*International Journal of Engineering Applied Sciences and Technology, 2019*
**Vol. 4, Issue 7, ISSN No. 2455-2143, Pages 80-87**
Published Online November 2019 in IJEAST (http://www.ijeast.com)

# DRIVER DROWSINESS DETECTION USING MATLAB

Hanojhan Rajahrajasingh
Department of Electrical & Electronic Engineering
ICBT Campus, Colombo, Sri Lanka

*Abstract*— **When a driver doesn't get proper rest, they fall asleep while driving and this leads to fatal accidents. This particular issue demands a solution in the form of a system that is capable of detecting drowsiness and to take necessary actions to avoid accidents.**

**The detection is achieved with three main steps, it begins with face detection and facial feature detection using the famous Viola Jones algorithm followed by eye tracking. By the use of correlation coefficient template matching, the eyes are tracked. Whether the driver is awake or asleep is identified by matching the extracted eye image with the externally fed template (open eyes and closed eyes) based on eyes opening and eyes closing, blinking is recognized. If the driver falling asleep state remains above a specific time (the threshold time) the vehicles stops and an alarm is activated by the use of a specific microcontroller, in this prototype an Arduino is used.**

*Keywords*— **Drowsiness Detection, Image Processing, Matlab, Eye Detection**

## I. INTRODUCTION

A micro sleep is something we all may have experienced at least once in our life, where we momentarily fall asleep and then immediately awake up. And it is a very common thing among vehicle drivers, they fall asleep and awake up immediately and continue with their previous activity and it will not be an issue as long as it is a straight road and there is no vehicle ahead of them or behind them but at the same time if the driver is driving along a curve or the motor vehicle ahead of the driver is applying a sudden brake then the reflex action by the driver will not be fast enough to take appropriate action and mostly result in an accident.

Accidents due to sleep deprivation is a common problem all around the world. According to a database study conducted by the Australian Transport Bureau in 2002 about fatality crashes [1] it is found that 16.6% of fatal crashes were caused by sleep deprivation and based on the report submitted by Ministry of Transportation of Ontario in 2004 [2], 25.5% of injury causing crashes and 17.8% of crashes with fatality related to sleep deprivation. All of the above statistics is an estimate because in severe cases the drivers will not admit the real cause to the Police.

The Sri Lankan law on driver fatigue is described in Section 169 of the Motor Traffic Act [3] [4], which reads as follows:

1. No individual shall drive, cause or allow any individual to ride any motor tricycle hired by him/her or subject to his/her instructions. Van, truck or motor coach:

- For any stretch of higher than four and a half hours continuously; or

- Consequently, the vehicle driver does not have a minimum 10 continuous hours of sleep in any twenty-four-hour period calculated from the beginning of any driving period.

2. Because of the purposes of the subsection of the first point, any two or more-time period will not be considered as a consecutive period of time, except when the time periods are splited by a break of minimum thirty minutes in which the driver will be able to get some refreshment and rest.

In reality the majority of the people does not follow any of the above law and it is very difficult even for a traffic Police officer to penalize the driver with any of the above law. And this is where technology comes in, there have been a lot of devices that were invented around the world to reduce this cause but most of them are very expensive or not available in the local market. This paper, discusses about a system that uses computer vison and image processing to control the damage caused by the above stated phenomenon.

## II. IMAGE PROCESSING

### A. IMAGE ACQUISITION –

The camera is used at this point to obtain the driver's picture. An adjustment is created in the camera center. The camera is adjusted at angle of 50 ° in order to capture a clear picture and it uses the light integrated with the camera when it is essential.

In this phase, which involves connection of the camera with MATLAB, a command named imaqhwinf is executed. This particular command is used to obtain the camera name in MATLAB, this is named “VoilaVideo” in this case. At last, the “imaqhwinf (‘ VoilaVideo ‘, 1)” line of code is executed in order to verify both the linked camera’s features and the resolution.

Two environments are selected to capture the pictures, the firstly in the day-time and then secondly in the night-time. The face of the driver is focused during the image acquisition stage.

### B. IMAGE PREPARATION –

According to (Machaca Et al, 2018) [6] at this point, when processed in MATLAB, the image is converted to a two-dimensional matrix. Each matrix element corresponds to an image pixel.

