

Development of Tomato Inspection and Grading System using Image Processing

N. M. Z. Hashim¹, N. H. Mohamad², Z. Zakaria³, H. Bakri⁴, F. Sakaguchi⁵

^{1,2,3}Faculty of Electronics & Computer Engineering, Universiti Teknikal Malaysia Melaka,
Hang Tuah Jaya, 76100 Durian Tunggal, Melaka Malaysia.

⁴Faculty of Mechanical Engineering, Universiti Teknikal Malaysia Melaka,
Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.

⁵Faculty of Engineering, University of Fukui,
3-9-1 Bunkyo, Fukui City, Fukui Prefecture, Japan. 910-8507.

nikzarifje@utem.edu.my¹

fieza_07@yahoo.com²

zahriladha@utem.edu.my³

hairul.bakri@utem.edu.my⁴

saka@dignet.fuee.fukui-u.ac.jp⁵

Abstract: Nowadays the food quality is become a major issue in health care. It is difficult to find the best food quality by ourselves in the market. This system was created to specify the quality of specific product. Quality is commonly related with product and it is very important to satisfy the customer's desire. The system is focus on the quality of food which has impact on our health. For this proposed system, a tomato is been used as the product that to be tested for food quality. This paper explained a technique for automatically detecting tomatoes skin surfaces in digital color images. The system describes two-step process which the first is detecting regions which are likely to contain tomatoes skin in the color images and then extracts information from these regions which might indicate the location of a tomato in the image. An inspection and grading system for tomato that has been loaded as an image and after that the image passed through the brightness process. After the image taken from webcam and already loaded onto the system then the system will process the both images (captured and input) by reading the color in each of pixel images. The processes of the images in this system needed to display the percentages value of color in order to classify the grade of tomato. Matlab software and its image processing toolbox have been used in images processing and analysis. As the result, Graphic User Interface (GUI) for tomatoes inspection and grading system by using Matlab software version 7.0.4 are achieved. For the future recommendation this system can be upgraded by adding the various types of fruits or vegetarians.

Keywords: Graphic User Interface (GUI), Image Detection, Image Processing, Matlab Software

1. Introduction

This project is to develop an inspection and grading system for tomatoes. A lot of tomato images are needed as an input to the system to build this project and save the images into the computer. The system will able to process the images and classify to a specific grade of tomatoes. Webcam is used to capture the tomatoes skin surface and send the image to the system by graphical user interface. The system will only operate when it received the captured images. Later it will make a comparison between the saved image and captured image to be classified into the respective tomato grades.

Hence, system will make a comparison by calculating the color percentage through tomatoes skin surface image into three grades. Three types of tomatoes grade are Grade A, Grade B and Grade C. Grade A is the best quality of tomatoes where it has the majority of red and striking orange color skin, Grade B is a good quality where it has the majority of orange

color while Grade C is a bad quality where it has the combination color of orange and brown. After the system verified the tomatoes skin color for each grade, the system will display both result of tomatoes image and type of grade on the screen.

The first literature review was about Simulator for Digital Simulation Techniques. The purpose of this project is to design a simulator for digital modulation techniques. This simulator can display the outputs for Amplitude-shift keying (ASK), Frequency-shift keying (FSK), Phase-shift keying (PSK) and Minimum-Shift Keying (MSK). The simulation built functioned as a platform for user to enter specific information (types of frequencies and types of modulations) that display the output automatically. This project can be an alternative program for user to run a digital simulation work effective and work efficient. This simulator was built with a Graphic User Interface (GUI) that helps users run the simulation easily. To make this project successful, another program must developed for purpose of data management because if users enter

information into the program, the information will be read and the types of output will be display. When this project is complete, it can be used as a trainer for student such as Communication Principle and Digital Communication Principles [1].

The second literature review was about Face Detection in Color Images. This project presents a technique for automatically detecting human faces in digital color images. The system relies on two step processes which first detects regions which are likely to contain human skin in the color image and then extracts information from these regions which might indicate the location of a face in the image. The skin detection is performed using a skin filter which relies on color and texture information. The face detection is performed on a grayscale image containing only the detected skin areas. A combination of thresh holding and mathematical morphology are used to extract object features that would indicate the presence of a face.

In this project, a prototype algorithm for automating the detection of human faces in digital photographs was developed and can serve as an introduction for future work in detecting people in images. Several systems designed for the purpose of finding people or faces in images have already been proposed by numerous research groups. Some of these programs, such as the Rowley, Baluja, and Kanade system developed at Carnegie Mellon, rely on training of a neural network and computing distance measures between training sets to detect a face.

