Driver Drowsiness Detection with Audio-Visual Warning

Nikita G. Prajapati
PG Student
Department of Computer Engineering
IIET-Dharmaj

Ms. Pooja M. Bhatt
Assistant Professor
Department of Computer Engineering
IIET-Dharmaj

Abstract

Driver drowsiness is one of the major causes of road accidents and it can lead to serious physical injuries, loss of human life, damage to property & loss of money. So a reliable driver drowsiness detection system is needed to be implemented, which could alert the driver before anything undesired happens. In this paper, design and implementation of 'Driver Drowsiness Detection System with Audio-Visual Warning' will be discussed. This system is to be developed for car driver, but the scope of this system is far more than it. It can be used in any situation where a person's drowsiness is needed to be monitored. The proposed system will use a camera that takes images of driver's face and monitors the driver's eyes in order to detect drowsiness of driver. When fatigue is detected, the alarm will be used to alert the driver. The proposed system will work in three main stages, in first stage the face of the driver is detected and tracked. In the second stage the facial features are extracted for further processing. In last stage, eye's status is monitored. In this last stage it is determined that whether the eyes are closed or open. On the basis of this result the warning is issued to the driver. For this Raspberry pi with raspbian (Linux) OS is used. The camera will be connected through USB port of Raspberry pi. The image processing will be done using OpenCV.
Keywords: Advanced Vehicle Safety, Driver Drowsiness Detection, Driver Fatigue, Raspberry-pi, Raspbian, Vehicle Accident Warning

I. INTRODUCTION

Drowsiness is a process in which one level of awareness is reduced due to lacking of sleep or fatigue and it may cause the driver fall into sleep quietly. When the driver is suffering from drowsiness driver loses the control of the car, so driver might be suddenly deviated from the road and hit an obstacle or a car to overturn. According to available statistical data, over 1.3 million people die each year on the road and 20 to 50 million people suffer non-fatal injuries due to road accidents [1]. Based on police reports, the US National Highway Traffic Safety Administration (NHTSA) conservatively estimated that a total of 100,000 vehicle crashes each year are the direct result of driver drowsiness. These crashes resulted in approximately 1,550 deaths, 71,000 injuries and $12.5 billion in monetary losses. In the year 2009, the US National Sleep Foundation (NSF) reported that 54% of adult drivers have driven a vehicle while feeling drowsy and 28% of them actually fell asleep. There are two methods for drowsiness detection. The first one is intrusive methods and the second one is non-intrusive methods. The intrusive methods include measurement of heartbeat rate, mind wave monitoring etc. It is most accurate, but it is not realistic, because sensing electrodes would have to be attached directly onto the user's body, and hence it would be annoying and distracting the user. While the non-intrusive methods include the yawn detection, eye closure, eye blinking rate, head pose etc. It is realistic because it does not irritate the user while driving because no sensing electrodes would be attached to user's body. Proposed system uses raspberry pi which is small in size, less power requirement and has low cost compare to other computers like desktop and laptop solves some problems of existing systems. Face and eye detection is done using Haar cascade classifier. Open CV is used to increase efficiency.

II. BACKGROUND

The Raspberry Pi is a basic embedded system having a credit card-sized single board computers developed in the UK by the Raspberry Pi Foundation. It is based on ARM processor, which is used in most smart phones and tablets. The raspberry pi is intended to be used as headless (Just a CPU), but is widely used with different Displays, Touchscreens and multimedia components. This proposed system uses raspberry pi 2 model B, which is latest at this time. The physical view of raspberry pi 2B is as shown in figure 1. The raspberry Pi board is likely just a CPU which has all necessary components on board for processing. The components on board are as shown in figure 1. It has 1 GB RAM, 40 GPIO for connecting inputs/outputs. The CPU is Quad core BCM2836, which runs at 900MHz frequency. The processor is based on ARM v7 architecture. Comparison between different raspberry pi models is given in table 1.

