International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 07 | July - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

Drowsiness Detection System using Eye Aspect Ratio Technique

Prathamesh Prakash Tatar
Computer Technology
Rajashri Shahu Maharaj Polytechnic
Nashik,India
prathameshtatar1310@gmail.com

Anushka Dilip Deore
Computer Technology
Rajashri Shahu Maharaj Polytechnic
Nashik,India
deoreanushka94@gmail.com

Akshat Sandeep Gaikar
Computer Technology
Rajashri Shahu Maharaj Polytechnic
Nashik,India
akshatgaikar8@gmail.com

Guided By
Prof P.D.Boraste
HOD,Computer Technology
Rajashri Shahu Maharaj
Polytechnic Nashik,India
prasad.boraste@rsmpoly.org

Snehal Ashok Shewale
Computer Technology
Rajashri Shahu Maharaj Polytechnic
Nashik,India
snehalshewale.05@gmail.com

***Abstract*— Transportation is widely used to allow user travel conveniently from place to place, for a personal of official purpose. Travel during peak hour or holiday, expose the driver to traffic jam for several hour, thus cause the drive to feel drowsy easily due to high concentration and lack of rest. This situation contributes the increasing of the percentage of car incident due to car driver fatigue is the primary origin of the car accident. In this paper, image detection drowsiness system is proposed to detect the state of the car driver using Eye Aspect Ratio (EAR) technique. A developed system that occupied with the Pi camera, Raspberry Pi 4 and GPS module are used to detect and analyse continuously the state of eye closure in real time. This system able to recognize whether the driver is drowsy or not, with the initial, wearing spectacles, dim light and microsleep condition experimental conducted successfully give 90% of accuracy. This situation can increase the vigilant of drivers significantly.**

***Keywords*—Drowsy, Car accident, Eye Aspect Ratio, Raspberry Pi 4, Transportation.**

I. INTRODUCTION

Transportation is a great invention that allows human beings to explore the other places for a long-range distance. In this contemporary era, traffic is a basic need for every human being and the number of transportations on the road is increasing obviously year by year. This situation causes a traffic jam which leads to the time of travel become longer. This maycause the driver feel drowsiness during the long term of traveling time.

Nowadays, statistic shows that road accidents are the primary origin of the number of people death, compare to other roots causes all over the world. There are a lots of sources to lead road accidents, which are (i) the situation of the road such as slippery and pot hole, (ii) the condition of the vehicle, the braking system problem and the main problem is (iii) the attitude of the driver. The attitude of the driver that maycontribute the drowsiness effect due to the driver does not have enough rest thus may cause the road accident. Each human being has the limit including the duration of the driving. Therefore, driver drives should be controlling in a standard period to avoid excessive fatigue and tiredness. Tiredness leads the driver to feel sleepy and loss of focus on the road. What is drowsiness? Drowsiness is a complex phenomenon that states a decrease in alerts and conscious levels of the driver [1]. Different technologies are developed to overcome this problem due to no direct mechanism to detect and measure the drowsiness.

One of the technologies is based on the vehicle like autonomous driving that can monitor steering wheel direction, keep the car in lane position and pressure on the accelerator continuous controlling by the engine control unit (ECU), however this is not the most accurate solution for this problem. On the other hands, the physiological of the driver, which continuously measure the driver heart rate and brain activity by electrocardiogram (ECG), electroencephalogram (EEG), electrooculogram (EOG) and electromyogram (EMG) using a particular custom device was invested, up till now it is an impracticable solution [2].

Another technology is based on the behaviour of the driver, which is tracked the blinking frequency of eye closure using a camera continuously in real-time. This is considered the best and ideal solution to overcome the drowsiness of a driver [3]. Nowadays, some car is occupied with accessories that able to track and analysed the eye closure, once the system detecting the eye, the number of eye closure is analysed. Thus, comparison data with the specified algorithm is performed to identify the eye condition. The driver is alerted through an alarm for any positive drowsy condition.