Other software packages exist which can recognize facial features in pictures known to contain a human face somewhere in the image. This project focused on face detection in arbitrary color images and differs from the first type of system in that it relies on a combination of color and grayscale information. Additionally, it does not require the time consuming process of training a neural net or computing distance measures between every possible region in the image. The developed system also differs from those software packages that recognize facial features because, in this scenario, the task is to detect a facial region in an arbitrary image, and not to analyze images known to contain a face.

The process for detection of faces in this project was based on a two-step approach. First, the image is filtered so that only regions likely to contain human skin are marked. This filter was designed using basic mathematical and image processing functions in Matlab and was based on the skin filter designed for the Berkeley-Iowa Naked People Finder. Modifications to the filter algorithm were made to offer subjective improvement to the output. The second stage involves taking the marked skin regions and removing the darkest and brightest regions from the map. The removed regions have been shown through empirical tests to correspond to those regions in faces which are usually the eyes and eyebrows, nostrils, and mouth. By performing several basic image analysis techniques, the regions with "holes" created by the thresh holding can be considered likely to be faces. This second stage was a combination of Khoros visual programming and Matlab functions. The entire system was entirely automated and required no user intervention save for indicating the correct file names to be processed at each stage. While not implemented in this project, a more advanced program could implement a third step to

discriminate between whole sizes and spatial relationships to make an even more robust detection system [2].

The third literature review was about Measurements of Trash Contents and Grades in Cotton using Digital Images Analysis. The paper presents the measurements of trash contents and grades in cotton using digital processing and analysis. A measuring system has been constructed and transmitted, reflected and compound light measurement mode has been tested, discussed and the optimum mode identified. A formula has been derived and used to translate the trash content by area into the trash content by mass. Matlab and its image processing toolbox have been used in images processing and analysis.

Existing standardized methods and means for measurement of trash content in cotton are mainly based on mechanical separation of the trash from cotton fibers using the Shirley Analyzer' and weighing the trash relative to the total mass of the cotton sample. Currently, the determination of cotton grades still depends on the subjective assessment of the cotton classers. These methods are time consuming, labor-intensive and therefore expensive. Furthermore, they are not suited to on-line or field measurements. Attempts were made by Lieberman and Patil' using computer vision and pattern recognition techniques to discriminate among three trash categories; bark, stick, and leaf pepper. However, the Lieberman and Patil's approach was not a feasible way of assessing the trash content and accordingly it was not successful in assessing the grades of cotton [3], [4].

The three systems that have been explained earlier used images digitization and processing method which quite similar with this project. From the systems, the similarity about the system function has been analyzed. The first review is about simulator for digital simulation techniques. The project used the Graphical User Interface by using Matlab to run and display the output of simulation. The project has many similarities with proposed tomatoes inspection and grading technique which also apply image processing technique.

The second review about face detection in color images which implement image processing technique. Although the project of tomatoes inspection and grading system does not contain face detection technique, the journal gave me a glimpse of view on the current technology of image processing. The project focused on face detection in arbitrary color images. For the first the project used mathematical and image processing functions in Matlab to filter the images. Second involves taking the marked skin regions and removing the darkest and brightest regions from the map. By performing several basic image analysis techniques, the regions with "es" created by the thresh holding can be considered likely to be faces. This second stage was a combination of Khoros visual programming and Matlab functions.

The third review is about cotton grading process which also involved image processing technique to grade the cotton. The project applied pattern recognition techniques and computer vision to differentiate the cotton quality. Image processing technique is used to discriminate among three trash categories; bark, stick, and leaf pepper. However, the project was not a feasible way of assessing the trash content and accordingly it was not successful in assessing the grades of cotton because of certain limitation.

In tomato inspection and grading system, the focus will be more on skin color of tomato surfaces. Firstly the project will load the input images (original images) onto the system. Second the brightness of the images can be adjusted in order to makes the images more clearly. Third the system will process both images between input images and captured images when receives the images from the webcam. The mathematical and images processing function in Matlab is used to process the images. Lastly the system will display the grade of tomatoes after the process of images was completed.

The objectives of the project are to develop an inspection and grading system for tomato, to process the image of tomato and classify a specific grade, to create automatic system inspection to replace manual system inspection and produce a great system for sauce manufacturing.

2. Methodology

2.1. System Overview

This tomato inspection and grading system mainly employs a simple input device and software. Since the software is executed on a computer, a webcam is used for capturing an image. Matlab software is chosen for the software development, as it provides a suitable GUI [5], [6], [7], [8]. The image taken from the m-files then will be digitized as 24-bit RGB data. After that the RGB data will be processed and analyzed. The tomato grade will be displayed once the process is complete. The methodology for carrying out the project is represented in the form of a flow chart below. The Figure 1 shows the flow of the entire project.

