# Real-Time Fatigue Detection System using OpenCV and Deep Learning

**K Vijaychandra Reddy[1], Sanjana Gadalay[2]**

*[1,2]UG student, Dept. of CSE, Mahatma Gandhi Institute of Technology , Telangana, India.*

---------------------------------------------------------------------***---------------------------------------------------------------------

**Abstract –** *With the rise in population, the occurrence of road accidents has been increasing. A myriad number of individuals drive on the highway all around the clock. People traveling long-distance and taxi drivers, bus drivers, truck drivers suffer from sleep deprivation. Due to which it becomes very perilous to drive when feeling sleepy. A recent survey showed that about half a million accidents occur in one year in India. Out of which 60% of these accidents are caused due to Driver's fatigueness.*
*Our objective is to provide a fatigue detection system that will alert the Driver with a loud alarm in the car. Fatigue detection is a safety technology that can prevent accidents that are caused by drivers who fall asleep while driving. Thus, to prevent these accidents, we will offer a system using a model of deep learning and OpenCV that warns the Driver when fatigued. We used OpenCV to gather the images from the webcam using Face Detection and feed them into a convolutional neural network (CNN) that will classify whether the person's eyes are 'Open' or 'Closed'. This system will alert the Driver when fatigue is detected.*

***Keywords:*** Fatigue Detection System, OpenCV, Convolutional Neural Network (CNN), Driver Drowsiness, Face Detection.

## 1. INTRODUCTION

Lack of sleep behind the wheel is driving a motor vehicle while being psychologically impaired by sleep deprivation. Lack of sleep behind the wheel is one of the leading causes of motor vehicle crashes. When a person does not get the necessary amount of sleep, his capacity to function correctly is affected. When their capability to function is affected, they have a longer response time, and their judgment is weakened. Numerous studies have shown that sleep deficiency can affect driving as much as alcohol intoxication.

About 20% of the individuals have admitted to falling asleep at the wheel, with 40% of them confessing that this has taken place at least once in their driving careers. Moreover, research shows that 60% of highway crashes or near crashes occur in India due to drowsy driving. Given these statistics, it is imperative to develop a fatigue detection system.

The discovery of driver fatigue can be an innovation in automotive safety that avoids accidents when the Driver becomes lazy. The real-time sleepiness behaviors associated with fatigue are eye closure, nodding, or brain inactivity. Henceforth, we can either measure change in physiological signals, such as brain waves, heart rate, and eye blinking, to monitor drowsiness or consider physical changes such as drooping posture, sloping of Driver's head, and open or closed state of eyes.

Improving technologies to recognize or avoid lazy driving could be a significant challenge in the field of accident-avoidance frameworks. Given the risk of laziness on the street, strategies had to be put in place to counteract its influences. Driver negligence may be the result of a need for acuity in driving due to driver fatigue and distraction. Diversion occurs when an event or opportunity attracts the attention of an individual. Unlike driver diversion, driver laziness includes no activation opportunity but is characterized by a dynamic withdrawal of consideration from the street and activity requests. Driver fatigue and diversion, in all cases, can have similar impacts, including decreased driver performance, longer response times, and increased risk of accidents.

The former technique is more accurate but not realistic since highly sensitive electrodes would have to be attached directly on the Driver's body, which can be irksome and distracting to the Driver. In addition, long time working would result in sweat on the sensors, diminishing their ability to monitor precisely. The other technique is to measure physical changes (i.e., open/closed eyes to detect fatigue) is well suited for actual-world conditions since it is non-intrusive by using a video camera to detect changes. Moreover, micro-sleeps that are short sleep periods of 2 to 3 minutes are good indicators of fatigue. Thus, the Driver's sleep status can be detected by continually monitoring the Driver's eyes, and a timely warning is issued.

## 2. EXISTING SYSTEM

Previously an infrared sensor (IR sensor) was placed on the eye for fatigue detection. The problem with this system is using the user in complex placement of a sensor over the eye directly. Along with the IR sensor, we have to use some modules in an existing system like a buzzer, raspberry pi processor, and a jumper wire for hardware connections.

The main drawbacks of the existing system include:

- Require mandatory hardware modules which is not cost effective.
- In the hardware, certain connections will be loosened resulting in false detections.

- We must appropriately organize the circuit each time to carry it out. This is a time-consuming process.
- Performance accuracy is poor in detecting sleepiness.