The drowsiness can be identified through some natural action such as blinking of eyes, yawning, eye closure and head pose [4]. This can be done by installing a camera in front of the driver to capture the real time images of the driver [5]. The driver's images are then further processed to detect the drowsiness of the driver. This can be done by performing live monitoring of Eye Aspect Ratio (EAR) by application of image processing. The real time images are processed using pre-trained Neural Network based Dlib functions. In landmarks retuned Dlib predictor function, each eye is represented by 6 (x, y) coordinates starting at the left-corner of the eye and working clockwise around the remainder of the region [6].

II. METHODOLOGY

For this project, there are many criteria that can be added and optimized in order to increase the effectiveness of the system. The first criteria would be road user's connection and the stand-alone mobile application. The workflow of proposed system of detecting a sleep driver in order to avoid an accident is shown in Fig. 1. In addition, an alert message can be generated to notify its live location of the drowsy driver via telegram to alert other road user's and perform a quick response if something unwanted happen.

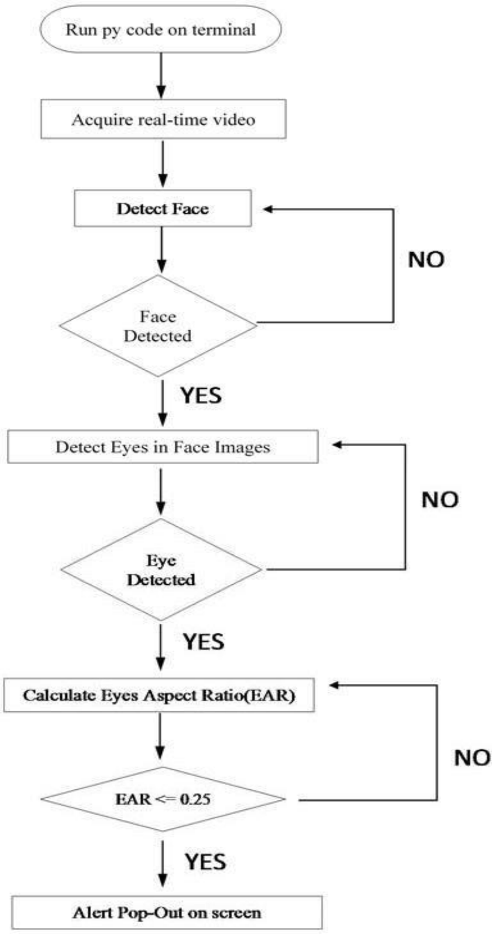


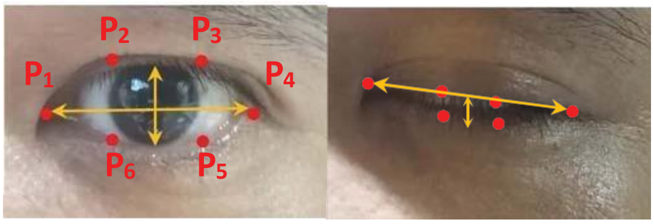
Fig. 1. Block diagram of proposed system.

A. Eyes Aspect Ratio (EAR)

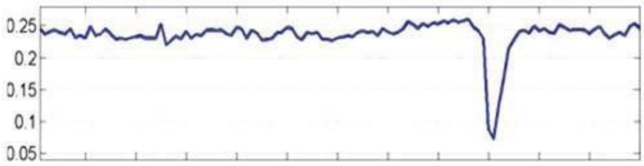
Drowsiness detection system is dominantly requiring detection of eye blink movement. The recorded blink detection through computer webcam can be done by calculating EAR using OPENCV platform and Dlib prediction, also known as Dlib pre trained Neural network-based prediction. The EAR can be calculated based on the eye coordinates returned from OPENCV using the EAR formula as shown in (1) [7][8].

$$\text{EAR} = \frac{|P_2 - P_6| + |P_3 - P_5|}{2|P_1 - P_4|} \quad (1)$$

For (1), the numerator of this equation is the distance between the vertical eye landmarks while the denominator is the distance between horizontal eye landmarks. It contains P_1, P_2, P_3, P_4, P_5 and P_6, as shown in Fig. 2 (a). They are the 2D facial landmark locations. The denominator is weighted appropriately as there are two sets of vertical point and only one set of horizontal points. The EAR is almost constant when the eye is opened while it rapidly fall to zero when a blink .takes place. Therefore, in this project the EAR value is set to be <= 0.25, due to it worked best for this application.