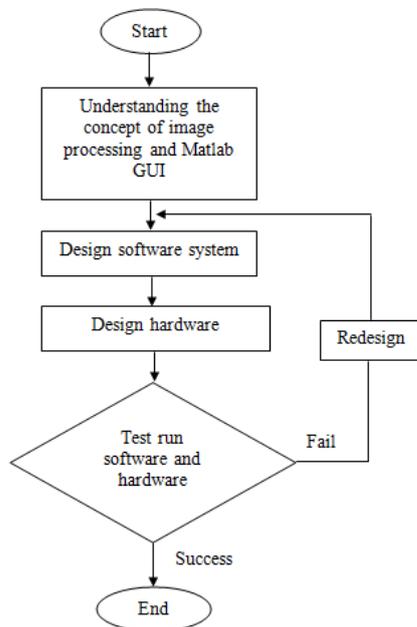


Figure 1: Overall Project Flow chart

2.2. Method to Process Images

Image processing is the manipulation and analysis of images by a computer. By applying image processing concept to this project it will be easier to read the image. This system will use a lot of digital image to operate and needs to read an image of pixel color in other to gain the best result to classify the types of grade [7].

2.2.1. Image Detection

Generally, detection refers to identifying a location or identifying and registering components of a particular object class at various levels detail, which is the foundation of image recognition. In this project, the HSV color space has been used, instead of the RGB space. According to that model, Hue dimension represents the “color”, Saturation dimension represents the dominance of that color and the Value dimension represents the brightness. Therefore, the color detection algorithm can search in terms of color position and color “purity”, instead of searching for specific RGB values. In order to detect the desired color, it is needed to define a tolerance in each HSV dimension.

The returned value (HSV) is an $M \times 3$ matrix (M is the number of images), each row of that matrix corresponds to the median HSV value of the selected areas in the respective image.

2.2.2. Images Composes

Each image is composed of an array of $M \times N$ pixels (contraction of “picture element”) with M rows and N columns of pixels [10]. Each pixel contains a certain value for red, green and blue. Varying these values for red, green, blue (RGB) user can get almost any color.

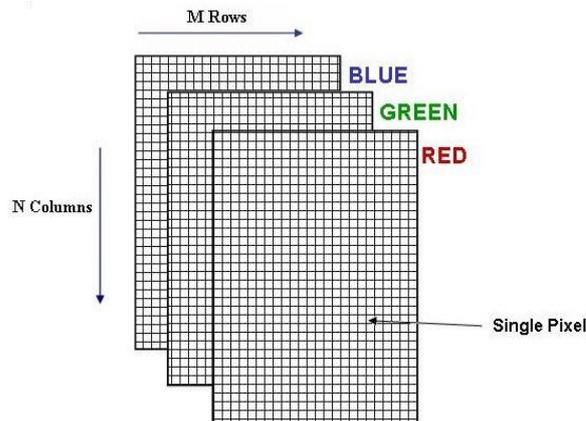


Figure 2: Image storage in Matlab [10]

2.2.3. Color maps

MATLAB uses a numerical array with three columns to represent color values [3]. This array is called a colormap, and each row in the matrix represents an individual color by using numbers in the range 0 to 1. The numbers in each row indicate the intensity of red, green and blue those make up a specific color. The following table illustrates the correspondence between numerical values in colormap and colors as in Table 1.

Table 1: Correspondence between numerical values in colormap and colors

Red	Green	Blue	Color
1	0	0	Red
0	1	0	Green
0	0	1	Blue
1	1	0	Yellow
1	0	1	Magenta
0	1	1	Cyan
0	0	0	Black
1	1	1	White
0.5	0.5	0.5	Medium gray
0.67	0	1	Violet

1	0.4	0	Orange
0.5	0	0	Dark red
0	0.5	0	Dark green

The first column in a color map is the intensity of red, the second column is the intensity of green and the third column is the intensity of blue. Colormap values are restricted to the range from 0 to 1.

A *colormap* is a sequence of rows containing red-green-blue (RGB) values that vary in some prescribed way from the first row to the last. Matlab provides a number of predefined *colormaps* as shown in the following table.