- A grayscale image is a matrix of m x n size in which each element of the matrix is an illumination value image pixel.
- An RGB color image is a matrix of m x n x 3 size.
- A hyperspectral image is a matrix of m x n x l.

Since this video input is used in this project, in addition to the above-mentioned dimensions, a time dimension will be added. The basic thing is to prepare the image for efficient detection. The method starts with changing the image's features to obtain an image enhancement. The features are: the noise and the contrast in the image. At the same time, before detection, several filters are applied to the image to reduce noise.

In order to apply filters to an image firstly, the 2D spatial domain image should be converted to a 2D frequency domain. This is done by using DFT.

The Discrete Fourier Transform is basically a sampled Transform and because of this, it does not include all image-making frequencies, instead it contains only a group of sampled frequencies which are sufficient enough to give a detailed account about the image of the spatial domain. At this point, each number of the frequencies relates to each number of the pixels of the image of the spatial domain, given that the image is of the same size in the spatial domain and Fourier.

The equation below is used to convert a square image with size N×N into its two-dimensional Discrete Fourier Transform [5].

$$F(k,l)=\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} f(a,b)e^{-t2\pi\left(\frac{ka}{N}+\frac{b}{N}\right)}$$

In which f(a, b) is the spatial domain image and the exponential term is the elemental function of each of the points F(k, l) in Fourier space. Therefore, this equation can be interpreted by multiplying the spatial image with the respective base function and after that summing up the outcome, the value of each of the points F(k, l) is acquired.

### C. DETECTION

An algorithm is developed in the software to generate a frame with a square for the face, indicating a face image for image processing as shown in Figure 1. And this particular method is performed to detect drowsiness patterns more efficiently. To make face detection more accurate, feature-based technique was selected rather than other techniques. Of all the haar-like features, face detection is regarded easy and efficient. Because of the face's haar-like characteristics, we used the face detection method based on the cascade of adaboost technique-trained classifiers. There are several methods, such as cascade and integral image, for better face detection results.

The integral image at k, l contains the sum of the above and left pixels of k, l, including:

$$ii(k,l)=\sum_{k'\leq k,l'\leq l} i(k',l')$$

Where ii(k,l) is the integral image and i(k^',l^') is the original image, using the following repetition pair:

$$S(k,l)=S(k,l-1)+i(k,l)$$
$$ii(k,l)=ii(k-1)+s(k,l)$$

Where S(k,l) is the sum of cumulative rows.

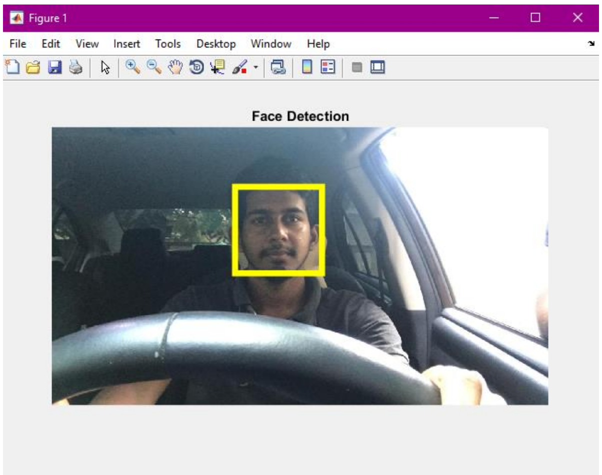


Figure 1 Face detection

Once the face has been detected, it is cropped out in order to detect the eyes on it. Eye region receives an evident function on the face's horizontal projection curve, according to research. Therefore, the face region is removed from the image to obtain the location of the eyes and then the face horizontal projection is worked out [10].

For an image with X number of rows and Y number of columns, the horizontal projection is a sum of the intensities of the pixel in each line.

$$Vp = \{Vp_k | 1 \le k \le X\}$$

$$Vp_k = \sum_{l=1}^{Y} f(k,l)$$

Where Vp is the value of horizontal projection.

By the application of Harris method, the pupil is roughly located. After the pupil is identified in the initial frame, the region of eye is located by kallman filtering.

Firstly, the extraction of the eye candidates happens by using the following steps. It starts with identifying the values denoted by A, B, C by using the below equation.

$$A = (I_x)^2, B = (I_y)^2, = (I_x I_y)$$

Where $I_x, I_y$ represents the gradient of the image in the respective, x and y directions.

