Implementation of Haar Cascade Classifier and Eye Aspect Ratio for Driver Drowsiness Detection Using Raspberry Pi

Nora Kamarudin[1], Nur Anida Jumadi[1,2,*], Ng Li Mun[1], Ng Chun Keat[1], Audrey Huong Kah Ching[1], Wan Mahani Hafizah Wan Mahmud[1], Marlia Morsin[1], Farhanahani Mahmud[1]

[1]Department of Electronic Engineering, Universiti Tun Hussein Onn Malaysia, Malaysia
[2]Institute of Integrated Engineering, Universiti Tun Hussein Onn Malaysia, Malaysia

Abstract   Driver's drowsiness is one of the leading contributing factors to the increasing accidents statistics in Malaysia. Therefore, the design and development of driver drowsiness detection based on image processing using Raspberry Pi camera module sensor interfacing with Raspberry Pi 3 board are proposed in this paper. To achieve the aim of the research, the Haar Cascade Classifier algorithm is implemented for eyes and face detection whereas for eyes blink (open and close) detection, the Eye Aspect Ratio (EAR) algorithm is employed. From several experiments conducted on six recruited subjects, the findings revealed that the accuracy of Haar Cascade classifier to detect the eyes and faces was subjected to correct sitting position (head must facing to the camera) as well as the eyes must not be covered with glasses or shades. Meanwhile, the range of average EAR value detected by the system was between 0.141 (eyes closed) and 0.339 (eyes opened). In conclusion, the image processing-based Haar Cascade and EAR algorithms utilized on Raspberry Pi platform have been successfully executed. For future improvement, the current board can be replaced with Raspberry Pi Touch Screen to minimize the hardware setup and the physiological based analysis using alcohol and heart rate sensors can be added.

Keywords   Drowsiness, Eye Aspect Ratio, Haar Cascade Classifier, Raspberry Pi, Open CV, Python

1. Introduction

The accident statistics as published in the website of Malaysia Road Safety Department showed an increment from 476,196 cases (2014) to 521,466 cases (2016). The driver drowsiness has been one of the main reasons of road accidents in Malaysia and it can lead to physical injuries or worst; death. Drowsiness is a condition where the person experiences a decreased level of consciousness due to lack of energy. The term drowsy means the sensitivity to fall asleep [1]. Driver fatigue is a major cause in road accident due to long working hours, lack of sleep and medical conditions [2-4]. Driver fatigue can be characterized by behavioural response such as high frequency of eyes blinking and head nodding. Another factor is because of consuming too much alcohol. Alcohol related to crashes could have similar fatality rate with accidents caused by driver drowsiness [5]. Besides, driving for long hours such as in highway driving can be one of the examples for task related factors that con-tribute to driver drowsiness [6]. Because of those factors, drivers and other road users may contribute to a collision because these factors affected driver's behavior, visual, decision-making ability and reaction speed [7].

Researchers have attempted to determine driver drowsiness using the following measures such as vehicle-based measures, behavioural measures and physiological measures [8]. Nowadays, there are current technologies designed for driver's fatigue detection such as the system implemented into the Driver Assistant in Ford cars [9]. It examines speedy steering movements, driving onto lines separating lanes, uneven and hasty braking or acceleration. Besides, there are researchers that used simulated environments for drowsiness manipulation. The main benefits for using simulators driver are experimental control, efficiency, low cost, safety and easy of data collection [10]. The driving simulators can be classified into low-level simulators, mid-level simulators and high-level simulators that provide 360-degree with an extensive moving base [11]. Another technology to

safeguard driver against drowsy driver is by monitoring the car based on metrics of driving performance such as the detection of lane departures, large lateral deviations within the lane and cessation of steering corrections [12].

Eye blinking is one of the signs of driver fatigue and is the major contributing feature in road accidents. Detecting the frequency of eye blinks (open and close) is significant to notice driver drowsiness. There are several previous projects that implemented eye blink detection for instance, it is embedded in the human operator vigilance [13], where it will notify when the user is not blinking for a long time when watching at the screen. Another example is face recognition with eye blink detection to avoid the dry eye and for anti-spoofing protection [14]. Another advanced system is the Optalert Alertness Monitoring System (OAMS) that utilised infra-red (IR) reflectance oculography which is one of the tools to monitor eye-lid movement [15]. It has been used by the driver drowsiness in mining, road transport industries and pilot drowsiness detection in aviation for detection and monitoring them while driving [16-18]. The system used sensor mounted on a spectacle frame and IR emitter to continuously measure eye blink velocity from which level of drowsiness are derive.

Previously, there were several research implement Haar Cascade Classifier and Eye Aspect Ratio on their project. For instance, the Haar Cascade Classifier was investigated for the training procedure on the performance of the author's face and eyes with the goal to design efficient face and eye Haar Cascade Classifier detectors be-fore combining them into the hierarchical system [19]. This algorithm was able to detect and localize area of the eyes based on the set of test images. . On the other hand, another research was implementing the Eyes Aspect Ratio for driving monitoring system [20]. The experiment was being carried out through a video camera that captured image frames of the driver face to be as the input of the system. The purpose was to monitor driver awareness by using face detection and facial landmarks.

