Proceedings of the Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)
IEEE Xplore Part Number:CFP20OSV-ART; ISBN: 978-1-7281-5464-0

Driver Drowsiness Detection System based on LBP and Haar Algorithm

Yuvraj Suryawanshi
MTech Student
Dept. of E&TC Engineering
Govt College of Engineering Aurangabad
Aurangabad, India
yuvisuryawanshi12795@gmail.com

Sushma Agrawal
Associate Professor
Dept. of E&TC Engineering
Govt College of Engineering Aurangabad
Aurangabad, India
agrawal.sushma1@gmail.com

Abstract - **Drowsiness is one of the main reasons for road accidents in the last few years. With the improvement in technology, various accident prevention technologies are evolving. The primary objective of avoidance of road accidents can be achieved through real-time drowsiness detection of a driver using video capturing with face detection. After capturing and detecting the drowsiness by using a camera, the alarm will buzz. The position of head and blinking of eyes are used as the features to detect whether the driver is drowsy or not. The camera captures the real-time drowsiness by using Local Binary Pattern to detect the face and Haar cascade to detect the eyes. A custom eye blinking file has been developed for eye blinking detection and AdaBoost is used to focus on eye movements at the same instant of time.**

Keywords—LBP Algorithm, Haar Algorithm, AdaBoost Algorithm, Drowsiness Detection, OpenCV

I. INTRODUCTION

The effective count of showing cars and automobiles running on the road is constantly increasing in advanced countries. The problem with the increase in the use of cars and vehicles is that it furnishes an increase in the number of road accidents. Reasons for these road accidents are highway traffic, over speeding, use of mobile phones while driving, drunk drivers, and drowsiness of the driver [8]. Drowsiness is defined as the middle stage between wakefulness and sleep. As the research, in India in 2018, 1.51 hundred thousand lost their lives in roadway disasters due to over speeding of vehicles and sleepiness [11]. Hence drowsiness prevails as a primary source regarding road disasters. Effective count of accidents that occurred were 4,67,044 and deaths recorded were 1,51,417 and caused injuries to 4,69,418 in 2018. An average of 1280 accidents and 415 deaths occurred were every day in 2018. The WHO Global Report on Road Safety says that India ranks 1st in the world for road accidents in 2018. India accounts for almost 11% of the accident recorded in the world. The working-age groups from 18-60 years were almost as high as 84.7% of total deaths in road accidents. The total no of males and females contributed to 86 % and 14% respectively in road accidents. But research shows that 50% or more of road accidents are due to drowsy driving. The major contributor of the road accidents is lack of sleep as it appears in all deaths and fatal injuries occurred in road accidents. With the advancement in technology, it is possible to design a system to monitor the behavior of the driver and generate an alert to the occupants of the moving vehicle when sleepiness or drowsiness is detected. Such a system can help to reduce roadway disasters to a great extent. Drowsiness Detection is carried out in two ways as a Vehicle-based approach, and a Behavioral-based approach. In the Vehicle-based approach, the system will observe the steering rotation location, lane location, also clutch pressure. In the proposed system behavioral-based approach is used with the monitoring of eye, eye blinking, and head position. The real-time drowsiness is detected only during daylight using the camera and Python. This method is used as a software-based output it does not require Raspberry Pi.

II. RELATED WORK

Fouzia, Roopalakshmi R et al. [1] have detected drowsiness by counting the eye blinks. A stand-alone system is implemented by using Raspberry Pi, camera, and alerts were sounded off using a buzzer and vibrator. The Haar cascade is used to recognize an effective face with eyes [12]. If affecting the blinking rate of the eyes is more than 2 seconds, then drowsiness condition is detected. When the driver is detected as drowsy at that instant the buzzer will buzz, and the vibrator will try to awake the driver. Md Yousuf Hossain, Fabian Parsia [2] proposed a non-intrusive and real-time system for drowsiness detection using Raspberry Pi, Camera module, and buzzer. The Haar cascade classifier is used to detect facial landmarks. The human eyesight has an eye aspect ratio (EAR) ratio of 0.25 for the normal eye. When the EAR ratio of the normal human eye falls below 0.25 then the system detects that the driver is drowsy while driving. As soon as the driver is detected drowsy then the system this connected through IOT sends an email to the owner of the car and the buzzer continuously buzzes till the driver is fully awake. Akalya Chellappa et al. [3] have implemented driver drowsiness detection using Raspberry Pi 3 & Raspbian camera wherein it catches the driver's face and eyes. Haar Cascade Classifier is used to calculate eye blink duration and Eye Aspect Ratio (EAR). Raspberry Pi 3 with Raspbian OS which is Linux Based Performing method with Python IDLE. It involves programming in software with extensions of OpenCV computer vision installed. To execute the program, libraries like NumPy, OpenCV, play sound, argparse, dlib, distance, timer, client, ApiClient, and pi-camera were imported. The two conditions are measured, first is eye open and the other is eye closed. The threshold value of EAR is set for 0.5-second when the eye is closed, then it will sound alarmed to wake up the driver and also send the message through the cloud to the owner with vehicle number. In this way, the Drowsiness is detected.

