Prevention from Road Accidents by Detecting Driver Drowsiness

S.Priyadharsini
Department of Computer Science and Engineering, Mepco Schlenk Engineering College, Sivakasi, India.
****Corresponding Author***
E--mail Id:- *priyadharsini@mepcoeng.ac.in*

ABSTRACT
Driver lethargy is one of the main explanations for traffic accidents and the associated fiscal losses. Existing drowsiness detection techniques does not concentrate on all the key factors of drowsy drivers. The proposed system designed for the analysis and detection of drowsiness uses visual based features. The eye state, eye blinking frequency, eye closure duration, redness level detection, mouth state, yawning frequency are the key factors for detecting drowsiness. Systems that use this technique usually monitor eye states and the position of the iris through a specific time period to estimate the eye blinking frequency and the eye closure duration. On the other hand, mouth analysis and tracking the yawning frequency of a driver is an alternative way of detecting the drowsy driver. These techniques will identify the drowsing state of the driver and if he is drowsy, then an alert message is sent to the driver stating that the driver is no longer capable of driving the vehicle safely thus preventing accidents.

Keywords:- *Drowsiness, Accident Prevention, Eye Closure State, Blinking Frequency, Eye Closure Duration, Eye Redness Level, Mouth State, Yawning Frequency*

INTRODUCTION
Driver weariness is a colossal traffic wellbeing issue and is broadly accepted to be one of the biggest supporters of fatalities and extreme wounds in rush hour gridlock today, either as an immediate reason for nodding off at the worst possible time or as a contributing variable in bringing down the consideration and response season of a driver in basic circumstances. Mishaps with business weighty vehicles are risky as well as exorbitant and the countering of driver weakness is profoundly significant for development of street security. It is a critical consider an enormous number of vehicle mishaps. Late measurements gauge that yearly 1,200 passings and 76,000 wounds can be ascribed to weakness related crashes.

Driver tiredness discovery framework is a vehicle wellbeing innovation which forestalls mishaps brought about by the driver getting sleepy. The point of this paper is to foster an application for the sluggishness location framework. The spotlight will be put on planning a framework that will precisely screen the open or shut condition of the driver's eyes continuously. By observing the eyes, it is accepted that the side effects of driver exhaustion can be recognized sufficiently early to stay away from an auto crash. Detection of fatigue involves the observation of eye movements, redness detection, and yawning state detection in a sequence of images of a face.

LITERATURE SURVEY
This section focuses on the study of the eye blinking rate, yawning state of mouth and redness level that are used for the detection of the drowsiness level of the driver. Mario I.Chacon [2] discussed about Drowsiness Fatigue Detection

(DFD) System. The main function of DFD is to screen the driver's form and take action accordingly. This algorithm is based on the geometric methods as they establish very particular traces from iris. Eyelid closure metrics - PERCLOS (Percentage of Eye Closure) and AVECLOS (Average Eye Closure) are used. But this kind of DFD systems is more complex and expensive, since it uses both visual and non-visual systems.
Sinan Kaplan [1] provided a comprehensive insight into the well-established techniques for driver inattention monitoring in their paper. In this literature, eye state analysis mostly exploits the PERCLOS value as drowsiness metric, which shows the percentage of time in a minute that the eyes are 80% closed. If the driver is tired, eye closure duration will increase and the value of PERCLOS is higher than awake periods of driver. However, there are some limitations in extracting those visual features.

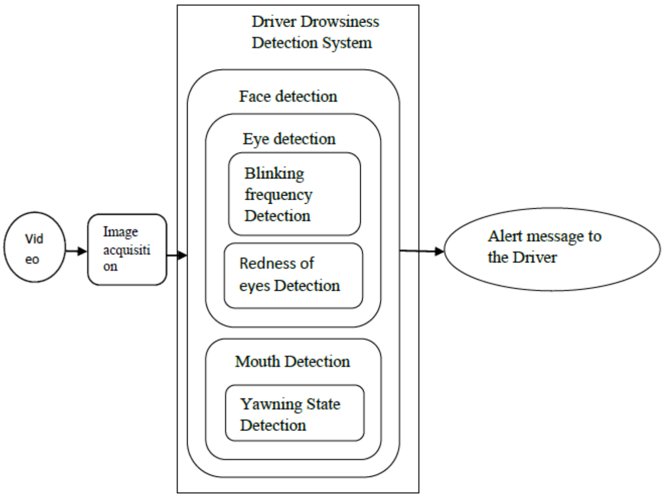


Fig 1: *Driver Drowsiness Detection System*

Azim, 2009 detected fatigue state by analysing the mouth and the eye state. The face is tracked using the Kalman filter. It guesses the position and indecision of the moving objects in the next time frame. The system fails to detect the face properly when the head posture changes.

