

Drowsy Driver Detection using MSER Feature Detection and Binarization on MATLAB

Asma Jamesh*

Biomedical Engineering, Sri Ramakrishna Engineering College, Coimbatore, India

*Corresponding author

doi: <https://doi.org/10.21467/proceedings.114.31>

Abstract

Every year, 1.5 lakh people die in road mishaps in India. Among these accidents, 40% are due to 'Drowsy or Sleep Driving'. According to several statistics, almost all commercial private drivers tend to drive continuously for 10 hours a day. Nearly all road accidents caused due to lack of sleep and drowsiness are highly hazardous and fatal. Drowsy Driver Detection Algorithm acquires real-time video and captures snapshots using an external Webcam. Using the Viola-Jones Algorithm, the face and the eyes of the driver are detected. The original RGB eye image is converted to a Gray image and then into a Binary image. Two techniques, Maximally Stable Extremal Regions Feature Detection and Binarization are deployed to determine the status of the driver. This research paper focuses on the development of a MATLAB algorithm to alert the driver or the co-passenger on-time by plotting the MSER Features and thresholding the acquired real-time images.

Keywords: Real-time video, Viola-Jones Algorithm, MSER Features, Binarization.

1 Introduction

According to National Sleep Foundation (NSF), 37% of people tend to fall asleep behind the wheel. These sleep-related crashes have been higher in younger people age 18-27. The National Highway Traffic Safety Administration estimates that about 100,000 accidents occur due to the direct result of a drowsy driver, each year. In India, 65% of the total freight transport takes place through trucking. According to 'Motor Transport Acts, 1961' in India, the maximum working hours for drivers are restricted to only 8 hours a day but, 36% of the drivers violate these rules. Accidents caused due to speediness, alcohol consumption, and drowsiness result in an economic loss of close to 3% of the country's total GDP.

Various factors need to be considered when analyzing a drowsy driver like age, education, marital status, etc. A few reasons like lack of adequate sleep, driving long distances alone, and the presence of untreated or unrecognized diseases like Narcolepsy are proved to cause drowsiness. A study that took place stated that most of the accidents are caused between 12 am to 4 am and in the post lunchtime of 3 pm to 4 pm.

According to the USA legal limit, driving continuously for 20 hours without sleep is equivalent to driving with a Blood Alcohol Concentration (BAC) of 0.08%. Severe drowsy driver accidents tend to occur at high speeds on highways or other major roads.

Major warning signs that are symptoms of drowsiness:

- 1) Yawning
- 2) Inability to keep eyes open
- 3) Nodding off or unable to keep your head up
- 4) Ending up too close to other cars on the lane

Drowsy Driver is as dangerous as drunken driving and it is vital to prevent such tragic incidents. At the end of the day, drowsy drivers not only put themselves at complete risk but every single person on the road. It is crucial to develop a system that can detect drowsiness patterns and alert the driver. Drowsy Driver Detection detects the status of the driver and alerts when the driver is heavy-eyed, by analyzing the captured image using a MATLAB Algorithm.



The paper is organized as follows: Section 1 states the Introduction, Section 2 explains the methodology or procedures followed in processing the real-time image, Section 3 provides a detailed explanation of MSER Feature detection and Section 4 elucidates how the algorithm works and drowsiness is detected. The last section involves the presentation of results and the conclusion of the paper.

2 Methodology

Drowsy Driver Detection deploys a detailed interpretation of various image processing techniques to detect the nature of the driver. MATLAB R2020b (Matrix Laboratory), an advanced computational and analysis platform that has several toolboxes like Image Acquisition, Image Processing, and Computer Vision, has been utilized to interpret the result. An external Webcam is positioned in front of the driver to record a video as soon as the car starts.

2.1 Image Acquisition

An external Webcam is placed at an angle of 45° in front of the driver’s seat to acquire clear images of the driver. Using the ‘Image Acquisition Toolbox Support Package for OS Generic Video Interface’ MATLAB toolbox, an interface between the algorithm and Webcam is created. [1]

$$vid=videoinput('winvideo',1); \tag{1}$$

$$start(vid) \tag{2}$$

The above-mentioned MATLAB command called ‘video input(‘winvideo’)’ is executed to begin the recording of the video. Real-time is obtained and multiple snapshots are captured and processed. Images from the live videos are used to analyze various features.

2.2 Face Detection

The face of the driver is detected from the original snapshot, which is obtained from the real-time video. The MATLAB toolbox, ‘Computer Vision’ is utilized to localize the face separately.