$$EAR = \frac{(p2 - p6) + (p3 - p5)}{2(p1 - p4)}$$

S. S. Kulkarni, A. D. Harale et al. [4] have detected drowsiness by Raspberry pi, 5 MP digital camera, Microcontroller, Buzzer, and GSM Module. The 5-megapixel camera is used with Raspbian OS. The Haar cascade and adaptive boost are used to recognize the effective face and eye blinking and the threshold value is set to 6 blinks per second. When the threshold value is crossed then raspberry pi sends a signal to the controller to send a message to the person whose mobile number is saved in the system. A buzzer will buzz continuously so that driver can take a rest, and a relay is turned OFF to stop the vehicle. Roshni Rajan K. et al. [5] proposed that the driver's drowsiness is detected at night time driving. The infra-red night vision camera is used to observe the fatigue of the driver by monitoring the eye. If the driver's eye remains closed for more than normal time, then the vehicle is parked to the left side and a buzzer is sounded off to alert the driver. Another feature is to detect the accident and alerting the hospital. The accelerometer is used to detect the accidents and the camera output is given to microcontroller and the controller will detect output and enables Bluetooth to send the message through the Android Application. The android application is developed using Java language applying powerful Android Software Development Kit. Anjali K U, et al. [6] has proposed a stand-alone real-time system for drowsiness detection using a Raspberry Pi, camera, buzzer, and vibrator. The main points are related to face detection and eye blinking relation. Face and eye are detected using the Haar Cascade Classifier. The eye-detection algorithm is used to spot eyes open or closed state. If eyes were closed for periods lasting to 3 to 4 seconds then drowsiness state is detected and the buzzer along with the vibrator turns 'ON' to wake up the driver. Athira Gopal and V.V. Vineeth [7] have used an alcohol sensor to detect the presence of alcohol in a driver's breath and monitor the driver's eyes to detect fatigue. The open or closed state of eyes is analyzed to understand drivers driving behavioral style. The captured image is converted to binarization due to which eyes are detected [9]. The removal of noise in a binary image is necessary else the face detection will be affected [10]. The state of the blinking of the eye are been plotted with the intensity changes in plot horizontal average of y-coordinates. The eye close intensity is compared with the eye open and drowsiness is detected. If 5 continuous image frames of the closed eye then an alarm is buzzed.

III. PROPOSED METHOD

The vital purpose is to observe the sleepiness state of the driver. The proposed work uses a behavioral-based approach to intercept the sleepiness of the driver in real-time. Drowsiness detection of the driver is achieved by video capturing in real-time. The computer vision library of Python is used to implement the algorithm. Methodology for Drowsiness Detection in the form of a flowchart is given in Fig.1.

