UNIVERSITI MALAYSIA PAHANG

BORANG PENGESAHAN STATUS TESIS♦

JUDUL: **DROWSINESS DETECTION FOR CAR ASSISTED DRIVER SYSTEM USING IMAGE PROCESSING ANALYSIS**

SESI PENGAJIAN: **2010/2011**

Saya **ANI SYAZANA BT JASNI (880918-01-5782)**
(HURUF BESAR)

mengaku membenarkan tesis (Sarjana Muda/~~Sarjana~~ /~~Doktor Falsafah~~)* ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Universiti Malaysia Pahang (UMP).
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (√)

□ **SULIT** (Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

□ **TERHAD** (Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)

☑ **TIDAK TERHAD**

Disahkan oleh:

(TANDATANGAN PENULIS)	(TANDATANGAN PENYELIA)

Alamat Tetap:

455, JLN WP A 14/1, **BANDAR WARISAN PUTERI,** **71770 SEREMBAN,** **NEGERI SEMBILAN DARUL KHUSUS**	**DR. KAMARUL HAWARI GHAZALI** (Nama Penyelia)
Tarikh: **29 NOVEMBER 2010**	Tarikh: **29 NOVEMBER 2010**

CATATAN:
* Potong yang tidak berkenaan.
** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh tesis ini perlu dikelaskan sebagai atau TERHAD.
♦ Tesis dimaksudkan sebagai tesis bagi Ijazah doktor Falsafah dan Sarjana secara Penyelidikan, atau disertasi bagi pengajian secara kerja kursus dan penyelidikan, atau Laporan Projek Sarjana Muda (PSM).

DROWSINESS DETECTION FOR CAR ASSISTED DRIVER SYSTEM USING IMAGE PROCESSING ANALYSIS – SIMULATION OF REAL TIME MONITORING SYSTEM

ANI SYAZANA BT JASNI

UNIVERSITI MALAYSIA PAHANG

“I hereby acknowledge that the scope and quality of this thesis is qualified for the award of the Bachelor Degree of Electrical Engineering (Electronics)”

Signature : ___________________

Name : DR. KAMARUL HAWARI

Date : ___________________

DROWSINESS DETECTION FOR CAR ASSISTED DRIVER SYSTEM USING IMAGE PROCESSING ANALYSIS

ANI SYAZANA BT JASNI

This thesis is submitted as partial fulfilment of the requirement for the award of the Bachelor Degree of Electrical Engineering (Electronics)

Faculty of Electrical & Electronics Engineering

University Malaysia Pahang

(UMP)

NOVEMBER, 2010

“All the trademark and copyright use herein are property of their respective owner. Reference of information from other sources is quoted accordingly; otherwise the information presented in this report is solely work of the author.”

Signature : ___________________

Author : ANI SYAZANA BT JASNI

Date : ____________________

Dedicated to my beloved parents, siblings, supervisor and all of you for giving constant source of support and encouragement

ACKNOWLEDGEMENT

Thanks to Allah for giving me this opportunity, the strength and the patience to complete my project finally, after all the challenges and difficulties.

First and foremost, I would like to express my greatest gratitude to my supervisor Dr. Kamarul Hawari for his encouragement guidance, consistent supports and enthusiasm given in finishing my final year project. This appreciation is also dedicated to my academic advisor Ms. Najidah Hambali for motivation and supports.

I would like to thanks my parents for supporting me mentally and physically not just during finishing this task but also during my whole studies in order to become a good Muslims.

Finally, I would like to take this opportunity to thank all my friends and colleagues who have given their support and help to accomplish this final year project.

Hopefully, this final year project will not be the end of my journey in seeking for more knowledge to understand the meaning of life.

ABSTRACT

The current technology in digital computer system allows researchers around the world to study the fatigue behavior. Although the current technology of drowsiness detector has been created, it is lack of efficient since the detection is used ordinary sensor. This project is to develop a driver drowsiness detection system by using histogram analysis. It is known that a driver is under drowsiness influences by looking at the eyelid. Based on the previous research, there is none used histogram for analysis. The result can be not accurate because histogram analysis analyzed the whole image. Therefore, if the analysis area is not specified, the result will be not accurate and efficient. The retina movement shows the fatigue level of the driver. For example, if the driver's eyes are closed about more than 5 seconds in the last 60 seconds, the driver considered as drowsiness. Based on the fact that driver's eye movement can be used to recognize the level of drowsiness, a sensor can be developing by using image processing analysis in MATLAB. The image processing analysis that will be used is histogram analysis. This system will be developing only on software part.

ABSTRAK

Kini dalam zaman teknologi, sistem komputer membolehkan penyelidik di seluruh dunia mendalami perilaku manusia yang berada dalam kepenatan. Walaupun teknologi alat pengesan mengantuk telah dicipta, tetapi alat tersebut kurang berkesan kerana alat tersebut menggunakan sensor biasa. Projek ini adalah untuk membangunkan sebuah sistem yang boleh mengesan mata pemandu yang mengantuk. Ianya boleh dikesan dengan melihat kelopak mata pemandu. Gerakan tutup dan buka kelopak mata menunjukkan tahap mengantuk seorang pemandu. Sebagai contoh, jika mata pemandu ditutup selama 5 saat dalam 1 minit, pemandu tersebut dikira mengantuk. Tetapi, jika mata pemandu hanya tertutup selama 2 saat, pemandu hanya dalam keadaan biasa sahaja. Berdasarkan fakta bahawa gerakan tutup dan buka kelopak mata pemandu boleh digunakan untuk mengenalpasti tahap mengantuk dan dapat berkembang dengan menggunakan analisis di MATLAB. Analisis yang akan digunakan adalah analisis histogram. Sistem ini akan berkembang hanya pada bahagian perisian.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

Safe driving is a major concern of societies all over the world. Thousands of people are killed, or seriously injured due to drivers falling asleep at the wheels each year. Recent studies show those drivers' drowsiness accounts for up to 20% of serious or fatal accidents on motorways and monotonous roads, which impair the drivers' judgment and their ability of controlling vehicles [6]. Therefore, it is essential to develop a real-time safety system for drowsiness-related road accident prevention.

Many methods have been developed and some of them are currently being used for detecting the driver's drowsiness [7], including the measurements of physiological features like EEG, heart rate and pulse rate, eyelid movement, gaze, head movement and behaviors of the vehicle, such as lane deviations and steering movements. Among those different technologies, ocular measures, such as eye-blinking and eyelid closure, are considered as promising ways for monitoring alertness.

Typically, after long hours of driving or in absent of alert mental state, the eyelids of driver will become heavy due to fatigue. The attention of driver starts to lose focus, and that creates risks for accidents. These are typical reactions of fatigue, which is very dangerous. Usually many exhausted drivers are not aware that they are in falling asleep. In fact, many such drivers can fall asleep any time during their driving. In an image fatigue detection, correct and real time decision is important. Therefore, in this project eyelid closure is chosen to be the method for drowsiness detection when driving.

The current technology in digital computer system allows researchers around the world to study the fatigue behavior. Although the current technology of drowsiness detector has been created, it is lack of efficient since the detection is used ordinary sensor. This project is to develop a driver drowsiness detection system by using histogram analysis. It is known that a driver is under drowsiness influences by looking at the eyelid. Based on the previous research, there is none used histogram for analysis. The result can be not accurate because histogram analysis analyzed the whole image. Therefore, if the analysis area is not specified, the result will be not accurate and efficient. The retina movement shows the fatigue level of the driver. For example, if the driver’s eyes are closed about more than 5 seconds in the last 60 seconds, an alarm will sound to alert the driver. Based on the fact that driver’s eye movement can be used to recognize the level of drowsiness, a sensor can be developing by using image processing analysis in MATLAB. The image processing analysis that will be used is histogram analysis. This system will be developing only on software part.

1.2 Objective

The objective of this project is to:

i. To develop a system that able to detect drowsiness of a driver based on eyelid detection in digital image.
ii. To make analysis of the eyelid by using histogram features.

1.3 Scope of Project

The scopes that need to be proposed in this project:

i. Focus on image processing tool which is histogram.
ii. Develop on software only.

1.4 Problem Statement

This project is to develop a driver drowsiness detection system by using histogram analysis. It is known that a driver is under drowsiness influences by looking at the eyelid. Based on the previous research, there is none used histogram for analysis. The result can be not accurate because histogram analysis analyzed the whole image. Therefore, if the analysis area is not specified, the result will be not accurate and efficient.

1.5 Thesis Outline

Chapter 1 will describes the introduction of this system, the objectives of this project, the problem statement, the work scopes and overview of this project.

Chapter 2 will review about the information found on all the material or data used in the development of the system.

Chapter 3 will explain all the method used to develop this system. This chapter will explain briefly about MATLAB in designing the drowsiness detection system.

Chapter 4 will include all the results and the explanations about the results after all development process has done.

Chapter 5 will show the summary after all and come up with some recommendations for some improvements.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

There are some previous researches that are used to be referred to develop a real-time system on drowsiness detection. These are researches and journals that are related to this project either directly or indirectly.

For the first journal titled “Detecting eye blink states by tracking iris and eyelids” by Huachun Tan and Yu-Jin Zhang (2005). This paper works on the simultaneous exploitation of intensity and edge information for detecting the eye state as well as the record of the patterns of eyelids before closing for tracking the reopened eyes. In the method proposed by Tian et al. (2000), the open eye boundary is tracked by two key points of the eye template (the center points of the upper and the lower eyelids). The eye template parameters are obtained from the two points and the eye corners.

However, in the author’s experiments, there are problems in tracking eyelids. When the iris is redetected, the eye boundaries are tracked through the center points of the upper and the lower eyelids. However, the center points of the upper and the lower eyelids are the same point because of the line model. The patterns of the two center points are same, which lead to the two points moving to the same point in the following frames. Thus, at least one of the eyelids (upper or lower) trackers would be in error. And in the following image sequences, the error would be propagated.

Figure 2.1 illustrates the problem of tracking eyelids. From Figure 2.1, it can be found that when eye is reopened, the upper eyelid is failed to be tracked because the pattern of center point of upper eyelid is changed. In the following sequences, the error is propagated, even enlarged.

