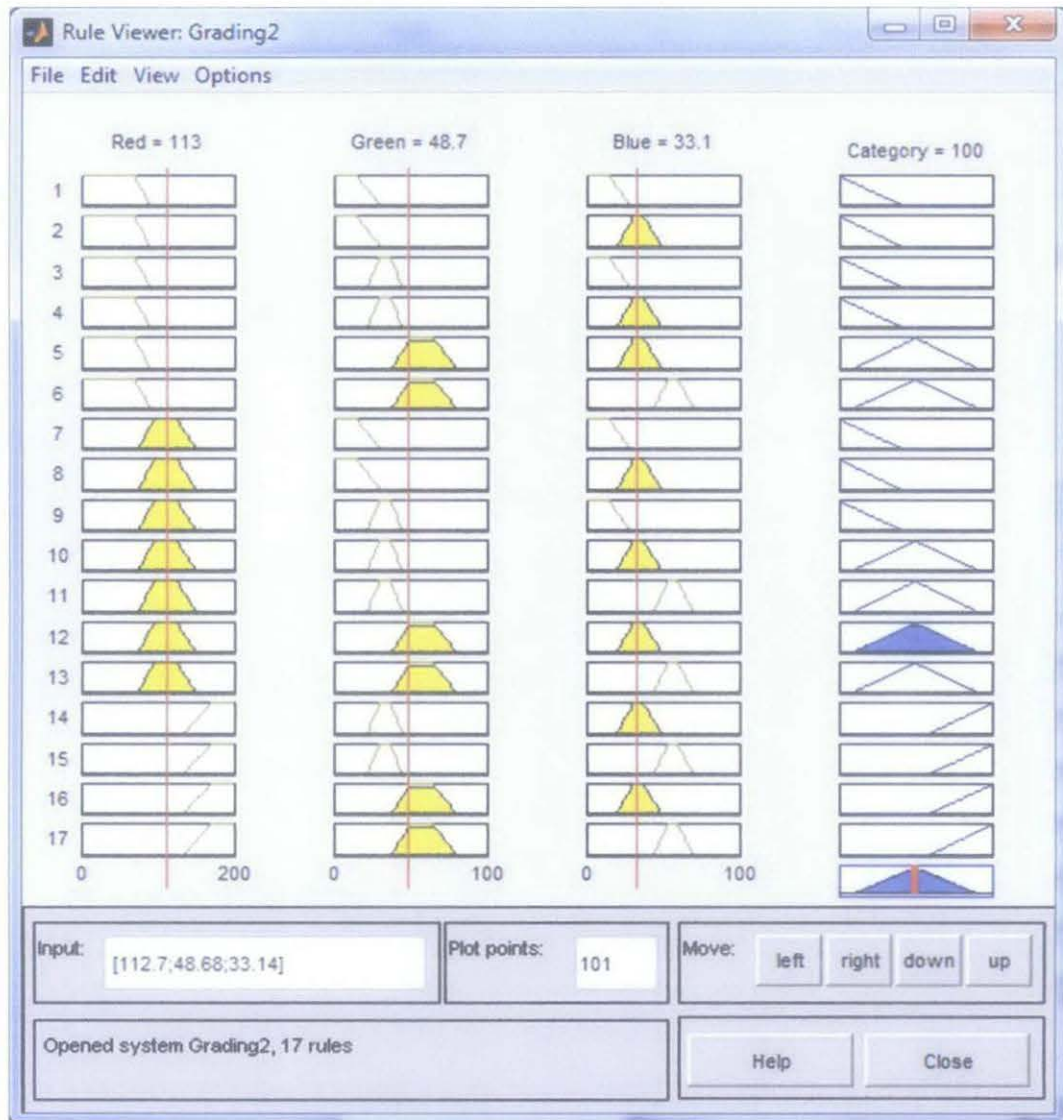


Figure 18: Rule viewer of the fuzzy system

In this case, the inputs satisfied Rule 12 where the conditions are red is medium, green is high and blue is medium. The last column indicates the end result of the system and the end result value can be obtained from the output Category. The classification of the oil palm fruit is determined based on the crisp logic given in Table 7.

Table 7: Fuzzification Algorithms

Defuzzification Output	Oil Palm Fruit Category
(Category $\leq 65$ )	Underripe
(Category $\geq 66$ ) && (Category $\leq 155$ )	Ripe
(Category $\geq 156$ )	Overripe

In this case, the Category value is 100. Therefore, the oil palm fruit used in this experiment is classified as ripe category.