Table – 1
Comparison of Different Raspberry Pi models

Raspberry-Pi	*A*	*A+*	*B*	*B+*	*Pi 2B*
RAM	*256 Mb*	*256 Mb*	*512 Mb*	*512 Mb*	*1 Gb*
Processor Cores	*Single*	*Single*	*Single*	*Single*	*Quad*
Processor Speed	*700 Mhz*	*700 Mhz*	*700 Mhz*	*700 Mhz*	*900 Mhz*
Voltage And Power	*300ma,5v*	*200ma,5v*	*700ma,5v*	*600ma, 5v*	*650ma@5v*

Fig. 1: Raspberry Pi 2B [2]

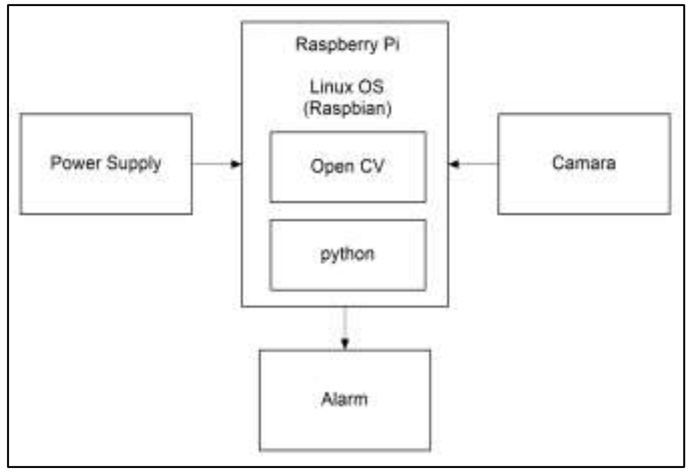

Fig. 2: block diagram of proposed system

III. PROPOSED METHODOLOGY

A. *Overview of Proposed Work:*

There are many signs by which we can detect the drowsiness of drivers. They are as given below:

- Driver may be yawn frequently.
- Driver is unable to keep eyes open.
- Driver catches him nodding off and has trouble keeping head up.

Driver drowsiness detection system is a system that is implemented using image processing to detect drowsiness of the driver. This application can be very useful to reduce the accidents, because most accidents occurs due to drowsiness of drivers. With use of this application the driver's status can be monitored, like yawning, fatigue, closure of eyes etc. In this system, the camera is placed in front of driver. The status of driver's face is continuously monitored using the camera. The alarm can be used to give an alert, if any sign of drowsiness is detected.

The proposed system will work in three main stages.

1) In first stage the face of the driver is detected and tracked.
2) In the second stage the facial features are extracted for further processing.
3) In last stage, eye's status is monitored. In this last stage it is determined that whether the eyes are closed or open. On the basis of this result the warning is issued to the driver.

For this system I am going to use Raspberry Pi with raspbian (linux) OS. The camera will be connected through USB port of raspberry pi. The image processing will be done using OpenCV.

B. *Flow Chart of Proposed System:*

The flowchart of image processing part is shown as figure 3. The camera is connected on Raspberry Pi using USB port. The camera needs to be initialized at the start-up of code. The modules of camera will be loaded and it will be ready to take images. The captured images are processed to detect face of the driver. If the face is not detected in the processed image, it will continue to capture image and process it. If the face is detected in- the frame, it will be processed to extract facial features. The image will be process to detect open eyes. If the eyes are open, the code will continue to capture & process image. If the eyes are closed, the system will check first, if the timer is ON or OFF. If the timer is OFF, it will be turned ON. And if it is ON, it will check if the time is greater than 2 seconds. It time is less than 2 seconds, it will go back and continue to capture and process image. If the timer exceeds time of 2 seconds, the alarm will be turned ON. The alarm will be played using GPIO port of Raspberry Pi. One pin will be initialized as Output and hardware alarm will be connected on it. Pin will be set ON to turn on the alarm.

After turning ON the alarm, the General Purpose Input Output module of raspberry Pi will be initialized to take input from user. One button will be connected on GPIO module to take input. The alarm will be stopped if the button is pressed. By this mechanism the driver can turn the alarm off when he/she is awakening, and the system will return to capture & process the images.