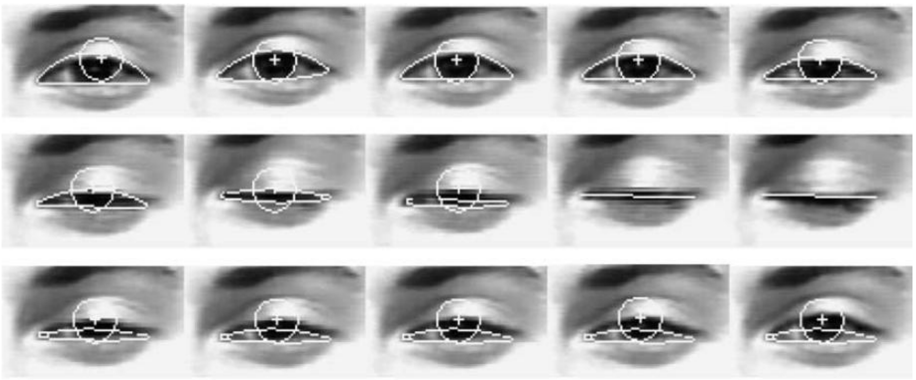


Figure 2.1: The problem of tracking eyelids of the method proposed by Tian et al. (2000), where the upper eyelids are not correctly tracked when the eye is reopened.

The problem of tracking eyelids lies in the wrong pattern when the iris is redetected. Assume the illumination is invariable in the process of blink, the patterns when the eye is open before blinking can be used to predict the patterns of eyelids when the eye is re-open. In addition, in order to track the eyelids more robust, we use AR model to predict the patterns of center points of eyelids when the iris is redetected. That is, the patterns of eyelids are predicted by

$$p_n = \sum_{i=1}^{m} w_i p_{n-i},$$

where *Pn* is the pattern of center point of eyelid in the nth frame where the eye is open; m is the number of frames that account for prediction;
Wi is the prediction coefficient.

In the author's system, they set the order of the model as 2. Then the prediction process can be described as follows;

$$p_n = wp_{n-1} + (1-w)p_{n-2},$$

where *W* is the prediction coefficient.

Figure 2.2 shows an example of eyelid tracking using proposed method for the same image sequence in Figure 1, where w is set as 0.5. Due to the correctness of the patterns used to track the eyelids, both the lower and upper eyelids are tracked accurately.

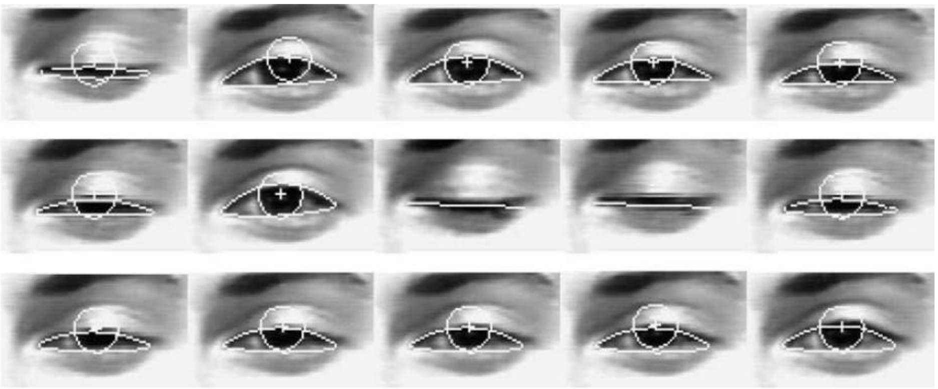


Figure 2.2: An example of eyelid tracking using proposed method, where the eyelids can be correctly tracked.

For the second journal is titled “Development of a drowsiness warning system based on the fuzzy logic image analysis” by Jian-Da Wu and Tuo-Rung Chen (2007). In this paper, a vehicle driver drowsiness warning system using image processing technique with fuzzy logic inference is developed and investigated. The principle of the proposed system is based on facial images analysis for warning the driver of drowsiness or inattention to prevent traffic accidents. The facial images of driver are taken by a charge coupled device (CCD) camera which is installed on the dashboard in front of the driver. A fuzzy logic algorithm and an inference are proposed to determine the level of fatigue by measuring the blinding duration and its frequency, and warn the driver accordingly.

An image which taken inside a vehicle includes the driver as well as other items as shown in Figure 2.3.

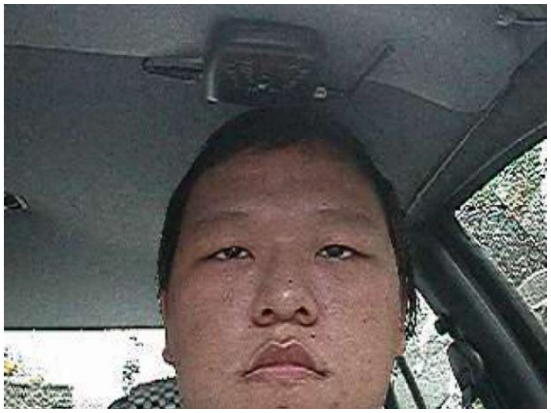


Figure 2.3: An image taken by CCD in vehicle

Hence, the proposed system applies a facial detection technique to single out the driver’s face from the rest of the image [8]. Typically, CCD takes images within the RGB model (Red, Green and Blue). However, the RGB model includes brightness in addition to the colors. When it comes to human’s eyes, different brightness for the same color means different color. When analyzing a human face, RGB model is very sensitive in image brightness. Therefore, to remove the brightness from the images is necessary.

To accomplish this, the images are converted from RGB to HIS colour model, where H stands for hue, S stands for saturation, and I stands for intensity of image. The characteristics of skin among different human races are similar. The only major difference between a people with a lighter complexion versus one who has a darker complexion is in the different lighting conditions. Hue and saturation will determine the facial skin in an image. Equations (1) and (2) can be used to determine the skin tone.

$$0.23 \leqslant \text{Saturation} \leqslant 0.63, \quad 10^\circ \leqslant \text{Hue} \leqslant 50^\circ \tag{1}$$

The formula to convert RGB to HSV is as followed:

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

$$H = \begin{cases} \theta & B \leqslant G \\ 360-\theta & B > G \end{cases}$$

$$S = 1 - \frac{3 \cdot \text{Min}(R,G,B)}{R+G+B}$$

$$I = (R+G+B)/3$$

After to determine the skin tone, the author’s then use vertical and horizontal projection. Projection basically uses every pixel in the skin to progress vertically and horizontally until it reaches a specific threshold value. When projection is done, the driver’s face area can be determined as shown in Figure 2.4. The area with the most of dense projection is the center of the face.

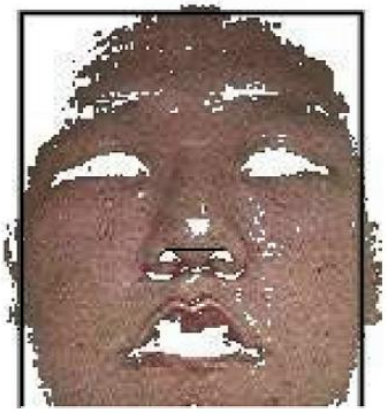


Figure 2.4: Facial skin region after facial detection

Since the center of every person's face is around the nose area and that blinking of the eyes usually happen concurrently, the author's assumed that the right eye will be positioned at the upper left-hand side of the center of the face is showed as Figure 2.5. Therefore, calculation will be based on only one eye. By taking these assumptions, the search for the eye will be limited to the area between 70° and 90°. This limited area will make the search more efficient and converting this particular area to grayscale will speed up the search even more.

Normally, a grayscale image can be divided with colors in the image into 256 values. Besides having 0 as black and 255 as white, there are 254 different shades of gray in the middle. Since a color image is three times the size of a grayscale image, the author's decided to use the following grayscale formula to facilitate the calculations:

$$\text{Grayscale value } (Y) = 0.299R + 0.587G + 0.114B \qquad (3)$$

The eye is in a 20 · 15 matrix within the eye region. Therefore, the author's used a 20 · 20 mask to begin search. The four corners of search area are smaller than the threshold value but the center area is larger than the threshold value. From such search, the author's determined the smallest matrix for the eye.

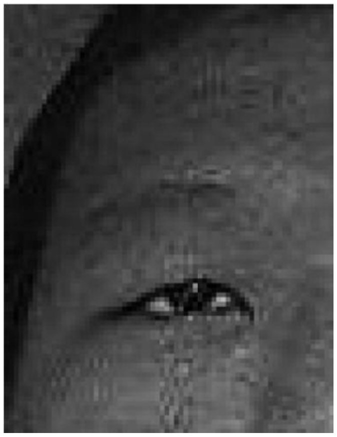


Figure 2.5: Eye region before eye detection

Figure 2.6 shows the search results. Only the lower right-hand side image fits the criteria.

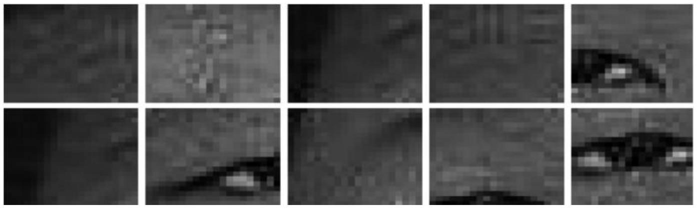


Figure 2.6: Searching the eye region

In this section, the concept of threshold is used to detect the openness of the driver's eyes. Thresholding is the operation of setting a pixel to black if the value is below a given threshold. On the other hand, a pixel is set to white if the value is above the threshold. From this process, we can generate a black-and-white image. In the following formula, *n* is the thresholding value.

$$n = \sum_{i=1}^{t} g(x, y) \tag{4}$$

where g is the image, t is the number of pixels, g(x,y) is pixel coordinates and (x,y) is pixel's grayscale.

$$\text{If } (x,y) > n, \text{ then } (x,y) = 255$$
$$\text{If } (x,y) < n, \text{ then } (x,y) = 0$$

This paper requires the calculation of the eyelid closure duration. After obtaining the thresholding value based on tests, set the skin pixel to white and the pupil to black. Based on the number of black pixels in an image, we are able to determine whether the eye is open or closed as illustrated in Figure 2.7.

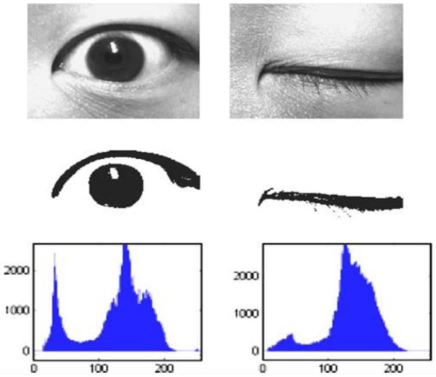


Figure 2.7: Thresholding the eye area

For the third journal is titled “Face detection for video summary using illumination-compensation and morphological processing” by Jae-Ung Yun, Hyung-Jin Lee, Anjan Kumar Paul, Joong-Hwan Baek (2009). This paper presents a simple and robust face detection algorithm that can be utilized to video summary. Because the characteristic of the face can be changed from unexpected condition like as shadows or lighting, firstly process the illumination-compensation for maintaining face components. Then analyze color-based face region in YCbCr space to obtain the skin color. Also, apply the morphological processing to improve the detection performance and reducing the number of false detected face regions. This process finds the candidates in face region and removes noise from the non-face region. For localization of the face region, make a proper face ratio based on golden ratio. Then evaluate the algorithm in the various genres. Experimental results demonstrate the effectiveness of the face detection algorithm that leads 96.7% in precision ratio on the average. The proposed method is applicable to video summary because of these high performances with low complexity.