The results of the grading system are evaluated against the human graders who have a lot of experience on the oil palm fruit. Five samples are chosen randomly from each category and the total is 15 samples.

Each sample is graded by the human graders as well as the automated grading system. It shows that the automated fuzzy grading system achieved 86.67% accuracy compared to the human graders. 13 samples are graded correctly while the remaining samples are wrong graded.

The reasoning behind is because the color of the fruit is lies between the boundaries of the two categories which result in difficulty for the system to graded correctly. But for the normal case, the system can perform the grading properly and gives the good results.

## 4.2 Discussion

Based on the result of range of RGB intensity of each category shown in Table 6, it is clear that red pixel play an important role in order to classify the oil palm fruit. The oil palm fruit is depends on the range value of red pixel. The average value of red pixel for ripe category is around 100 while for overripe category the red pixel is 30% more than ripe category and for underripe category it is less 30% compared to ripe category.

Furthermore, the green pixel for ripe category is around 40 to 80 but for underripe category it appears less green pixel while for overripe category it appears more or less the same like ripe category. Although this information is not very accurate but it is still considered as one of the parameter in order to determine the category of oil palm fruit.

Moreover, for the blue layer, the percentage is similar to all categories which mean all categories have the relatively similar intensity of blue pixel. But when it is compared with other layers like red and green, blue layer has the minimum amount of percentage over the whole fruit.

These three layers (R, G & B) can provide the useful information in order to set the range of the input value for the fuzzy logic system. The fuzzy logic use the Mamdani method because it is intuitive since the fuzzy rules are set based on the human knowledge. The Mamdani method also has a widespread acceptance and it has been used for many years with different application and it provides a good result[5][10].



## **CHAPTER 5**

### **CONCLUSION AND RECOMMENDATIONS**

#### **5.1 Conclusion**

As a conclusion, the grading system of oil palm fruit in the industry has been studied and the color for each category has been analyzed. The range values of RGB for each layer for each category are determined in this project.

The project was done by using RGB color model and one of the artificial intelligence methods which is Fuzzy Logic. In Fuzzy Logic, this project was done using Mamdani type with different type of membership function such as triangle, trapezium and Gaussian. However, the trapezium type of membership function gives a higher accuracy compared to other and it is later used in this project.

The three project objectives aimed for this project were successfully achieved. A new automated grading system using artificial intelligence equipped with graphical user interface was successfully developed. This system is user friendly and easy to use.

#### **5.2 Recommendations**

There are a few suggestions that can be recommended to improve the grading system. The grading system setup should have a consistence lighting setup to ensure the sample and the background is taken in the same condition with the same amount of light. The best way is to take the image under direct sunlight because it gives the actual color of the fruit. However, this is not the case since grading system using machine in done in the mill. It is crucial to select the proper lighting which can give the closest light condition as sunlight.

Even though the system did not get high accuracy classification on certain fruit which has a close boundary between two categories, but the system still give the correct classification on many normal case.

This project is only focused on RGB color model and Fuzzy Logic, there are several other color model that can be explored such as CMYK color model and CIE LAB color model. The CIE LAB color model is scientifically designed for computer calculation of color reproduction problems. There is also another artificial intelligent method like neural network which can perform the same goal for this project.

The combination of several techniques becomes popular or a preference because of the ability to improve the computation time which is very essential in any real time applications system and the ability to provide higher percentage accuracy.

## REFERENCES

- [1] Charles A. Poynton, "Digital Video and HDTV: Algorithm and Interfaces", Morgan Kaufmann, 2003.
- [2] Nicholas Boughen, "Lightwave 3d 7.5 Lighting", Wordware Publishing, Inc, 2003.
- [3] R. S. West, H. Konijin, S. Kuppens, N. Pfeffer, Q. V. Vader, Y. Martynov, T. Heemstra, J. Sanders, T. Tagi and G. Harbers, "LED backlight for large area LCD TV's", Proc. 10<sup>th</sup> Int. Disp. Works., Fukuoka, Japan, pp. 657-660, 2003.
- [3] Meftah Salem M. Alfatni, "Oil Palm Fruit Bunch Grading System Using Red, Green, and Blue Digital Number", University Putra Malaysia, Journal of Applied Sciences, Vol. 8, pp. 1444-1452, 2008.
- [5] Nursuriati Jamil, Azlinah Mohamed, Syazwani Abdullah, "Automated Grading of Palm Oil Fresh Bunches (FFB) using Neuro-Fuzzy Technique", Universiti Teknologi MARA, International Conference of Soft Computing and Pattern Recognition, 2009.
- [6] W.I. Wan Ismail, M.Z. Bardaie, and A.M. Abdul Hamid, "Optical properties for mechanical harvesting of oil palm FFB," J. Oil Palm Research , vol. 12, no. 2, pp. 38-45, Dec 2000.
- [7] A. Nureize , J. Watada, "Multi-criteria Fuzzy Regression Model for Evaluating Oil Palm Grading", Graduate School of Information, Production and System, Waseda University, 2-7 Hibikino, Wakamatsu, Kitakyushu, Japan, 2009.