The cheapest technology by using microcontrollers such as Arduino and other current technologies such as GSM Module and Bluetooth has been explored too in the previous projects [21-23]. Nevertheless, there are limitations that can be found in these projects. For example, the Arduino output pins were connected to several modules such as GSM Module and Bluetooth HC-05 Module. The GSM Module provides limited data capability and higher data rate for GSM advanced version devices. The input for the prototype also had too many sensors that did not have specific placement for the driver to use it.

From the literatures discussed previously, these previous attempts to detect driver drowsiness can only be found in specific brand car and it can be seen that the majority of the drowsiness detection was based on the driving performance metrics and not based on the behavioural responses of the driver when the drowsiness occurred. Moreover, the types of feedback to the driver in respond to the dangerous driving performance metrics were not mentioned as well.

Therefore, in this project, the design and development of eyes and face recognition as well as eye blinks detection based-behavioural measure for driver drowsiness detection are being proposed using two main components only; Raspberry Pi 3 and Raspberry Pi 8 Mega Pixel camera sensor. The objectives of this research are to implement the Haar Cascade Classifier algorithm for face and eyes recognition, to employ the Eye Aspect Ratio algorithm for detecting eye blinks that indicate the drowsiness and to evaluate both algorithms on various situations using recruited subjects. To achieve the objectives, the scopes of the research have been outlined as follow. The Raspberry Pi 3 is used as the main microcontroller whereas Raspberry Pi 8 Mega Pixel camera sensor is used to capture the images of the subjects. The algorithms are coded using Open CV in python environment. The buzzer is used as simple feedback mechanism to notify the driver when the drowsiness is detected. The prototype, which consists of Raspberry Pi and Pi Camera is casing together and can be located inside the car.

2. Image Processing System: Hardware and Software Apparatus

Figure 1 shows the basic block diagram of driver drowsiness detection based on image processing. It consists of three main stages namely as input, process and output. The main concept of this system is to do real time video monitoring on a driver's face from a camera and is able to accurately calculate the level of driver drowsiness. The input stage contains of Raspberry Pi camera sensor to take video of the eyes and face of the drivers. The process stage involves the software of Raspberry Pi B using Python with Open CV for image processing execution using Haar Cascade and Eye Aspect Ratio algorithms. The output is the drowsiness detection and buzzer rings as a method of alerting the driver. The placement of Pi camera in the car is at the dashboard in front of a steering wheel. It will not disturb the driver as the prototype in small size.
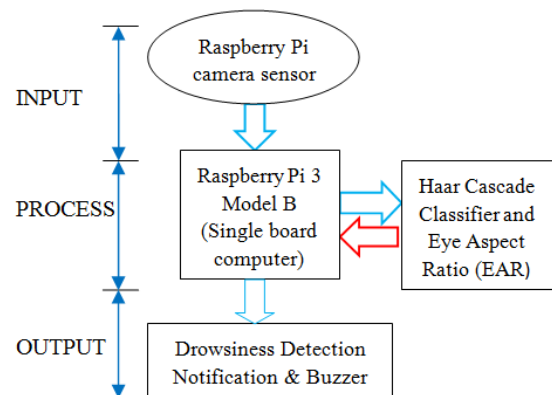

Figure 1.   Block diagram of proposed device

2.1. Hardware Apparatus

The main apparatuses to develop the prototype-based image processing are the Raspberry Pi 3 Model B and Raspberry Pi 8 Mega Pixel camera sensor. Figure 2 shows the Raspberry Pi 3 Model B, which was created by the Raspberry Pi Foundation. The Raspberry Pi 3 Model B is the third generation of Raspberry Pi. Whilst maintaining the popular board format, the Raspberry Pi 3 Model B brings more powerful processer that is 10 times faster than the first-generation Raspberry Pi. Moreover, it is equipped with wireless LAN and Bluetooth connectivity making it the ideal solution for powerful connected designs.

Figure 2. Raspberry Pi 3 Model B

Meanwhile, Figure 3 shows the Raspberry Pi camera sensor that is equipped with a high quality of 8-megapixel Sony IMX219 image sensor. It is capable to capture 3280 x 2464-pixel static images and supports 1080p30, 720p60 and 640x480p60/90 video. Its small size and light (weighs just over 3 g), make it suitable for portable applications. It can be connected to Raspberry Pi through a short ribbon cable.