Following that, the value of R is obtained by using the equation below.

$$R = Det(M) - KTr^2(M)$$

$$Tr = \alpha + \beta = A + B, Det(M) = \alpha\beta = AB - C^2$$

The points with the highest R values are selected as the suitable eye candidates points in the specified region. Among these points, the eye area is well detected in the next phase. Color entropy is used to remove irrelevant ones in the eye region.

The eye image has many characteristics. Variations in the concentrations of lightness and the shape of the eye element make it difficult to detect these characteristics. One helpful factor for identifying eye location is the uneven variation in sclera lightness concentrations. In this region, the blood vessels add to this entropy on the sclera surface. Some points with the largest R values are indicated to guarantee right pupil detection. These points are regarded as candidates for pupils. Color entropy is assessed for each candidate to eliminate the ones irrelevant to the pupil, and then color entropy calculation happens. Entropy probability is then derived from equation given below.

$$H(S) = -\sum_{i=1}^{n} p(x_i) log_2^{p(x_i)}$$

Where $p(x_i)$ represents the probability of the pixel of $x_i$. Each pixel's probability function, $p(x_i)$, is the pixel intensity frequency in the selected area divided by the total pixel frequency. Two regions in the candidates for eye with the highest entropy are concluded as the region including the pupil. The center of these regions is considered as the pupil's center.

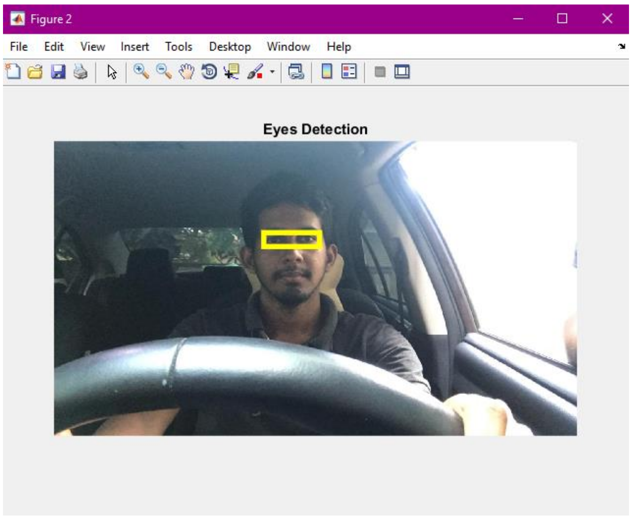


Figure 2 Eyes detection

After the eyes have been detected, the system will continue to identify patterns of drowsiness. The separation of the ROI as the eyes will start processing the patterns. Then the extraction method starts, this analyze the closing time of the eye and the opening distance of the eye.

As it can be seen in Figure 4, the program requires 4 frames, in which the first frame captures the driver's eyes, places the eyes inside second one, and lastly, the program analyzes that the eye is closed in the third frame and the externally fed template is in the fourth frame. The template is used as a reference to see if the eyes are opened or closed.

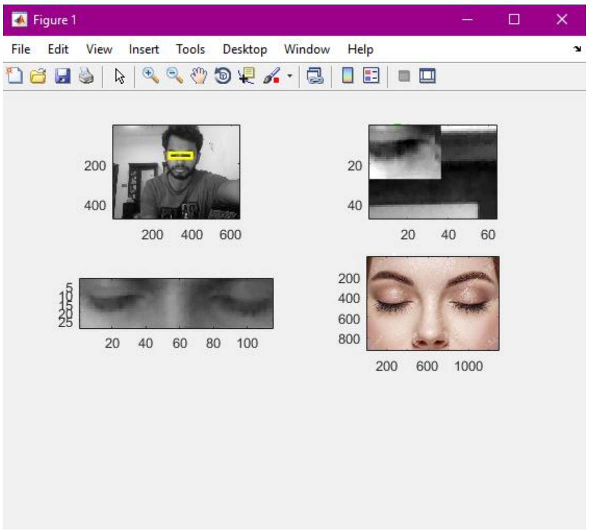


Figure 3 Image processing with closed eyes.

The process demonstrated by Figure 4 same as the earlier one, except the eyes are open in this situation which can be identified by a green circle in the third frame.