- [8] M.Z Abdullah, L.C Guan and B.M.N Mohd Azemi, "Stepwise Discriminant Analysis For Colour Grading of Oil Palm Using Machine Vision System", University Science Malaysia, Institute of Chemical Engineers, Trans ICheme, Vol. 79, Part C, 2001.
- [9] G. RENNICK, Y. ATTIKIIOUZEL and A. ZAKNICH, "Machine Grading and Blemish Detection in Apples", University of Western Australia, Fifth International Symposium on Signal Processing and its Applications, ISSPA, 1999.
- [10] I. KAVDIR, D.E. GUYER, "Apple Grading Using Fuzzy Logic", Turk J Agric For 27, pp. 375-382, 2003.
- [11] Nur Badariah Ahmad Mustafa, Nurashikin Ahmad Fuad, Syed Khaleel Ahmed, Aidil Azwin Zainul Abidin and Zainul Abidin Md Sharif. "Image Processing of an Agriculture Produce: Determination of Size and Ripeness of a Banana", Universiti Tenaga Nasional, 2008.
- [12] Michael Recce and John Taylor, "High Speed Vision-Based Quality Grading of Oranges", Department of Anatomy and Developmental Biology, University College London, London, UK, 1996.
- [13] Nur Badariah Ahmad Mustafa, Nurashikin Ahmad Fuad, Syed Khaleel Ahmed, Aidil Azwin Zainul Abidin and Zainul Abidin Md Sharif. "Agriculture Produce Sorting and Grading using Support Vector Machines and Fuzzy Logic", Universiti Tenaga Nasional, IEEE International Conference on Signal and Image Processing Applications, 2009.
- [14] Antonio Carlos Loureiro Lino, Juliana Sanches, Inacio Maria Dal Fabbro, "Image Processing Techniques for Lemons and Tomatoes Classification", *Bragantia*, Campinas, v.67, n.3, p.785-789, 2008.

## APPENDIX A

### PROGRAM CODE

#### GradingSystemTraining.m

```
clear all;
close all;
clc

files = dir('*.jpg');
num_files = numel(files);
images = cell(1, num_files)
for m = 1:num_files
    images{m} = imread(files(m).name);
end

%% image processing part

for a = 1:num_files %1

data = images{a};

% image resolution
res = 640*480;

%convert rgb2gray
I = rgb2gray(data);

%Thresholding
for i=1:480
    for j=1:640
        if (I(i,j)>200)
            I(i,j)=0;
            res = res - 1;
        else
            I(i,j)=I(i,j);
        end
    end
end

end

%figure, imshow(I)

for i=1:480
    for j=1:640
        for k=1:3
            if(I(i,j)==0)
                data(i,j,k)=0;
            else
                data(i,j,k)=data(i,j,k);
            end
        end
    end
end
```

```

        end
    end
end

% separate into 3 layers (R,G,B)
R = data(:,:,1);
G = data(:,:,2);
B = data(:,:,3);

% calculate the average for each layer
meanR = sum(sum(R)) / res;
meanG = sum(sum(G)) / res;
meanB = sum(sum(B)) / res;

% display result
Result(a,1) = meanR;
Result(a,2) = meanG;
Result(a,3) = meanB;

end

% show image
figure, imshow(R)
figure, imshow(G)
figure, imshow(B)

%% show results

Result

Rmax = max(Result(:,1))
Rmin = min(Result(:,1))

Gmax = max(Result(:,2))
Gmin = min(Result(:,2))

Bmax = max(Result(:,3))
Bmin = min(Result(:,3))