**Figure 3.** Raspberry Pi 8 Mega Pixel camera sensor

Raspbian Operating System is a free operating system based on Debian optimized for the Raspberry Pi hardware. It contains a set of basic programs and utilities that run the Raspberry Pi. Python is a programming language that can be used to write computer program in Raspberry Pi while IDLE is an application that creates and runs the Python programs. Python IDLE software with extensions of Open CV computer vision is used to program the Haar Cascade and EAR algorithms. In order to initiate the program execution, the following libraries such as NumPy, Open CV, and picamera are needed to be installed. The Open CV is released under a BSD license and free for both academic and commercial use. It has C++, C, Python and Java interfaces and supports Windows, Linux, Mac OS, IOS and Android. It is designed for computational efficiency and with a strong focus on real time application. It is an open-source computer vision project that aims to provide a development platform for computer vision algorithms with collection of libraries and applications. It provides I/O libraries for easy video data acquisition and manipulation from multiple camera inputs.

3. Development of Image Processing System

3.1. Image Processing

Face and eye detection is a very important and challenging matter in the field of image processing. It is also a crucial step of face recognition. Open Source Computer Vision Library (Open CV) is used to implement the Haar Cascade Classifier. In this project, the driver drowsiness detection requires a video sensor to detect the faces of drivers. Then the drowsiness level of drivers can be determined by checking the eye blinking rate. Figure 4 illustrates the methods involving the implementation of detection of face and eye including eye blinks using Haar Cascade Classifier and Eye Aspect Ratio, respectively. In Haar Cascade Classifier, there are four main steps that need to be executed. The steps are Haar Feature, Integral image, AdaBoost and Cascade Classifier. As for Eye Aspect Ratio, it is used to detect the eye blinks (eyes open and close) using the ratio formula based on the eyes width and height.
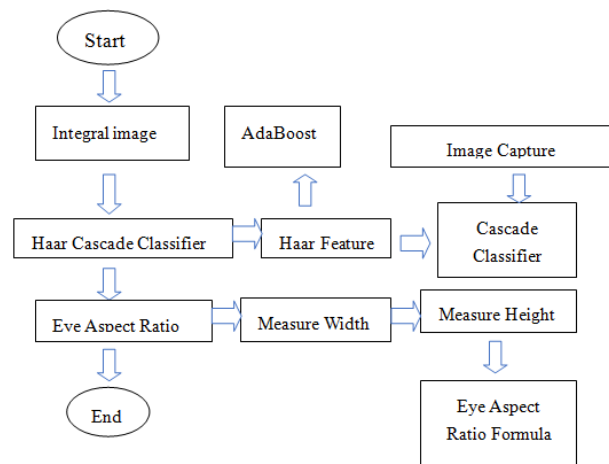
Figure 4. Methods involving face and eye as well as eye blinks detection using Haar Cascade classifier and Eye Aspect Ratio, respectively

On the other hand, Figure 5 shows the overall flowchart of face and eye including EAR detection when the system is operated. The acquisition of real-time video from the camera is done by using command raspistill. As explained previously, the Haar Cascade Classifier is used to detect face and eyes images whereas the EAR formula is being implemented to detect the blinking of eyes. Finally, the buzzer is implemented in this prototype as a simple mechanism to alert the driver when the EAR value output is approximate to 0 for two seconds long.
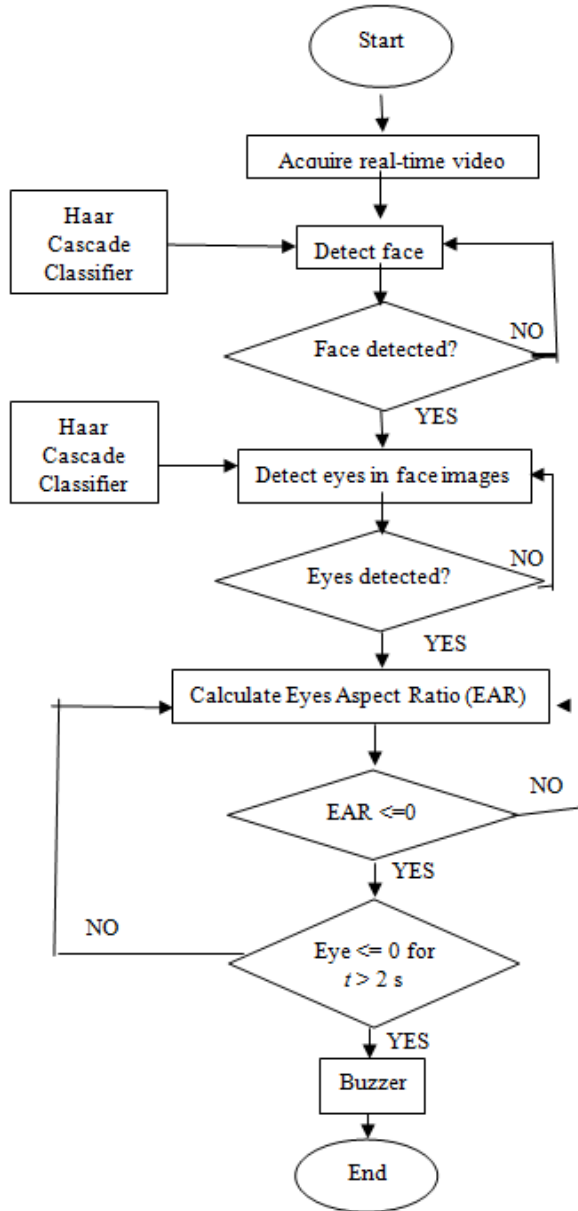
3.2. Haar Cascade Classifier

