



e-ISSN: 2319-8753 | p-ISSN: 2347-6710

IJRSET

International Journal of Innovative Research in
SCIENCE | ENGINEERING | TECHNOLOGY



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH


IN SCIENCE | ENGINEERING | TECHNOLOGY

Volume 12, Issue 9, September 2023

ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA

Impact Factor: 8.423

 9940 572 462

 6381 907 438

 ijrset@gmail.com

 www.ijrset.com



Real Time Driver Fatigue Detection System Using Facial Landmarks

Santhosh T^[1], Hemashree H C^[2], Manasa G R^[3], Madhuri Martis^[4]

Assistant Professor, Department of Information Science & Engineering, Bapuji Institute of Engineering & Technology, Davanagere, Karnataka, India^{[1][2][4]}

Assistant Professor, Department of Computer Science & Engineering, NMAM Institute of Technology, Nitte, Karnataka, India^[3]

ABSTRACT: Around 75% of commuters in India drive day and night continuously for long distances usually will be having insufficient sleep. Most of the accidents will happen due to the fatigue of the driver. To prevent such kind of mishaps drowsiness identification system is necessary. In this work we have developed a real time driver drowsiness identification system, which is based on facial features, i.e., the eyes. This detection system is implemented using a simple algorithm to calculate the eye aspect ratios for both the eyes, with minimal hardware requirements for computation purposes. This detection system is web camera based, which capture the sequence of images, and then captured images are stored as a series of frames in system for further processing and these frames used as an input to build in detection functions like facial landmarks that operate in a dlib environment. Processed frames are then compared to detect the drowsiness of a driver while driving the vehicle. The system alerts the driver about fatigue by giving suitable warning. In the current work histogram of oriented gradients for feature extraction and linear support vector machine (LSVM) algorithm is used to implement the proposed system.

KEYWORDS: Drowsiness, Facial features, Web camera, Linear support vector machine.

I. INTRODUCTION

Drowsy driving is riding a vehicle when person can't remain alert while driving due to fatigue. This leads to road accidents. Around 21% of all mishaps are due to fatigue. Driver drowsiness detection system is a car safety technology, assists in prevention of accidents caused by the drowsiness of drivers. Various studies have suggested that around one-fifth of all road's related accidents are caused due to fatigue, and up to half of them on certain roads. According to 2018's world road safety status report, the WHO has spotted road traffic accidents as a concerning source of death and injury. Because of road traffic accidents annually, around 1.35 million people die, and millions further handicapped. Figure 1 shows road deaths between 2001 and 2013.

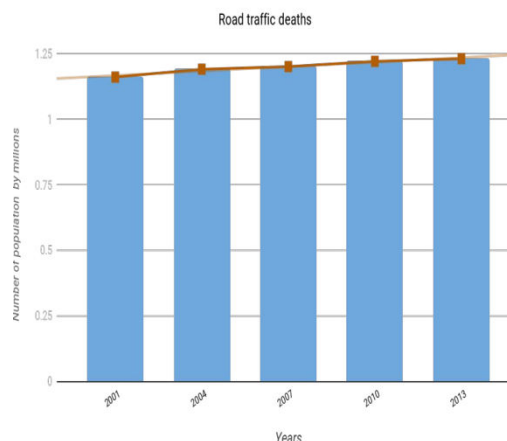


Fig. 1. Road traffic deaths between 2001 and 2013

Some of the key points to be noted here are:

- Sleep deprivation or disruption causes drowsiness during times when a person should be fully awake.
- If a person can't sleep properly for a single night that can cause high short-term sleepiness, and a pattern of disrupted sleep can lead to chronic sleepiness.
- If a person sleeps less than four hours also that leads to the individual driving performance
- Numerous factors, some of which are beyond a person's control and others of which are the result of personal decisions, contribute to sleep loss and sleep disruption. Long job hours, adjusting to new shifts, caring for one's family, participating in social activities, and illnesses including stress and sleep disorders are a few examples.
- Sleepiness slows reaction time
- Being sleepy reduces vigilance, alertness, and concentration, which makes it difficult to complete tasks that require attention.