```

## GRAPHICAL USER INTERFACE CODE

### GradingSystem.m

```
function varargout = GradingSystem(varargin)
% GRADINGSYSTEM M-file for GradingSystem.fig
%   GRADINGSYSTEM, by itself, creates a new GRADINGSYSTEM or
raises the existing
%   singleton*.
%
%   H = GRADINGSYSTEM returns the handle to a new GRADINGSYSTEM
or the handle to
%   the existing singleton*.
%
%   GRADINGSYSTEM('CALLBACK', hObject, eventData, handles,...)
calls the local
%   function named CALLBACK in GRADINGSYSTEM.M with the given
input arguments.
%
%   GRADINGSYSTEM('Property','Value',...) creates a new
GRADINGSYSTEM or raises the
%   existing singleton*. Starting from the left, property
value pairs are
%   applied to the GUI before myAdder_OpeningFunction gets
called. An
%   unrecognized property name or invalid value makes property
application
%   stop. All inputs are passed to GradingSystem_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 GradingSystem

% Last Modified by GUIDE v2.5 18-Apr-2011 08:16:32

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @GradingSystem_OpeningFcn,
                  ...
                  'gui_OutputFcn',  @GradingSystem_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 GradingSystem is made visible.
function GradingSystem_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 GradingSystem (see
VARARGIN)

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

% Update handles structure
guidata(hObject, handles);

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

% --- Outputs from this function are returned to the command line.
function varargout = GradingSystem_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;

function input1_editText_Callback(hObject, eventdata, handles)
% hObject    handle to input1_editText (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 input1_editText
as text
%         str2double(get(hObject,'String')) returns contents of
input1_editText as a double
input = str2num(get(hObject,'String'));
if (isempty(input))
    set(hObject,'String','0')
end
end

```



```

guidata(hObject, handles);

% --- Executes during object creation, after setting all
properties.
function input1_editText_CreateFcn(hObject, eventdata, handles)
% hObject    handle to input1_editText (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 input2_editText_Callback(hObject, eventdata, handles)
% hObject    handle to input2_editText (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 input2_editText
as text
%         str2double(get(hObject,'String')) returns contents of
input2_editText as a double
input = str2num(get(hObject,'String'));
if (isempty(input))
    set(hObject,'String','0')
end
guidata(hObject, handles);

% --- Executes during object creation, after setting all
properties.
function input2_editText_CreateFcn(hObject, eventdata, handles)
% hObject    handle to input2_editText (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 add_pushbutton.
function add_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to add_pushbutton (see GCBO)
% eventdata  reserved - to be defined in a future version of
MATLAB

```

```

% handles      structure with handles and user data (see GUIDATA)

a = get(handles.input1_editText, 'String');
b = get(handles.input2_editText, 'String');
% a and b are variables of Strings type, and need to be converted
% to variables of Number type before they can be added together

total = str2num(a) + str2num(b);
c = num2str(total);
% need to convert the answer back into String type to display it
set(handles.answer_staticText, 'String', c);
guidata(hObject, handles);

% --- Executes during object creation, after setting all
properties.
function input3_CreateFcn(hObject, eventdata, handles)
% hObject      handle to input3 (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 browse_image.
function browse_image_Callback(hObject, eventdata, handles)
% hObject      handle to browse_image (see GCBO)
% eventdata    reserved - to be defined in a future version of
MATLAB
% handles      structure with handles and user data (see GUIDATA)

% get a file to read
[FileName, FilePath] = uigetfile({'*.jpg'; '*.jpeg'; '*.bmp';
'*.gif'; '*.tiff'; '*.png' }, 'Select an image...');
I = imread([FilePath, FileName]);
setappdata(GradingSystem, 'MyImage', I);
setappdata(GradingSystem, 'FileNameAndPath', [FilePath,
FileName]);
%figure, imshow(I);

set(handles.file_name, 'String', [FilePath, FileName]);

% display image

```

```

Image = getappdata(GradingSystem, 'MyImage');
ppp = uipanel('Units', 'Pixels');
set(ppp, 'position', [50 190 500 400]);
ax1 = axes('parent',ppp, 'position', [0 0 1 1], 'Units',
'normalized');
him1 = image(Image, 'parent', ax1);

function file_name_Callback(hObject, eventdata, handles)
% hObject    handle to file_name (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 file_name as
text
%          str2double(get(hObject,'String')) returns contents of
file_name as a double

% --- Executes during object creation, after setting all
properties.
function file_name_CreateFcn(hObject, eventdata, handles)
% hObject    handle to file_name (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 red_value_Callback(hObject, eventdata, handles)
% hObject    handle to red_value (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 red_value as
text
%          str2double(get(hObject,'String')) returns contents of
red_value as a double

% --- Executes during object creation, after setting all
properties.
function red_value_CreateFcn(hObject, eventdata, handles)
% hObject    handle to red_value (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 green_value_Callback(hObject, eventdata, handles)
% hObject    handle to green_value (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 green_value as
text
%          str2double(get(hObject,'String')) returns contents of
green_value as a double

```

```

% --- Executes during object creation, after setting all
properties.
function green_value_CreateFcn(hObject, eventdata, handles)
% hObject    handle to green_value (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 blue_value_Callback(hObject, eventdata, handles)
% hObject    handle to blue_value (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 blue_value as
text
%          str2double(get(hObject,'String')) returns contents of
blue_value as a double

```

```

% --- Executes during object creation, after setting all
properties.