Haar Cascade Classifier is one of the few object detection methods with the ability to detect faces. It offers high-speed computation depending on the number of pixels inside the rectangle feature and not depending on each pixel value of the image. This method has four steps for detecting an object namely as Haar-like feature, integral image, AdaBoost learning and Cascade Classifier [24]. For the detection of the face, Haar features are the main part of the Haar Cascade Classifier. The Haar features are used to detect the presence of feature in given image. Each feature results in a single value, which is calculated by the sum of pixels under black rectangle. The Haar-like feature is a rectangular feature providing specific indication to an image for rapid face detection [25]. Figure 6 shows the examples of common variety of Haar-like features.
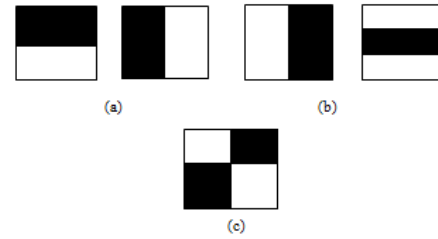

Figure 6. (a) Edge feature, (b) Line feature and (c) Four-Triangle feature

In obtaining object detection value, Haar-like feature value was calculated using integral image. It starts scanning the image for the detection of the face from the top left corner and ends the face detection process at the right bottom of image in order to detect the face from an image as shown in Figure 7(a). The integral image could calculate values accurately and relatively quick by creating new presentation of image by using value of region previously scanned by specific Haar-like feature as depicted in Figure 7(b). The value at any point (x,y) is the summed area table of the sum off all the pixels above and to the left of (x,y), inclusive as shown in Equation (1).

$$I(x, y) = \sum_{\substack{x' < x \\ y' < y}} i(x', y') \tag{1}$$

where i(x,y) is the value of pixel at (x,y) whereas I(x,y) is the sum of integral of pixel values. The value of integral image, I(x,y) is obtained by sum value previous index, starting from the left top until right bottom. Moreover, the summed-area table can be computed efficiently in a single pass over the image, as the value in the summed-area table at (x,y) in Equation (2) [26].

Figure 5.   Overall face and eye including EAR detection flowchart

$$I(x, y) = i(x, y) + I(x, y-1) + I(x-1, y1) + I(x-1, y-1) \quad (2)$$

	1	2	3	4
1	3	7	7	3
2	1	3	3	1
3	5	9	9	5
4	3	6	6	3

(a)

|	1	2	3	4
1	3	10	17	20
2	4	14	24	28
3	9	28	47	56
4	12	37	62	74

(b)

Figure 7. (a) Input image and (b) Integral image [26]

Once the summed-area table had been computed, evaluating the sum of intensities over any rectangular area requires exactly four array references regardless of the area size. Equation (3) shows the sum of i(x,y) over the rectangle spanned by A, B, C and D.

$$\sum_{\substack{x0<x<x1 \\ y0<y<y1}} i(x, y) = I(D) + I(A) - I(B) - I(C) \quad (3)$$

The notation in the Figure 8 shows the example of the description of computing a sum in the summed-area table data structure having A=(*x0,y0*), B=(*x1,y0*), C=(*x0,y1*) and D=(*x1,y1*).
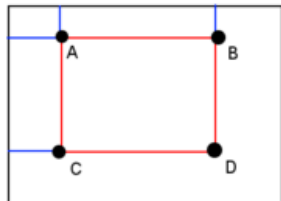
Figure 8. Description of computing a sum in the summed-area table data

Value which has been calculated by using integral image would then be compared with threshold value of specific features provided by AdaBoost. This should be completed to find potential features because not all features were relevant to use for specific object detection. AdaBoost combines potential features called weak classifier to become strong classifier. The Cascade classifier can be divided into two which are strong classifier and weak classifier. Weak classifier means less accurate or also irrelevant prediction and strong classifier means more accurate or relevant prediction. Strong classifier made by AdaBoost can detect object level by level on a cascade.