## 3. PROPOSED SYSTEM

Nowadays, the Driver's safety in the car is one of the most desired systems for avoiding accidents. Our goal for the paper is to ensure the security system. A system that can check the Driver's condition for fatigue and alert the Driver before it is too late is desired.

In our proposed system, we are used a driver assistance system using a camera that will focus on the open or closed state of the Driver's eyes by monitoring the state of the eyes, Calculating the EAR ratio continuously for each frame detection drowsiness is easy. In this technique, we will detect the Driver's fatigue state and alert the Driver using an alarm.

Facial detection is achieved through OpenCV face detection. The eye ball is monitored for fatigue detection. The control unit controls every part in this system; if fatigue is detected, the system will give the alarm using the buzzer. Detection in real-time is the major challenge in the field of accident prevention systems.

Advantages of the proposed system are:

- Driver assistance system with cameras focusing free of charge.
- M2M (Machine-to-Machine) communication systems.
- No external hardware apart from web camera.
- It takes just one PC to run the program.
- Caution based on human sensing.

## 4. LITERATURE SURVEY

A variety of methodologies proposed by researchers for the detection of drowsiness and blinking in recent years are:

Manu B.N [1], in 2017, has proposed a method, "Facial Features Monitoring for Real-Time Drowsiness Detection," that detects the face using Haar feature-based cascade classifiers.This article described an effective method for detecting sleepiness through three well-defined phases. These three phases are the detection of facial features with the aid of Viola-Jones, eye monitoring and the detection of yawning.

Amna Rahman [2], in 2016, has proposed a method. "Real-Time Drowsiness Detection using Eye Blink Monitoring." In this method, firstly, the image is converted into grayscale, and corners are detected using the Harris corner detection algorithm. Next, each image calculates the distance 'd' from the mid-point to the lower point to determine the eye state. If the distance is zero or near zero, the eye condition is categorized as "closed"; otherwise, the eye condition is identified as "open".

Anil Kumar, Mansoor Ahmed, Sahana R [3], in 2016 had suggested a method to design drowsiness, a Heart Beat detection system, and an alertness indicator for Driver safety. In this method R-Peak detection algorithm, Frame difference algorithm is used to detect drowsiness. This system provides safety and security for the Driver along with the passengers in real-time implementation. The limitation of this system is that it takes more time to detect drowsiness.

Chih-Hao Ding, Chih-Jer Lin[4] in 2015 proposed "Development of a real-time drowsiness warning system based on embedded system." In this, the Electroencephalography technique is used to detect drowsiness. Electroencephalography-based brain signal analysis can predict the Driver's drowsiness to produce a warning to the Driver.

Y.Ishii, Ogitsu, H.Takemura, and Mizoguchi [5] in 2014 proposed "Real-Time Eyelid Open/Closed State Recognition based on HLAC towards Driver Drowsiness Detection." In this Higher-order, Local Auto-Correlation(HLAC) is used. In this system, it allows the recognition of the eyelid state even from low-resolution images. A Support Vector Machine distinguishes between open and closed eyelids based on the HLAC function.

Aouatif Amine, Nawal Aliousa, Frederick Zann [6] in 2013 proposed "Monitoring Drivers Drowsiness using a wide-Angle Lens" using Circular hough Transform (CHT). The proposed schema starts by extracting the face from the video frame using a skin-color-based face detector. Then, the proposed system using a wide-angle lens makes it possible to detect sleepiness by extracting micro-sleep using the presence of the iris in the eyes of the frontal and profile surfaces. Finally, the analysis of the state of the eyes is based on the circular transformation of Hough (CHT) and is applied to the extracted areas of the eyes.

Ping Wang, Lin Shen [7] in 2012 proposed A Method of "Detecting Driver Drowsiness State Based on Multi-Features of Face" using Adaboost algorithm. They combined the state of the eyes with the state of the Driver's mouth to judge whether the Driver is fatigued, thus solving the traditional challenge of wearing glasses. The AdaBoost algorithm is used to detect the facial area because of its high correct throughput. AdaBoost algorithm is used to detect face region due to its correct high rate.