MATLAB command called ‘vision.CascadeObjectDetector(‘FrontalFaceCART’)’ has been deployed. [12]

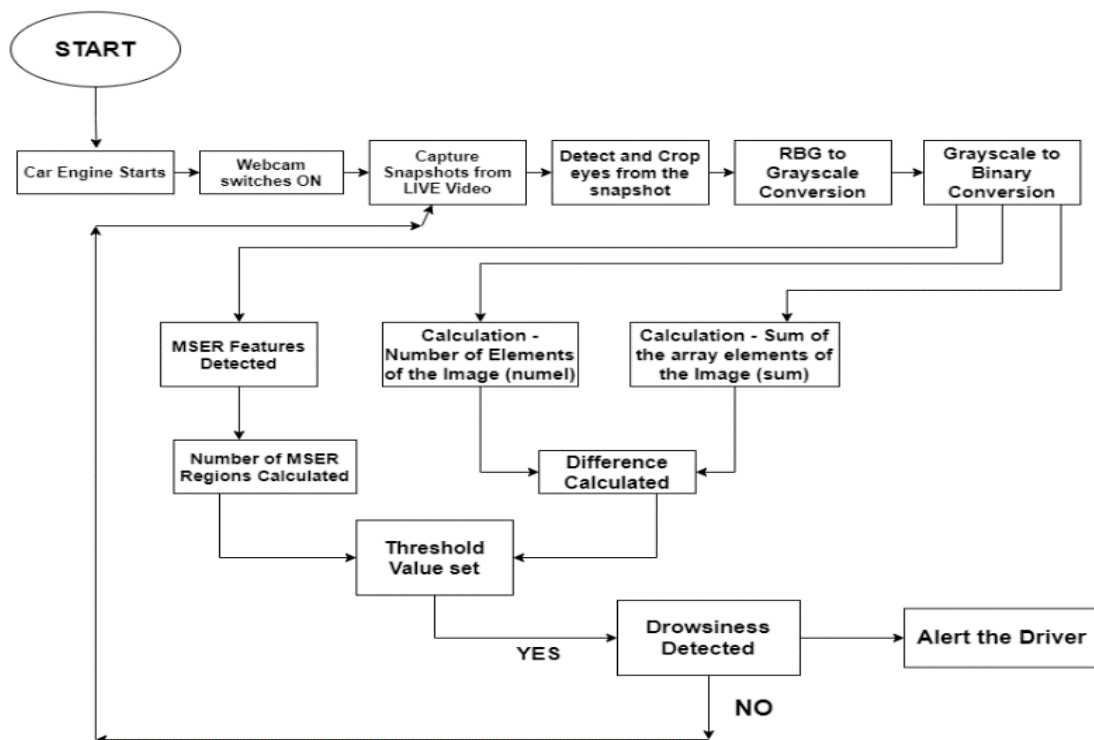


Figure 1: A Flowchart of the Drowsy Driver Detection Algorithm

$$Fc=vision.CascadeObjectDetector('FrontalFaceCART') \quad (3)$$

The above MATLAB command is executed to detect faces that are upright and forward-facing. The detection process is done based on 'Classification and Regression Tree Analysis.

The images in Fig.2 and Fig.3 show a sample face detection that has been employed for the Drowsiness Detection. For a given period, the 't', 'n' number of snapshot samples are acquired from the live video. Eventually, the driver or the co-passenger will be alerted on time.

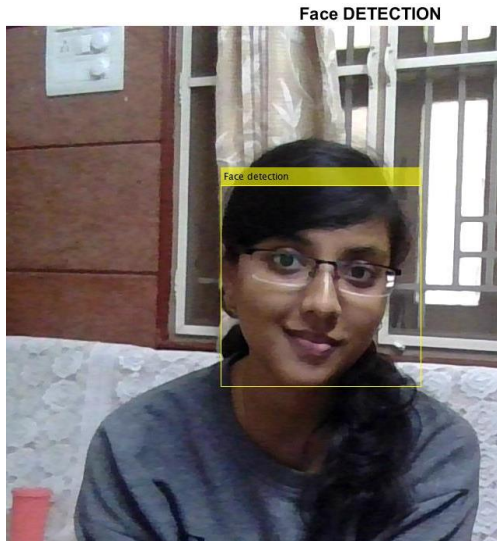


Figure 2: A sample Face detection using Viola-Jones Algorithm from a camera.