Mandeep Singh [3] discussed a method that detects the drowsiness in eyes using the mean sift algorithm. This approach analyses the images to detect physical changes of drivers, such as eyelid movement. But it is difficult to give the eye closure percentage value for all the people.
The aim of the system developed by Mehrdad, 2012 [4] is to trace, track and analyse both the driver's face and eyes to compute a drowsiness index to prevent accidents. Both face and eye detection is performed by Haar-like features and AdaBoost classifiers. The face detection process is done using Viola and Jones method. The eye is located using the AdaBoost algorithm. The eye state is analyzed using the SVM training set. One drawback to this object tracking method in face tracking process is the size of surrounding rectangle is not adaptive.

Thus the system study reveals that the earlier techniques don’t concentrate on all the key factors for drowsiness detection. In the proposed work, Viola-Jones, 2001 algorithm is used to detect the face region and extract the eye and mouth region separately. The Binarization technique changes over the dark scale picture into highly contrasting picture. The consequence of Optical Character Recognition (OCR) profoundly relies on the binarization. Filtering technique is used for modifying or enhancing an image. The color histogram provides a compact summarization of the distribution of pixel values in an image. The red pixel value is estimated in the detected eye region. The red pixel value in the iris region is calculated. Based on the red pixel value a threshold value is set. If the detected eye region has red pixel value exceeding the threshold value, then it detects it as a drowsy condition for the driver and generates an alarm.

Proposed Methodology

Fig. 1 shows the block diagram of Driver Drowsiness Detection System.

Face Detection

Face acknowledgment is a functioning yet testing point in PC vision applications. Albeit the facial pictures have a high dimensionality, they normally lie on a lower layered subspaces or sub-manifolds. Here Viola-Jones calculation is utilized to independently identify the face area and concentrate the eye and mouth locale. In Viola-Jones framework, a straightforward Haar highlight is utilized. Viola and Jones pursue note that decision of elements is significant because of the advantage of specially appointed space encoding. If there should be an occurrence of face location this is especially significant. Elements can be utilized to address both the measurably close facial data and scantily related foundation information in an example picture. In its most straightforward structure the elements can be considered pixel power set assessments. This is where the amount of the luminance of the pixels in the white locale of the component is deducted from the amount of the luminance in the excess dim segment.

This distinction esteem is utilized as the element esteem, and can be consolidated to shape a frail speculation on districts of the picture. Inside the execution four of the Haar-like elements are picked, the first with flat division, the second an upward, the third containing two vertical divisions and the last containing both the level and vertical division. The features are called Haar like because of their resemblance to Haar-basis functions. Figure 2 depicts the features of Haar basis functions.

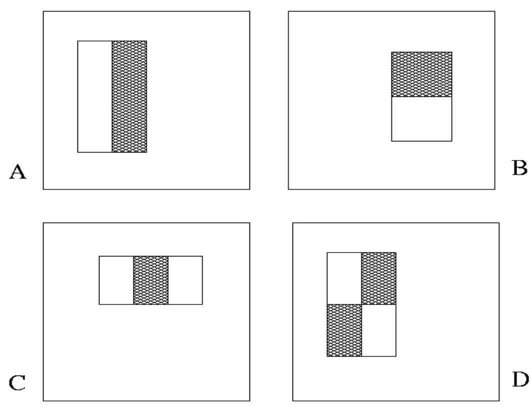


Fig.2:- *A,B Two-rectangle feature: difference between the sum of the pixels within two rectangular regions. C. Three-rectangle feature: sum within two outside rectangles subtracted from the sum in a center rectangle. D. Four-rectangle feature: difference between diagonal pairs of rectangles.*

To find success a face identification calculation should have two key elements, exactness and speed. There is for the most part a compromise between the two.

Using another picture portrayal, named vital pictures, Viola and Jones depict a method for quick element assessment, and this ends up being a compelling means to accelerate the characterization errand of the framework. In this framework a variation of AdaBoost is utilized both to choose the elements and to prepare the classifier. From the detected face eyes and mouth region are extracted.