Table 2: Number of predefined *colormaps*

Color map Function	Description
Hsv	Hue-saturation-value color map, begin and end with red
Jet	Variant of hsv that starts with blue and end with red
Hot	Black to red to yellow to white
Cool	Shades of cyan and magenta
Summer	Shades of green and yellow
Autumn	Shades of red and yellow
Winter	Shades of blue and green
Spring	Shades of magenta and yellow
White	All white
Gray	Linear gray scale
Bone	Gray with a tinge of blue
Pink	Pastel shades of pink
Copper	Linear copper tone
Prism	Alternating red, orange, yellow, green, blue and violet
Flag	Alternating red, white, blue and black
Lines	Alternating plot line color
Color cube	Enhanced color cube

By default, each of these colormaps generates a 64-by-3 array specifying the RGB descriptions of 64 colors. Each of these functions accepts an argument specifying the number of rows to be generated. For example, hot (m) generated an m-by-3 matrix containing the RGB values of colors ranging from black, through shades of red, orange and yellow to white.

2.2.4. Displaying Colors

The number of bits per screen pixel determines the display's screen bit depth [12]. The screen bit depth determines the screen color resolutions, which are how many distinct colors the display can produce. Most computer displays use 8, 16, or 24 bits per screen pixel. Depending on our system, user might be able to choose the screen bit depth user want to use. In general, 24-bit display mode produces the best results. If user need to use a lower screen bit depth, 16-bit is generally preferable to 8-bit. The integer Matlab returns represents the number of bits per screen pixel as in Table 3.

Table 3: Displaying color

Value	Screen Bit Depth
8	8-bit displays support 256 colors. An 8-bit display can produce any of the colors available on a 24-bit display, but only 256 distinct colors can appear at one time. (There are 256 shades of gray available, but if all 256 shades of gray are used, they take up all the available color slots.)
16	16-bit displays usually use 5 bits for each color component, resulting in 32 (i.e., 2^5) levels each of red, green, and blue.

	This supports 32,768 (i.e., 2^{15}) distinct colors (of which 32 are shades of gray). Some systems use the extra bit to increase the number of levels of green that can be displayed. In this case, the number of different colors supported by a 16-bit display is actually 64,536 (i.e. 2^{16}).
24	24-bit displays use 8 bits for each of the three color components, resulting in 256 (i.e., 2^8) levels each of red, green, and blue. This supports 16,777,216 (i.e., 2^{24}) different colors. (Of these colors, 256 are shades of gray. Shades of gray occur where R=G=B.) The 16 million possible colors supported by 24-bit display can render a lifelike image.
32	32-bit displays use 24 bits to store color information and use the remaining 8 bits to store transparency data (alpha channel).

Regardless of the number of colors our system can display, Matlab can store and process images with very high bit depths: 224 colors for uint8 RGB images, 248 colors for uint16 RGB images, and 2159 for double RGB images. These images are displayed best on systems with 24-bit color, but usually look fine on 16-bit systems as well.

2.3. Acquiring Images from the Webcam in Matlab

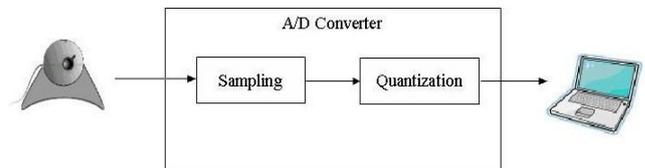


Figure 3: Acquiring images from the webcam in Matlab

For Matlab to recognize the video camera it needs to create it as an object using the command `obj=videoinput('winvideo')[11]`. Matlab will automatically find the webcam connected to computer. Once it is an object in workspace settings can be edited, such as the number of frames per second, to optimize it for this project. `Preview()` and `getframe()` are two useful functions for determining if the camera has been positioned properly. The first allows us to see what the camera sees, without collecting any data from it, and the second acquires a single snapshot and stores it as in image.

2.4. Calculation of Image Color

The process of performing the color inspection is carried out in Matlab [1]. The image captured by the webcam is first being dithered or reduced to only 4 colors. The dithering process increases the apparent number of colors in an image. This dithered image is then cropped into a matrix of 16×16 . The RGB of each cropped portion are calculated. The mean value for each portion is then calculated in order to obtain the final result on the object inspected. The cropping is expressed in a matrix form as shown in equation (1).

$$RGB_{i,j} = \begin{bmatrix} RGB_{1,1} & RGB_{1,2} & \dots & RGB_{1,16} \\ RGB_{2,1} & RGB_{2,2} & \dots & RGB_{2,16} \\ \dots & \dots & \dots & \dots \\ RGB_{16,1} & RGB_{16,2} & \dots & RGB_{16,16} \end{bmatrix} \quad (1)$$

The RGB mean values for the cropped portion can be calculated using equations (2), (3), and (4).