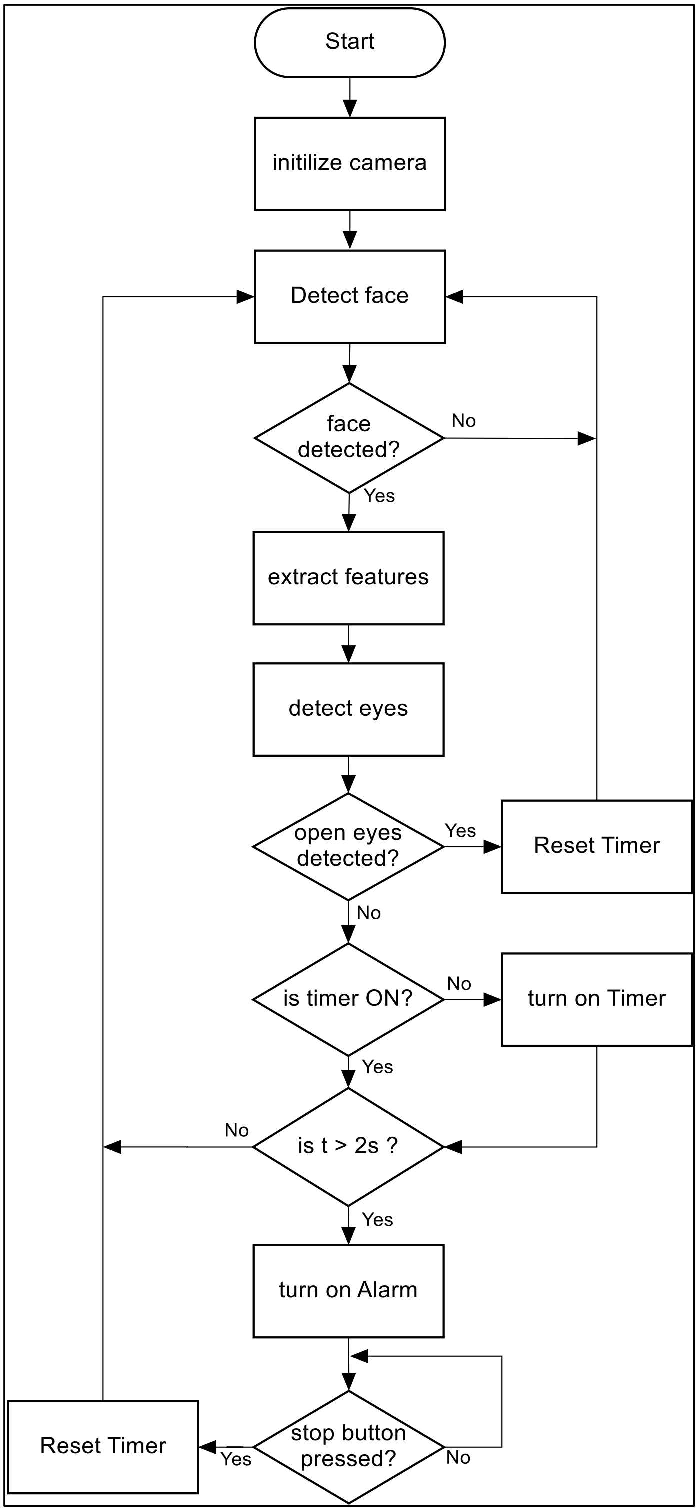

Fig. 3: flowchart of proposed system

C. *Performance Evaluation:*

1) *Speed*

The speed of system is defined as the frequency of total program instructions for one particular cycles. Higher the frequency, better the performance.

$$\text{Speed} = \text{cycles per second}$$

$$\text{Speed} = 1 / Tm$$

Where, Tm is time taken by one complete cycle of total program instructions.

2) *Accuracy:*

The accuracy is defined by the ratio of correctly detected eye state, by the total samples taken by camera. One particular cycle can be explained as one sample.

$$Accuracy = Nc / Nt$$

Where, Nc = number of samples in which eye state is detected correctly

Nt = Total number of taken samples

IV. WORK RESULTS

A. Face and Eye Detection in Image:

Following figure4(a) & 4(b) shows the detected face and eye from the image. The face and eye detection is done using Haar Cascade Classifier technique. The blue square is drawn in result image automatically after the execution of python code.