Figure 3: A sample Eye Detection using Viola-Jones Algorithm

2.3 Eye Detection

After the detection of the face of the driver, the eyes of the driver are detected. Viola-Jones Algorithm is utilized to segment the eyes of the automobile driver from the detected face. MATLAB Command called 'vision.CascadeObjectDetector('EyeBigPair')' is executed to segregate the eyes. The bounding box is drawn around the eye pair and is cropped from the original image. [13]

$$eye=vision.CascadeObjectDetector('EyeBigPair'); \quad (4)$$

The above-stated MATLAB command is deployed to detect a pair of eyes from the real-time video.

$$bbox2=step(eye,a); \quad (5)$$

$$out = insertObjectAnnotation(a,'rectangle',bbox2,'Face detection'); \quad (6)$$

The bounding box is drawn in the form of a rectangle to annotate the detected eyes using the code mentioned above. The purpose of using a cropped eye image is to reduce the noise and the size of the image to speed up the analyzing process. Fig. 3 shows a sample image of Eye Detection from an original snapshot captured from the external Webcam.

2.4 Conversion of RGB Image to Grayscale Image

RGB images on MATLAB are saved in the form of m x n matrices. A MATLAB command, `rgb2gray` is used to convert original RGB images to Grayscale Images by eliminating the hue and saturation information.

Cropped Eye RGB



Figure 4: A cropped RGB image of the eye

Cropped Eye

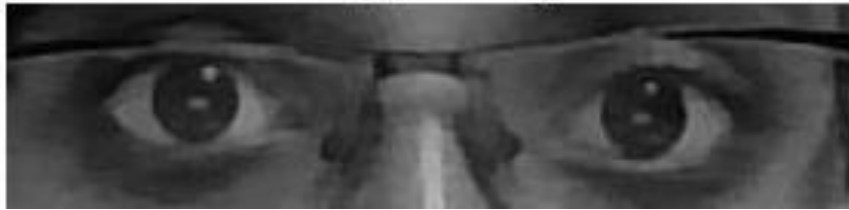


Figure 5: A cropped Grayscale image of the eye

$$\text{Gray_Image}=\text{rgb2gray}(\text{Image}); \quad (7)$$

2.5 Conversion of Grayscale Image to Binary Image

The conversion of a Grayscale image to a Binary format is the most crucial and critical aspect of the image processing technique. Here, ‘imbinarize’ is the MATLAB command that is deployed to create a binary image from a 2D or 3D Grayscale image. A threshold value is set to specify the matrix of the luminance values.

$$\text{Bin_img}=\text{imbinarize}(\text{Image}); \quad (8)$$

Binary Image



Figure 6: A cropped Binary image of the eye

3 MSER Feature Detection

Maximally Stable Extremal Regions is a technique of the Computer Vision Toolbox that aids in the blob detection of images. MATLAB command called detectMSERfeatures (I) has been executed to return an object that contains information on MSER features and MSER regions. [11]

Table 1 illustrated above states the advantages and limitations of various feature extraction techniques. MSER Features of the real-time eye snapshot is obtained. Feature extraction with the MSER technique is proved to be reliable and stable. Here, the number of MSER Regions or the count of MSER Regions is obtained.

$$\text{det2}=\text{detectMSERFeatures}(\text{gcrop}); \quad (9)$$

$$\text{plot}(\text{det2},\text{'showPixelList'},\text{true},\text{'showEllipses'},\text{false}); \quad (10)$$

Table 1: Comparison of various Feature Extraction techniques

Sno.	Techniques	Advantages	Limitations
1.	HAAR Classifier	High Calculation Speed	Costly
2.	SURF (Speeded up Robust Feature)	Robust	Sensitive to Illumination
3.	MSER (Maximally Stable Extremal Regions)	Reliable and Stable, Invariant to affine transformation	Slow
4.	Min-Eigen	Invariant to rotation	Doesn't support scaling and illumination changes

**Figure 7:** MSER Feature Detection of Pre-loaded Images (Open and Closed Eyes).

The images from Fig.7 depict the MSER Features that are detected on the two eye images. The eye images used here are pre-loaded reference images. The correspondence of image elements between two different image elements is analyzed and matched.

4 Drowsiness Detection Algorithm

Once the image of the eye pair is segregated from the captured snapshot, the next step is to detect if the eyes are close or not. The eyes are analyzed to understand if the driver is drowsy or sleepy.