$$\bar{R} = \frac{\sum_{j=73i=53}^{88} \sum_{i=53}^{68} R_{i,j}}{256} \quad (2)$$

$$\bar{G} = \frac{\sum_{j=73i=53}^{88} \sum_{i=53}^{68} G_{i,j}}{256} \quad (3)$$

$$\bar{B} = \frac{\sum_{j=73i=53}^{88} \sum_{i=53}^{68} B_{i,j}}{256} \quad (4)$$

Color quantization is then performed to obtain the color range for the color of the object. This process reduces the number of colors presented in an image. The last part of the system operation is the conversion of the calculated RGB mean values in other to appear the percentage of color. The conversion is performed through equation (5), (6) and (7), where R, G and B represent the mean value of the three primary colors of RGB in hexadecimal.

$$\frac{R}{16} = X + \frac{Y}{16} \quad (5)$$

$$\frac{G}{16} = X + \frac{Y}{16} \quad (6)$$

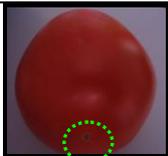
$$\frac{B}{16} = X + \frac{Y}{16} \quad (7)$$

X is the whole number, 0-15; and Y is the remainder represented as a whole number (0-15) over 16 [1].

2.5. Types of Tomato Skin Surface

To classify the tomatoes grade, a lot of tomatoes images were needed as an input to the system. Captured tomatoes images must be saved in the personal computer (PC) because these images will be used in the system to compare and match the both image; captured and original image through their skin surface to being classify the grade of tomatoes. The image below shows that the tomatoes skin surface for the grade A, B and C and specification of each grade [13], [14], [15].

Table 4: Category of tomato skin surfaces

GRADE A	GRADE B	GRADE C
		
The image above show that the tomato skin surface for grade A. The skin surface of this image very smooth and the color of its surface is striking red.	The image shows that the tomato skin surface for grade B. The skin surface of this image also very smooth but the color of its surface different from grade A the color is red orange striking.	The image shows that the tomato skin surface for grade C. The skin surface of that image not smooth it has some defect mark and so many scratches mark on its surface. The color of this image same like grade A and also grade B.
The specification of tomatoes grade A is more than 80% red color.	The specification of tomatoes grade B is more than 80% orange color	The specification of tomatoes grade C is less than 80% red and orange color

2.6. GUI Creation Fundamentals

The fundamentals steps of GUI creation as follows:

- Understand what users want the GUI to do. This is often the most difficult step. In many cases, what users want the GUI to do changes as new ideas and options are discovered during GUI construction.
- Sketch the GUI layout on a piece of paper. This often skipped step saves time in the long run, because iterating hand sketches of potential GUI layouts on paper is much faster than building and rebuilding GUI prototypes in Matlab.
- On the basic of the final sketch, create the GUI using UI objects. This step can be done by directly entering the creation code in to an M-file. Alternatively, the Matlab graphical user interface development environment (GUIDE) can be used.
- Create the Matlab code that is executed when a user interacts with the GUI. The callback code is most often contained in the same M-file as the code that creates the GUI.
- Test and debug the GUI. In particular, if the GUI will be used by others, try interacting with the GUI in ways that users never intended, yet in ways a less knowledgeable user may actually use it [5], [16], [17], [18].

2.7. System Development Flowchart

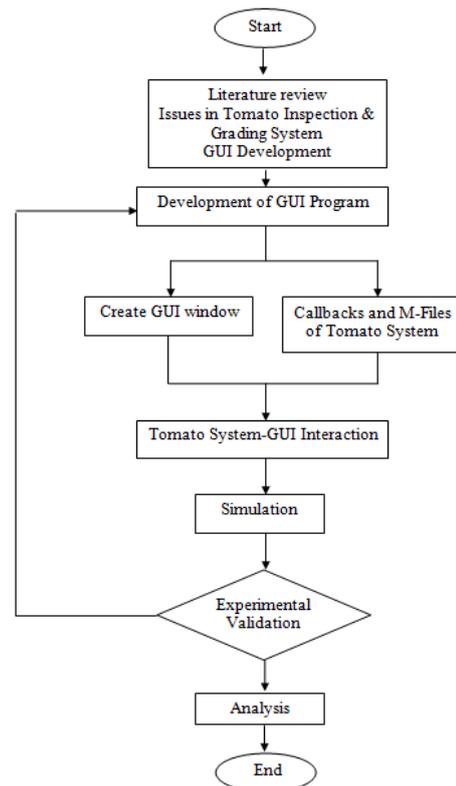


Figure 4: System development flowcharts

3. Results/ Findings and Discussion

At the end of the project, this system is complete and function same as the planning.