3.3. Eye Aspect Ratio (EAR)

After detecting the face of the driver, the calculation of drowsiness level of the driver is based on eye blink rate. The Eye Aspect Ratio (EAR) formula, which was proposed in [27] is able to detect the eye blink using the scalar value. For instance, if driver blinks eyes more frequently, it means that the drivers are in the state of drowsiness. Thus, it is necessary to detect the eyes shape accurately in order to calculate the eye blink frequency. From the landmarks detected in the image with face, the EAR is used as an estimate of the eye openness state. For every video frame, the eye landmarks are detected between height and width of the eye that had been computed. The eye aspect ratio can be defined by the Equation (4).

$$EAR = \frac{\left| p2 \quad p6 \right| + \left| p3 \quad p5 \right|}{2 \left| p1 \quad p4 \right|} \quad (4)$$

Equation (4) shows the eye aspect ratio formula where p1 until p6 are the 2D landmark locations. The p2, p3, p5 and p6 are used to measure the height whereas p1 and p4 are used to measure width of the eyes in meter (m) as shown in Figure 9(a). The eye aspect ratio is a constant value when the eye is opened, but rapidly falls approximately to 0 when the eye is closed as shown in the Figure 9(b).

(a)   (b)
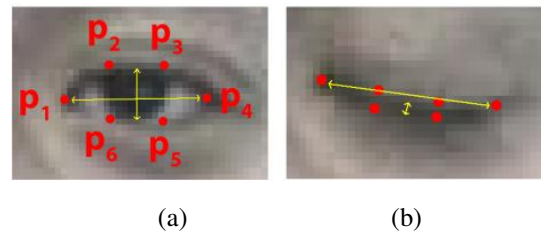
Figure 9. (a) Open and (b) Close eye with landmarks

$$EAR = \begin{cases} x > 0; & \text{eyes open} \\ 0 & ; \text{eyes close} \end{cases} \quad (5)$$

Equation (5) shows the EAR output range during eyes open and eyes close. During eyes closed the EAR result will be approximately to 0 whereas during eyes open the EAR can be any integer number which is *x* greater than 0.

## 4. Results and Discussion

The results and discussion are presented in this section.

The main findings resulted from the Haar Cascade algorithm and EAR formula are discussed based on the various experimental analysis. Figure 10(a) and Figure 10(b) show the Raspberry Pi operating setup with User Interface Development and the camera sensor interface with Raspberry Pi respectively. To emphasize, this experiment setup is not designed for the real driving situation as the main purpose of this research is to investigate the implementation of Haar-Cascade and EAR algorithms in the Raspberry Pi environment. That is why the desktop is used to display the real-time video from the camera. Total six subjects (3 male and 3 female) were recruited for the designed experiments and the informed consents were obtained. All subjects were given brief explanation on the procedures involved prior to the experiments. There were four experiments that were carried out in order to evaluate the performance of the Haar Cascade classifier and the EAR algorithms.

(a)
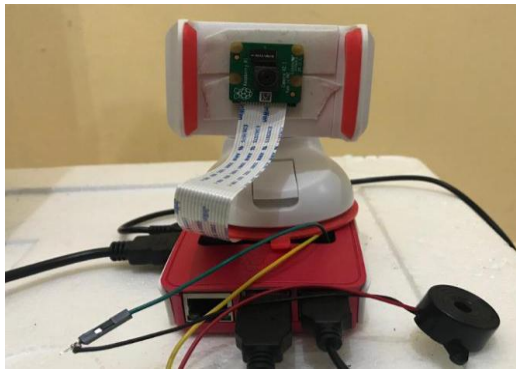
(b)

Figure 10.   (a) Raspberry Pi Operating System Setup and (b) Camera Sensor Interface with Raspberry

4.1. Real-Time Video Acquisition from Pi Camera

The first experiment was to examine whether or not the real-time video could result in good quality of video images. Figure 11 shows the screenshot results from the real-time video using different subjects; female and male using Raspberry Pi camera module. The image was successfully captured in the size of 1920 pixels wide by 1080 pixels high.


Figure 11.   Successful screenshot video images from Raspbery Pi Camera

4.2. Face and Eye Detection using Haar Cascade Classifier

The second and third experiments were to assess the accuracy of the Haar Cascade Classifier to detect eyes and face by asking the subjects to sit in various positions as well as by wearing glasses and shades, respectively. The positions of subject to be observed such as when standing straight, head moving to the left, head moving to the right, head nodding to the left and head nodding to the right. Besides that, the subjects were asked to wear spectacles and shades to assess whether or not the Haar Cascade Classifier can still detect the eyes shape.

4.2.1 Different Positions of Subjects

Different position of subjects is important to analyse the detection of face and eyes using Haar Cascade Classifier. The results based on these different positions can be referred to Figure 12(a) and 12(b) for standing straight, 12(c) head nodding to the left and 12(d) head nodding to the right whereas 12(e) head moves to the left and 12(f) head moves to the right. These findings showed that the Haar Cascade classifier was unable to detect the face and eyes when the subjects were not in the position of sitting straight looking at the camera. Any movements done by the subjects will affect the resulted output.