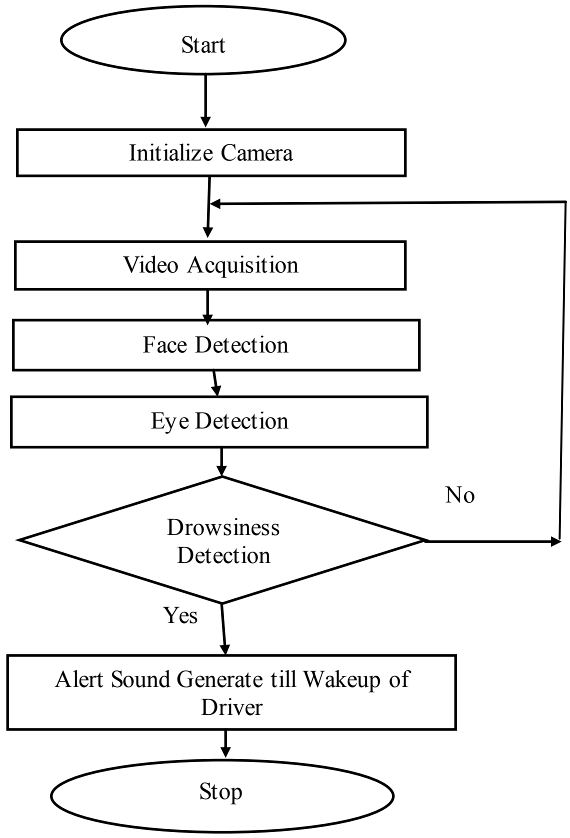

Fig 1. Flow Chart

As shown in the flow chart, the camera continuously monitors the driver's state by capturing his/her video. The system's camera captures video at a high definition of 720p and 30 fps. The time sleep is set to 1 second to detect the drowsiness of the driver with sound buzz at an instant. The drowsiness detection process will continue until the vehicle is being driven. The sound will buzz until the driver is fully awake in real-time. This is confirmed by the system which keeps taking input of the driver's face till the driver is fully awake. The methods for face, eye, and drowsiness detection are explained in the following sections.

A. *Face Detection*

Face spotting live is achieved by using the Local Binary Pattern Algorithm. Local Binary Pattern (LBP) is achieved by using cv2.CascadeClassifier command of OpenCV library. Local Binary Pattern cascade classifier is combined with the Histogram of Oriented Gradients (HOG) descriptor used in computer vision for particular object detection. The image is divided into cells for the encoding of features. LBP uses 9 picture elements (3x3 aperture) at a time. The midpoint picture element is compared to each neighborhood pixel in the aperture. These picture elements can be greater than, less than, or equal to the midpoint picture element. If the value of the nearby pixel is greater than the central pixel then it is assigned 1 and otherwise 0.

Hence this aperture translates into a picture element value that is 0 either 1 inside a forward or backward direction, and then pattern a binary number. The LBP tag picture element on threshold value impressive 3x3 nearby touching the midpoint picture element evaluate, while shown in Equation 1.

$$LBP(x_{p,}y_p) = \sum_{n=0}^{7} s(i_n - i_p)2^n \qquad (1)$$

while (x_p, y_p) be expressive picture element about a figure n perform effective nearby picture element, also affecting separate grey leveled about the nearby besides midpoint picture element with s(x) could remain shown in Equation 2.

$$s(x) = \begin{cases} 1, & x < 0 \\ 0, & x \geq 0 \end{cases} \qquad (2)$$

As an effective of the above two equations, the midpoint picture element are calculated through the nearby picture element to shape an empty value. For each picture element, this task provides an 8-bit string. Using this 8-bit string of nearby pictureelements, the final value between 0 to 255 are represented as the grayscale value. The grayscale value is to detect a face within a rectangle box. Local Binary Pattern has a characteristic of texture analysis used to recognize the face of a person to detect it and to avoid other things in the background. So, such classifiers are used for face detection.

B. Eye Detection

Face detection is achieved by the Local Binary Pattern. After the face is detected now eye pupil is detected through the Haar algorithm. Haar algorithm uses the OpenCV libraries which detect eyes. The computer vision uses to preprocess the features of the Haar algorithm and has the inbuilt haar wavelets. The Haar wavelet's major property is the real-time detector of a precisely described object in a rectangle window frame. A haar algorithm has a property of adjoining parallelogram place on a described position now a detection aperture, sums upward effective image element potency within a particular place, including computing effective divergence connecting those aggregate used to identify subsection or the target of an image. Here is some Haar feature as shown below in Fig 2.