Multiple eyelid movement features to deal with strong collinear relations between eyelid movement features and this is based on partial least square regression. predictive accuracy and resilience are validated, combining many parameters for improving capacity to recognize and anticipate the state of sleepiness [1]. Large dataset of 90 hours of real road drives have considered for detecting the fatigue using eye tracking system which is based on camera in vehicle. The eye measurements like blinking, opening, closing is assessed later statistically [2]. Changes in illumination and driver position problems are addressed by robust eye- detection- tracking algorithm. Six measures are calculated those measures are closing of eyelid, maximum closure time, frequency of blinking eyes, average of opened eyes, velocity of opened and closed eyes. Then measurements are integrated with fisher's linear discriminated functions in order to lessen the correlations and to obtain an independent index using a stepwise method. Driving simulator experiments demonstrated by six participants. Video- based drowsiness recognition method provides accuracy of 86% [3]. Hidden Markov Model (HMM) method has used for analyzing facial expressions which is based on dynamic modeling to determine the drowsiness [4]. Electrooculography, Electrocardiogram, Electroencephalogram are used to acquire the data from the different parts of the body through sensors. Theta, delta, and alpha signals are three important signals used to gauge fatigue of the driver. When a driver is sleeping, first two signals spike, while the alpha signals marginally increase. This technique has comparatively high accuracy return rates, of over 90%. The disadvantage in these techniques is its high intrusiveness. It may be uncomfortable for the driver to have numerous sensors attached to their body. Non-intrusive methods, on the other hand, tend to be less accurate. And then we have another class of techniques to extract the facial features is computer vision, this method makes use of actions including eye closure, yawning duration, head motions, and gaze or facial expression [5]. The Euclidean distance between the eyelids is used to measure three stages of drowsiness. This computation takes blinks per minute into account as the driver's drowsiness increases number of blinks per minute also increases [6]. The extraction of the region of interest from facial features was done using a multitask cascaded convolutional network. To identify the states of the eye and mouth movements from the ROI of photos, EM convolution neural network has been introduced. Degree of mouth opening, and percentage of eyelid closure are used as indicators of drowsiness, and the proposed model has a 93.643 accuracy rate [7]. CNN has used to identify the objects like opened and closed eyes from real time video streams. In the mobile net CNN model single shot multibox detector technology is used for the detection of objects. The model developed does not require expensive hardware [8]. To measure a driver's state of drowsiness, wearable glasses are incorporated with eye blink and heart rate sensors are used. The device features a system for messaging family members or friends about the location of a sleepy driver. This can recognize three levels of tiredness and respond accordingly by buzzing the driver to awaken them. When a high level of tiredness is identified, ignition lock will be activated. As a result, when a harmful degree of drowsiness is identified, the vehicle will be slowed down and eventually stopped as a safety measure [9]. CNN has been used to achieve the better accuracy and fastness in drowsiness detection. For eye status classification three different pretrained networks are used those are fully Designed NN, VGG16 and VGG 19 and to extract the features and these networks have given 98.15%,95.45%, and 94.46% accuracy respectively.[10]

Among many concepts that play a factor in the process of real- time driver drowsiness detection, some include eye tracking, eye detection, face detection, facial landmark detection. These concepts come into tangent at almost any integral process in image processing, computer vision, machine learning, contextually. Dlib is an integral toolkit used in the aforementioned related domains. Dlib is a framework for creating practical C++ machine learning and data analysis applications, according to its github page. While the library is originally written in C++, it has portable, scalable, implementable Python bindings.

The frontal face detector works well with this Dlib toolkit, due to its simplicity. Detector is based on histogram of



oriented gradients and linear SVM. The HOG, SVM frontal face detector is mainly used for frontal face detector applications, whereas the face detector-based CNN can recognize faces from any angle. CNN has the flaw of not being ideal for real-time video. It is intended for use with a graphics processing unit (GPU). Proposed system uses a frontal face detector leaning towards the HOG, linear SVM, as to run a CNN face detector, extra hardware requirements are needed i.e., a powerful Nvidia GPU or its equivalent. When building scripts for the CNN approach, two command-line inputs are needed: input picture and model weights. In contrast, there is no file that must be supplied to initialize the HOG based one. Since it has already been constructed inside of Dlib, calling the function should be sufficient. The only distinction is that we must supply the weights file to initialize with for the CNN based one. Detection of facial landmarks is a subset of the shape prediction problem. Given an image and normally a region of interest (ROI) that specifies an object of interest, a shape predictor attempts to localize key points of interest along the shape. There are two steps that can be taken to identify face landmarks: The first stage is to identify the face in the image, and the second is to identify the important facial structures on the region of the face that is of interest. A pre-trained HOG + Linear SVM object detector, deep learning techniques, or OpenCV's built-in Haar cascades can all be used to recognize faces in the first stage. Any of the aforementioned methods must be used to extract the face bounding box, or the x, y coordinates of the face in the image. There are two steps that can be taken to identify face landmarks: The first stage is to identify face in an image, and second step is to identify an important facial structure of the face that is of interest. HOG, Linear SVM, deep learning techniques, or OpenCV's built-in Haar cascades are used to recognize faces in the first stage. Any of the methods must be used to extract the face bounding box, or the x, y coordinates of the face in the image. Step 2 can then be applied after we have the face region. Methods used by facial landmark detectors primarily aim to identify and localize the following facial regions are mouth, eyebrows, jaw, eyes, and nose. One millisecond face alignment with an ensemble of regression trees of two steps is used.