(a)   (b)

(c)   (d)

(e)   (f)
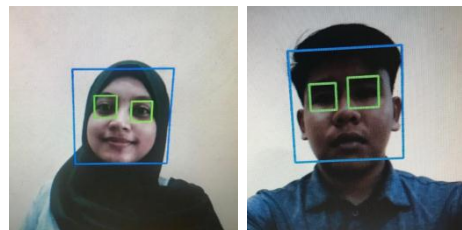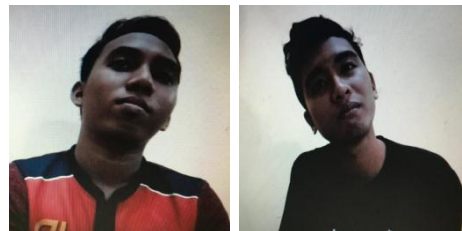
Figure 12.   (a) and (b) are for standing straight, (c) head nodding to the left, (d) head nodding to the right, (e) head moving to the left and (f) head moving to the right

4.2.2. Subjects Wearing Spectacles and Shades

Figure 13(a) and 13(b) show the comparison results when the subjects wore spectacles and shades. The findings revealed that when the subject wore spectacles, the Haar Cascade classifier could only detect face (unable to detect eyes) whereas when the subject wore shades, the algorithm could detect eyes and face but for eyes detection was inaccurate as it detected the size of the shades instead of eyes (as outlined by the green box in Figure 13(b)).


(a)   (b)
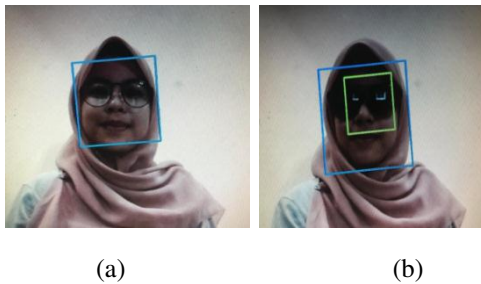
Figure 13.   (a) Image of wearing spectacles and (b) Image of wearing shades

4.3. Eyes Open and Close using EAR formula

The fourth experiment was to evaluate the EAR formula by measuring the eye landmarks detection. Based on the formula, it can be seen that if the EAR value is suddenly dropped, the driver may close their eyes. The EAR was calculated based on Equation (4) for each consecutive video frame and a threshold for EAR was set in the code. The numerator of this equation computes the distance between the vertical eye landmarks while the denominator computes the distance between horizontal eye landmarks. Then, the EAR values for both the eyes were calculated and then the mean for both eyes' EAR values was displayed on the screen.

As mentioned previously, during eyes closed, the EAR result will be approximately to 0 whereas during eyes open, the EAR can be any integer number which is x greater than 0. Figure 14 shows the real time calculation of EAR during eyes opened and closed... It can be seen that the value of EAR appeared on the screen as in Figure 14(a) that was 0.367 when the eyes opened (not drowsy). Meanwhile, the value of EAR was low (0.081) when the eyes closed

(indicate drowsiness). A drowsiness alert notification can be seen in Figure 14(b).

(a)

(b)
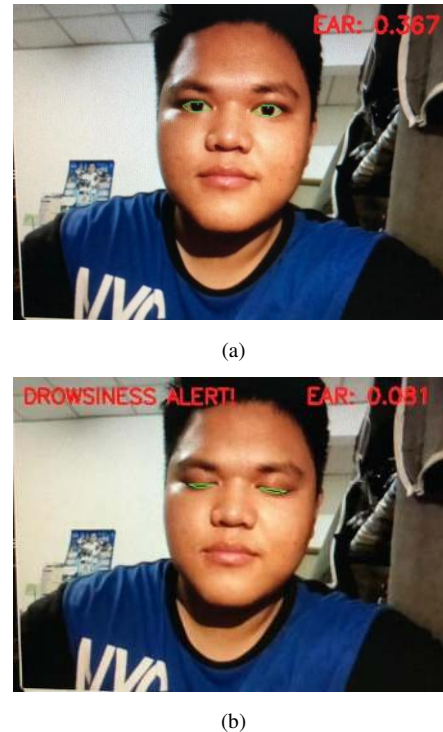
Figure 14.   Example of real time EAR calculation during (a) eyes opened (EAR=0.367) and (b) eyes closed (EAR=0.081). A notification of drowsiness detection was popped out in the screen during eyes closed.

Table 1.   The average value of EAR during eyes opened and closed from six subjects

Subjects	EAR during eyes open	EAR during eyes closed
1	0.367	0.081
2	0.321	0.083
3	0.362	0.154
4	0.343	0.182
5	0.305	0.177
6	0.335	0.171
Average	**0.339**	**0.141**

Table 1 shows the calculated average value of EAR obtained from the six subjects. The average eye aspect ratio is 0.339 and 0.141 when the eyes opened and closed, respectively. From this experiment, it can be noticed that zero average EAR value could not be obtained during eyes closed. Nevertheless, based on the EAR formula, it can be deduced that if the EAR value is suddenly decreased, it implies that the driver probably in the state of closes his eyes. Therefore, the decreasing value of average EAR from 0.339 to 0.141 indicates the sign of eyes blink or eyes close as proved in Figure 14.