**Table -1**: Literature Survey

| S NO | Year | Title | Author | Algorithm | Disadvantages |
|---|---|---|---|---|---|
| 1 | 2017 | Facial Features Monitoring for Real-Time Drowsiness Detection | Manu B.N [1]. | Haar cascade classifier technique. | It is more complex and makes a driver uncomfortable and complicated to set up. |
| 2 | 2016 | Real-Time Drowsiness Detection using Eye Blink Monitoring | Amna Rahman [2] | Harris corner detection algorithm. | It is not cost-effective. |
| 3 | 2016 | Design of Drowsiness, Heart Beat Detection System, and Alertness Indicator for Driver Safety | Anil Kumar Mansoor Ahmed, Sahana R [3] | R-peak detection algorithm, Frame difference algorithm. | It takes more time. |
| 4 | 2015 | Development of a real-time drowsiness warning system based on an embedded system. | Chih-Hao Ding, Chih-Jer Lin [4] | Electroencephalography Technique. | Performance is low. |
| 5 | 2014 | Real-Time Eyelid Open or Closed State Recognition. | Y.Ishii, Ogitsu, H.Takemur Mizoguchi [5] | Higher-order Local Auto-Correlation (HLAC). | It is less efficient. |
| 6 | 2013 | Monitoring Drivers Drowsiness using a Wide-Angle Lens | Aouatif Amine, Nawal Alioua, Frederick Zann. [6] | Circular Hough Transform (CHT) | Depending on the speed factor. |
| 7 | 2012 | A Method of Detecting Driver Drowsiness State Based on Multi features of Face | Ping Wang, Lin Shen [7] | AdaBoost algorithm | Performance is decreased. |

## 5. METHODOLOGY

Initially, we take pictures as input from the webcam installed. An endless loop is deployed using OpenCV with the cv2.VideoCapture(0) command; this makes it possible to capture every frame.

To identify the face in the image, we need to change the image to grayscale. OpenCV calculation for Face Detection takes gray images as the input. We do not require color data to identify the objects. We shall use a haar cascade classifier to identify the faces. To set the classifier we used, face=cv2.CascadeClassifier('way to our haar cascade XML file'). At that stage, we perform the detection utilizing faces = face.detectMultiScale(gray). This returns x, y coordinates, width, and height of the boundary box of the object. Now we can iterate on the faces and draw bounding cells for each face.

The same procedure for face detection is used for eye detection. Firstly, we define the eye cascade classifier in the leye and reye variables, respectively, and then perceive the eyes using left_eye = leye.detectMultiScale(gray). We just have to extract the eye data from the entire image. This can be attained by extracting the border box from the eye, and then we can extract the picture from the eye from the frame with this code. leye = frame[ y : y+h, x : x+w ]

leye contains only picture data from the left eye. This information will be transmitted to our CNN classifier to determine whether the eyes are open or closed. Similarly, we will extract the right eye into reye.

The CNN classifier is used to predict the condition of the eye. To integrate our image into the model, we have to pre-process the image. First, we convert the color image to gray levels, and then we resize the image to 24*24 pixels as our model was formed on 24*24 pixels images. We normalize our data for better convergence (All values will be between 0-1). Extend the dimensions to feed our classifier. After loading the model, we predict each eye with our model

lpred = model.predict_classes(leye).

If the value of lpred[0] = 1 is an indication that the eyes are open, if the value of lpred[0] = 0 is an indication that the eyes are closed.

The score is fundamentally a value that we will use to determine how long the individual has closed their eyes. Therefore, if both eyes are closed, we will continue to raise the score, and when the eyes are open, we decrease the score. Using cv2.putText() function, we will determine the number of seconds the person has closed their eyes and display it on the screen. A threshold is defined if the score exceeds 20, which means that the person's eyes are closed for a long time. This is when we emit an audible beep with sound.play() command.

## 5.1 SYSTEM DESCRIPTION

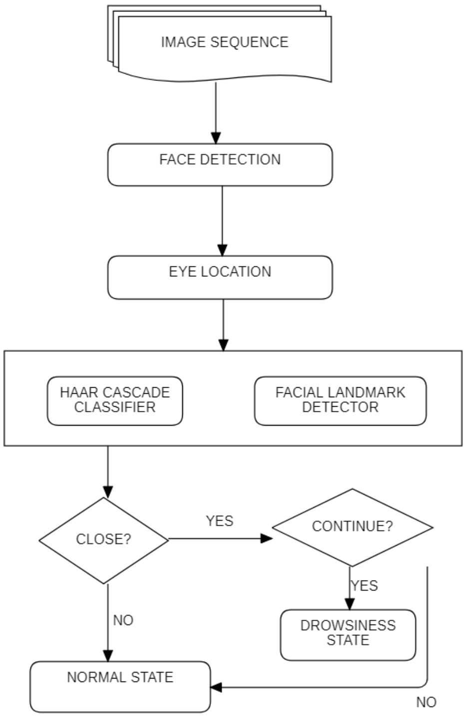


**Fig -1**: System Architecture

The various detection steps are described below:

**Face Detection:**

[8]For facial detection it uses cascading classifiers based on Haar features which is a efficient object detection method proposed by Paul Viola and Michael Jones in their paper, "Rapid Object Detection using a Boosted Cascade of Simple Features" in 2001. This is a machine learning approach whereby a cascade function is formed from numerous positive and negative images. It is then used for the detection of objects in other pictures.

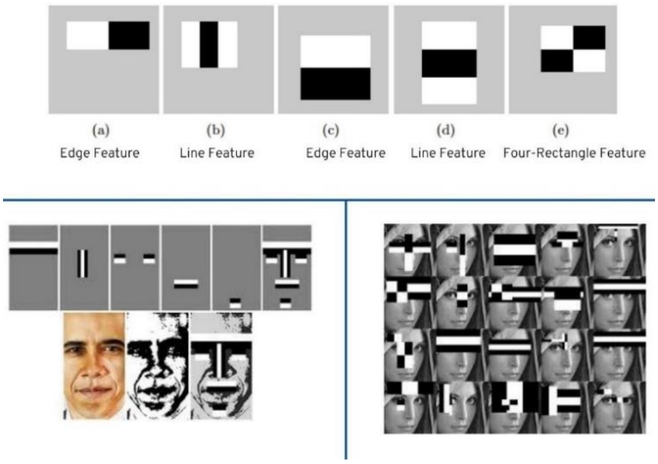


**Fig -2:** Haar Cascade Features

To begin with, the algorithm requires numerous positive (facial images) and negative (faceless images) images to form the classifier. Next, we must extract the features. This is done with the Haar features shown in Fig. 2. These are like our convolutional kernel. Each feature is a single value acquired by deducting the sum of pixels below the white rectangle from the sum of pixels below the black rectangle. Fig. 2 represents five Haar-like features & examples.

**Eye detection:**

In this technique, we have used facial landmark prediction for eye detection. Facial landmarks are used to confine and represent prominent areas of the face, such as:

- Mouth
- Nose
- Jawline
- Eyes
- Eyebrows

For the detection of blinks, the estimation of the posture of the head, the change of the face, the alignment of the face, the facial landmarks were applied.

The identification of facial markers is therefore a two-step process:

- Locate the face within the picture.
- Identify key facial structures within the facial region of interest.

Locate the face within the picture: The face image is localized by Haar feature-based cascade classifiers which were discussed in the first step of our algorithm i.e., face detection.

Identify key facial structures within the facial region of interest: Various face landmarks are available, but all methods are intended primarily to locate and label the following facial regions:

- Nose
- Mouth
- Left eye
- Right eye
- Left eyebrow
- Right eyebrow

**Facial Landmark Detection**:

[9] The facial landmark detector included in the dlib library is based on "One Millisecond Face Alignment with an Ensemble of Regression Trees" paper by Kazemi and Sullivan (2014).

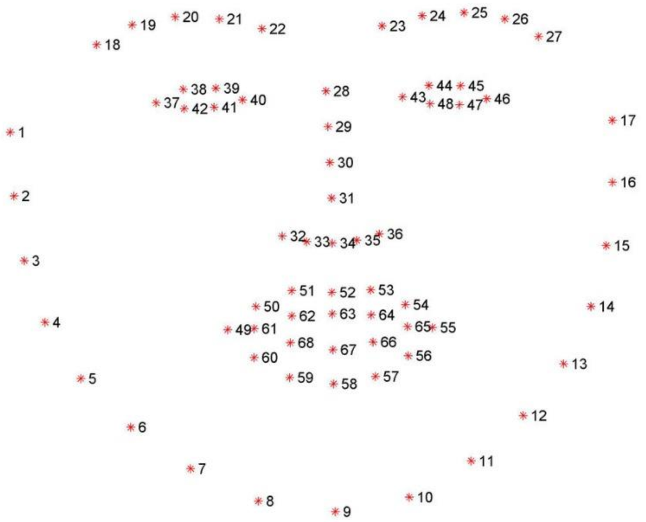


**Fig 3**: Facial Landmark Detection

[10] Facial landmark detection begins with:

1. A training set of pictures with labeled facial landmarks. These pictures are labeled manually. They provide the specific coordinates (x, y) of the areas around each facial structure.