In this paper, we present a robust face detection method through the illumination-compensation algorithm using YCbCr space and morphological processing as supplement that is able to remove noise and can help to find clear face. Using some attribute of face, this gives the information about face of interest (FOI) that helps increasing the performance on face databases. Figure 2.8 shows our face detection system.

This paper is organized as follows. Firstly we introduce the illumination-compensation algorithm to detect even under the unclear brightness conditions in Section 2. Section 3 describes the skin tone analysis and shows the processing of FOI to improve detection ratio. Localization of face is discussed in Section 4. Section 5 presents the detection results of our algorithm on several face databases. Conclusion and future works are given in Section 6.

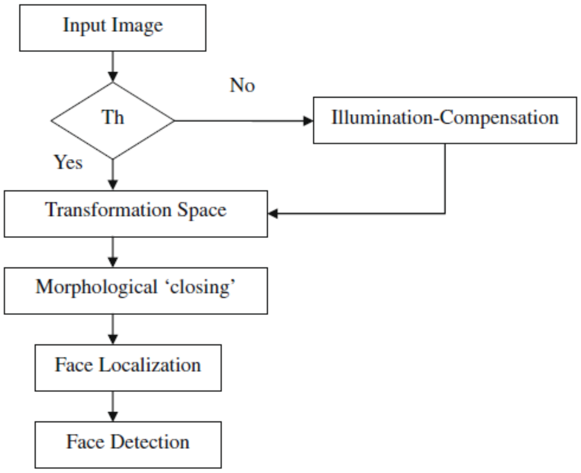


Figure 2.8: Flow diagram of the face detection system.

The skin tone color depends on the illumination conditions like an artificially created light inside, strong sunlight as brightness cases and like the shadow, fading effects as dark cases. In this paper, firstly introduce the compensation method to remove the illumination effects. The illumination-compensation algorithm shown in figure 2.9 consists of 4 steps.

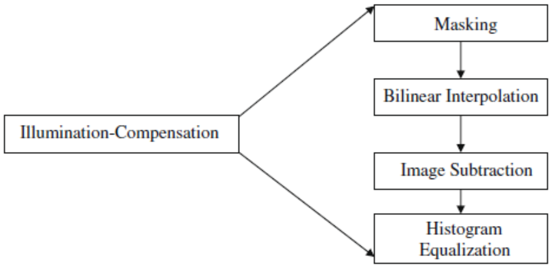


Figure 2.9: Illumination-compensation processes.

They use a 20 x 20 window to compute the minimal value which generates 16 x 12 minimal brightness plane (the size of the original image: 320 x 240) shown in figure 2.10c. These brightness planes are resized by bilinear-interpolation method in figure 2.10c. At last, the illumination-compensation images are further normalized by applying histogram equalization.

The output intensity levels, out _level, are obtained by the following transformation on input levels:

$$out\_level = \int_0^x p_x(w)dw \qquad (5)$$

where, w is the integral parameter and $p_x(w)$ denotes the probability density function (PDF).

$$out\_level_j = \sum_{i=1}^{j} p_x(x_i) = \sum_{i=1}^{j} n_i/n \qquad (6)$$

For j = 1,2,3, . . . ,L, where out_level_j is the output intensity values corresponding to value x_j in the input image (Figure 2.10d).

The illumination-compensation can be utilized alternatively when the image is too bright or dark. They use the ‘‘reference value” to decide the criteria of irregular conditions. If the average values of the components are smaller (<260) or larger (>550) than some threshold values in the image, the illumination-compensation carries into execution before skin tone detection. In case of uniform distribution of luminance, the compensation step may be skipped. Figure 2.10 illustrates that brightness or darkness in images can be changed to detect face more precisely.

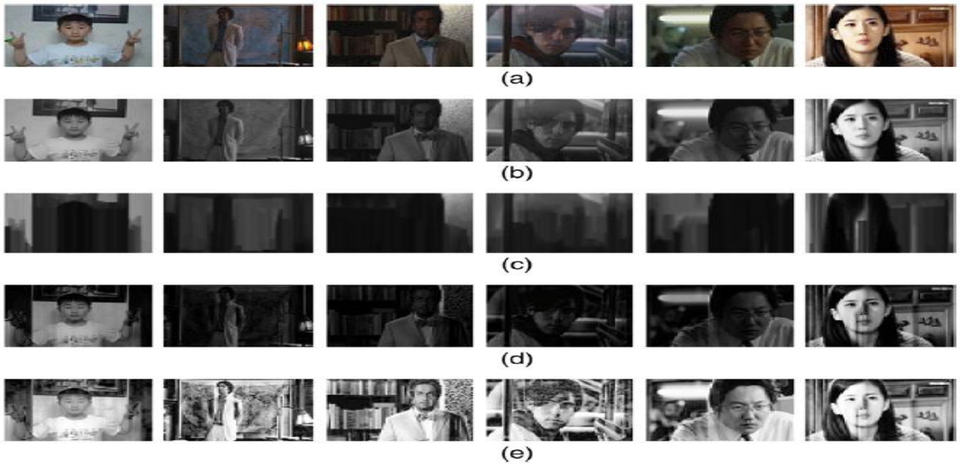


Figure 2.10: Examples of the illumination-compensation. (a) Original images. (b) Y component images. (c and d) Bilinear interpolated images by the minimal value in 20 x 20 window. (e) Illumination-compensated images.

The first step of the skin tone detection method is the transformation of color space from RGB to YCbCr space because RGB is not necessarily the most efficient representation of color due to the HVS (Human Visual System) is less sensitive to color than to luminance. After transformation, they extract the skin tone and carry out the morphological processing for face region detection.

Skin color modeling requires choosing an appropriate color space (Hsu and Mohamed, 2002). Comparison of nine different color spaces for face detection, the tint-saturation-luma (TSL) space provides the best results for two kinds of Gaussian density models (Terrillon et al., 2000). Nevertheless, we adopt the YCbCr space because the RGB color space contains color information that is less sensitive to HVS than luminance. The colors are equally important and the luminance is present in all three color components in RGB color space. On the other hand, the YCbCr space can represent a color image more efficiently by separating the luminance from the color information (Richardson, 2003). The equation of transformation between RGB to YCbCr is given as the following Eq. (7):

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & 0.331 & 0.500 \\ 0.500 & -0.419 & -0.081 \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{7}$$

They collect 100 and 200 samples for training skin tone without and with the illumination-compensation, respectively. They are from various genres such as drama, movie, news and actor images and the sizes are 258 x 228 and 96 x 86 pixels. Statistical distributions of CbCr components of the training data are shown in figure 2.11a and b. Figure 2.11a shows the Cb and Cr histogram without the illumination-compensation process, and figure 2.11b is with the illumination-compensation. Based on Cb and Cr information, they can detect the skin tone region shown in figure 2.12.

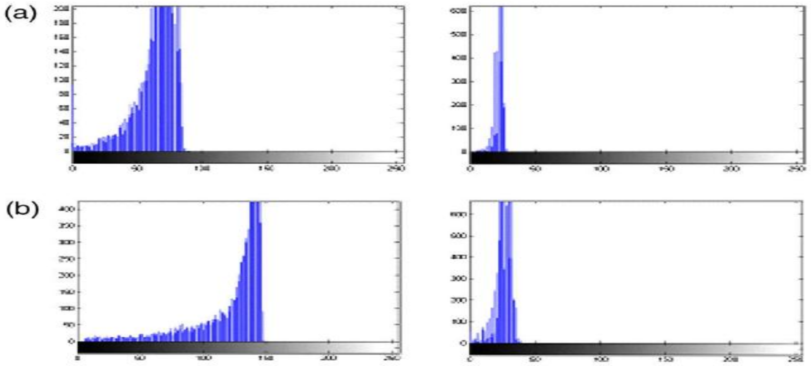


Figure 2.11: Cb and Cr histograms for skin tone detection. (a) Without the illumination-compensation. (b) With the illumination-compensation.

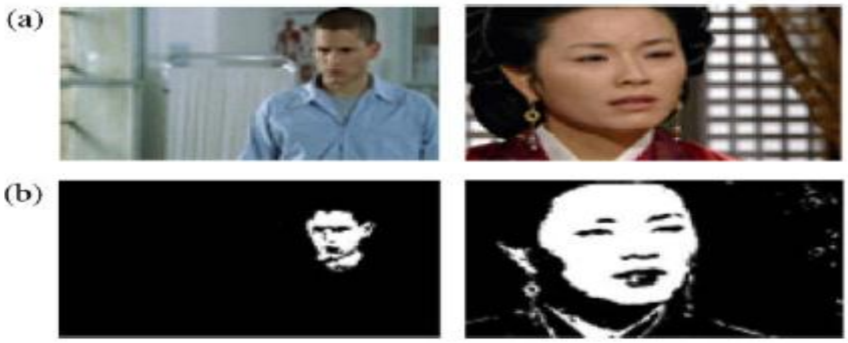


Figure 2.12: Results of the skin tone region detection. (a) Original images. (b) Results.

To improve the efficiency of face detection, we apply the morphological process as a combination of dilation and erosion. The operation of dilation and erosion are fundamental to morphological processing and are used most often in various combinations in practical processing such as the opening, closing and hit-or-miss (Gonzalez et al., 2004). We utilize the closing method to remove the noise from face region and non-face regions.

The closing of A by B, as denoted below is a dilation followed by erosion:

$$A \cdot B = (A \oplus B)\Theta B \qquad (8)$$

Like opening, morphological closing smoothes the contours of objects. Unlike opening, however, it generally joins narrow breaks, and fills holes smaller than the structured element. The process is illustrated in figure 2.13. Using the closing method, we can remove the noise and make the face region detection process more robust.