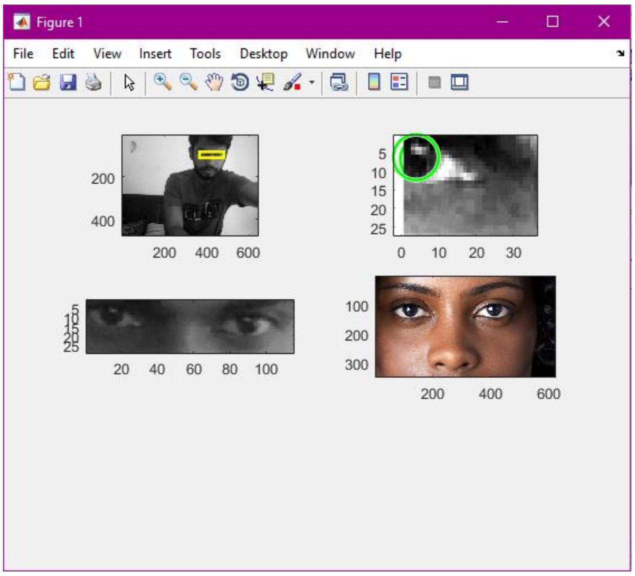


Figure 4 Image processing with open eyes.

Blinking rate of eyes is used as the key element to identify drowsiness pattern. Blinking of eyes are influenced by two major factors, they are, rate of blinking and opening of the eye. The below two equations were developed based on the two factors mentioned,

Rate of blinking or the flicker frequency is the number of blinks made by the driver during a given time. According to the below equation

$$Rate\ of\ blinking = \frac{number\ of\ blinking}{time\ taken}$$

In an individual with drowsiness, the blinking frequency is about twenty-one per minute whereas the individual blinks only about fifteen times per minute in the ordinary state.

Opening of the eye is the amplitude in the closing point between the two eyelids when the eyes are opened and closed. The process starts when the eyelids cover the pupils of the eyes. That the eyelids above and below are open.

And PERCLOS is used to calculate this factor to determine the percentage of eyelid closure. The following mathematical formula is used as described in equation 2.

$$Percentage\ of\ closing\ of\ the\ eye = \frac{t3 - t2}{t4 - t1}\ X\ 100\%$$

## III. HARDWARE IMPLEMENTATION

Once the drowsiness in driver is detected, necessary steps should be taken immediately to avoid fatal accidents. In this project two different techniques are used to handle this situation. Firstly, an alarm will be activated and then the vehicle itself will be controlled using the accelerator and the brake.

### D. ALARM-

The device will generate an alarm sound after detection to alert the individual for falling asleep, by the use of Arduino and a buzzer in this prototype. The alarm will vary depending on the pattern identified corresponding to level of sensitivity detected in the individual. The driver is therefore not accustomed to the monotonous tone of the alarm and as a result ignore the warning.

### E. SPEED CONTROL-

Once the driver is identified as falling asleep the speed of the vehicle should be reduced to considerably lower rate. This is achieved by first locking the accelerator by using a dc motor. In this prototype, the shaft of the motor is attached to a rectangular plastic block with a curved edge, which is aligned at 60 degrees from the accelerator underneath it. The moment the driver is identified as not fit to drive, this block will turn 60 degrees upwards, locking the accelerator in the upward position (no acceleration). Following this another block which is aligned 40 degrees from the brake above it, will turn 40 degrees below, pressing the brake and bringing down the speed to a considerably lower rate. Along with these processes, two proximity sensors attached to the front of the vehicle and underneath it will monitor the distance between obstacles and the vehicle. If the obstacle is less than 15 cm from the vehicle (forward) then the brake of the vehicle is pressed fully and the vehicle is brought to rest. Similarly, if the reading from the proximity sensor underneath the vehicle is above 2 cm (this value is for the prototype used in this project, this may vary from vehicle to vehicle) it suggests that the vehicle is losing contact from the road, in this scenario also the brake is pressed completely and the vehicle comes to a rest.

### F. SYSTEM-

The system includes a camera that captures the image. The type of camera integrated in this prototype is model DC-B615 belongs to the brand HP and the camera has a higher resolution and excellent clarity as well as linked to a laptop via USB. Using MATLAB software, the computer will perform image processing to detect sleepiness in the individual. At last, an alarm and a braking system is integrated to the system by using an Arduino. Figure 5 shows the system.