2. In advance, more specifically, the probability of the distance between the input pixel pairs. The pre-trained face landmark detector assesses the location of 68 (x, y) coordinates that match the facial structures on the face. The indices of the 68 coordinates can be seen in the image in Fig. 3.

We can detect and access the eye region through the following facial landmark.

- The right eye using [36, 42].
- The left eye with [42, 48].

Regardless of the dataset used, the same dlib framework may be used to train an input training data form predictor.

**Recognition of Eye's State:**

The eye area can be estimated from the optical flow, by sparse monitoring or by the difference in image-to-image intensity and adaptive threshold. Moreover, finally, a decision is reached if the eyes are or are not covered by eyelids. We offer a simple but effective algorithm to detect eye blinks using a newer facial landmark detector. In addition, a unique scalar amount that reflects an eye-opening level is derived from the landmarks. Finally, with a frame-by-frame series of eye-opening estimates, eye blinks are found by an SVM classifier that is constructed on examples of blinking and non-blinking patterns.

**Eye Aspected Ratio Calculation:**

For every video frame, the eye landmark is detected. The eye aspect ratio (EAR) between width and height of the eye is computed.

$$\mathrm{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

**Fig 4**: EAR Formula

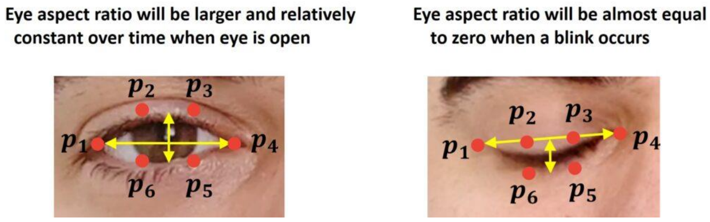


**Fig 5**: Eyes (Open and Closed) with landmarks

Where p1, . . ., p6 are the 2D landmark locations, depicted in Fig. 5. The EAR is essentially constant when eyes are open and approaches zero while closing the eyes. It is partially person, and the head poses insensitive. The eye aspect ratio of the open eye has a slight variance among individuals. It is totally invariant for uniform image scaling and rotation in the facial plane. As the two eyes blink synchronously, the EAR in the two eyes is averaged.

EAR is a sophisticated remedy which requires a simple estimation based on the relationship of the distances between the facial landmarks of the eyes.

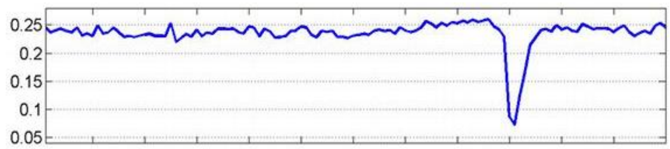


**Fig 6**: EAR for Single Blink

The eye aspect ratio EAR was plotted for several frames of a video sequence. This is the EAR ratio for single blink User flexibility. This interface enables the user to set the EAR accordingly, ranging from low to high percentage.

**Eye State Determination:**

Finally, the decision for the eye state is made based on the EAR calculated in the previous step. If the distance is zero, the eye state is classified as "closed"; otherwise, the eye state is identified as "open." We set a time-out of five seconds. If the eyes remain closed for five seconds or more, drowsiness is detected, and an alarm is activated.

## 6. RESULTS

In the beginning, the webcam is set up to monitor the stream of faces. With facial detection, it detects the face and a boundary box is drawn for each face. Through facial landmark detection, we extract the eyes and calculate the EAR ratio. In Fig. 7, the fatigue detection system senses that the eyes are open with an EAR score of 0.