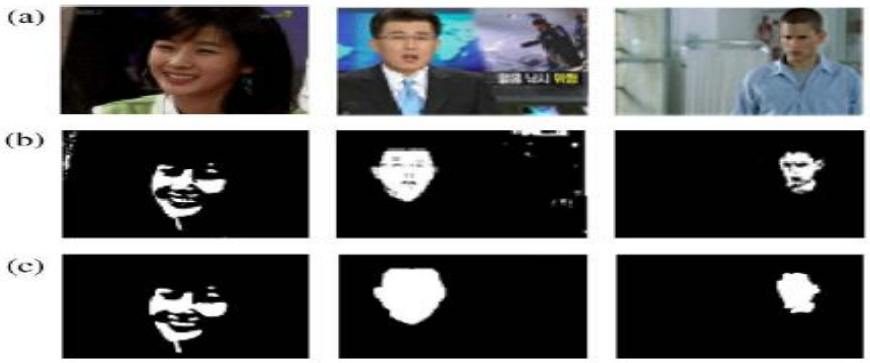


Figure 2.13: Morphological image processing. (a) Original images. (b) Skin tone detected images. (c) Results of the closing process.

The face candidate regions are found by using closing algorithm as mentioned before. The purpose of this research is to achieve high detection efficiency using low computational complexity. The eye and mouth regions are the most important features for face detection and recognition. However, they locate the face directly using the area of FOI and simple geometric features. This task can be divided into (3) subtasks:

(1) pre-processing, (2) calculation of the area, and (3) face region detection. At first, for the localization of face, they fill holes in the candidate regions in the pre-processing stage.

A hole is a set of pixels in the binary image with similar color of the background; but, originally, holes do not belong to the background. In second step calculate the area of filled region and then obtain the centroid of that area. It can give important information for determining the face region sizes. The face region detection step draws boundary of face using geometrical feature ‘golden ratio’. The face has a general ratio between width and height. Therefore, utilize this information to make a face boundary. For an ideal face, the golden ratio is denoted as Eq. (9):

$$\varphi = Height/Width = \left(1+\sqrt{5}\right)/2 \approx 1.618033989 \tag{9}$$

Since the face database in our research contains both frontal and side face images, they make an empirical modified face ratio from golden ratio, height:width = 1.28:1. The empirical experiments were performed by 30 images randomly chosen from dataset (see Figure 2.14)

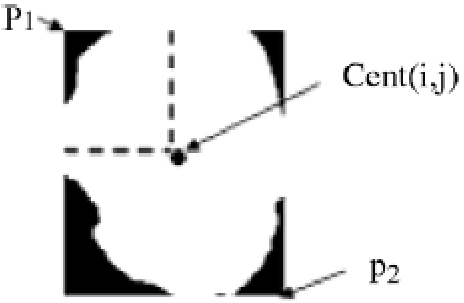


Figure 2.14: New golden radio

$$P_1(i,j) = Cent\left(i-\sqrt{area}/2, j-\sqrt{area}/2\right)$$
$$P_2(i,j) = Cent\left(i+\sqrt{area}/2, j+\sqrt{area}/2\right)$$
$$area = H \cdot W$$

where, Cent(i,j): centroid of the face.

P_1: start point of the face.

P_2: end point of the face.

H,W: height and width of the face.

area: area of the face region.

P1 and P2 are important points to obtain face line, and the resulting region will be a rectangle size of H x W. The figure 2.15 shows the processes of face region detection.

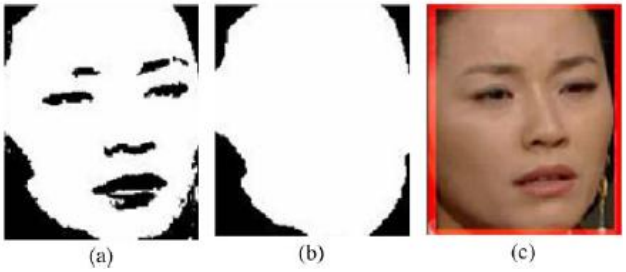


Figure 2.15: Face detection result. (a) Candidate of face image. (b) Localization step. (c) Final result.

2.2 Data Acquisition

Data acquisition is the sampling of the real world to generate data that can be manipulated by a computer. Sometimes abbreviated DAQ or DAS, data acquisition typically involves acquisition of signals and waveforms and processing the signals to obtain desired information. [15].

Acquired data is displayed, analyzed, and stored on a computer, either using vendor supplied software, or custom displays and control can be developed using various text-based programming languages such as BASIC, C, Fortran, Java, Lisp and Pascal. LabVIEW offers a graphical programming environment optimized for data acquisition. MATLAB provides a programming language but also built-in graphical tools and libraries for data acquisition and analysis [15].

2.3 Webcam

Started in 1991, the first webcam was pointed at the Trojan room coffee pot in the computer science department of Cambridge University. This webcam is now defunct, as it was finally switched off on August 22, 2001. The final image captured by the camera can still be viewed at the webcam's homepage [2].

Webcams connected to PCs can act as web-possible cameras with certain software; the software uploads pictures to a server, from which can produce an input to system. Usually, this kind of software is programmed to work with almost every webcam. This software can be configured in many ways, and will often include options for image size and quality, overlaying logos and time stamping images.

In this project, the webcam is eye of this system to capture or record the eyelid of the driver and sends back to system to analyse the data. The webcam will capture or record an image for 30 seconds that will be set in the system.

2.4 Image Processing

Image processing is any form of information processing for which the input is an image, such as photographs or frames of video; the output is not necessarily an image, but can be for instance a set of features of the image. Most image processing techniques involve treating the image as a two-dimensional signal and applying standard signal processing techniques.

An image defined in the "real world" is considered to be a function of two real variables, for example, a(x,y) with a as the amplitude (e.g. brightness) of the image at the real coordinate position (x,y). An image may be considered to contain sub-images sometimes referred to as regions-of-interest, ROIs, or simply regions. This concept reflects the fact that images frequently contain collection of objects each of which can be the basic for a region. Figure 2.8 show the pixel on image that converts in dot matrix. In a sophisticated image processing system it should possible to apply specific image processing operations to selected regions. Thus one part of an image (region) might be processed to suppress motion blur while another part might be processed to improve colour rendition [15].

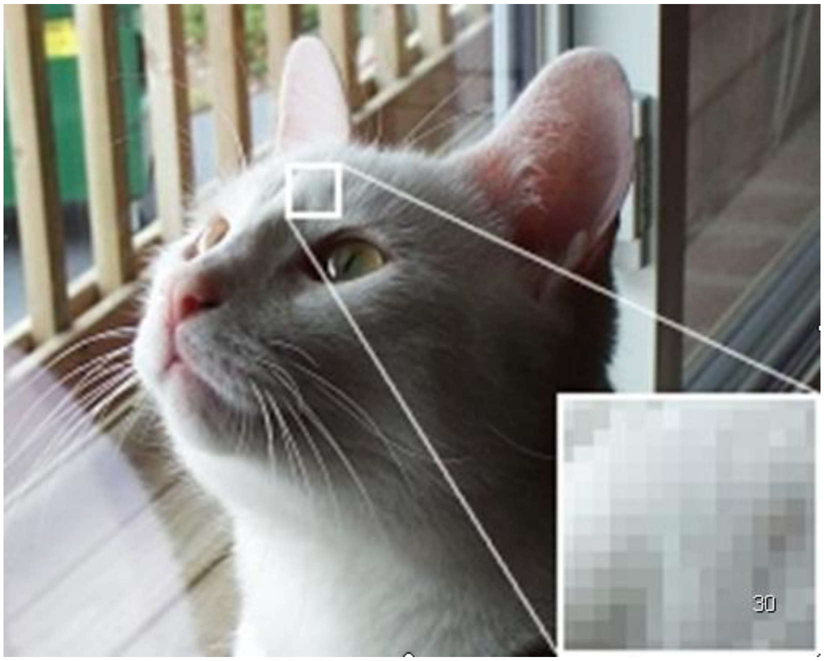
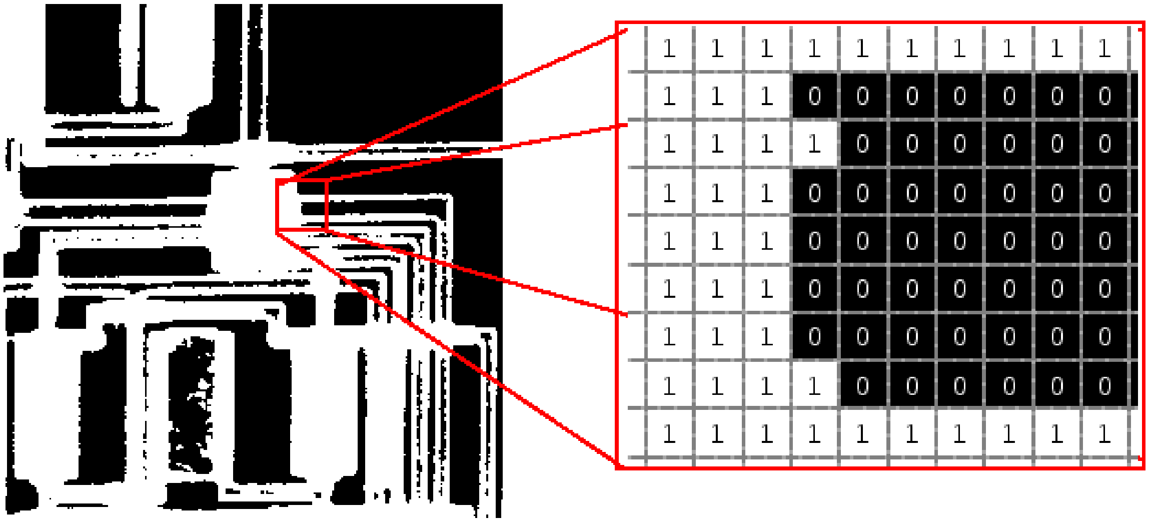


Figure 2.16: Pixel on image

The amplitudes of a given image will almost always be either real numbers or integer numbers. Figure 2.9 show that pixel value in a binary image. The latter is usually a result of a quantization process that converts a continuous range (say, between 0 and 100%) to a discrete number of levels. In certain image-forming processes, however he signal may involve photon counting which implies that the amplitude would be inherently quantized. In other image forming procedures, such as magnetic resonance imaging, the direct physical measurement yields a complex number in the form of a real magnitude and a real phase [7].



1	1	1	1	1	1	1	1	1	1
1	1	1	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	1	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1

Figure 2.17: Pixel values in binary image

In image processing, must consider the amplitudes as real's or integers unless otherwise indicated. Visual Basic, the visual-development environment that prefer to work in, often falls short in supporting these requirements, particularly when it comes to fast image plotting. However, it is possible to achieve efficient image processing by improving disk access and image plotting disk access and image plotting speeds using the techniques.