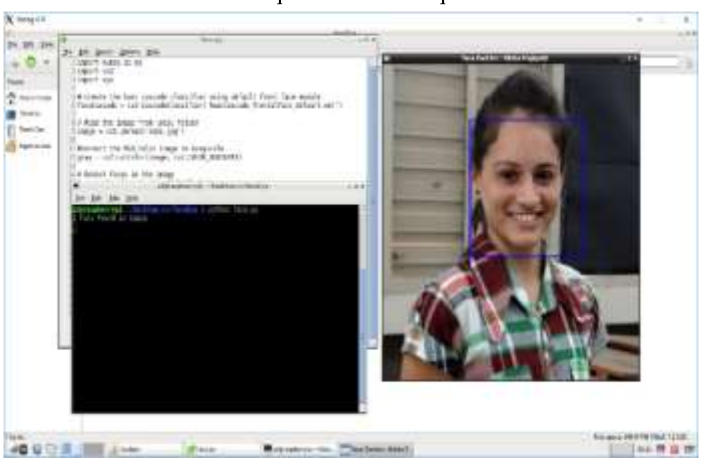

Fig. 4(a): Detected face in an image

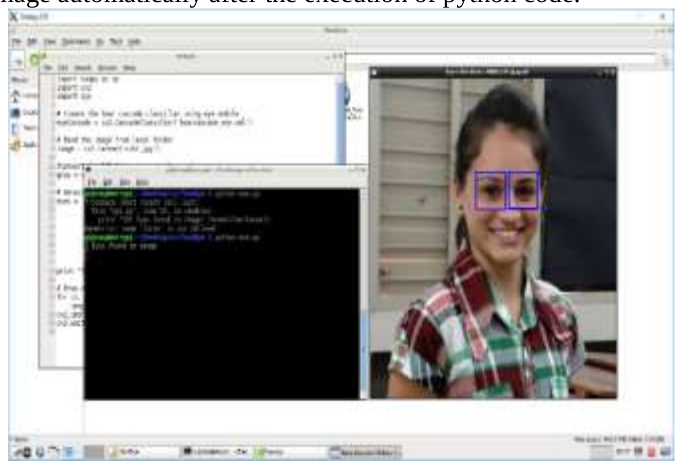

Fig. 4(b): Detected Eyes in an image

B. Mouth and Nose Detection in image:

Mouth and Nose detection is shown in figure 5(a) & 5(b). The purpose of mouth detection is to detect yawning if possible for future scope. In the result of nose detection, the nose is detected in face region correctly, but it is also detected in background, which is an error. So we can remove this error by limiting the image search area. First the face will be detected, and from that area of face, the other features like eyes, nose & mouth will be detected. This will solve the problem.