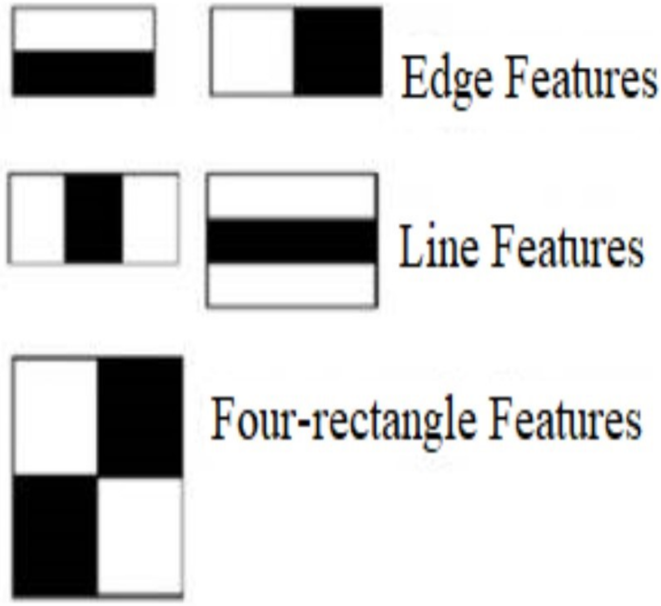

Fig 2.Haar Cascade Feature Extraction

As the Edge and Line feature will only be able to detect a substance with plain edges and lines. This feature targets the characteristics such as affecting the place about eyes gets black than an effective area about nose and cheeks. Another part depends on the powerful characteristics of eyes and nose and cheeks will remain black. The Four rectangle features are used to detect coordinates of a rectangle box as ex, ey, ew, eh to each eye position with the oblique lines to both eyes. This target of an image is used to target the eye region in real-time with having a high speed with accuracy in the rectangle window frame. The Haar Cascade property is used to detect the eye located on the face and to avoid other background objects while capturing video. So the Haar algorithm endures applying through recognition of impressive eye part in real-time but unable to recognize the blinking of eyes. So, to recognize the blinking of eyes the Adaboost Algorithm is utilized.

C. Drowsiness Detection

Adaptive Boost continues to deploy toward merge and study or boost a poor algorithm. The Custom Blink Cascade file uses the Adaptive boost (AdaBoost) algorithm together with Haar cascade. As the AdaBoost used to train the poor algorithm of the Haar feature algorithm. The AdaBoost algorithm is a merging of a poor (Haar feature) learning algorithm into a bulk quantity, this produces a powerful concluding solution about effective advance classifiers as a powerful classifier. The AdaBoost algorithm can be presented in Equation (3).

$$F_T = \sum_{t=1}^{T} (f_t(x)) \qquad (3)$$

Where each value is poor in Haar feature learning algorithm particularly holding a phenomenon inserted including the recovery of a merit to specify affecting group about the object. Adaptive Boost features are shown in Fig 3.

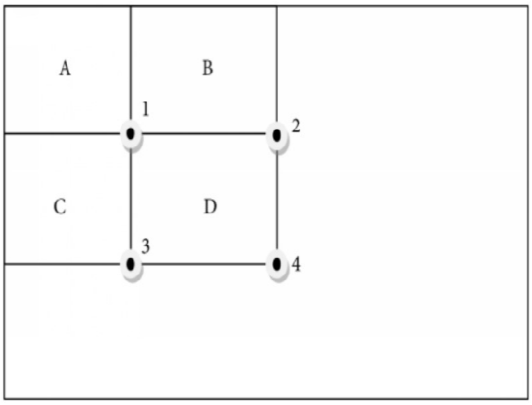

Fig 3.AdaBoost Feature

Astonishing aggregate regarding picture element now parallelogram D conceivable computed on Position 1 (total out from picture element into parallelogram A), on Position 2 (A + B), on Position 3 (A + C), on Position 4 (A + B + C + D), Conclusively affecting total coming from parallelogram D is ((4 + 1) – (2 + 3)). This feature is used to boost the opening and closing movement of the eye with speed and accuracy as the rectangular frame of eye detection by using the Haar cascade. The drowsiness detection system is a real-time operating systemuse to detect the sleepiness of the driver at instant so no need of utilizing the available dataset. The proposed system has been tested on people's for which 90% accuracy is achieved on people's drowsiness is detected correctly. When the eye is closed for 1 second the drowsiness is detected then a sound will give buzz till the driver is not fully awake. When the driver is awake then the camera will continue to as it was capturing before.