Step 1: Define facial landmarks on an image for training and label images manually, specify particular (x, y) coordinates of a particular region that surrounds each facial structure.

Step 2: The probability of the distance between input pixel pairs is calculated. In this technique, feature extraction is performed statically by training a group of regression trees to approximate the positions of the face landmarks from the pixel intensities [11]. Estimating the locations of 68 (x, y)-coordinates that correspond to facial structure points on the face is done using the pre-trained facial landmark detector present in the dlib library. A simple illustration for the indexes of the 68 coordinates (from the iBUG 300-W dataset) is given in figure 2.

Histogram of oriented gradients (HOG) is used for feature extraction in the human detection process, while linear support vector machine (SVM) is used for the purpose of classification. To create an object detector, a HOG feature descriptor combines fine-scale gradient computation methods from the field of computer vision with the linear support vector machine. When it comes to image recognition, the gradient intensities of a picture can provide some useful local information. Pixel intensities and gradient intensities play a key role in defining an image.

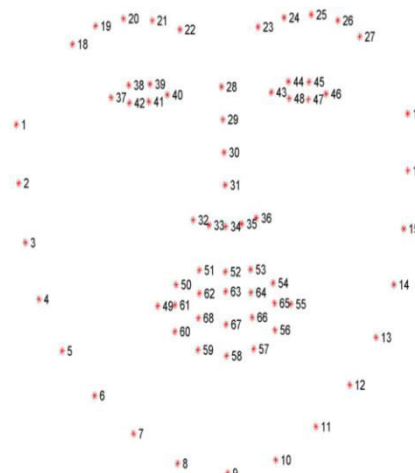


Fig. 2. 68 facial landmark coordinates

A feature descriptor tries to extract the essential details from an image and keeps unnecessary information behind a hidden level of abstraction. Then the necessary information plays its role in image recognition and object detection. So basically, face detection works by computing HOG features and then classifying them with SVM.

Therefore, an effective frontal facial landmark detector-based technique for detecting eye blinks is proposed. SVM classifier is used to classify 2 types of patterns one blinking and second one is non-blinking. Everybody blinks in a slightly unique pattern. The pattern varies in blink duration, degree of eye squeezing, and speed of closing and opening. The eye blinks for between 100 and 400 milliseconds. The technique described in this paper uses a facial landmark detector to locate the contours of the eyes and eyelids. We determine eye aspect ratio (EAR), which is utilized as approximation of eye-opening condition, from landmarks found in the image illustrated in figure 3. The per-frame EAR might not be able to consistently discriminate eye blinks, thus a classifier considers a larger temporal window is developed.

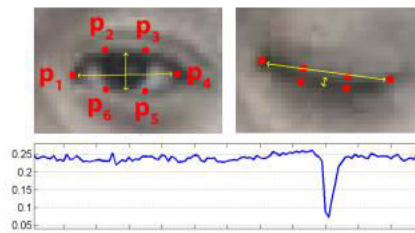


Fig. 3. Open and closed eyes with landmarks p_i .