(a)



(b)

Fig. 2. (a) The 6 (x, y) coordinate labelling, and (b) EAR over time [8].

According to Fig. 2 (b), the EAR is constant initially and then rapidly drops near zero but increases again afterwards. This indicates that a blink has taken place [8]. Fig. 2 (b) shows the EAR decreases until it is approaching to zero if a person blink. If the eye of a person is opened, the EAR is relatively constant over time.

B. Python Programming for Proposed method

```
from scipy.spatial import distance
from imutils import face_utils
import imutils
import dlib
import cv2
```

Fig. 3. Installed packages.

Fig. 3 shows basic packages requirement to run the simulation. SciPy package can compute the Euclidean distance between facial landmarks points in the EAR calculation (not strictly a requirement but need to have SciPy installed if intend on doing any work in the computer vision, image processing, or machine learning space). The imutils package is for computer vision and image processing functions to make working with OpenCV easier.

The Dlib library is a histogram of oriented gradients-based face detector with facial landmark predictor. Fig. 4 is a part of code combines both the numerator and denominator to make them become the final EAR as stated in (1).

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 07 | July - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

```
def eye_aspect_ratio(eye):
    A = distance.euclidean(eye[1], eye[5])
    B = distance.euclidean(eye[2], eye[4])
    C = distance.euclidean(eye[0], eye[3])
    ear = (A + B) / (2.0 * C)
    return ear
```

Fig. 4. Python function for EAR calculation.

```
thresh = 0.25
frame_check = 20
detect = dlib.get_frontal_face_detector()
predict = dlib.shape_predictor(".\shape_predictor_68_face_landmarks.dat")# Dat
file is the crux of the code
```

Fig. 5. The facial landmarks code.

Fig. 5 shows a path to Dlib pre-trained facial landmarks detector. Facial landmark prediction is the process of localizing key facial structures on a face, including the eyes, eyebrows, nose, mouth, and jawline. Detecting facial landmarks is a subset of the shape predictionproblem. Given an input image (normally an ROI that specifies the object of interest), a shape predictor attempts to localize key points of interest along the shape.

```
ret, frame=cap.read()
frame = imutils.resize(frame, width=450)
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
subjects = detect(gray, 0)
for subject in subjects:
    shape = predict(gray, subject)
    shape = face_utils.shape_to_np(shape)#converting to NumPy Array
    leftEye = shape[lStart:lEnd]
    rightEye = shape[rStart:rEnd]
    leftEAR = eye_aspect_ratio(leftEye)
    rightEAR = eye_aspect_ratio(rightEye)
    ear = (leftEAR + rightEAR) / 2.0
    leftEyeHull = cv2.convexHull(leftEye)
    rightEyeHull = cv2.convexHull(rightEye)
    cv2.drawContours(frame, [leftEyeHull], -1, (0, 255, 0), 1)
    cv2.drawContours(frame, [rightEyeHull], -1, (0, 255, 0), 1)
    if ear < thresh:
        flag += 1
```

Fig. 6. Python BGR to gray and NumPy setup code.

Fig. 6 shows the conversion code for original image from the BGR color space to gray, that working with code “COLOR_BGR2GRAY”. BGR is a true color image in which each pixel is specified by three values, which one each for the red, blue, and green components of the pixel scalar. BGR and GRAY image can be converted to binary form easily. Binary form is used to detect the landmarks of face. The code “face_utils.shape_to_np(shape)” is used to detected faces, that apply Dlib facial landmark detector. The code “left EAR = eye_aspect_ratio(left eye)” and “right EAR = eye_aspect ratio (right eye)” is using NumPy array slicing that can extract the (*x*, *y*) coordinates of the left and right eye, respectively. From the (*x*, *y*) coordinates for both eyes, which then compute their EAR.