IV. Programming Algorithm

Python is effective to get the computerized output. The camera of the system is used to capture the video. The obtained complete result considering sleepiness awareness is displayed including effective assistance attributed to a flow diagram appear current fig1. The various Python library installed are OpenCV, NumPy, time, Tkinter, and Pygame to execute the programming algorithm to detect the drowsiness. The OpenCV and NumPy have utilized through catch expressive driver Face and Eye movement at that instant using the LBP algorithm and Haar algorithm with the custom blink cascade. The time library is used to calculate time sleep at the instant of 1 second for a closed eye, it will turn on the sound at that instant till the driver is fully awake. The sound will play through the Pygame library. The Tkinter library is GUI (Graphical User Interface) in python which is used to create the web page with few buttons and a message box.

V. Result Analysis

The OpenCV library is used to get the output in real-time with higher accuracy. This library is used to detect the face, eye, and eye movements with the platform are run in Python IDLE. LBP algorithm used to detect the face, the Haar algorithm used to detect eye position on the face, Adaboost algorithm used to boost the Haar cascade to produce efficient eye blinking is traced by open and close movements and generate efficient system quick suitable to work in real-time.

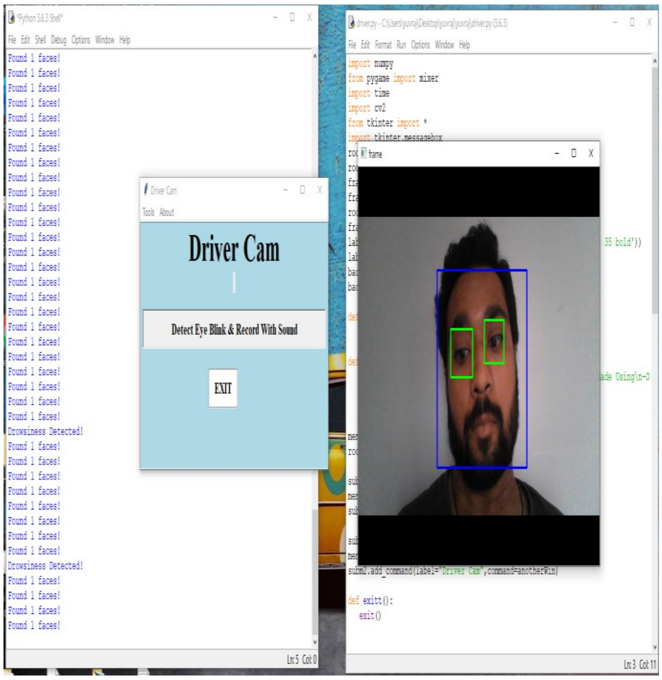

Fig 4(a).Output of System

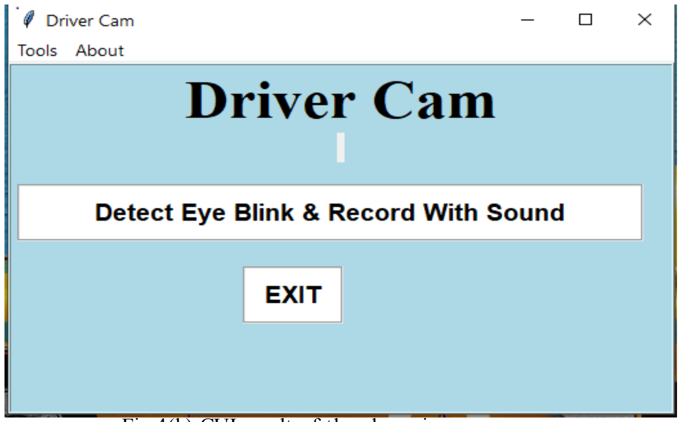

Fig 4(b).GUI result of the above image

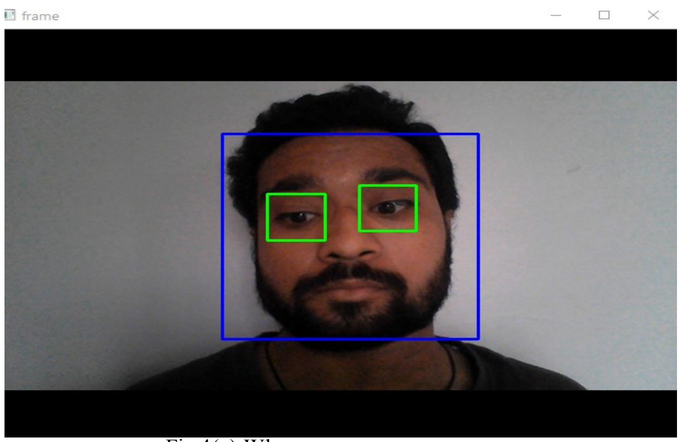

Fig 4(c).When eyes are open

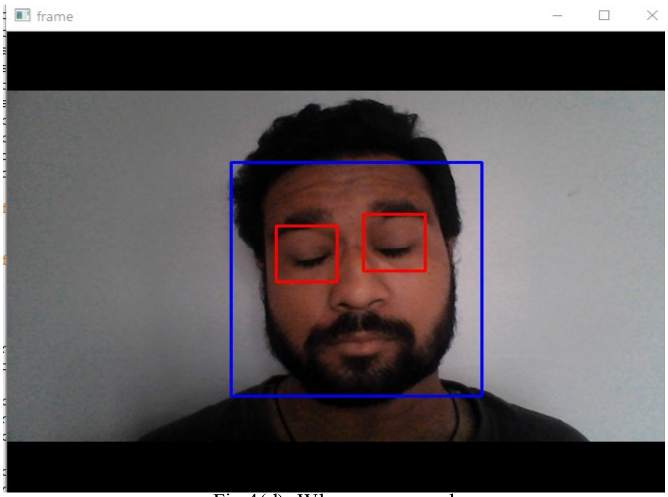

Fig 4(d). When eyes are close

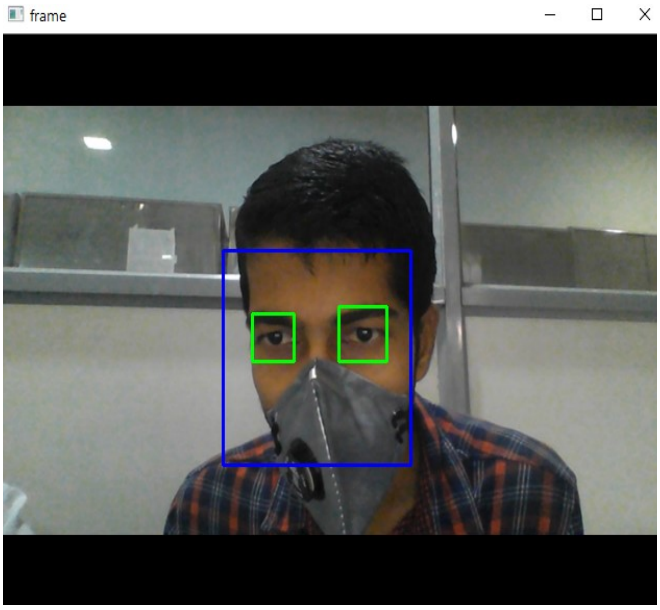

Fig 4(e).When eyes are open

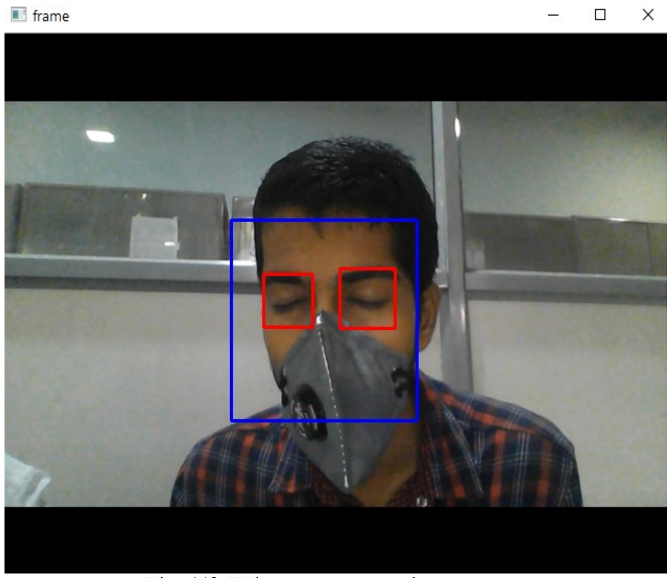

Fig 4(f).When eyes are close

The above Fig 4(a), 4(b), 4(c), 4(d), 4(e), 4(f), display effective output related to the system. Fig 4(c), 4(d), 4(e), and 4(f) show the eye open and close in the frame capture at the same time as eye close then the alarm will buzz to wake up driver. The blue rectangle box uses to locate the face while the green rectangle box shows the driver is awake and the red rectangle box shows the driver is sleeping at that moment.