2.5 Edge Detection

The goal of edge detection is to mark the points in a digital image at which the luminous intensity changes sharply. Sharp changes in image properties usually reflect important events and changes in properties of the world. These include:

i. Discontinuities in depth
ii. Discontinuities in surface orientation
iii. Changes in material properties
iv. Variations in scene illumination

Edge detection of an image reduces significantly the amount of data and filters out information that may be regarded as less relevant, preserving the important structural properties of an image. There are many methods for edge detection, but most of them can be grouped into two categories, search-based and zero-crossing based. The search-based methods detect edges by looking for maxima and minima in the first derivative of the image, usually local directional maxima of the gradient magnitude. The zero-crossing based methods search for zero crossings in the second derivative of the image in order to find edges, usually the zero –crossings of the Laplacian or the zero-crossings of a non-linear differential expression [7].

Most edge detection methods work on the assumption that an edge occurs where there is a discontinuity in the intensity function or a very steep intensity gradient in the image. Using this assumption, if we take the derivative of the intensity values across the image and find points where the derivative is a maximum, we will have marked our edges.

A number of researches have considered the problem of measuring edge detector performance. In fact, it is difficult since we don't really know what the underlying features are that we wish to detect. However, if we assume that they are step edges corrupted by Gaussian noise, then some criteria can be set for evaluating performance. Such criteria are usually the following:

- The probability of false edges
- The probability of missing edges

- The error in estimating the edge angle
- The mean square distance of the edge estimate from the true edge
- The algorithm's tolerance to distorted edges and other features such as corners and junctions.

Currently, the canny operator (or variations of this operator) is most commonly used edge detection method. A large number of edge detection operators have been published but so far none has shown significant advantages over the Canny-type operators in general situations. In his original work, Canny studied the problem of designing an optimal pre-smoothing filter for edge detection, and then showed that this filter could be well approximated by a first-order Gaussian derivative kernel. Canny also introduced the notion of non-maximum suppression, which means that edges are defined as points where the gradient magnitude assumes a maximum in the gradient direction [8].

2.6 Threshold Image

Threshold requires a suitable reference image and the selection of an appropriate detection threshold. Several threshold selection methods are investigated, and an algorithm based on hysteresis thresholding is shown to give acceptably good results over a number of test image sets.

To performing the automatic thresholding of difference images is to assume particular distribution models for the difference of image samples and the noise. The image was threshold at multiple intensities, and the connectivity value of each calculated. The threshold was selected from an intensity range that produced a stable set of connectivity values. Rather than measuring connectivity, the number of regions may be more appropriate [12].

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter consists of the development of the system using MATLAB. Each of the boxes in flow chart will be explained in detail. The overall of the system is shown in Figure 3.1 below.

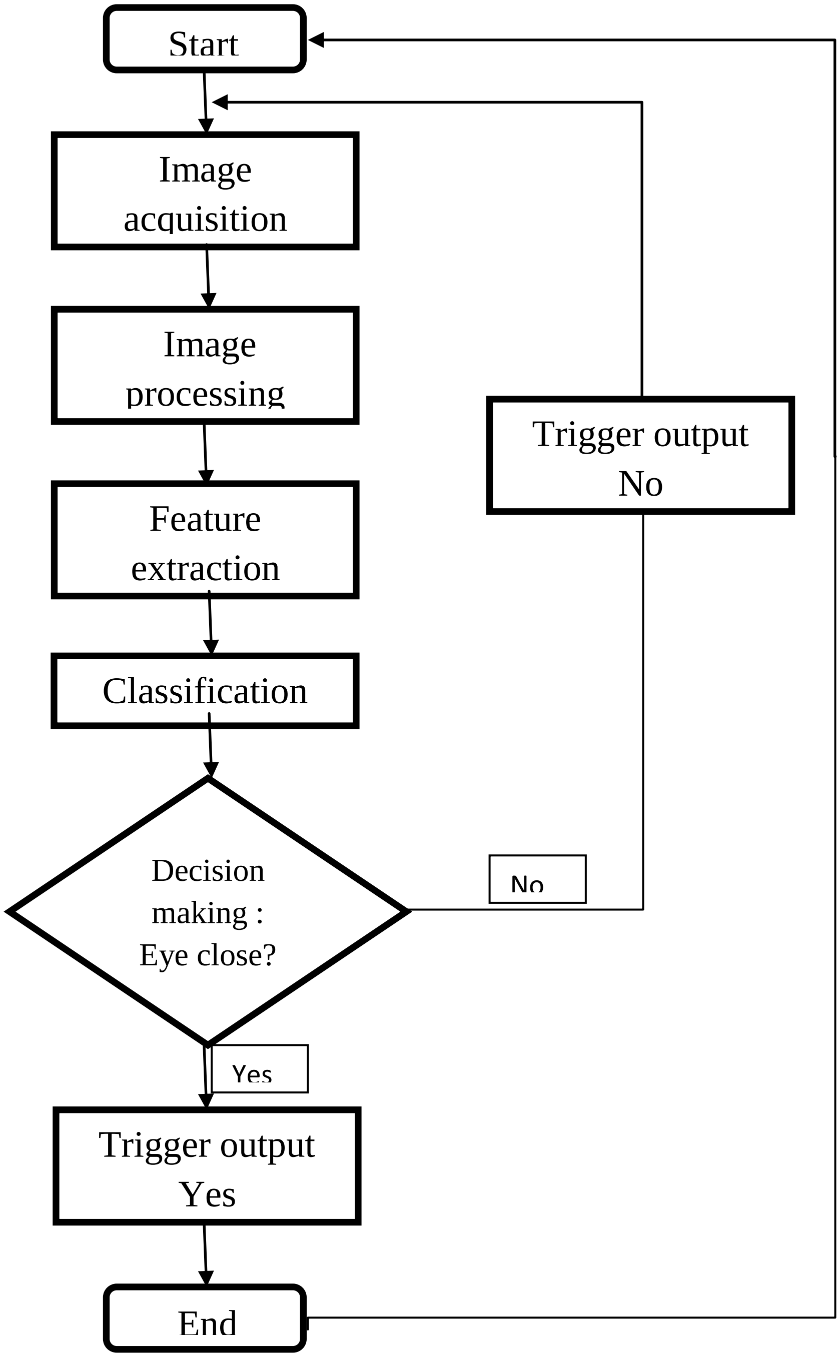


Figure 3.1: Flow chart on drowsiness detection in offline system

3.2 Image acquisition

The first steps in developing of this system start with the image acquisition. For this purpose eyelid area analysis begins with using data cursor to find suitable coordinates and precisely locating its inner and outer boundaries. However it is since for this time we will work with the offline system, therefore a series of eyelid image is collected like what in the standard eyelid database and after that each image will be test one by one by the next sequential process to be use as a training image.

In this part, the video camera Sony 12 MP will capture or recorded the image of the eyelids' driver. The video camera is mounted on the dashboard in front of the driver and it must focus on the eye area during the recording. The video will be converted to a hundred of images for analysis. There are several images are blurred due to auto-focus of the camera. But still, the eye images can be detected as close or open.

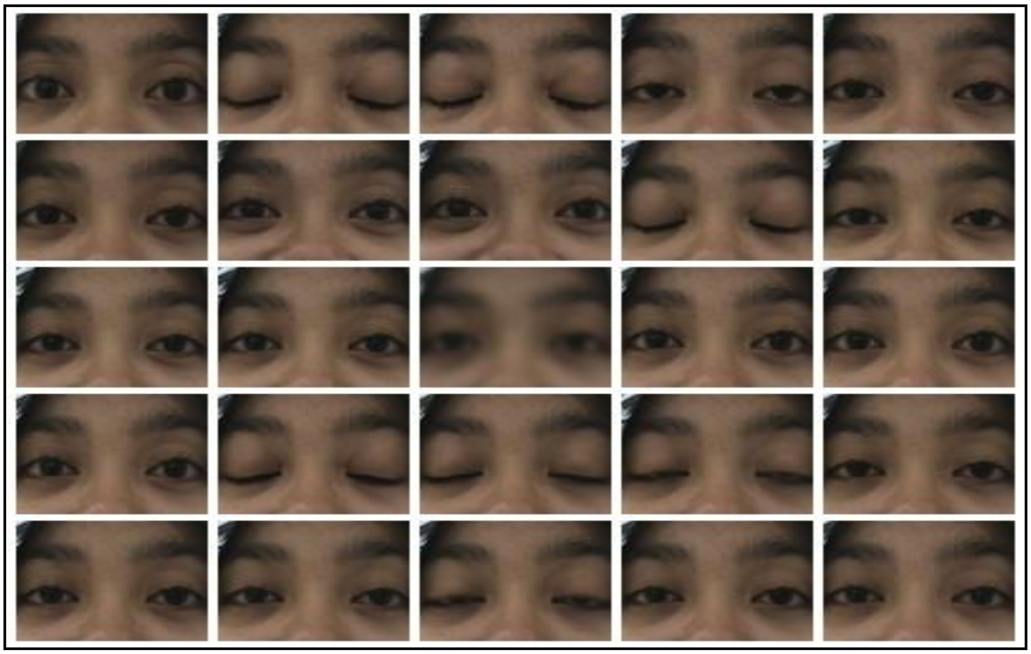
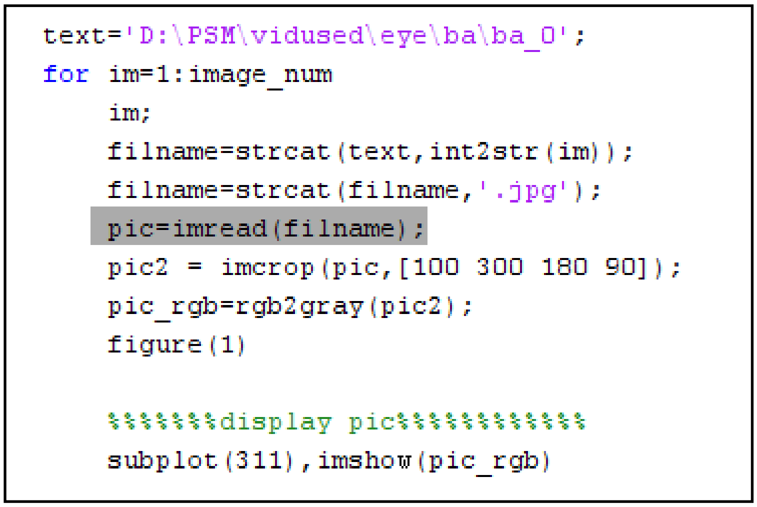


Figure 3.2: Sample of eye images

3.3 Image Preprocessing

First step for starting preprocessing is loading the close open eyelid images by using “imread” function in editor M-file. The function will read and load the image from directory. Below is the coding to read the image as shown in figure 3.3.



```
text='D:\PSM\vidused\eye\ba\ba_0';
for im=1:image_num
    im;
    filname=strcat(text,int2str(im));
    filname=strcat(filname,'.jpg');
    pic=imread(filname);
    pic2 = imcrop(pic,[100 300 180 90]);
    pic_rgb=rgb2gray(pic2);
    figure(1)

    %%%%%%%display pic%%%%%%%%%%%%
    subplot(311),imshow(pic_rgb)
```

Figure 3.3: Coding to read the image

Function of “for” is used to loop each images provided. All the programming of cropping images, convert to grayscale and others are inside the loop.