Meanwhile, Table 2 shows the time taken by buzzer to respond when detecting eyes closed from the same six subjects and the average time was 0.039 s. The quickest respond time (less than 1 second) has proved that the buzzer is sufficient to provide simple feedback mechanism for alerting the driver.

Table 2. The average time taken by buzzer to respond when the eyes closed from six subjects

Subjects	Time taken by buzzer to respond after detect the eyes closed (s)
1	0.030
2	0.039
3	0.040
4	0.044
5	0.043
6	0.036
Average buzzer respond time	**0.039**

5. Conclusions

In conclusion, all objectives that were set in the beginning of the research were successfully achieved. The implementation of image processing operations based on Haar Cascade Classifer and Eye Aspect Ratio (EAR) algorithms on Raspberry Pi to detect drowsiness has been successfully developed and executed in this study. This research was conducted to propose a prototype of hardware platform using Raspberry Pi 3 Model B and Raspberry camera sensor to run image processing for detection of driver drowsiness. These two main algorithms were coded using Python with Open CV. In order to assess the performance of Haar Cascade Classifier algorithm in detecting face and eyes, six subjects were recruited to perform several specific experiments. It turned out that the Haar Cascade classifier was unable to detect the face and eyes if the subjects moved or wore glasses/shades. As for eye blink detection; the resulted EAR value was around 0.339 when the eyes opened whereas when the eyes closed, the EAR obtained was 0.141. Although, zero average EAR value could not be obtained, the sudden dropped value in average EAR indicates that the driver was in the state of drowsiness. As mechanism to alert the driver, the drowsiness notification was popped out on the screen and the buzzer immediately rang in less than 1 second after detecting the eyes closed.

As for recommendation, the current set-up for user interface includes monitor, mouse and keyboard, HDMI cable, power supply micro USB B and 8 GB micro SD card that can be replaced with Raspberry Pi 7" Touch Screen Display to minimize the hardware setup during real driving situation. Implementation of designed prototype to be used in the car can be done if Raspberry Pi Touch Screen is employed since the current set-up requires the use of screen monitor. Besides that, sensors such as alcohol sensor and heart rate sensor to detect alcohol and pulse rate of the driver can be added for enhancement in physiological-measure analysis. Other than that, a better face and eye recognition algorithm should be implemented since the Haar Cascade is not giving accurate results whenever the subjects make any specific movements. Finally, once the prototype is improved, its performance can be investigated in a real driving environment.

Acknowledgements

REFERENCES

[1] A. Sahayadhas, K. Sundaraj, and M. Murugappan, (2012). "Detecting driver drowsiness based on sensors: A review," Sensors (Switzerland), vol. 12, no. 12, pp. 16937–16953, doi:10.3390/s121216937

[2] Fischer, Frida Marina, Claudia Roberta, De Castro Moreno, Flavio Notarnicola, and Fernando M Louzada, (2000). "Implementation of 12-hour shifts in a Brazilian Petrochemical Plant: Impact on sleep and alertness." 17, 521-537, doi:10.1081/CBI-100101062

[3] Dawson, Drew, and Kirsty McCulloch, (2005). "Managing fatigue: It's about sleep." 365, 80, doi:10.1016/j.smrv.2005.03.002

[4] Smolensky, Michael H, Lee Di, Maurice M Ohayon, and Pierre Philip, (2011) "Sleep disorders, medical conditions, and road accident risk." Accident Analysis & Prevention, 533-548, doi:10.1016/j.aap.2009.12.004

[5] Pack, Andrew M, Eric Rodgman, Allan I Pacic, David F Dinges, and C William Schwab, (1995) "Characteristics the driver of crashes attributed having fallen asleep to" 27, 769-75, doi:10.1016/0001-4575(95)00034-8

[6] Williamson, Ann, David A Lombardi, Simon Folkard, Jane Stutts, Theodore K Courtney, and Jennie L Connor, (2011) "The link between fatigue and safety. " Accident Analysis & Prevention 43, 498-515, doi:10.1016/j.aap.2009.11.011

[7] Saeed, Seyed, Hashemi Nazari, Ali Moradi, and Khaled Rahmani, (2012) "Original Article A Systematic Review of the Effect of Various Interventions on Reducing Fatigue and Sleepiness While Driving," Chinese Journal of Traumatology. - English Ed., vol. 20, no. 5, pp. 249–258, doi:10.1016/j.cjtee.2017.03.005

[8] Sedighi, Zahra, Mohammad Ali, Alamdar Yazdi, Lora A Cavuoto, and Fadel M Megahed, (2017) "A data-driven approach to modeling physical fatigue in the workplace using wearable sensors," Applied Ergonomics., vol. 65, pp. 515–529, doi:10.1016/j.apergo.2017.02.001