The Drowsiness Detection Algorithm makes use of three important conditions:

1. The number of array elements in the binary eye image.
2. The sum of the pixels of the binary eye image.
3. The count of MSER Regions detected in the Grayscale eye image.

Since all the Images on MATLAB are stored in the form of matrices, $m \times n$, the MATLAB Command, `numel (Image)` is executed to return the number of elements present in the matrix of the binary eye image. The binary image is produced by thresholding a grayscale, to separate an object from an image in the background. The MATLAB Command, `sum (Image (:))` is executed to measure the sum of all the pixels present in the binary eye image [5].

The difference between the number of elements of the binary image and the sum of all the pixels of the binary image is calculated. This difference provides the number of Black pixels present in the Binary eye image.

$$No_Black_pixels = numel(image) - sum (image(:)); \quad (11)$$

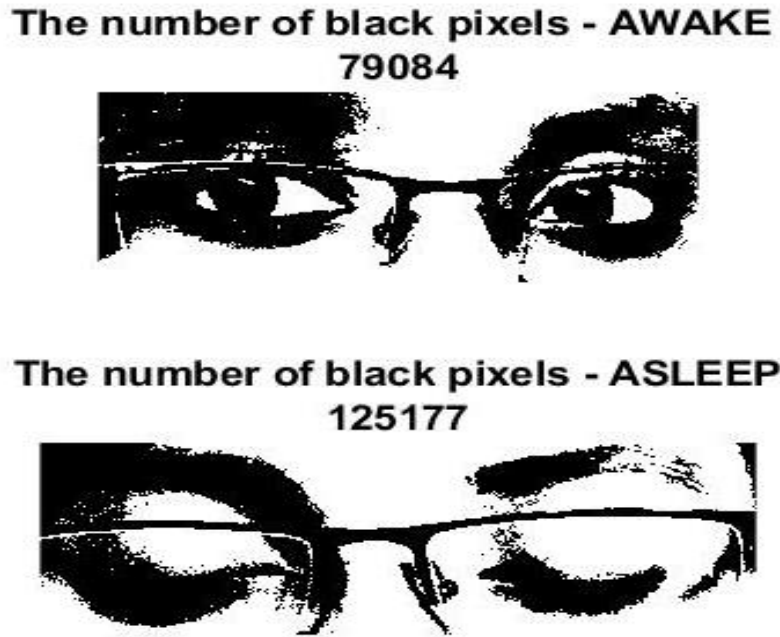


Figure 8: Number of Black pixels of pre-loaded reference images of the driver (Open and Closed Eyes)

Fig.8 shown above displays the number of black pixels present in the pre-loaded reference images. Here, both eyes open and eyes closed image is utilized.

Additionally, to improve the accuracy and efficiency of drowsiness detection, the MSER Regions are also detected. Here, the count of the MSER Regions of eye open and eye close images are measured and compared.

$$regions = detectMSERFeatures(gcrop); \quad (12)$$

$$A = regions.Count; \quad (13)$$

The MATLAB Commands, (11) and (12), mentioned above are executed to provide the count of MSER Regions of the grayscale real-time image.

5 Results

Numerous samples of snapshots have been utilized to test the Drowsy Driver Detection Algorithm. A few sample images that were tested are shown in Fig. 9 and Fig.10. Sample Images from a driver, in two states – Conscious (Eyes open) and Unconscious (Eyes Closed) state have been tested and are shown in Fig 9 and Fig 10.

In Figure 9, the MATLAB algorithm can detect the MSER Features of the person with open eyes. The count of the number of the MSER Regions on the grayscale eye image is also calculated. Additionally, the number of black pixels is measured for the open-eyed binary snapshot.

By comparing the analyzed parameters, **the number of black pixels, and the count of MSER Regions**, the algorithm was able to successfully detect the current status of the driver. In this case, the driver is seen to be with open eyes and the algorithm has rightly displayed **‘Person Awake!!’**.

Similarly, for a driver with close eyes, the MATLAB Algorithm is successfully able to detect drowsiness. **The number of black pixels, as well as the count of the MSER Regions of the eye image**, is calculated. In this incident, the algorithm analyses the image and displays **“Person Asleep!!”**.