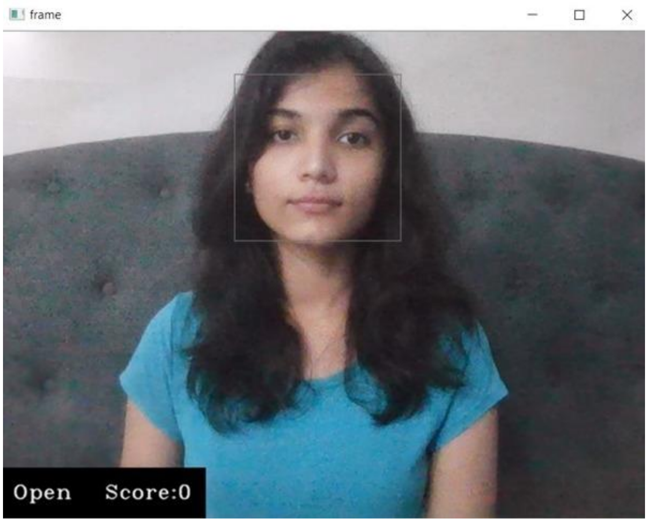


**Fig 7**: Fatigue Detection system when the eyes are open.

When the Driver closes his/her eyes, the EAR score increases. Once the limit of 20 is reached, our system triggers an alarm to alert the Driver, as indicated in Fig. 8.

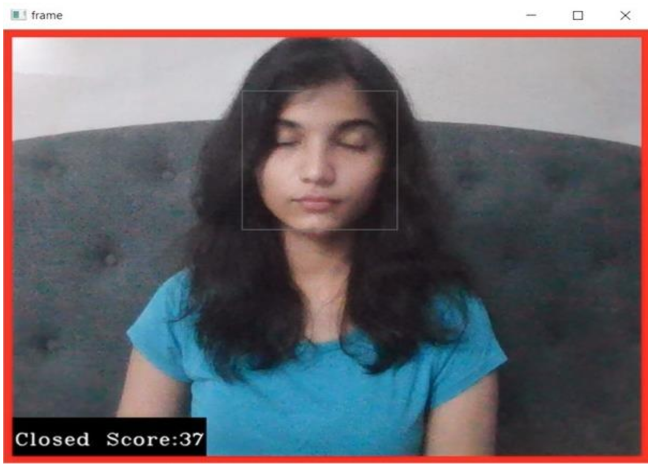


**Fig 8**: Fatigue Detection system with an alarm in background when the eyes are closed.

As a result of the COVID-19 pandemic, face masks have become mandatory. Hence, this proposed technique works well with the inclusion of masks as well. This can be demonstrated by Fig. 9 and Fig. 10.