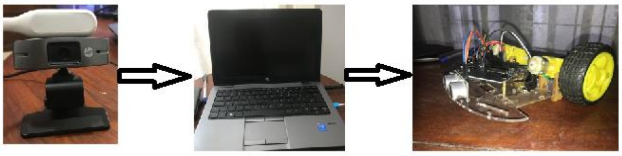


Figure 5 System integration

## IV. EXPERIMENT AND RESULT

The number of blinks an individual does in different kinds of situation is shown in the table below. [6]

Table 1 Number of blinks per minute.

| Scenario | Accuracy |
|---|---|
| Individuals when wide awake | ≤ 15 blinks per minute |
| Individuals with moderate drowsiness | > 15 blinks per minute |
| Individuals with heavy drowsiness | ≥ 21 blinks per minute |

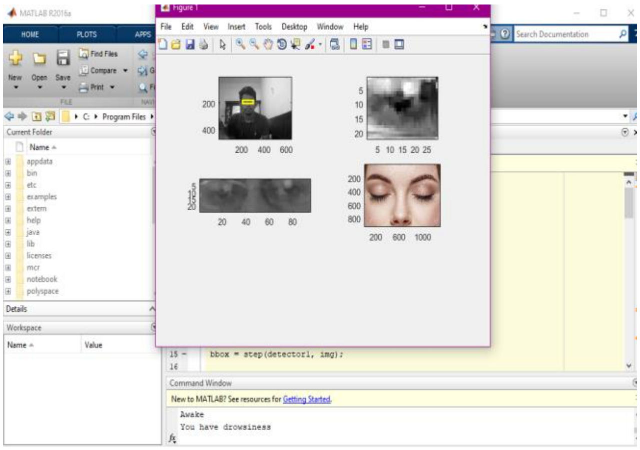
This data is essential since it is required to be expressed coherently in MATLAB as a consequence of the research of the system phases, we can get the hierarchical understanding of how the system will operate.

Figure 8 (appendix) shows the result of a test run. The person is said to have drowsiness based on the blinking rate.

Tests were conducted in a well-lit room as well as a less lit room (refer to figure 9 in appendix). The results suggested that the light has a very little impact in the process since both setting had a similar percentage of accuracy.

When tested with different types of individuals, it was found that the system is unable to detect drowsiness pattern accurately for people with spectacles. Since the system uses blinking rate as the key factor, it is a challenge for it to detect the eyes under spectacles, especially in a less lit room.

The figure below shows a falsely identified blink, when tested with an individual wearing spectacle.



```
15 -    bbox = step(detector1, img);
16
```

```
Awake
You have drowsiness
```

Figure 6 Incorrect detection with glasses.

Table 2 Test results

| Scenarios | Number of tests | False identification | Accuracy |
|---|---|---|---|
| Eyes open | 17 | 1 | 94.12% |
| Eyes closed | 17 | 3 | 82.35% |
| Eyes open (with glasses) | 17 | 4 | 76.47% |
| Eyes closed (with glasses) | 17 | 7 | 58.82% |

The table above states that the average accuracy of the system is 77.94% and the average accuracy when tested with people without wearing glasses is 88.24% and the average accuracy for people wearing glasses is 67.50%.

It is clear that the accuracy is higher with people not wearing glasses. The above test was run by placing the camera 48cm away from the driver's eye, in order to test the limitations, the camera was taken further away (60 cm) from the driver's eye and the results are summarized in the table below.

Table 3 Test results (long distance)

| Scenarios | Number of tests | False identifications | Accuracy |
|---|---|---|---|
| Eyes open | 17 | 8 | 52.94% |
| Eyes closed | 17 | 6 | 64.71% |

| | | | |
|---|---|---|---|
| Eyes open (with glasses) | 17 | 12 | 29.41% |
| Eyes closed (with glasses) | 17 | 4 | 76.47% |

This result suggests that the closed eye with glasses is detected with a higher accuracy than when tested with a shorter distance but this may not be the case because, when the distance was increased even open eyes were detected as closed eyes therefore the data above maybe biased.

Therefore, in order to obtain high accuracy, the camera should be placed in a much shorter range, preferably on the dashboard with an alignment of about 50 degrees.

## V. CONCLUSION

In this project, a driver drowsiness detection system based on eye blinking rate using image processing is proposed. Three conceptual designs were developed with different approaches to detect drowsiness, including: facial features, sudden change velocity, yawning, eye blinking and opening of the eye. After considering the pros and cons [11] of each design the design with eye blinking and eye opening was selected and implemented in this project.