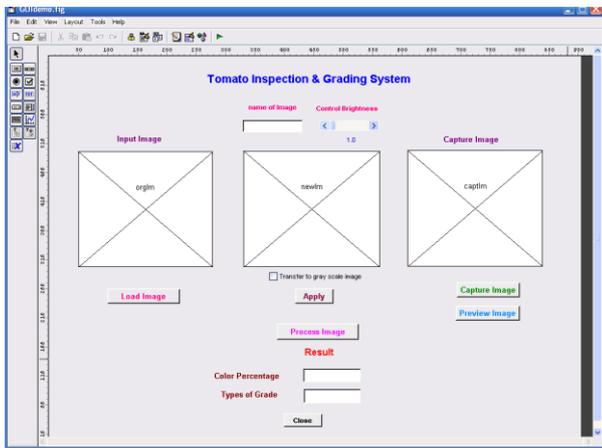


Figure 5: GUI layout

The figure below represents the input image through the process control brightness. The process control brightness is needed in this system because the input image will be used as a reference image during making the comparison with captured image. So when the reference image is dark or image blur it can be clear when user setting the brightness value of the images.

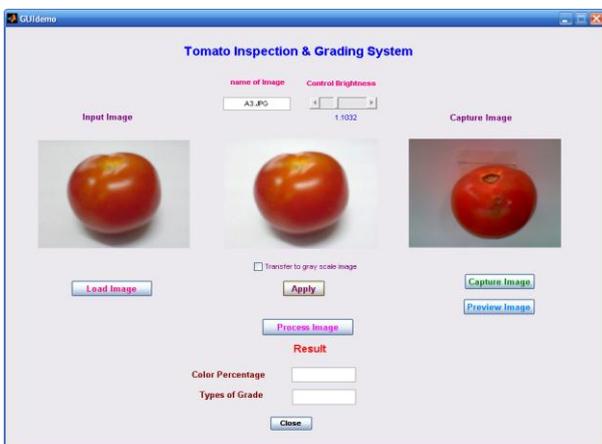


Figure 5: The brightness of input image

This diagram below shows the process of brightness and contrast adjustment. The moving the left or right changes the display brightness. This diagram explained how the control brightness button functions.

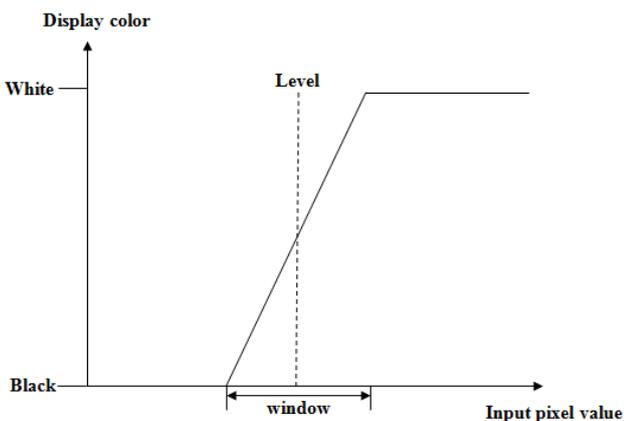


Figure 6: Diagram of process brightness and contrast adjustment

The figure below represents the image of tomato grade A after user clicked the process image button. The process image button executed the comparison between two image of surface area by calculate both color and then matching that color from database. From the observation based on figure below, the surface of captured image is good surface area and red color is more than 80% so that can be declared the image is tomato grade A.

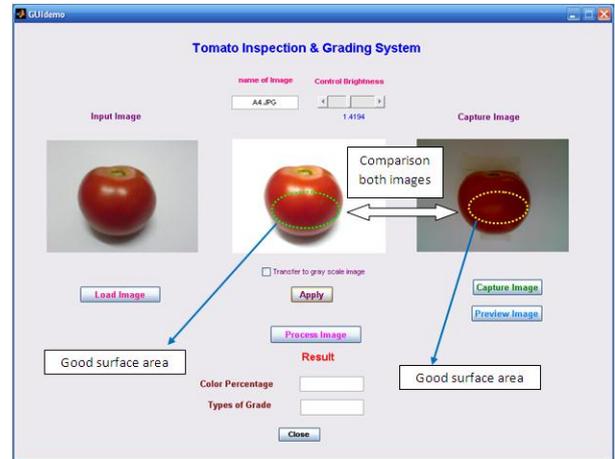


Figure 7: Tomato grade A

The figure below represents the image of tomato grade B after user clicked the process image button. The process image button executed the comparison between two image of surface area by calculate both color and then matching that color from database. From the observation based on figure below, the surface of captured image is good surface area but 70% orange color more than red color so that can be declared the image is tomato grade B.