Image processing technique has variety of methods that can be used especially in this project. One of the methods that I used in this project is histogram analysis. To use histogram analysis, the RGB image must converted to grayscale image. Below is the original image as shown in figure 3.4.

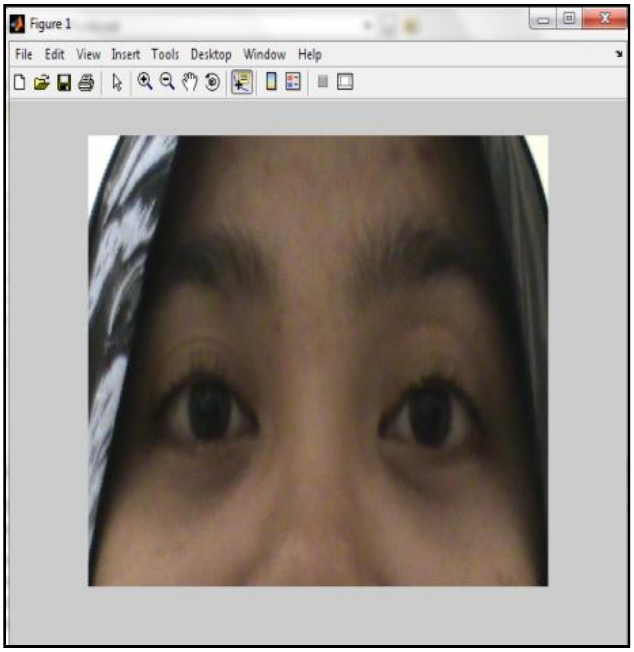


Figure 3.4: RGB image

Grayscale is very important for easy analysis with two channel of colour rather than RGB image, with three channel of colour. “rgb2gray” function had been applied, meaning that the parameter of hue and saturation is eliminated while the value of luminance is still maintained. This will result in figure 3.5 below.

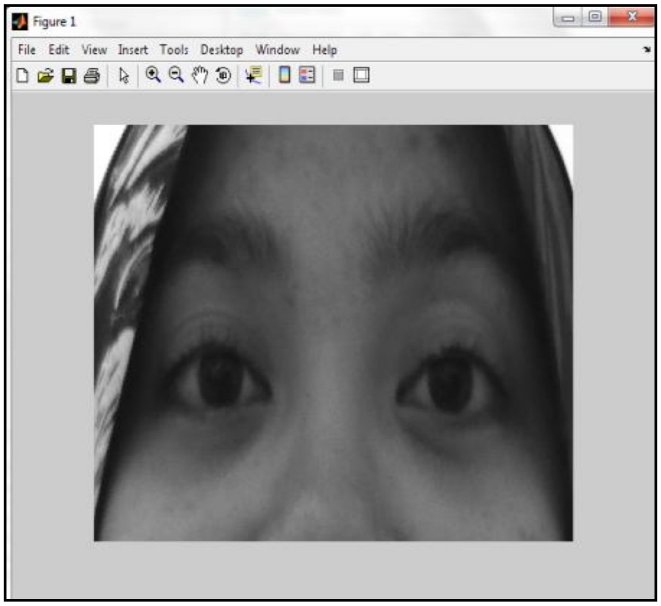


Figure 3.5: Grayscale image

Then the grayscale image is been analyzed by using data cursor to know the coordinates of the eyelid area. It shown as figure 3.6 below.

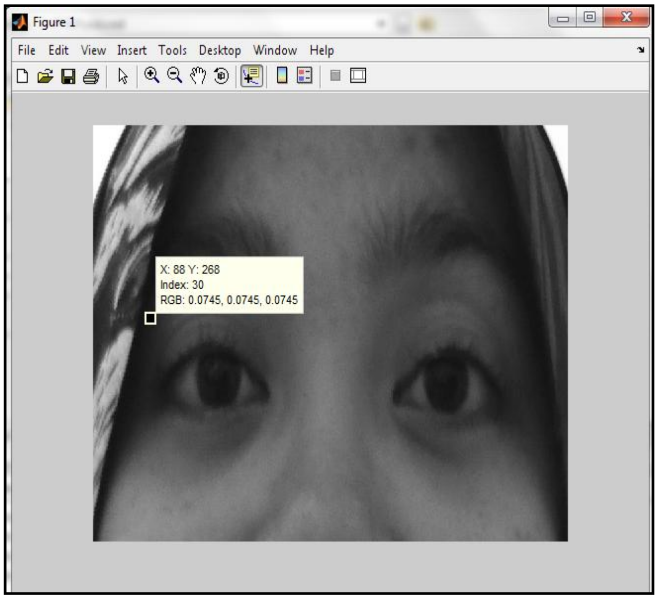
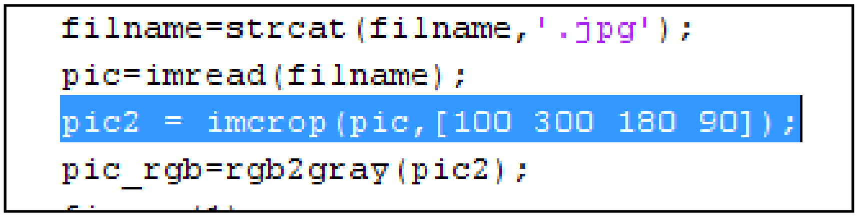


Figure 1
File Edit View Insert Tools Desktop Window Help
X: 88 Y: 268
Index: 30
RGB: 0.0745, 0.0745, 0.0745

Figure 3.6: One of eyelid the coordinates

By using the coordinates, the eye area can be narrowed using cropping function which is "imcrop" as shown in figure 3.7. Therefore only eyelid and pupil can be seen as shown in Figure 3.8 below.



```
filname=strcat(filname,'.jpg');
pic=imread(filname);
pic2 = imcrop(pic,[100 300 180 90]);
pic_rgb=rgb2gray(pic2);
```

Figure 3.7: Cropping function using "imcrop" along with the coordinates

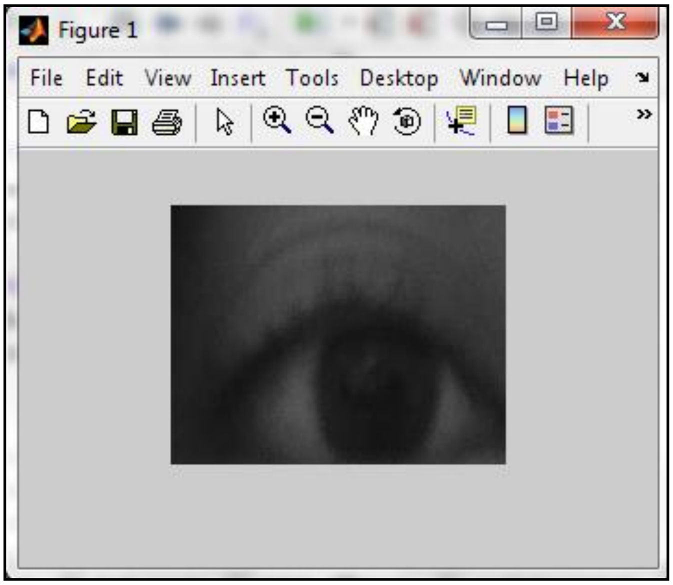
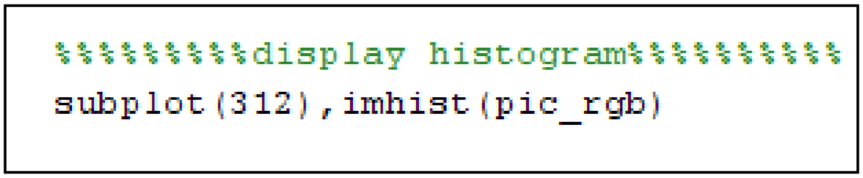


Figure 3.8: Eyelid that has been cropped

3.4 Feature extraction

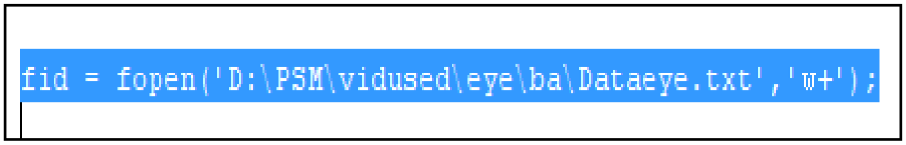
For the analyzing process histogram analysis will be used. After cropping the eyelid to appropriate size, the function of histogram which is “imhist” is used in programming as shown in figure 3.9.



```
%%%%%%%%%display histogram%%%%%%%%%%
subplot(312),imhist(pic_rgb)
```

Figure 3.9: Coding to display histogram

From the histogram it can be analyzed the changes of shape and values when eyelid is closed or opened. All the value can be precisely taken from the “Dataeye.txt” in notepad format. To get the data, function “fopen” is used as shown in figure 3.10. The values also can be used for threshold value to make a decision.



```
fid = fopen('D:\PSM\vidused\eye\ba\Dataeye.txt','w+');
```

Figure 3.10: Coding for function “fopen”

3.5 Classification

At this final stage, the image that has been extracted must be classified whether the eyelid is closed or opened. Decision making at this stage is crucial to know this system function or not. Therefore, the values and the shape from the histogram is used to detect either the eyelid is closed or opened.