III. Results and Analysis

Fig. 7 (a) and (b) show the result of the initial setup of the experimental process. The findings show that EAR can be detected with the left and right eye respectively. For the case of the eyes closed, the aspect ratio of the eyes result is drastically decreasing as close to 0, whereas throughout the open eyes, the aspect ratio of the eyes produced any whole number which is ‘*x*’ larger than 0. During the eyes closed, the system notified the driver to alert them which can be seen in Fig. 7 (b).

(a)

(b)

Fig. 7. Initial setup (a) Normal eye detection, (b) pop out notification on frame webcam.

Fig. 8. Eye detection on wearing spectacles.

Fig. 8 shows the person wearing spectacles. By wearing spectacles, it can cause the system to do not recognize the eyes correctly and have an error from detecting eyes of driver. This is due to short-sighted problem. To overcome this problem, system is required to run more sample of image with spectacles so that system can be train and learn, thus make system more efficient in detecting the sleepyeyes.

Fig. 9. Eye detection in dim light surrounding.

Fig. 9 shows the person in dim light surrounding. The most obvious constraint when using an image-based method for the system is the lighting. Generally, cameras achieve its maximum capability during the evening where there is not much light. To bypass this constraint, analysts utilized the infrared light-emitting diode (LED). Even though these works quite decently at evening, LEDs are considered less useful during the day. Besides, a large portion of the techniques have been tried on information acquired from drivers emulating drowsiness as opposed to on genuine video information in which the driver gets drowsy naturally from driving.

For the most part, picture is obtained by utilizing a solitary charge-coupled gadget (CCD) or web camera during the day. Meanwhile the infrared camera is used at night. Both cameras operate at 30 fps to have good enough video and not use a ton of space for every time it records a video. Dim light may happen when a driver drives through a tunnel or under a shaded object or at night. Once the system detects the driver is under a dim light environment, the system still worked and act base on the closure of the eyes. After the calculation, the system can give an alert when a prolonged eye closure has occurred. Although the image taken is not clear and in low quality as the picture is taken by a web camera which is more economical and contains only a small number of pixels. Base on the results provided, the system can identify the occurrence of dim light due to the eye that is only one organ which reflect the light in the dim light environment.

Fig. 10. Eye detection in microsleep condition.

Microsleeps are the condition of a person does not pay attention at a certain moment of time. It can identify by occasion as head wobbling, and which they stare blankly into the distance. It usually occurs during tired but attempt to do task that is repetitive or for a long time like driving or staring at computer screen. In any case, these could lead to the most hazardous outcomes of sleep deprivation. Once the system alerts the driver, the system has found out the driver has undergone microsleep. The system takes an action based on the closure of the eyes. As shown in the Fig. 10, the system is still able to give alert for the case of a prolonged eye closure is occurred although the image taken is not good in quality. Based on the results provided, the system can identify the occurrence of microsleep easily.

The application of image processing is depends on calculations of the ratio of the eyes using Python interpreter Anaconda and PyCharm in order to recognize drowsiness managed to be developed and executed in this project. This project was done in order to suggest a simulation of the system prototype. Based on the calculation of the ratio of the eyes, for the low average value it is shows that the driver was drowsy and felt exhausted. As a response the system alert the driver as shown in Fig. 7. Eye detection summarisation is shown as in Table 1.

TABLE 1: RESULTS OF EYE DETECTION

Experimental setup	No of test	Detected	Undetected	Detection Percentage (%)
Initial	10	10	0	100
Wearing Spectacles	10	9	1	90
Dim Light	10	9	1	90
Microsleep Condition	10	9	1	90