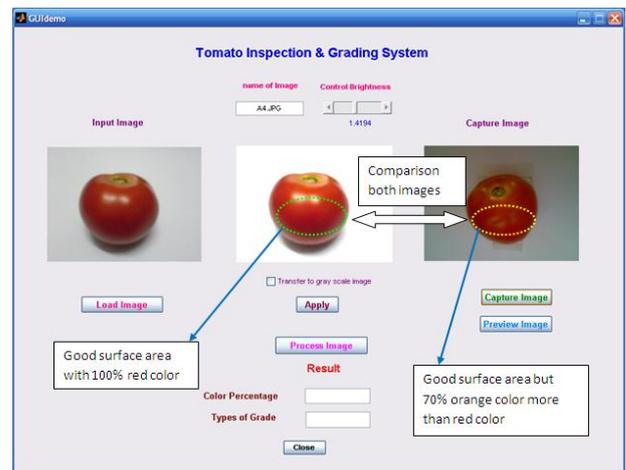


Figure 8: Tomato grade B

The figure below represents the image of tomato grade C after user clicked the process image button. The process image button executed the comparison between two image of surface area by calculate both color and then matching that color from database. From the observation based on figure below, the surface of captured image is bad area and has a lot of scratches marks. Then when the system compares with input image the percentage of red color is less than 60% so that can be declared the image is tomato grade C.



Figure 9: Tomato grade C

In order to develop the project, the project is divided into two categories which are software and hardware. The software to be used must suitable with the operation, thus the research need to be involve on the software concept first before making any decision to select software can be made. For the hardware, the material was indentified to build the project as it is important because both software and hardware need to be fully understood so that the interface can be determined whether it is suitable or not to be used in this project.

Before building the tomato inspection and grading system, the first step need to be done is to understand the whole project's concept. After the entire basic concept is understood, the research about the project is initiated to find the information and coding example that can be implemented to create the system.

4. Conclusions

As a conclusion this project is almost fully completed successfully. The entire GUI is functioned correctly however the improvement in the phase three which is comparison between both images to classify into respective tomato grades need to be improved. For the future recommendation this system can be upgraded by adding the multi types of fruits or vegetables. The GUI layout which can be added by developed the histogram of Red, Green, and Blue (RGB) colors in other to calculate the surface area of the input image and captured image and displays that values in both decimal and hexadecimal to the final result also recommended as the future work. By adding the GUI layout, the result is become clearer and detail for the user.

5. Acknowledgement

We are grateful to Universiti Teknikal Malaysia Melaka (UTeM) for their kind help for supplying the electronic components and giving their laboratory facility to complete this study.

References

[1] Ling Mei Chan, Rodney Tan, Gilbert Thio, "Design of Visual-based Color Classification System" , University College Sedaya International (UCSI), Research papers, January 2007.

[2] G. Hippleheuser, B. Marek, B. McMeel, C. Motsinger, T. Ngo, "Camera Controls Group Electrical Engineering Senior Design", final report, May 6 2008.

[3] S. Laykin, V. Alchanatis and Y. Edan, Image Processing, Algorithms for Tomatoes Classification, the XIV Memorial CIGR, World Congress 2000, Tsukuba, Japan.

[4] Richard W. Conners, D. Earl Kline and Philip A. Araman, Parts Color Matching Scanner for Edge Gluing, Hardwood Symposium Proceedings, 8-11 May 1996, National Hardwood Lumber Association, Memphis

[5] Patrick Marchard and O.Thomas Holland (2003). Graphics and GUIs with Matlab third Edition. Chapman & Hall/CRC

[6] Duane Hanselman and Bruce Littlefield (2005). Mastering Matlab 7 International Edition. Prentice Hall

[7] Rafael C. Gonzalez and Richard E. Woods (2002). Digital Images Processing Second Edition. Prentice Hall

[8] www.mathwork.com/matlab

[9] <http://www.mini.pw.edu.pl/~kotowski/Grafika/Images/Ind ex.html>

[10] <http://cnx.org/content/m15696/latest/>

[11] <http://madan.wordpress.com/2007/03/23/image-capture-using-webcam-in-matlab/>

[12] <http://www.mathworks.com/access/helpdesk/help/toolbox/images/index.html?/access/helpdesk/help/toolbox/images/f8-12781.html&http://www.google.com.my/search?q=Displaying+Colors&btnG=Search&hl=en&sa=2>

[13] N. M. Z. Hashim, S. H. Husin, A. S. Ja'afar, N. A. A. Hamid, "Smart Wiper Control System", International Journal of Application or Innovation in Engineering & Management (IJAIEM) Volume 2 Issue 7, pp. 409-415, 2013.