Binarization.
The Binarization Method changes over the dim scale picture (0 up to 256 dim levels) into highly contrasting picture (0 or 1). The consequence of Optical Character Recognition (OCR) exceptionally relies on the binarization. The great binarized picture can give more exactness in character acknowledgment as thought about unique picture since clamor is available in the first picture.

As a matter of fact issue is what binarization calculation is proper for all pictures. The choice of most ideal binarization calculation is troublesome, in light of the fact that different binarization calculation gives different execution on various informational collections. This is particularly obvious on account of authentic reports pictures with variety conversely, and light.

The Binarization algorithms can be divided into two categories.

a) Global Binarization
b) Local Binarization.

The worldwide binarization devices used single verge value for whole image and the local binarization method where the threshold value calculated locally pixel by pixel or region by region.

Filtering
Pictures are in many cases debased by irregular varieties in force, brightening, or have unfortunate differentiation and can't be utilized straightforwardly. Separating change pixel force values to uncover specific picture attributes. Middle sifting is broadly utilized in advanced picture handling on the grounds that, under specific circumstances, it jam edges while eliminating commotion. Middle separating is a nonlinear technique used to eliminate clamor from pictures. It is generally utilized as it is exceptionally viable at eliminating clamor while saving edges. It is especially successful at eliminating 'salt and pepper' type commotion. The middle channel works by traveling through the picture pixel by pixel, supplanting each worth with the middle benefit of adjoining pixels. The example of neighbors is known as the "window", which slides, pixel by pixel over the whole picture pixel, over the whole picture. The middle is determined by first arranging all the pixel values from the window into mathematical request, and afterward supplanting the pixel being considered with the middle(median) pixel esteem.

Identifying the eye closure state
If the eye is closed, the number of black pixels in the eye region is increased after binarization. The black pixel level is estimated in the detected filtered eye region. A threshold value is set. If the detected eye region exceeds the threshold value for more than 3consecutive images, then it is considered as a drowsy condition and the alarm is generated.

Algorithm for identifying the blink rate of the eyes is given below which is based on Matlab implementation.

```
Algorithm BlinkRate(face)    {
Input: -Input face
Output: Detect the closed state of eyes
for each frame
EyeDetect = vision.CascadeObjectDetector('EyePairBig');
//The CascadeObjectDetector creates a System object detector that detects //objects using the Viola-Jones algorithm. EyePairBig is a valid model //character vector to detect the eye.
BB=step(EyeDetect,face);
//The EyeDetector detects eyes from all the frames in the video. The //boundary box BB is used to annotate the eye region for all the frames.
eye=imcrop(face,BB(i,:));
//The detected eye region is cropped using the imcrop function.
eye_gray=rgb2gray(eye);
//The detected eye is binarized using the rgb2_gray.
data=imresize(eye_gray,[112,112]);
//The image is resized in n*n format.
diff_in = medfilt2(data, [3 3]);
//Then median filter is applied in the binarized images.
The size of the matrix value is 3*3
eye_bw = im2bw(diff_in,0.17);
//The images are filtered and converted into black and white images to //detect the pupils region.

  If Black_Pixel_count is less than the Threshold value
  // Threshold value is set as 20
  Flag=Flag+1;   //Increment the Flag Value
  Else
  Flag = 0;    //Reset the Flag value to 0
  If Flag >= 3  //consider it as closed state eyes
  sound(y,Fs);   //An alert message sounded through an audio clipping.
}
```

Detecting the Redness of Eyes

Here the eye region is detected, cropped separately for each frame and color histogram is applied. Pink eyes refers the redness, swelling of membrane that lines the eyelid. Redness of eyes is detected in the outermost layer of the pupil and inner surface of eye lids.

The red pixels are identified by using color histogram technique. Color histogram characterizes the number of pixels that have colors in each of a fixed list of color ranges. It determines the brightness distribution of each individual Red/Green/Blue channel. By setting the threshold value, the redness level is determined.

The red pixel value is estimated in the detected eye region. The red pixel value in the iris region is calculated. Based on the red pixel value a threshold value is set. If the detected eye region has red pixel value exceeding the threshold value, then it detects it as a drowsy condition for the driver and generates an alarm. Figure 3 depicts the flowchart for calculating the redness level of eyes.