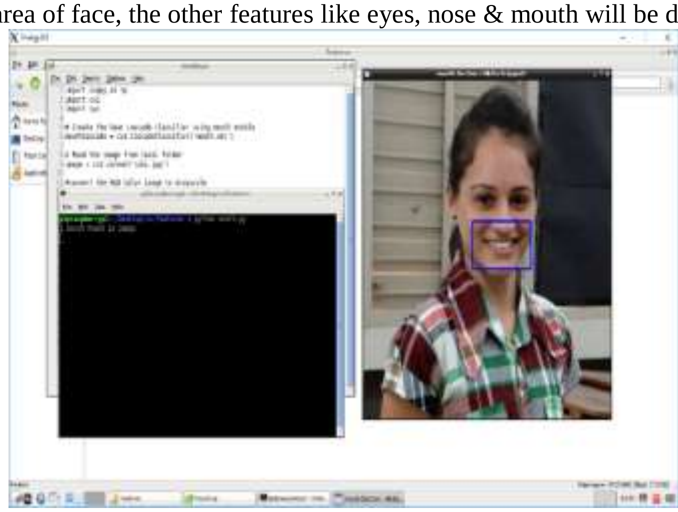

Fig. 5(a): Detected Nose in an image

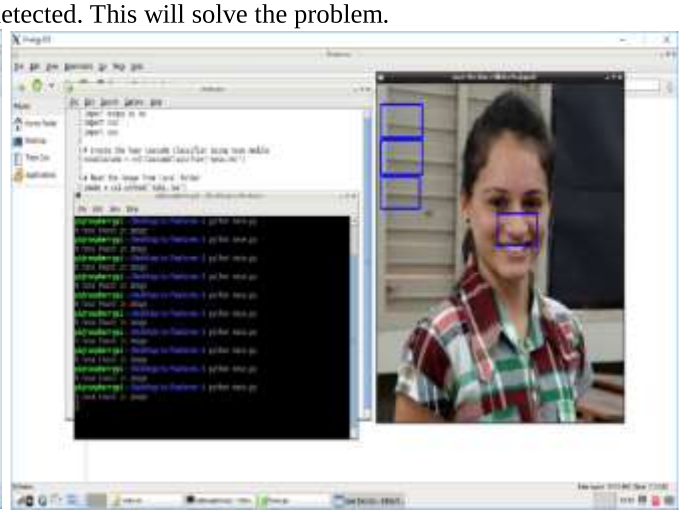

Fig. 5(b): Detected Mouth in an image

C. Face, Eye & Mouth Detection in Image:

In the figure 6, the detection of Face, Eye & mouth is shown. It is implemented using the method explained in previous section. The result is better and accurate this way, because it reduces the possibility of error by limiting the search area to only face region.

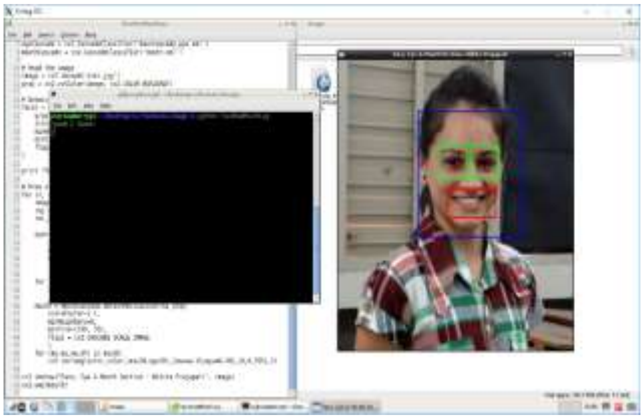

Fig. 6: Detected Face, Eyes & Mouth in an image

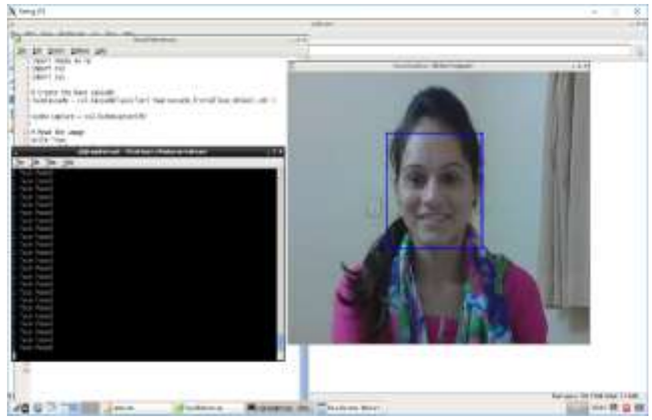

Fig. 7: Detected Face through webcam

D. Face Detection using Live Webcam:

The detection of face is done using live webcam as shown in figure7. In this code, the USB webcam is configured for taking images and that image is taken for processing using same method as above Face detection is accurate using Haar cascade technique compare to other like LBP.