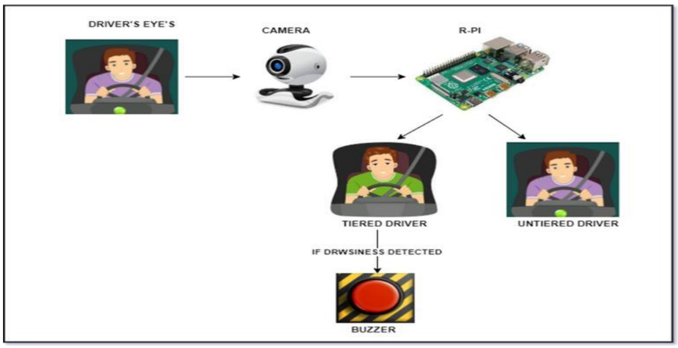


Fig 11.System Architecture

This paper proposes a drowsiness detection system based on EAR. The role of the system is to detect eyes location from images and calculate the value of EAR. In this method each eye is labelled with 6 (x, y) coordinates in landmarks retuned Dlib predictor function. The labelling is starting at the left-corner of the attention, then working clockwise round the remainder of the region. Meanwhile, there's a relation between the distance of those coordinates. Thus, it derives an equation for this relation called the attention ratio and also known as Eye Aspect Ratio (EAR). According to the experimental results, it successfullydetects person during drowsy condition. However, there is still space for the performance improvement. The further work will focus on detecting the distraction and yawning of the driver. Other than that, using sensors, for example, liquor sensor and pulse sensor to distinguish liquor and heartbeat pace of the driver can be included for improvement in physiological-measure analysis.

ACKNOWLEDGMENT

We extend our heartfelt gratitude to Prof P.D.Boraste Sir who played pivotal roles in the development of our drowsiness detection system. Our deepest appreciation goes to our dedicated advisors for their unwavering guidance and profound insights, which were invaluable in shaping the direction of our research. We are also indebted to Rajashri Shahu Maharaj Polytechnic for providing the essential resources and facilities that facilitated our experiments and the overall research environment.

We are thankful to our collaborators for their valuable contributionswhich made this research possible. Furthermore, we express our sincere thanks to the participants whose involvement in our experiments was critical. Lastly, we acknowledge the support and encouragement received from friends and family throughout this journey. Every individual and organization mentioned here played a crucial role in the successful completion of this project.

REFERENCES

1] C. B. S. Maior, M. J. D. C. Moura, J. M. M. Santana, and I. D. Lins, “Real-time classification for autonomous drowsiness detection using eye aspect ratio,” Expert Systems with Applications, vol. 158, 2020

2] K. Kaida, M. Takahashi, T. Åkerstedt, A. Nakata, Y. Otsuka, T. Haratani, and K. Fukasawa, “Validation of the karolinska sleepiness scale against performance and EEG variables,” Clinical Neurophysiology, vol. 117, no. 7, 2006

3] K. Arai, and R. Mardiyanto, “Real time blinking detection based on Gabor filter,” International Journal of Recent Trends in Human Computer Interaction (IJHCI), vol. 1, no. 3, 2010, pp. 33-45

4] R. O. Mbouna, S. G. Kong, and M. G. Chun, “Visual Analysis of Eye State and Head Pose for Driver Alertness Monitoring” IEEE Transaction on Intelligent Transportion Systems, vol. 14, no. 3. 2013

5] A. Dasgupta, A. George, S. L. Happy, and A. Routray “A Vision-Based System for Monitoring the Loss of Attention in Automotive Drivers” IEEE Transaction on Intelligent Transportion Systems, vol. 14, no. 4, 2013

6] T. Soukupova and J. Cech “Real-Time Eye Blink Detection using Facial Landmarks,” Center for Machine Perception, Department of Cybernetics Faculty of Electrical Engineering, Czech Technical University in Prague. Prague, Czech Republic. [Electronic]. https://vision.fe.uni-lj.si/cvww2016/proceedings/papers/05.pdf

7] A. Rosebrock , Eye blink detection with OpenCV, Python, and dlib, (2017) [Electronic] https://www.pyimagesearch.com/2017/04/24/eye-blink-detection-opencv-python-dlib/

8]S. Tereza, and J. Cech, "Eye Blink Detection Using Facial Landmarks." 21st Computer Vision Winter Workshop, Rimske Toplice, Slovenia. 2016