MATLAB was used in this project. Regardless of its simple language and easy visualization and debugging features, the software takes more time to process real time video and to execute Arduino functions. As a result, snaps taken from previous frames overlaps with current frame, this leads to incorrect detection. This impeded the quality in contrast with the first objective of this project.

The algorithm used to detect drowsiness in drivers, detects with an average accuracy of 73.74% and it has a higher accuracy of 89.34% with people not wearing glasses. The system was able detect drowsiness even under less lighting conditions and it was established that the brightness level has a very little impact in the procedure.

After drowsiness has been identified, speed of the vehicle is controlled by an externally built system that controls the accelerator and the brake. Proximity sensors are used in front and underneath the vehicle to avoid collision. The sensor underneath the vehicle, monitors the distance between the surface of the road and the vehicle, if the distance increases then it is implied that the vehicle is losing contact with the road but this could be biased due to irregular roads and produce inaccuracy. This speed controlling system was only used in the prototype and was not tested in a real vehicle.

In the near future, the system will be modified to overcome the limitations of the current results. The immediately planned developments are mentioned below;

- As Cheng, W.-C., & Xu, Y.-S. (2013) [7] addressed in their work, the system will be developed using OpenCV to avoid delay in processing.

- The system will include two modes; day light and night mode. These modes will be automatically set by the use of a specific sensors, depending on the light available.

- As A. K. Jain, (2018) [8] addressed in his work, after drowsiness has been detected the vehicle will be set to self-driving mode [9] and will be controlled using CNN.

## VI. APPENDIX

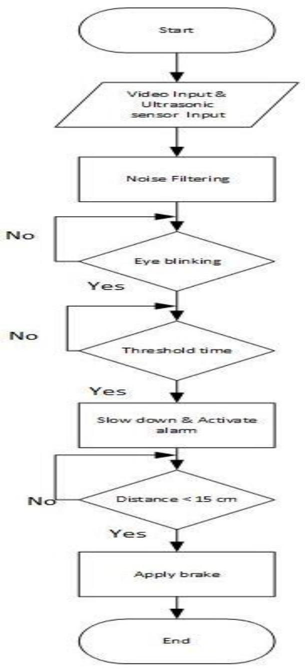
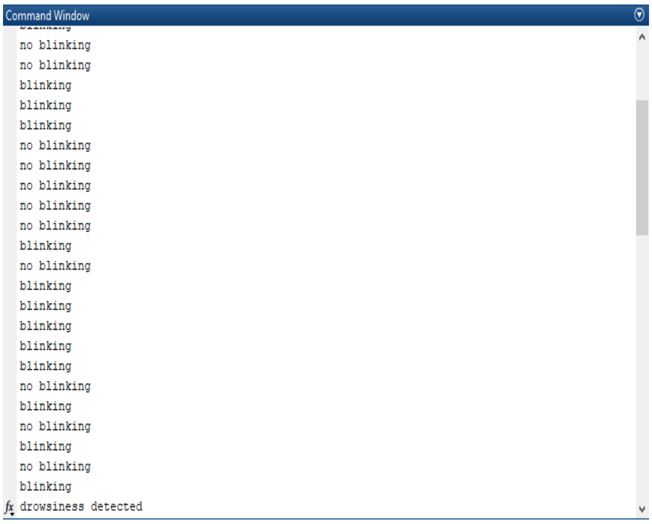


Figure 7 Flow chart of the design.



```
no blinking
no blinking
blinking
blinking
blinking
no blinking
no blinking
no blinking
no blinking
no blinking
blinking
no blinking
blinking
blinking
blinking
blinking
blinking
no blinking
blinking
no blinking
blinking
no blinking
blinking
fx drowsiness detected
```

Figure 8 Drowsiness detection in matlab.