```

```

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

% get the image from workspace
Image = getappdata(GradingSystem, 'MyImage');

% image resolution
res = 640*480;

% convert rgb2gray
I = rgb2gray(Image);

%Thresholding
for i=1:480
    for j=1:640
        if (I(i,j)>185)
            I(i,j)=0;
            res = res - 1;
        else
            I(i,j)=I(i,j);
        end
    end
end

for i=1:480
    for j=1:640
        for k=1:3
            if(I(i,j)==0)
                Image(i,j,k)=0;
            else
                Image(i,j,k)=Image(i,j,k);
            end
        end
    end
end

% separate into 3 layers (R,G,B)
R = Image(:,:,1);

```

```

G = Image(:,:,2);
B = Image(:,:,3);

% calculate the average for each layer
meanR = sum(sum(R)) / res;
meanG = sum(sum(G)) / res;
meanB = sum(sum(B)) / res;

aR = num2str(meanR);
aG = num2str(meanG);
aB = num2str(meanB);

set(handles.red_value, 'String', aR);
set(handles.green_value, 'String', aG);
set(handles.blue_value, 'String', aB);

%set(handles.category_value, 'String', 'Overripe');

aR=str2num(get(red_value,'String'));
aG=str2num(get(green_value,'String'));
aB=str2num(get(blue_value,'String'));

options = simset('SrcWorkspace','current');
sim('GradingSystemPlant',[],options);

%guidata(hObject, handles);

function category_value_Callback(hObject, eventdata, handles)
% hObject    handle to category_value (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 category_value
as text
%         str2double(get(hObject,'String')) returns contents of
category_value as a double

% --- Executes during object creation, after setting all
properties.
function category_value_CreateFcn(hObject, eventdata, handles)
% hObject    handle to category_value (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

```

## simulateButton.m

```
%notice that this m-file is NOT a function because
%simulink only allows you to use variables that are within the
main
%workspace
clear all

%make the handles structures available to the main workspace
h =gcf;
handles = guidata(h);

%set the axes to which the data will be plotted to
%axes(handles.axes1);

% load fuzzy logic file
load Grading2.mat

% get the image from workspace
Image = getappdata(GradingSystem, 'MyImage');

% image resolution
res = 640*480;

% convert rgb2gray
I = rgb2gray(Image);

%Thresholding
for i=1:480
    for j=1:640
        if (I(i,j)>185)
            I(i,j)=0;
            res = res - 1;
        else
            I(i,j)=I(i,j);
        end
    end
end

for i=1:480
    for j=1:640
        for k=1:3
            if(I(i,j)==0)
                Image(i,j,k)=0;
            else
                Image(i,j,k)=Image(i,j,k);
            end
        end
    end
end

% separate into 3 layers (R,G,B)
R = Image(:,:,1);
G = Image(:,:,2);
B = Image(:,:,3);
```

```

% calculate the average for each layer
meanR = sum(sum(R)) / res;
meanG = sum(sum(G)) / res;
meanB = sum(sum(B)) / res;

aR = num2str(meanR);
aG = num2str(meanG);
aB = num2str(meanB);

set(handles.red_value, 'String', aR);
set(handles.green_value, 'String', aG);
set(handles.blue_value, 'String', aB);

%simulate the system
options = simset('SrcWorkspace','current');
sim('GradingSystemPlant', [], options);

% output
cat = max(CAT.signals.values);

if(cat<=65)
    {set(handles.category_value, 'String', 'Underripe');}

elseif(cat>=66 && cat<=155)
    {set(handles.category_value, 'String', 'Ripe');}

elseif(cat>=156)
    {set(handles.category_value, 'String', 'Overripe');}

end

```



**APPENDIX B**

**GANTT CHART**

<b>WEEK</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>11</b>	<b>12</b>	<b>13</b>	<b>14</b>
<b>TITLE</b>														
Select the topics from list														
Confirmation of the topic														
Proposal														
Preliminary Report														
Material Selection/Design														
Analysis														
<b>Model</b>														
Build up system														
Testing														
Finishing														
Final Touch Up														
Report/Posters														