E. Face, Eye, Mouth & Nose Detection using Live webcam:

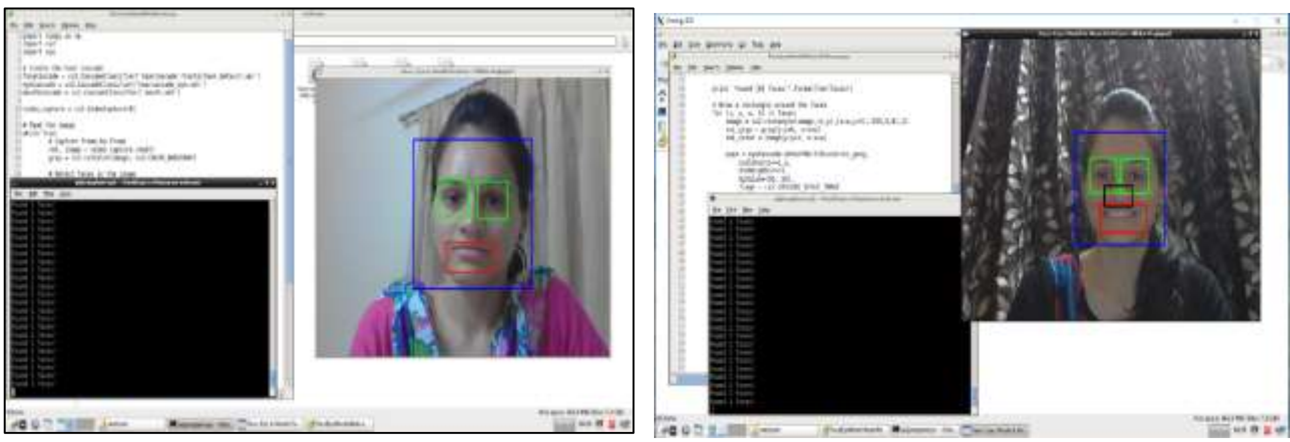

Fig. 11: Detected Face, Eyes, Mouth and nose through webcam

In the figure above, the detection of Face, Eye, Mouth and Nose using live webcam is shown. This also using the same method. It also works in low lighting condition as shown in figure. All the results are accurate for different users.

F. Open and Closed Eye Detection:

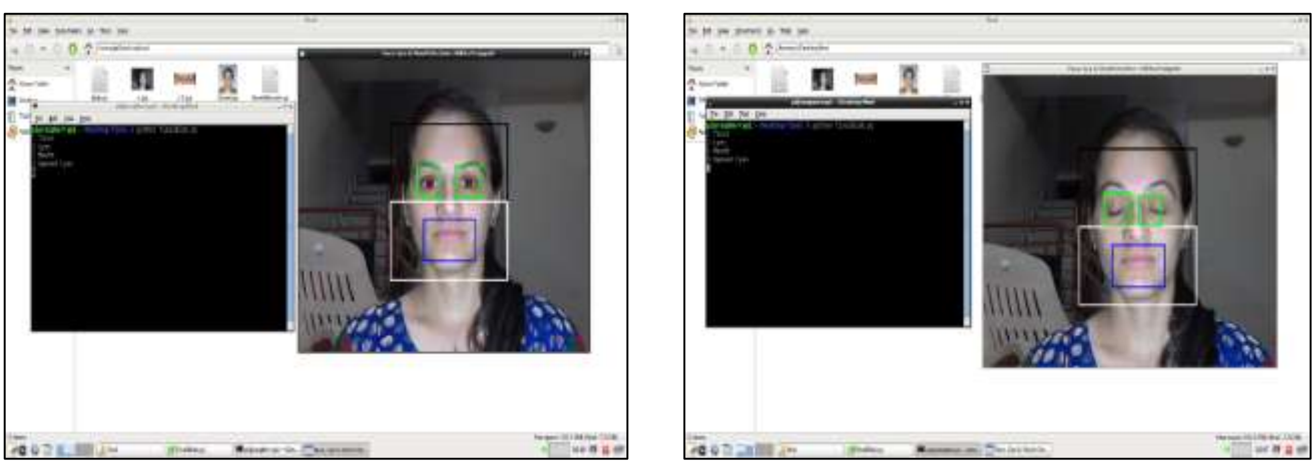

Fig. 12: Detected Open and Closed Eyes

The eye status monitoring is done using Blob detection. Figure shows the detected open and closed eyes. It draws circles around eye region as shown in figure. For face and eye detection is done using Haar cascade methods. Only eye status monitoring is done using blob detection to check eye status whether it closed or opened. Also it is accurate.