[14] N. M. Z. Hashim, A. S. Jaafar, N. A. Ali, L. Salahuddin, N. R. Mohamad, "Traffic Light Control System for Emergency Vehicles Using Radio Frequency", IOSR Journal of Engineering (IOSRJEN) Volume 3 Issue 7, pp. 43-52, 2013.

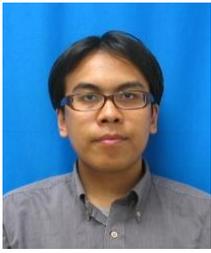
[15] N. M. Z. Hashim, M. S. Sizali, "Wireless Patient Monitoring System", International Journal of Science and Research (IJSR) Volume 2 Issue 8, pp. 250-255, 2013.

[16] N. M. Z. Hashim, N. A. Ali, A. S. Jaafar , N. R. Mohamad, L. Salahuddin, N. A. Ishak, "Smart Ordering System via Bluetooth", International Journal of Computer Trends and Technology (IJCTT) Volume 4 Issue 7, pp. 2253-2256, 2013.

[17] N. M. Z. Hashim, S. N. K. S. Mohamed, "Development of Student Information System", International Journal of Science and Research (IJSR) Volume 2 Issue 8, pp. 256-260, 2013.

[18] N. M. Z. Hashim, N. A. M. M. Arifin, "Laboratory Inventory System", International Journal of Science and Research (IJSR) Volume 2 Issue 8, pp. 261-264, 2013.

Author Profile



N. M. Z Hashim received the B.Eng. and M.Eng. degree in Electrical and Electronics Engineering from University of Fukui, Japan in 2006 and 2008, respectively. From 2008, he worked as Lecturer in Faculty of Electronics and Computer Engineering, Universiti Teknikal Malaysia Melaka (UTeM), Malaysia. He is acting the Head of Department of Computer Engineering

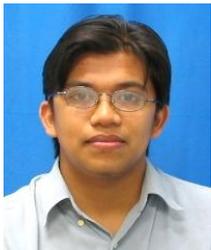
Department. His works are in Signal and Image Processing, Wavelet Transformation, Communication and Electronic Engineering. He joined Institute of Electrical and Electronic Engineers (IEEE), Board of Engineering Malaysia (BEM), Institute of Engineering Malaysia (IEM), Society of Photo-Optical Instrumentation Engineers (SPIE), International Association of Computer Science and Information Technology (IACSIT) and International Association of Engineers (IAENG) as member. (Email: nikzarif@utem.edu.my).

N. H. Mohamad received his B. Eng in Electronic Engineering (Computer Engineering) from Universiti Teknikal Malaysia Melaka, Malaysia. (Email: fieza_07@yahoo.com).



Z. Zakaria, PhD, MIEEE, BEM, Grad IEM - was born in Malaysia in 1975. He received the B. Eng. and M. Eng. in Electrical and Electronic Engineering from the Universiti Teknologi Malaysia in 1998 and 2004 respectively. He obtained his Doctorate from The University of Leeds in 2010 in the field of Microwave Engineering. From 1998 to 2002, he was with STMicroelectronics, based

in Malaysia where he worked as Product Engineer. He is currently a Senior Lecturer and also the Deputy Dean (Academic) at the Faculty of Electronic and Computer Engineering, University Teknikal Malaysia Melaka, where he teaches Electronics, Microwave Engineering and Signal Processing. His research interests include a variety of microwave device developments such as planar and non-planar microwave filters, amplifiers and antennas. He also investigates radiowave propagation as well as energy harvesting for wireless sensor networks and signal processing. (Email: zahriladha@utem.edu.my).



H. Bakri received the B.Eng. in Mechanical Engineering and M.Sc. in Mechanical Engineering from Gifu University, Japan in 2007 and 2009, respectively. From 2009, he worked as a Lecturer of Structure and Material Department in Faculty of Mechanical Engineering, Universiti Teknikal Malaysia Melaka (UTeM), Malaysia. He joined as a member of Board of Engineering Malaysia (BEM) and Institute of Engineering

Malaysia (IEM). (Email: hairul.bakri@utem.edu.my).



F. Sakaguchi graduated in 1983 from the Faculty of Science, Kyoto University, where he obtained an M.S. in Geophysics in 1985. In 1988, he became a Lecturer at Mie University (Faculty of Humanity). He also has a Dr. of Eng. degree and, in 1992, became an Associate Professor of Electronic Engineering, Fukui University. He has been engaged in research on statistical signal

processing and information theory. He is a member of the Institute of Information Theory and Its Applications. (email: saka@dignet.fuee.fukui-u.ac.jp).