CHAPTER 4

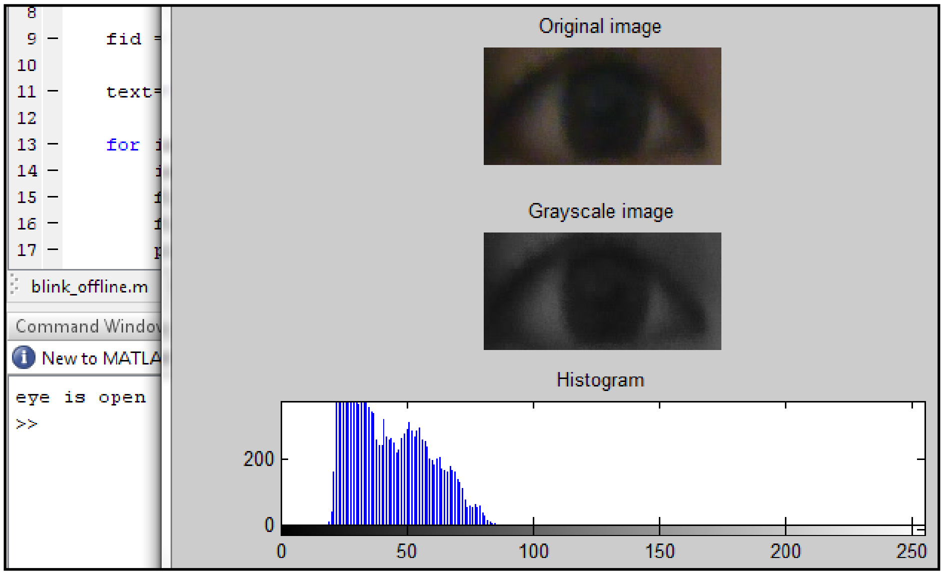
RESULT AND DISCUSSION

4.1 Introduction

In this chapter we will discuss in detail about the result and output for every single process that had been done. Each of steps will be provided with the discussion on itself. The technique that is histogram analysis will undergo a few minor processes. In order to build an effective system all of this step must be properly done and analyze. The effectiveness of this system should achieve at 100% effectiveness for a good usage.

4.2 Histogram pattern analysis

The result that had been obtained after all the process had been showed below. Each image with their process image and histogram had been representing by different result which is opened eyelid or vice versa. This result shows that the successfully of the method that previously been done.



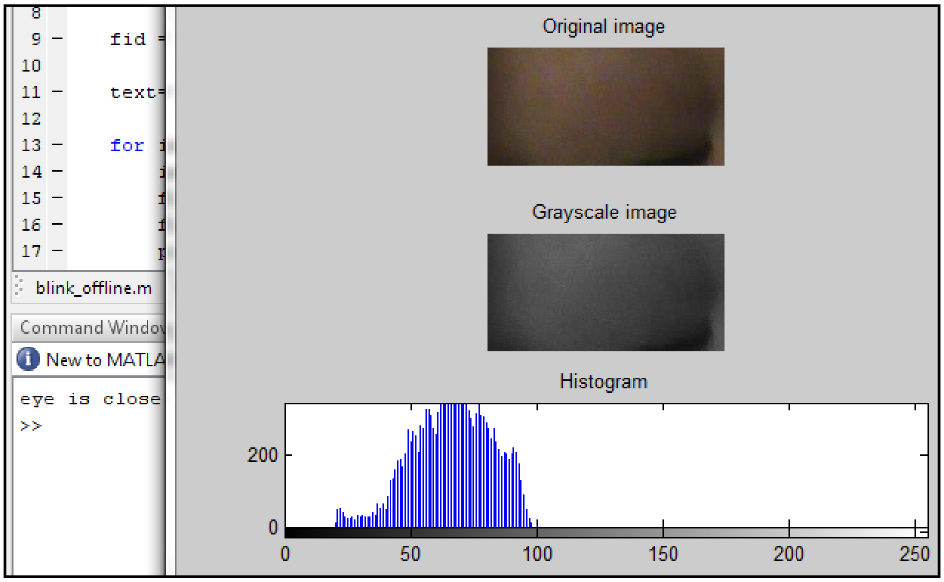
```
    fid =

    text=

    for i
        i
        f
        f
        p
```

```
eye is open
>>
```

Figure 4.1: Open eyelid



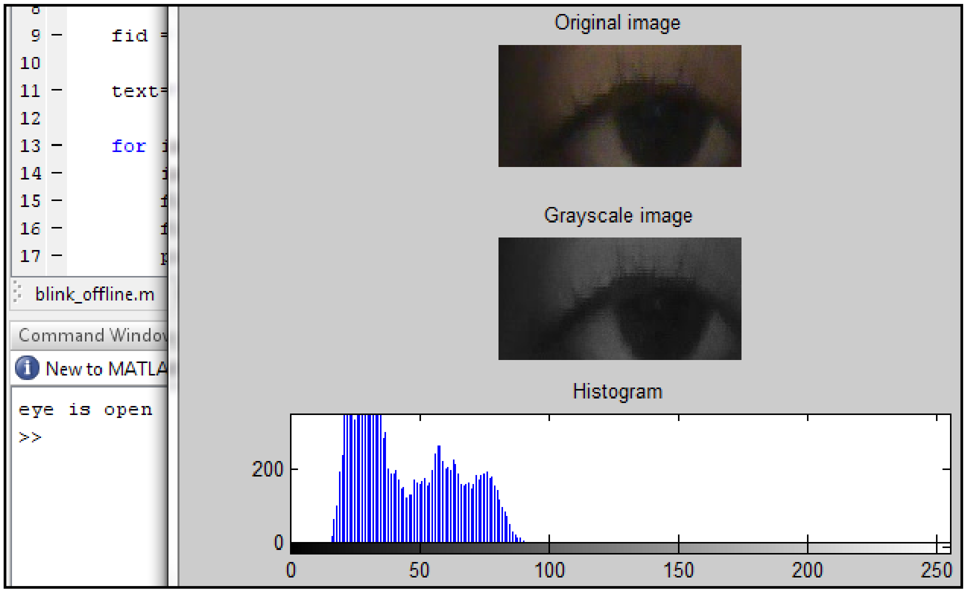
```
    fid =

    text=

    for i
        i
        f
        f
        p
```

```
eye is close
>>
```

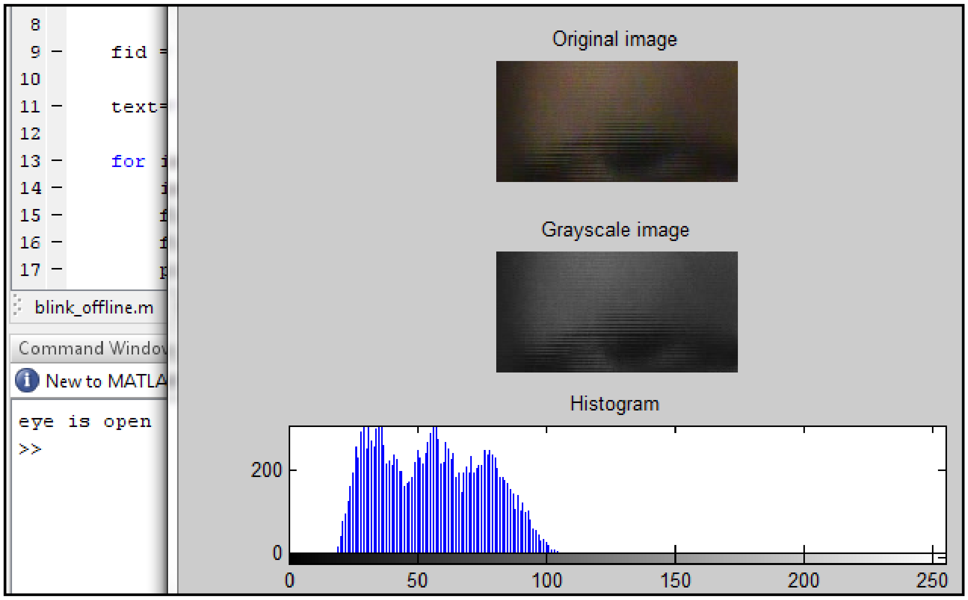
Figure 4.2: Close eyelid



```
fid =
text=
for i
    i
    f
    f
    p
```

```
eye is open
>>
```

Figure 4.3: Half close eyelid



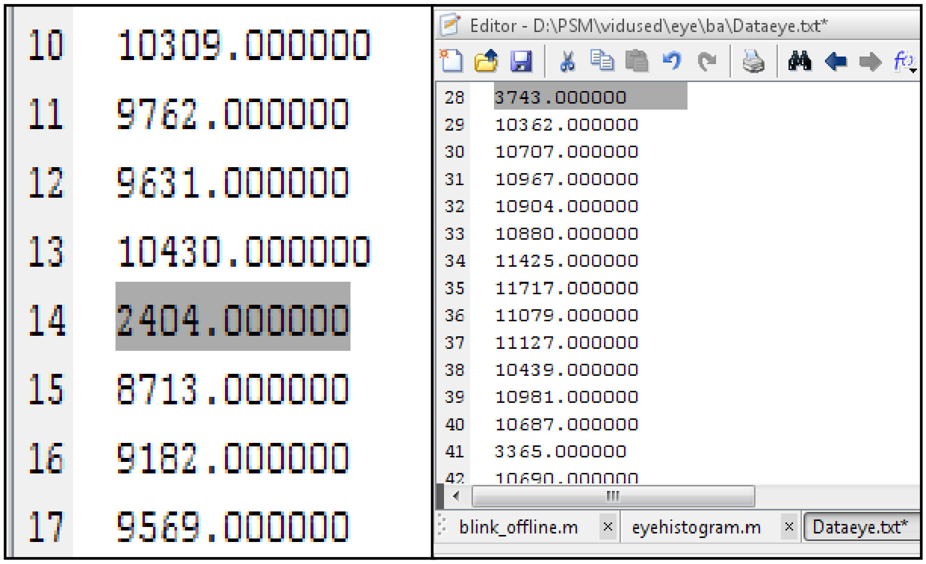
```
fid =
text=
for i
    i
    f
    f
    p
```

```
eye is open
>>
```

Figure 4.4: Quarter close eyelid

4.3 Histogram data analysis

Histogram data for each image is different from one another as shown in figure 4.5. Therefore by using the values, the images can be differentiate either it is closed or opened.



```
10   10309.000000
11   9762.000000
12   9631.000000
13   10430.000000
14   2404.000000
15   8713.000000
16   9182.000000
17   9569.000000
```

```
28  3743.000000
29  10362.000000
30  10707.000000
31  10967.000000
32  10904.000000
33  10880.000000
34  11425.000000
35  11717.000000
36  11079.000000
37  11127.000000
38  10439.000000
39  10981.000000
40  10687.000000
41  3365.000000
42  10690.000000
```

Figure 4.5: Histogram's data values for each image

Below are the data values for each image in histogram.