G. Setup of system

Fig. 13: Experimental setup of running system

The complete hardware configuration is as shown in above figure. There are total 4 LED, White, Yellow, Blue, Red are used as shown in above figure. White for power supply. Yellow LED will be on when system start running. Blue LED for face detection. If face is detected, then it will be turned on otherwise it will remain off. Red LED is used for eye status monitoring. When close eye is detected red LED will be turned on. If closed eye detected the Red LED will be turned off. There is also one switch to stop buzzer.

H. Result Analysis

The accuracy of frontal face and the eyes detection is as shown in figure 14. Face and eye detection is tested using Haar cascade algorithm and local binary pattern(LBP.) In graph, we can see that the accuracy of both face and eye detection is high using Haar cascade classifier algorithm as compare to the local binary pattern. Accuracy of frontal face using Haar cascade and lbp are around 100% and 92% respectively. And the accuracy of eye detection using Haar cascade and lbp are around 100% and 79% respectively as shown in following figure.

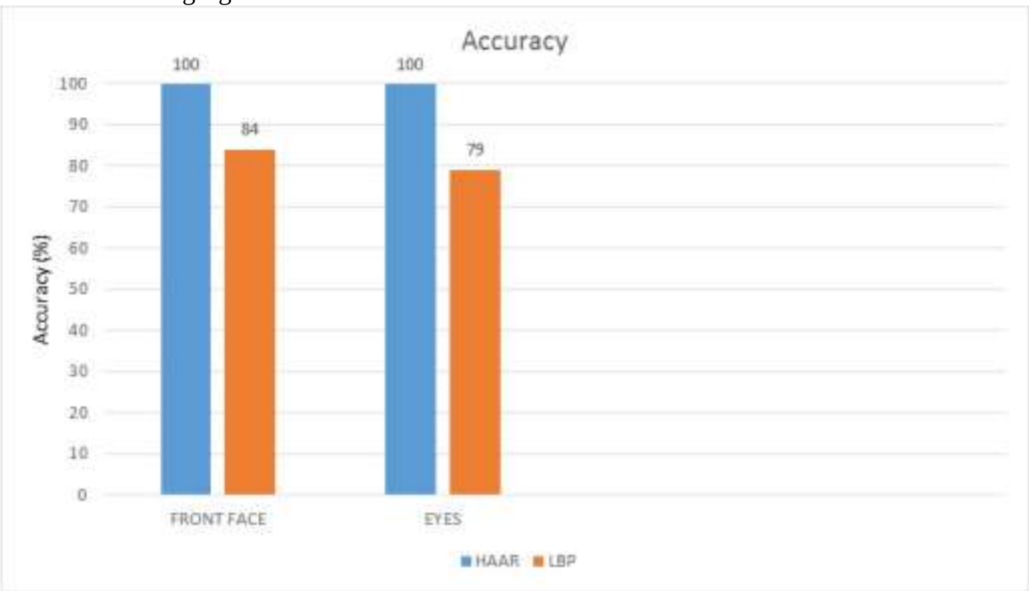

Fig. 14: Accuracy of face and eyes detection

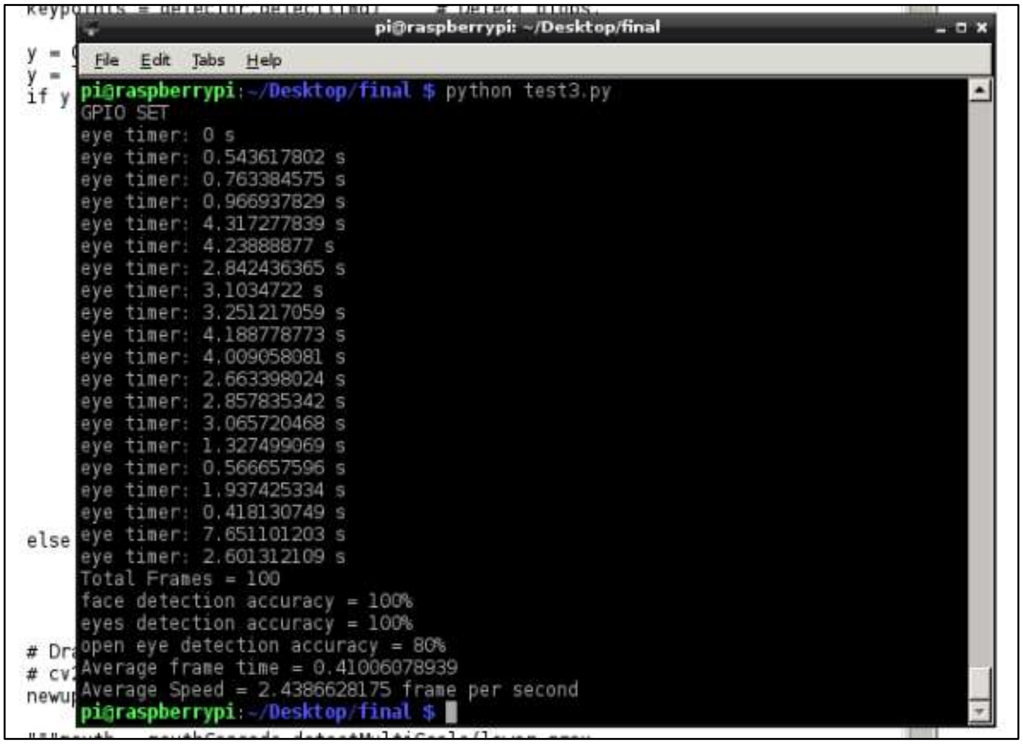

Fig. 15: Accuracy and Speed

V. CONCLUSION

There are several intrusive and non-intrusive methods to implement driver drowsiness detection system. From the study and design of proposed work it is clear that usage of raspberry pi and OpenCV is even more suitable for this particular application in terms of size, cost and power requirement The results are accurate and reliable for detection of face, eyes & mouth. The operations are performed on static