[9] Sałapatek, Damian, Jacek Dybała, Paweł Czapski, and Paweł Skalski, (2017) 'Proceedings of the Institute of Vehicles 3(112)', 41–48

[10] Auberlet, Jean-michel, Florence Rosey, Sébastien Aubin, Patrice Briand, Marie-pierre Pacaux, and Patrick Plainchault, (2012)"The impact of perceptual treatments on driver's behavior: From driving simulator studies to field tests - First results. vol. 45, no pp. 91–98, doi:10.1016/j.aap.2011.11.020

[11] N. Kaptein, J. Theeuwes, and R. Van Der Horst, (1996)

"Driving Simulator Validity: Some Considerations," Transp. Res. Rec. J. Transp. Res. Board, vol. 1550, no. November, pp. 30–36.

[12] Forsman, Pia M, Bryan J Vila, Robert A Short, Christopher G Mott, and Hans P A Van Dongen, (2013) "Efficient driver drowsiness detetion at moderate levels of drowsiness." Accident Analysis & Prevention 50, 341-350, doi:10.1016/j.aap.2012.05.005

[13] Danisman, Taner, Ian Marius Bilasco, Chabane Djeraba, and Nacim Ihaddadene. (2010) "Drowsy driver detection system using eye blink patterns." Machine and Web Intelligence (ICMWI), International Conference on. IEEE, 230-33, doi:10.1109/ICMWI.2010.5648121

[14] Pan, Gang, Lin Sun, and Zhaohui Wu, (2007) "Eyeblink-based anti-spoofing in face recognition from a generic web camera.1-8, doi: 10.1109/ICCV.2007.4409068

[15] Johns, Murray W, Monitoring Von Augen Augenlid and Zusammenfassung Fragestel, (2007) "Monitoring eye and eyelid movements by infrared reflectance oculography to measure drowsiness in drivers." Somnology Sleep Research and Sleep Medicine, 242, 234-242, doi:10.1007/s11818-007-0311-y

[16] Williamson, Ann, David A Lombardi, Simon Folkard, Jane Stutts, Theodore K Courtney, and Jennie L Connor, (2011) "The link between fatigue and safety." Accident Analysis & Prevention 43, 498-515, doi:10.1016/j.aap.2009.11.011

[17] Corbett, Mark, and M A Corbett, (2015) "Science & Technology Watch: A Drowsiness Detection System for Pilots: Optalert Science and Technology watch a Drowsiness Detection System for Pilots." 80: 149-149, doi:10.3357/ASEM.21001.2009

[18] Aidman, Eugene, Carolyn Chadunow, Kayla Johnson, and John Reece, (2015) 'Real-Time Driver Drowsiness Feedback Improves Driver Alertness and Self-Reported Driving Performance', Accident Analysis and Prevention, 81, 8–13, doi:10.1016/j.aap.2015.03.041

[19] Kasinski, Andrzej, and Adam Schmidt (2010) , 'The Architecture and Performance of the Face and Eyes Detection System Based on the Haar Cascade Classifiers', 197–211, doi:10.1007/s10044-009-01505

[20] Constantin, Leonte, Cristian Fosalau, Cristian Zet, and Daniel Petrisor, (2018) 'Driver Monitoring Using Face Detection and Facial Landmarks', International Conference and Exposition on Electrical And Power Engineering (EPE), 385–90.

[21] L. J. Joseph, (2016) "Arduino based real time driver drowsi-ness detection and mobile alert system using bluetooth," Int. J. Eng. Comput. Sci., vol. 5, no. 09, pp. 18106–18111.

[22] Ciu, Xiaoning, Zhang, Chen, Liu, (2018) "Pulse sensor based on electrically conductive natural fiber yarns,"Sensors and Actuator, vol 3, no.5, pp. 326-331.

[23] Souri, Hamid, Bhattacharyya, Debes (2018), "Wearable strain sensors based on electrically conductive natural fiber yarns," Materials and Design, vol 7, no.154, pp. 217-227, doi: 10.1016/j.matdes.2018.05.040.

[24] Viola, Paul, and Michael Jones, (2001) "Rapid object detection using a boosted cascade of simple features." Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on. Vol. 1. IEEE, doi:10.1023/B:VISI.0000013087.49260.fb.

[25] P. Viola and M. M. J. Jones, (2004) "Robust Real-time Face Detection," Int. J. Computer. Vis., Vol. 57, No. 2, Pp. 137–154.

[26] P. Viola and M. M. J. Jones, (2004)"Robust Real-time Face Detection," Int. J. Computer. Vis., Vol. 57, No. 2, Pp. 137–154.

[27] Soukupova, Tereza, and Jan Cech. (2016) "Eye Blink Detection Using Facial Landmarks." 21st Computer Vision Winter Workshop, Rimske Toplice, Slovenia.