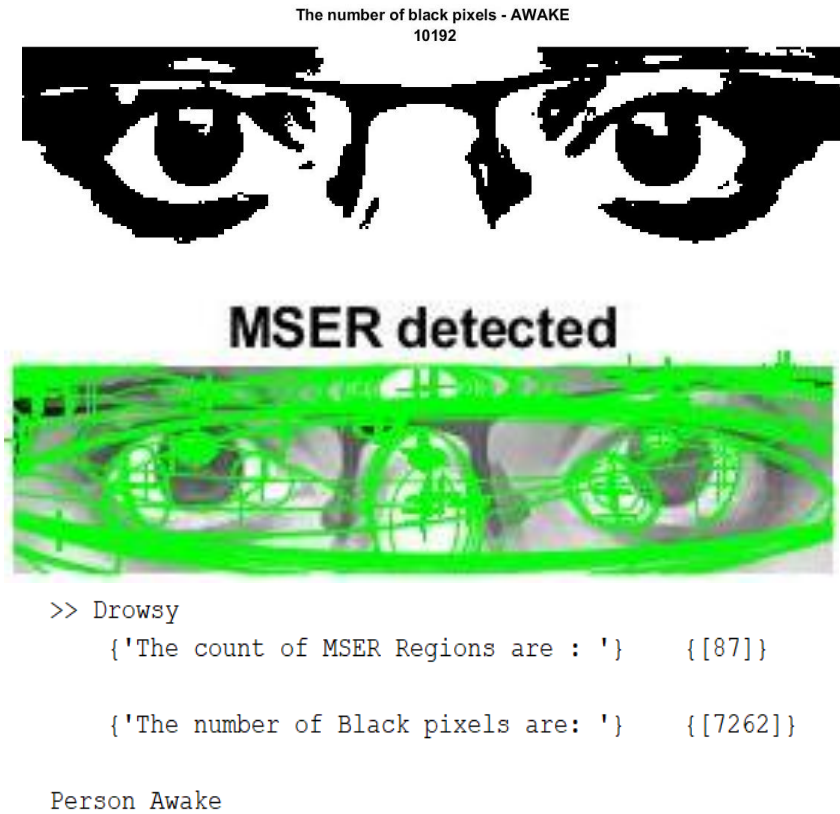


Figure 9: Drowsy Detection Algorithm tested on a conscious driver

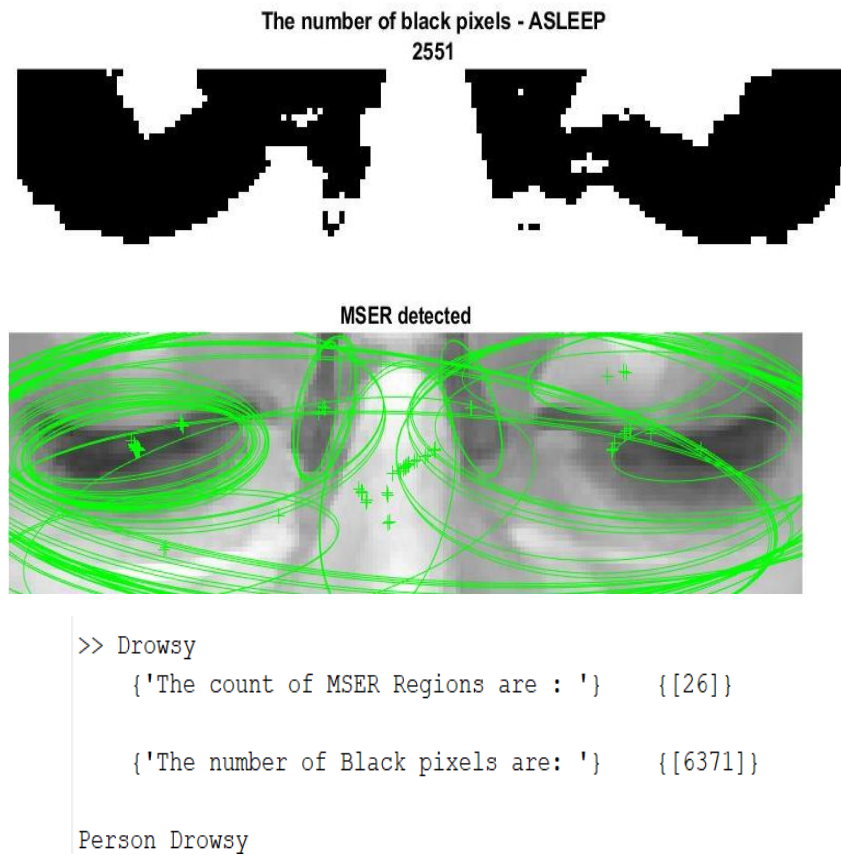


Figure 10: Drowsy Driver Detection Algorithm tested on an unconscious driver.