For each frame processed, its corresponding eye landmarks are detected. The eye aspect ratio (EAR) between height and width of the eye is computed.

$$EAR = (\|p2-p6\| + \|p3-p5\|) / (2 * \|p1-p4\|) \dots \tag{1}$$

where $p_1, p_2, p_3, p_4, p_5, p_6$ are the two-dimensional landmarks, depicted in figure 3. When the eyes are open, EAR is largely continuously constant simultaneously, the EAR is averaged when both eyes are considered. Additionally, it does not suggest that a person is blinking if the EAR has a low value. If a person closes his or her eyes for a long period of time, makes a facial expression, such as yawning, or the EAR records a brief fluctuation of landmarks, a low value in the EAR may also result. Thus, a classifier that takes a larger temporal window of a frame as input is proposed. This is implemented by the eye aspect ratio support vector machine which is trained from manually annotated sequences. Negative instances are drawn from areas of the dataset where no blinks occur, and positive examples are gathered as ground-truth blinks. A scanning-window approach is used when running a classifier. Except for the first and last frames of the input sequence, each frame is computed and classified using a 13-dimensional feature by this EAR SVM. The left and right eye landmarks are identified for each frame of the video. The eye aspect ratio (EAR) is calculated using the eye's height and width. EAR values from a frame's longer temporal window are given to SVM classifier to classify the blinking and, as the eye begins to close, it approaches zero. Considering that both eyes blink non-blinking patterns. Accuracy of the landmark detection is determined by average relative landmark localization error for profile image, which is defined by

$$\epsilon = \frac{100}{kN} \sum_{i=1}^N \|x_i - \hat{x}_i\|^2 \tag{2}$$

where

x_i - location of landmark i in the image

\hat{x}_i - approximate landmark location by a detector

N - number of landmarks

normalization factor k - Euclidean distance between eye centers in input image. ϵ is calculated for 12 landmarks of the eyes only. Thus, this proposed method makes use of the EAR thresholding, which is usage as a single image classifier to detect.

II. IMPLEMENTATION

The implemented algorithm uses EAR thresholding to establish conditions and decision statements to detect driver drowsiness.

Step 1: Define the eye aspect ratio function



- Step 2: Compute the Euclidean distance between vertical landmark coordinates and horizontal landmark coordinates
- Step 3: Define threshold values
- Step 4: Define the face detector and shape predictor functions
- Step 5: Perform preprocessing of dataset and image resizing
- Step 6: Perform RGB to Grayscale channel conversion
- Step 7: Convert the landmarks list to numpy array
- Step 8: Calculate the left eye's eye aspect ratio.
- Step 9: Calculate the right eye's eye aspect ratio.
- Step 10: Compute an eye aspect ratio, is an average of left EAR and right EAR
- Step 11: Invoke cv2 to perform convexHull methods on the left and right eyes
- Step 12: Invoke cv2 to perform draw Contours methods on the plotted convex hulls of left and right eyes.
- Step 13: Compare resultant EAR with the threshold value and consequently increment flag value, if flag value \geq frame_check then issue alert
- Step 14: After frame display, press 'q' to invoke destroyAllWindows method.

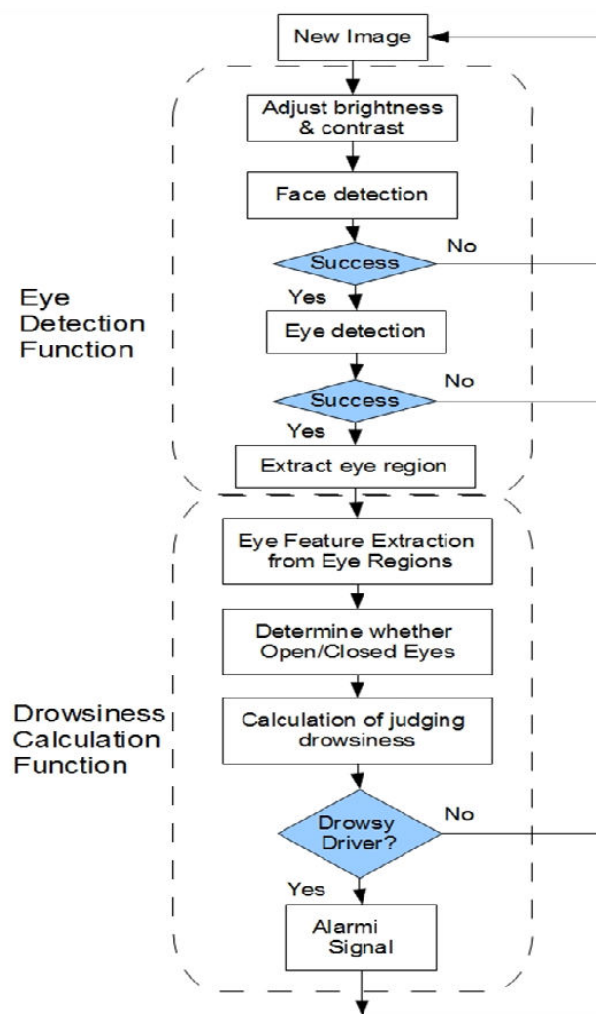


Fig. 4. Proposed System Design



Values Assignment:

Some of the variables that define with values include:

Thresh: When EAR thresholding, define a specific value as a boundary condition to determine if the person is drowsy or not. If resultant eye aspect ratio value is lesser than thresh value, it indicates that driver has an abnormal blink rate and shows sign of corresponding fatigue. It is set to the decimal value 0.20 by default.