VI.Conclusion

The proposed system gives us an analysis of drowsiness detection in real-time during daylight with efficient brightness towards the face. The main purpose was to detect the sleepiness of the driver while driving to avoid road accidents is achieved in this system. The face region was detected by the LBP algorithm and eye region was detected by the Haar algorithm and eye blinking was detected by the AdaBoost algorithm together with Haar cascade which was developed for eye blinking detection. As the system identifies the face, eye region, eye blinking gives us the output instantaneously in real-time. In lack of brightness, insufficient light, or night vision the system is unable to capture the drowsiness of the driver. In the future, it is possible to improve our system in every aspect of working conditions.

REFERENCES

[1] Fouzia, R. Roopalakshmi, J. A. Rathod, A. S. Shetty and K. Supriya, "Driver Drowsiness Detection System Based on Visual Features" Second International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, 2018, pp. 1344-1347, doi: 10.1109/ICICCT.2018.8473203.

[2] M.Y. Hossain & F.P. George, "IOT Based Real-Time Drowsy Driving Detection System for the Prevention of Road Accidents". *International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS), 2018, 3*, 190-195.

[3] A. Chellappa, M.S. Reddy, R. Ezhilarasie, S.K. Suguna & A. Umamakeswari, "Fatigue Detection Using Raspberry Pi" 3. *International journal of engineering and technology, 7*, 29.

[4] S. S. Kulkarni, A. D. Harale and A. V. Thakur, "Image processing for driver's safety and vehicle control using raspberry Pi and webcam," IEEE International Conference on Power, Control,

Signals and Instrumentation Engineering (ICPCSI), Chennai, 2017, pp. 1288-1291, doi: 10.1109/ICPCSI.2017.8391917.

[5] Roshni Rajan K, Neethu K. U, Riya Carmel T Isak, Sreehari K. S, Sibin Sunny, Vishnu E K, "Drowsiness Detection and Rescue System", International Journal of Science and Research (IJSR), https://www.ijsr.net/search_index_results_paperid.php?id=ART20171589, Volume 6 Issue 3, March 2017, 1663 – 1666

[6] Anjali K. U., A. K. Thampi, A. Vijayaraman, M. F. Francis, N. J. James and B. K. Rajan, "Real-time nonintrusive monitoring and detection of eye blinking in view of accident prevention due to drowsiness," International Conference on Circuit, Power and Computing Technologies 2016.

[7] Athira Gopal and V.V Vineeth, "Driver Drowsiness Detection System" International Journal of Advances in Electronics and Computer Science, **IJAECS-IRAJ-DOI-7366** ISSN: 2393-2835 Volume-4, Issue-3, Mar.-2017 http://iraj.in

[8] O. Khunpisuth, T. Chotchinasri, V. Koschakosai and N. Hnoohom, "Driver Drowsiness Detection Using Eye-Closeness Detection," *12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, Naples, 2016, pp. 661-668, doi: 10.1109/SITIS.2016.110.

[9] A. Ray Choudhury and V. R. Dahake, "Image Super Resolution Using Fractional Discrete Wavelet Transformand Fractional Fast Fourier Transform," 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), Palladam, India, 2018, pp. 557-564, doi: 10.1109/I-SMAC.2018.8653723.

[10] P. K. Kavitha and P. V. Saraswathi, "An Efficient Image Retrieval Using Fuzzy Based Clustering," Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), Palladam, India, 2019, pp. 551-555, doi: 10.1109/I-SMAC47947.2019.9032517.

[11] N. Borulkar, P. Pandey, C. Davda and J. Chettiar, "Drowsiness Detection and Monitoring the Sleeping Pattern using Brainwaves Technology and IoT," *2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud),* Palladam, India, 2018, pp. 703-706, doi: 10.1109/I-SMAC.2018.8653772.

[12] K. Vengatesan, A. Kumar, V. Karuppuchamy, R. Shaktivel and A. Singhal, "Face Recognition of Identical Twins Based On Support Vector Machine Classifier," *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)*, Palladam, India, 2019, pp. 577-580, doi: 10.1109/I-SMAC47947.2019.9032548.