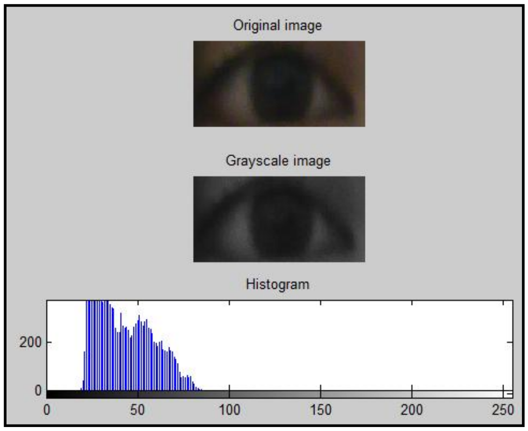


Figure 4.6: Open eyelid

For open eyelid, the data value is 10976.000000.

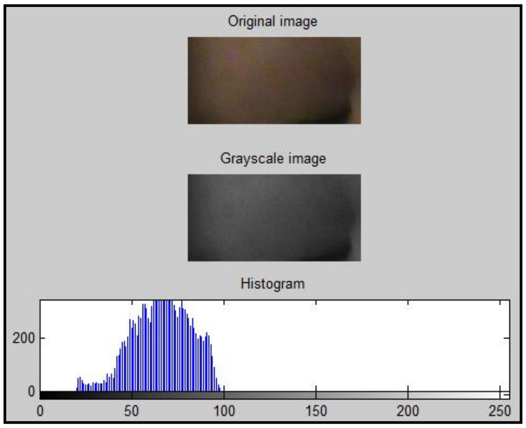


Figure 4.7: Close eyelid

For close eyelid, the data value is 3743.000000.

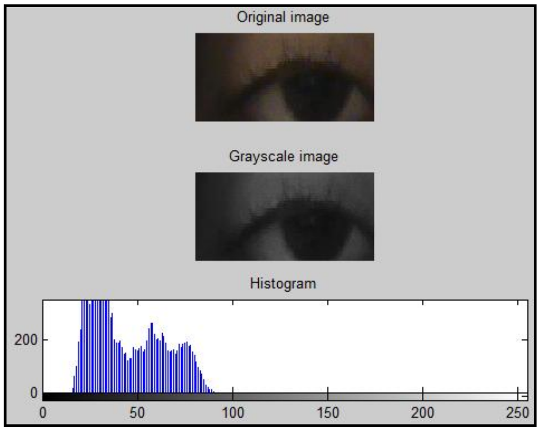


Figure 4.8: Half close eyelid

For half close eyelid, the data value is 10127.000000.

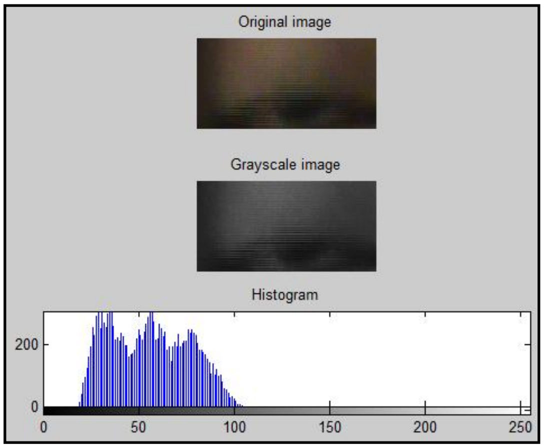
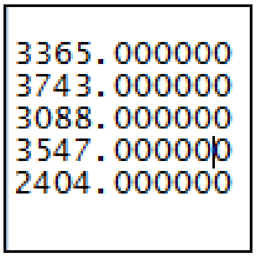


Figure 4.9: Quarter close eyelid

For quarter close eyelid, the data value is 6860.000000.



```
3365.000000
3743.000000
3088.000000
3547.000000
2404.000000
```

Figure 4.10: Close eyelid data values in histogram

Based on the data analysis above, the histogram's value of close eyelid is below 4000. If the value of images is below than 4000, the eyelid is considered closed and vice versa. Therefore that estimating value can be fitted in the programming to decide either the eyelid is closed or opened.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Introduction

This system had achieved a good accuracy in drowsy detection. This chapter summarizes the important finding from the analysis carried out in this project. It also includes some recommendation and suggestion for further work to improve the system and widely use the system.

5.2 Conclusion

In this project, a method of histogram analysis is used to detect the drowsiness of the driver. Based on the analysis and finding that had been done a few conclusions can be made with regard to the system in terms of complexity, accuracy, reliability, applications and many more.

I. The system that had been building is reliable for eye detection system with high percentage of accuracy.

II. The suitable value for certain process will determine the effectiveness of this system.
III. The purpose technique by using histogram can perform well and can identify with simplest method.
IV. Image processing is suitable method in develop this system.
V. This system can be compute within a few seconds and that shows the high performance of this system.
VI. All the method and analysis in this project had successfully done which is reliable for any input image.

Compared with existing methods, the space complexity of this method is considerably reduced when applied this method. The result of the analysis shown for this proposed approach is suitable for real-time applications although it is not being done in this project. Later on, for future work hopefully it can be done in real-time.

The main objective of this project had been achieved in order to build a system that can detect a drowsy driver by using histogram analysis.

5.3 Recommendation for future work

For this system it still can make further study for improvement and upgrading the effectiveness and application. This section describes some idea for extending to upgrade this system in this study. There are:

I. Suggested to improve the operation by work with online system whereby the special camera that have the high resolution for a wider application.
II. Beside that it is also able to improve the effectiveness of this system if make a combination recognition starting from face recognition and continue with eye detection and iris recognition. This is very secure for security system with high quality and performance.

REFERENCES

1. US Patent 7336804, Steffin, Morris
2. Fawky, A., Khalil, S., Elsabrouty, M.(2008). *Eye detection to assist drowsy driver.* Information and Communications Technology, 2007. ICICT 2007. ITI 5th International Conference. Available at: http://ieeexplore.ieee.org/
3. Citing Internet sources URL http://amath.colorado.edu/courses/5720/2000Spr/Labs/Worksheets/Matlab_tutorial/matlabimpr.html
4. Citing Internet sources URL http://www.eng.auburn.edu/~sjreeves/Classes/IP/IP.html
5. July 2004, Citing Internet sources URL http://instructional1.calstatela.edu/jdong2/Serenade/imageProcess.html
6. Citing Internet sources URL http://www.nhtsa.dot.gov/PEOPLE/INJURY/drowsy_driving1/drowsy.html
7. Huachun Tan, Yu-Jin Zhang (2005). *Detecting eye blink states by tracking iris and eyelids.* Available at : http://www.sciencedirect.com
8. Jian - Da Wu, Tuo – Rung Chen (2007). *Development of a drowsiness warning system based on the fuzzy logic images analysis.* Available at : http://www.sciencedirect.com
9. Zaini, S. N. (2007). *Automatic Vehicle Detection Car*, Degree Thesis, Universiti Malaysia Pahang.
10. Abdul Aziz, I. Z. (2007). Object's RGB Colour Extraction, Degree Thesis, Universiti Malaysia Pahang.
11. Citing Internet sources URL http://www.mathworks.com/matlabcentral/newsreader/view_thread/276950
12. S.Sirohey, A. Rosenfeld, Z. Duric (2002). *A method of detecting and tracking irises and eyelids in video.* Available at: http://www.sciencedirect.com

13. Guosheng Yang, Yingzi Lin, Prabir Bhattacharya (2010). *A driver fatigue recognition model based on information fusion and dynamic Bayesian network.* Available at: http://www.sciencedirect.com
14. Camus, T., Wildes, R. (2002). *Reliable and fast eye finding in close-up images.* Available at: http://www.sciencedirect.com
15. J. Deng and F. Lai (1997). *Region-based template deformation and masking for eye-feature extraction and description*. Available at: http://www.sciencedirect.com
16. Grauman, K., Betke, M., Gips, J., Bradski, G.R. (2001). *Communication via eye blinks—Detection and duration analysis in real time.* Available at: http://www.**ieee**.org
17. Pardas, M. (2000). *Extraction and ◂tracking▸ of the eyelids.* Available at: http://www.**ieee**.org
18. T. Morris, P. Blenkhorn and F. Zaidi (2002). *Blink detection for real-time eye tracking.* Available at: http://www.sciencedirect.com
19. Kazuhiko Sugiyama, Tomoaki Nakano, Shin Yamamoto (1999). *Method of detecting drowsiness▸ level by utilizing blinking duration*. Available at: http://www.sciencedirect.com

APPENDIX A

Source code for eye detection using MATLAB

```
clc
clear all

% D:\PSM\vidused\eye\ba\ba_0;100 300 180 90
% D:\PSM\vidused\eye\aani
% D:\PSM\vidused\eye\my
% D:\PSM\vidused\eye\wana
% D:\PSM\vidused\eye\aimi

image_num=100;

fid = fopen('D:\PSM\vidused\eye\ba\Dataeye.txt','w+');

text='D:\PSM\vidused\eye\ba\ba_0';

for im=1:image_num
   im;
   filname=strcat(text,int2str(im));
   filname=strcat(filname,'.jpg');
   pic=imread(filname);
   pic2 = imcrop(pic,[100 300 180 90]);
   pic_rgb=rgb2gray(pic2);
   figure(1)

   %%%%%%%display pic%%%%%%%%%%%%%
   subplot(311),imshow(pic_rgb)

   %%%%%%%%%display histogram%%%%%%%%%%
   subplot(312),imhist(pic_rgb)

   %%%%%%%%%threshold value%%%%%%%%%%%%
   a=sum(sum(pic_rgb<50));

   %%%%%%%%data
   data(im)=a;
   subplot(313),plot(data)

   %%%%%%%%%store data%%%%%%%
   fprintf(fid,'%f ',a);
   fprintf(fid,'\t');



%%%%%%%%%decision making%%%%%%%%%%
  if   a<(4500)
    disp('eye is close')
    else
    disp('eye is open')
  end

end

fclose(fid);
```

Appendix B

Table 1: Gantt chart for PSM 1 and PSM 2

No.	**MONTH ACTIVITIES /**	**Month 1**	**Month 2**	**Month 3**	**Month 4**	**Month 5**	**Month 6**	**Month 7**	**Month 8**	**Month 9**	**Month 10**
1	**Seminar Attend**										
	i.PSM1 briefing session by coordinator	■									
2	**Data Collection**										
	i.Find journal and related article of the title of the project	■	■	■	■	■	■	■	■	■	
	ii.Find video or image for data and program for references	■	■	■	■	■	■	■	■	■	
3	**Preparation Report 1**										
	i.Preparation report 1 with additional info.			■	■						
	ii.Submit the report and log book to supervisor				■						
4	**Thesis Writing**										
	i.Preparation of thesis report					■	■	■	■	■	■
	ii.Submit logbook									■	■
	iii.Submit thesis and check by supervisor.									■	■