Frame Check: This variable's significance is that if a person blinks, an ear value is computed and every time that it is less than thresh value, flag counter gets incremented by one. When the flag counter value is greater than frame check, indicates that the person has been blinking eyes for more than set consecutive frames. Here, the value of frame_check is set to 20 by default.

Cap: This variable sets up the primary webcam on system. If there are more sources of video capturing devices, the values may fluctuate.

Flag: This variable tracks the total number of subsequent frames with eye aspect ratios that are below the specified "thresh hold" value. By default, 0 is allocated to it.

Width: This variable aims to maintain the original aspect ratio of the image when evaluating the width or height argument.

Key: An input string of length 1 is passed to key. During comparison, if the value of key matches the pressed button on the keyboard, then an event occurrence is called. Here, upon pressing 'q', the camera and clean up the frame window are released.

Primary Methods:

The proposed system uses two primary methods for the purpose of detection of both face and facial landmark. First method is Shape Predictor which outputs a series of point positions to describe the pose of an object from an image region containing an object. Image of person face can be taken as an input and expected to locate mouth, eyes corners, tip of nose as key facial characteristics. Second method is Frontal Face Detector that retrieves the face data. Each face is an object that contains the points where the image can be found. Frontal Face Detector basically loads the detector into an acceptance state.

EAR definition:

Here a function needs to be defined that can calculate the eye aspect ratio of an eye. This is done firstly by acquiring the six landmark coordinates of the eye, p0 and p3 denoting the horizontal landmark coordinates, p1, p2, p4 and p5 denoting the vertical landmark coordinates. Euclidean distance between the first vertical and the last vertical landmark coordinate is taken, the Euclidean distance between the second vertical and the second last vertical landmark coordinate is taken, Euclidean distance between the two horizontal coordinates is taken, and consequently eye aspect ratio is computed according to equation 2.1. Eye aspect ratio is evaluated individually for both eyes and average of both is considered as the resultant eye aspect ratio. RGB to Grayscale channel conversion Here RGB to its equivalent Grayscale channel conversion takes place since a Grayscale image is an one-layer image, intensity ranges from 0-255. RGB has three channels, each containing range values for Red, Green and Blue, and they all need to be considered together. Because of the intrinsic complexity of grey level images is lesser than that of colour images, tool like image processing using grey level images been introduced. Here covert colour method to be used converts an image from one colour space to another colour space.

NumPy array and EAR Method:

In this stage, NumPy python library is used for numerical calculations. Here object shape is duplicated and the converted array replaces the original shape object which consisted of a list of landmark coordinates of the face. For the EAR method, initially the landmark coordinates for the left eye and right is passed from the shape. Then eye aspect ratio is invoked to find out EAR for the left eye, after which eye aspect ratio is invoked again to find out EAR for right eye. When the EAR's for both the eyes are computed, then their average is determined to be the resultant EAR.

III. RESULTS

There are two figures given below which illustrate the working of this proposed system. In Figure 5 the person has the green contour curves placed on his eyes, indicating a working image processing algorithm. In Figure 6 the alarm event takes place after identifying that the appropriate conditions for drowsiness have been satisfied.