image as well on feed of live webcam. It is observed that, the result varies due to un-even lighting conditions; however it is accurate even in low light conditions. Proposed system can be used in real working environment. In future yawning detection and head nodding can be possible.

REFERENCES

[1] Arun Sahayadhas, Kenneth Sundaraj and Murugappan Murugappan, "Detecting Driver Drowsiness Based on Sensors: A Review",2012 sensors
[2] Image taken from https://en.wikipedia.org/wiki/Raspberry_Pi
[3] Image taken from https://www.element14.com/community/community/raspberry-pi/raspberrypi2?ICID=rpimain-feature-products
[4] software win32DiskImager downloaded from http://win32diskimager.sourceforge.net/
[5] software bitwise SSH client downloaded from https://www.bitvise.com/ssh-client-license
[6] Syed Imran Ali, Dr. K.N. Modi, Dr. Prashant Singh Sameer Jain," An Efficient System to Identify User Attentiveness Based on Fatigue Detection", 2014, International Conference on Information Systems and Computer Networks.
[7] Mhammad Amin Assari, Mohammad Rahmati, "Driver Drowsiness Detection Using Face Expression Recognition". 2011, IEEE International Conference on Signal and Image Processing Applications (ICSIPA2011).
[8] Osama Rizwan, Hamza Rizwan, Mudassar Ejaz, "Development of an Efficient System for Vehicle Accident Warning", IEEE Transactions-2013.
[9] Chih-Jer Lin, Chih-Hao Ding, Chung-Chi Liu and Ying-Lung Liu, "Development of a real-time drowsiness warning system based on an embedded system",2015, IEEE Transactions.
[10] Arun Sahayadhas, Kenneth Sundaraj, Murugappan Murugappan, Rajkumar Palaniappan," A physiological measures-based method for detecting inattention in drivers using machine learning approach", 2014, Elsevier.