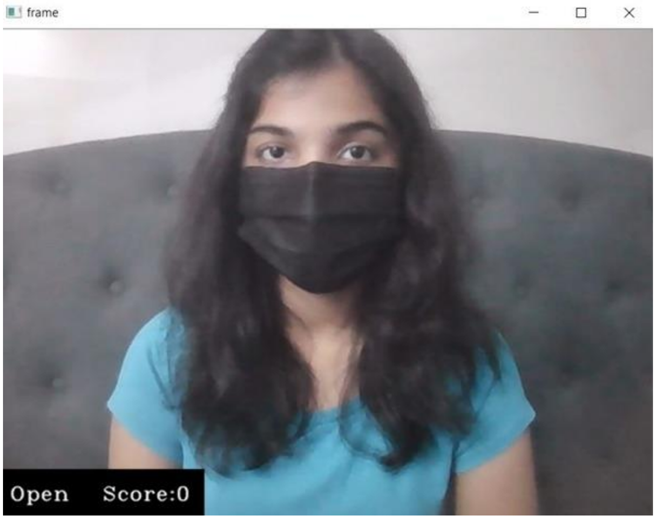


**Fig 9**: Fatigue Detection system along with the face mask.

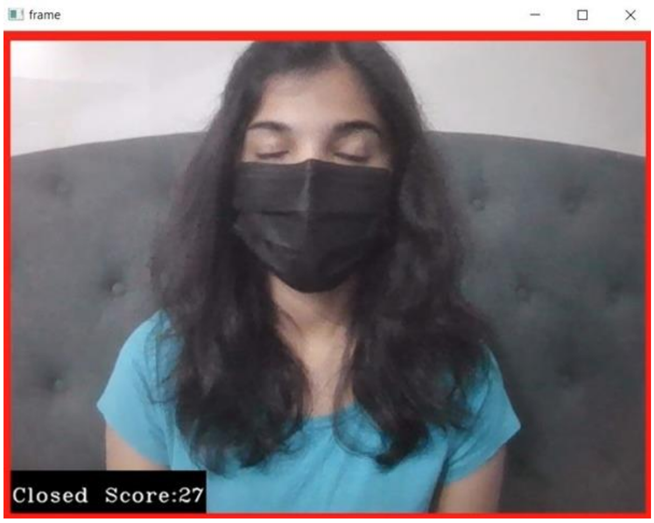


**Fig 10**: Fatigue Detection system along with the mask and the person is drowsy (eyes closed).

## 7. CONCLUSIONS

Driving in a fatigued state is a serious threat to drivers and traffic participants. Unfortunately, the previous systems lack essential features that provide non-reliable results and use complex IR sensors. In this paper, we proposed implementing a vision-based driver fatigue detection system to prevent the Driver from being fatigued.

This system will overcome previous drawbacks and provide accurate and reliable results. The overall workflow of our fatigue detection algorithm is relatively straightforward. First, a camera is setup to monitor the

Driver. After which, we apply facial landmark detection and extract the eye regions. We then compute the eye aspect ratio to determine if the eyes are closed. Video segments whose average eye state point exceeds the threshold value are detected as drowsy, and the Driver is alerted. This system can determine the driver state under actual day and night conditions using an IR camera. The results are adequate, with the possibility of improving facial detection through other symmetry calculation techniques.

## 8. FUTURE SCOPE

The system can also be used efficiently in locomotives and airplanes. It has a broad scope in the future and can be improved to meet excellence. In the future, this technique will be a part of a safety system being used in vehicles and help us save many lives. Soon, the project can be improved to detect passenger's faces and only focus on the Driver's face. The vehicle manufacturers can make this system inbuilt by using the dashboard screen and speakers. The system can be effectively used in locomotives and flights to detect driver drowsiness. The system can be improved to detect and track eyes even if the Driver is wearing shades.

This model is designed to detect the fatigue state of the eye and give an alert signal or warning in the form of an audio alarm. Nevertheless, the Driver's response after being alerted may not be sufficient to stop causing the crash. If the Driver reacts slowly to the warning message, the accident may occur. Hence to avoid this, we can plan and fit a motor-driven system and coordinate it with the warning signal so that the vehicle will slow down after immediately getting the warning signal.

We can also provide the user with an Android application that will provide the information of his/her drowsiness level during any journey. The user will know the Normal state, the Drowsy state, the number of times blinked the eyes according to the number of frames captured.

## REFERENCES

[1] Manu B.N "Facial Features Monitoring for Real Time Drowsiness Detection", 12th International Conference on Innovations in Information Technology (IIT), 2017.
[2] Amna Rahman "Real Time Drowsiness Detection using Eye Blink Monitoring" by Department of Software Engineering Fatima Jinnah Women University, National Software Engineering Conference (NSEC), 2016.
[3] Anil Kumar, Mansoor, Ahmed, SahanaR "Design of Drowsiness, Heart Beat Detection system and Alertness Indicator for Driver Safety", International Journal of Science Engineering and Technology Research (IJSETR), 2016.
[4] Chih-Hao Ding, Chih-jer Lin "Development of a real-time drowsiness warning system based on an embedded system", 12th International Conference on Innovations in Information Technology (IIT), 2015.
[5] Y.Ishii, Ogitsu H.Takemura, Mizoguchi "Real-Time Eyelid Open/Closed state Recognition based on HLAC towards Driver Drowsiness Detection" Department of Software Engineering 2014 National Software Engineering Conference (NSEC), 2014.
[6] Aouatif Amine,Nawal Alioua, Frederick Zann. "Monitoring Drivers Drowsiness using Wide Angle Lens", 11th International Conference on Innovations in Information Technology (IIT), 2013.
[7] Ping Wang Lin Shen. "A method of detecting driver drowsiness state based on multi-features of face", International Journal of Science, Engineering and Technology Research (IJSETR), 2012.
[8] P. Viola, M. Jones. "Rapid object detection using a boosted cascade of simple features", Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2001.
[9] Vahid Kazemi, Josephine Sullivan. "One millisecond face alignment with an ensemble of regression trees", IEEE Conference on Computer Vision and Pattern Recognition, 2014.
[10] Hakan Cevikalp; Bill Triggs; Vojtech Franc, "Face and landmark detection by using cascade of classifiers", 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), 2013.