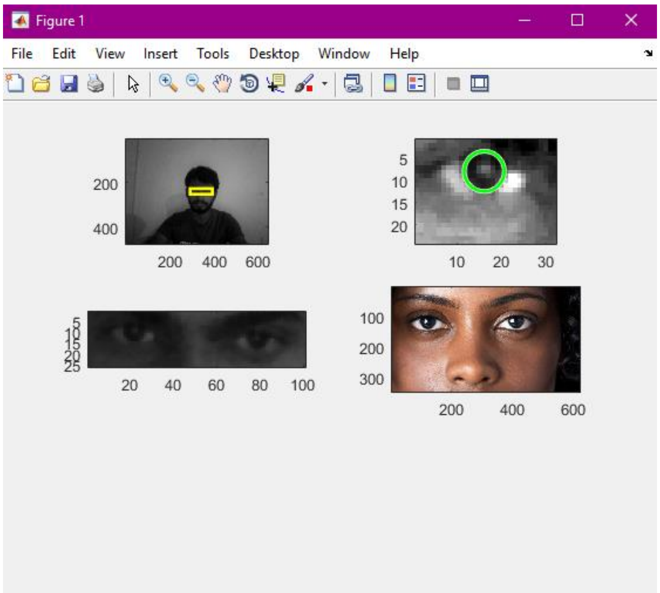


Figure 9 Accurate identification under less light.

## VII. ACKNOWLEDGMENT

I thank **Ms. Amudha Subramaniyam** for morally supporting me to pursue Science and **Mr. Chandran Subramaniyam** for supporting me financially to pursue my education.

**Mr. Dillum Fernando** from the International College of Business and Technology (ICBT) oversaw this project. Therefore, I thank him for offering understanding and knowledge that has significantly helped the research and results and for remarks that have significantly enhanced the report.

I thank **Ms. Mary Lanshiya** for assisting me with the project management aspect of the project. I thank **Mr. Shukri Razick** for assistance with designing a system to control the speed of the vehicle

## VIII. REFERENCE

[1] Australian Transport Safety Bureau, 2002. Australian Transport Safety Bureau Annual Review 2002. [Online] Available at: https://www.atsb.gov.au/media/36022/annual_review_2002.pdf [Accessed 21 January 2019].

[2] Safety Policy & Education Branch, Ontario, 2004. Ontario Road Safety Annual Report 2004. [Online] Available at: http://www.bv.transports.gouv.qc.ca/per/1022891/12_2004.pdf [Accessed 21 January 2019].

[3] Sunday Times, 2011. “Fatigue and Drowsy Driving”: An experts’ report. [Online] Available at: http://www.sundaytimes.lk/110501/News/nws_01102.html [Accessed 21 January 2019].

[4] Blackhall Publishing, 2009. Motor Traffic Act. [Online] Available at: https://srilankalaw.lk/Volume-V/motor-traffic-act.html [Accessed 21 January 2019].

[5] R. Fisher, S. P. A. W. a. E. W., 2003. Fourier Transform. [Online] Available at: https://homepages.inf.ed.ac.uk/rbf/HIPR2/fourier.htm [Accessed 26 June 2019].

[6] M. Yauri-Machaca, B. M.-C. N. V.-C. a. A. R.-G., 2018. Design of a Vehicle Driver Drowsiness Detection System Through Image Processing using Matlab. San Salvador, IEEE 38th Central America and Panama Convention (CONCAPAN XXXVIII). DOI: 10.1109/CONCAPAN.2018.8596513

[7] Xu, W. C. a. Y., 2013. Real-Time Driver Gaze Direction Detection Using the 3D Triangle Model and Neural Networks. 7th Asia Modelling Symposium, pp. 41-45.

[8] Jain, A. K., 2018. Working model of Self-driving car using Convolutional Neural Network, Raspberry Pi and Arduino. Coimbatore, India, 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA). DOI: 10.1109/ICECA.2018.8474620

[9] Manu, B. N., 2016. Facial features monitoring for real time drowsiness detection. 12th International Conference on Innovations in Information Technology (IIT), pp. 1-4.

[10] H. Singh, J. S. B. a. J. K., 2011. Eye tracking based driver fatigue monitoring and warning system. India International Conference on Power Electronics 2010 (IICPE2010), pp. 1-6.

[11] Iridiastadi, V. T. a. H., 2017. Challenges in detecting drowsiness based on driver's behavior. s.l., IOP

Conference Series: Materials Science and Engineering. DOI: 10.1088/1757-899X/277/1/012042

[12] X. Ma, L. C. a. K., 2017. Depth video-based two-stream convolutional neural networks for driver fatigue detection. Singapore, 2017 International Conference on Orange Technologies (ICOT). DOI: 10.1109/ICOT.2017.8336111