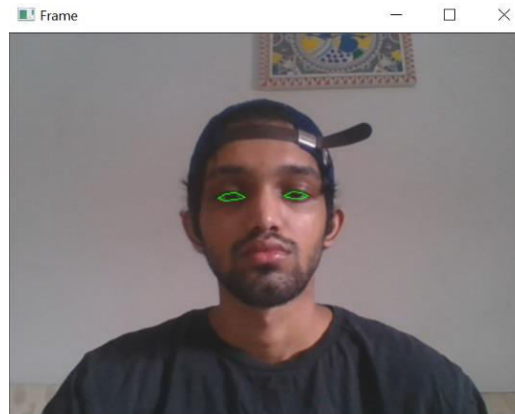


Fig. 5. Detection of the RoI

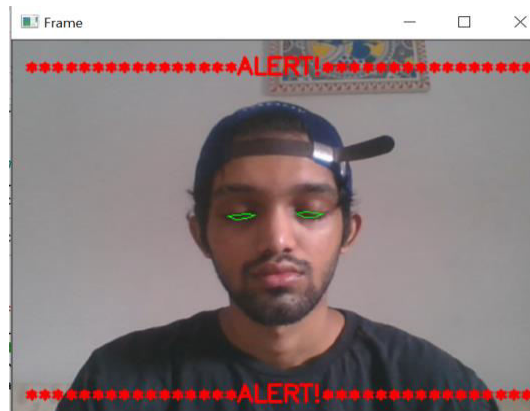


Fig. 6. Alarm event occurrence

The best performance results from the implementation of dlib regression trees. The ‘**normalized mean error**’ for the dlib facial detector starts increasing steadily and stabilizes after 3%. The landmark coordinates detected are compared with the ground-truth values of dlib 68 facial landmark coordinates dataset. If detected landmark values are denoted as ‘y’ and the ground-truth values are denoted as ‘y1’, the square of the differences, squared and summed over 1/N times, where N is the number of landmarks.

IV. CONCLUSION

Proposed system evaluates the driver’s drowsiness level based on frontal face detector method using HOG and linear SVM. System that can instantly spot signs of driver fatigue in a live video stream and sound an alarm if the driver seems sleepy. If the Eye Aspect ratio is less than 0.20 in 20 consecutive frames, an alert is triggered. It can be challenging to determine whether driver sleepiness is actually a contributing factor and to gauge their level of fatigue, currently not feasible to determine precise count of accidents that occur which are related to sleep. However, numerous studies have concluded that in monotonous roads, that probability of accidents that are related to drowsiness, is nearly one-fifth. So, the conclusion is that drowsy driving is a significant issue contributing to the increase in traffic-related accidents and road-related infrastructural damages .

REFERENCES

- [1] Fatigur, D. R. I. V. E. R. (2001). Driver fatigue and road accidents: a literature review and position paper. Edgbaston: The Royal Society for the Prevention of Accidents.
- [2] Su, H., & Zheng, G. (2008). “A partial least squares regression-based fusion model for predicting the trend in drowsiness”. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 38(5), 1085-1092.



- [3] Krajewski, J., Sommer, D., Trutschel, U., Edwards, D., & Golz, M. (2009, June). Steering wheel behavior based estimation of fatigue. In Driving Assessment Conference (Vol. 5, No. 2009). University of Iowa.
- [4] Friedrichs, F., & Yang, B. (2010, June). "Camera-based drowsiness reference for driver state classification under real driving conditions". In 2010 IEEE Intelligent Vehicles Symposium (pp. 101-106). IEEE.
- [5] Mardi, Z., Ashtiani, S. N. M., & Mikaili, M. (2011). EEG-based drowsiness detection for safe driving using chaotic features and statistical tests. *Journal of Medical Signals & Sensors*, 1(2), 130.
- [6] Zhang, W., Cheng, B., & Lin, Y. (2012). "Driver drowsiness recognition based on computer vision technology". *Tsinghua Science and Technology*, 17(3), 354-362.
- [7] Tadesse, E., Sheng, W., & Liu, M. (2014, May). Driver drowsiness detection through HMM based dynamic modeling. In 2014 IEEE International conference on robotics and automation (ICRA) (pp. 4003-4008). IEEE.
- [8] Maryam Hashemi,Alireza Mirrashid,Aliasghar Beheshti Shirazi1.(2020),Driver Safety Development: Real-Time Driver Drowsiness Detection System Based on Convolutional Neural Network., *SN Computer Science*
- [9] Zuopeng Zhao,Nana Zhou ,Lan Zhang, Hualin Yan,Yi Xu,2 and Zhongxin Zhang.(2020) Driver Fatigue Detection Based on Convolutional Neural Networks Using EM-CNN. *Hindawi*.
- [10] Md. Tanvir Ahammed Dipu, Syeda Sumbul Hossain, Yeasir Arafat, Fatama Binta Rafiq. (2021). Real-time Driver Drowsiness Detection



INNO SPACE
SJIF Scientific Journal Impact Factor
Impact Factor: 8.423



ISSN INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN SCIENCE | ENGINEERING | TECHNOLOGY

9940 572 462 **6381 907 438** **ijirset@gmail.com**



www.ijirset.com

Scan to save the contact details