The MSER Regions have been marked and illustrated for both the images and the MATLAB results have also been obtained.

6 Conclusion

The Drowsy Driver Detection Algorithm is a unique, efficient, and highly accurate MATLAB program that determines the state of the driver using a live image captured with the help of an external Webcam. To detect the drowsiness of the driver, two to three images are considered and analyzed. When all the three images are in a state of **'closed' eyes**, then an alert is signaled to the driver or the co-passenger. The algorithm developed using MATLAB software is efficient enough to detect with or without glasses of the driver. The two parameters are measured, simultaneously; The count of MSER Regions and the number of Black pixels provides contributes to the competence and the effectiveness of the algorithm. The MSER Region count is calculated from a grayscale eye image whereas the number of black pixels is measured from the binary eye image.

Limitations of the Drowsy Driver Detection Algorithm are:

1. Incorrect results are obtained due to improper posture and face position.
2. Inaccurate results are obtained due to an insufficient amount of light on the face of the driver.

For future work, auxiliary monitoring of the number of blinks of the driver's eye, the status of the mouth (open or closed), and the frequency of yawning can be monitored and analyzed. A significantly enhanced method of accident prevention can be facilitated by interfacing the drowsy detection algorithm with the car's engine. When the drowsiness of the driver is detected, the algorithm can be programmed in a way where the car's engine will be automatically turned off to prevent tragic and fatal events. [8]

References

- [1] Jang Woon Baek, "Real-Time Drowsiness Detection Algorithm for Driver State Monitoring Systems," 2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN), July 2018.
- [2] Anthony. D. McDonald, "A contextual and Temporal Algorithm for Driver Drowsiness Detection", Science Direct, Volume 113, April 2018, Pages 25-3.
- [3] Teik Jin Lim, "Eye Fatigue Algorithm for Driver Drowsiness Detection System, ICIPCN 2020: Image Processing and Capsule Networks pp 638-652.
- [4] Nordbakke, S., Sagberg, F.: Sleepy at the wheel: knowledge. Symptoms and behavior among car drivers. *Transp. Res. Part F Traffic Psychol. Behav.* **10**, 1–10 (2007)
- [5] Maninder Kahlon and Subramaniam Ganesan, "Driver Drowsiness Detection System Based on Binary Eyes Image Data", 2018 IEEE International Conference on Electro/Information Technology (EIT) May 2018
- [6] R. Pooneh. And R. Tabrizi, and A. Zoroofi, "Drowsiness Detection Based on Brightness and Numeral Features of Eye Image," Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," *IEEE Transl. J. Magn. Japan*, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
- [7] Ms. Shalini Kashyap1, Mr. V.K Sharma2, "Drowsiness Detection system using Matlab", *IJARSE*, Vol. No. 6, May 2017.
- [8] Namgiri Suresh and Gayam Akil, "Design Of A Vehicle Driver Drowsiness Detection System Through Image Processing Using MATLAB", *International Journal of scientific & technology research* VOLUME 9, ISSUE 04, APRIL 2020
- [9] Manan Arora1 and Nandika, "Driver's Sleep Detection", *International Research Journal of Engineering and Technology (IRJET)*, Volume: 06 Issue: 05 | May 2019.
- [10] Venkate Gowda Mn, "Driver Drowsiness Detection using MATLAB and Controller", April 2013, Conference: National Conference on Electronics and Communication Engineering 2013.
- [11] Anis-Ul-Islam Rafid, "A Brief Review on Different Driver's Drowsiness Detection Techniques", June 2020, *International Journal of Image, Graphics and Signal Processing*.
- [12] Bahajathul Fathema, "Real-Time Face Detection Using Matlab (Using Viola-Jones Algorithm)", *International Journal of Engineering Research & Technology (IJERT)*, Vol. 7 Issue 02, February-2018.
- [13] M. A. González, P. N. Posso, "Algorithm for eyes and mouth detection in face recognition using Simulink", *Research Gate*, June 2013
- [14] Hanojhan Rajahrajasingh, "Driver Drowsiness Detection using MATLAB", *International Journal of Engineering Applied Sciences and Technology*, 2019 Vol. 4, Issue 7, ISSN No. 2455 - 2143, Pages 80-87