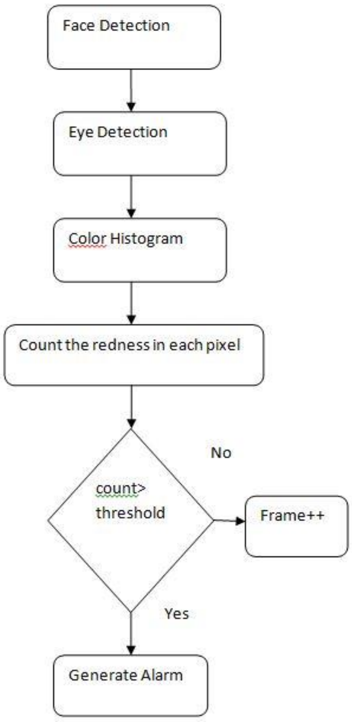


Fig.3:-*Flow Chart for detecting redness of eyes*

Algorithm for detecting redness level in the eyes is given below:
Algorithm Redness_detection(eye) {
Input: Detected eye region
Output: Redness of eye
for each detected eye region
redChannel = eye(:,:, 1); *//Detect the redness count for each pixel*
total_redness = sum(sum(redChannel));
//Sum the redness count for all the pixels.
If total_redness exceeds Threshold
//Threshold value is set to 15,00,000
Display the alert Message
}

Yawning State Detection

Yawning is a compulsory admission of breath through a totally open mouth; as a rule set off by exhaustion or weariness. This method is additionally one of the non-nosy strategies for identifying driver sluggishness by applying PC vision. Distinguishing tiredness includes two fundamental stages to break down the progressions in looks appropriately that suggest sluggishness. In the first place, the driver's face is recognized by utilizing overflow classifiers and followed in the series of edge shots taken by the camera. In the wake of finding the driver's face, the following stage is to recognize and follow the area of the mouth. For mouth identification the specialists have utilized the face recognition calculation proposed by Paul Viola and Michael J. Jones [2]. A while later, yawning has been dissected to decide the level of the sleepiness. This is ventured to be displayed with a huge vertical mouth opening and changes in the driver's mouth form. Mouth opens wide and the distance between its counters gets huger. The following is the algorithm for yawning state detection:

```
Algorithm YawningState_Detectio(face){
       Input: Input face
      Output: Detected yawning state of mouth
for each frame
        mouth_c=imcrop(fac,[30 80 40 50]);
    //The mouth region is cropped from the face by setting coordinates.
        mouth_gray=rgb2gray(mouth_c);
   //The detected mouth is binarized using the rgb2_gray.
       data=imresize(mouth_gray,[112,112]);
   //The image is resized in n*n format.
      diff_in = medfilt2(data, [3 3]);
  //Then 3*3 median filter is applied in the binarized images
     eye_bw = im2bw(diff_in,0.17);
 //The images are filtered and converted into black and white
    If Black_Pixel_count exceeds the Threshold value
// Threshold value is set as 7000
           Flag=Flag+1; //Increment the Flag Value
    Else
           Flag = 0; //Reset the Flag value to 0
    If Flag >= 3 //consider it as yawning state
          sound(y, Fs); //An alert message
}
```

Binarising the Detected Mouth Region

The Binarization Technique converts the grey scale image (0 up to 256 gray levels) into black and white image (0 or 1). The aftereffect of OCR exceptionally relies on the binarization. The top notch binarized picture can give more exactness in character acknowledgment as thought about unique picture since commotion is available in the first picture. Issue is what binarization calculation is suitable for, truth be told all the detected mouth regions.

Median Filter for binarised mouth region

Middle sifting is a nonlinear strategy used to eliminate commotion from pictures. It is generally utilized as it is exceptionally powerful at eliminating clamor while saving edges. It is especially compelling at eliminating 'salt and pepper' type commotion. The middle channel works by traveling through the picture pixel by pixel, supplanting each worth with the middle benefit of adjoining pixels. The example of neighbors is known as the "window", which slides, pixel by pixel over the whole picture pixel, over the whole picture. The middle is determined by first arranging all the pixel values from the window into mathematical request, and afterward supplanting the pixel being considered with the center (middle) pixel esteem. By this technique, the noisy areas are converted into black and the other regions are converted into white.

Calculation of black pixels in the filtered mouth region

The number of black pixels is estimated in the detected filtered mouth region. A threshold value is set. If the black pixel count in the detected mouth region exceeds the threshold value for more than 3 consecutive images, then it is considered as a drowsy condition and the alarm is generated. Figure 4 depicts the flowchart for the identification of the yawning rate of the mouth through various steps such as the detection of the mouth region from the face region, cropping the mouth region separately, binarizing the cropped mouth into grayscale and finally applying the median filter to convert the noisy area into black and other regions into white.

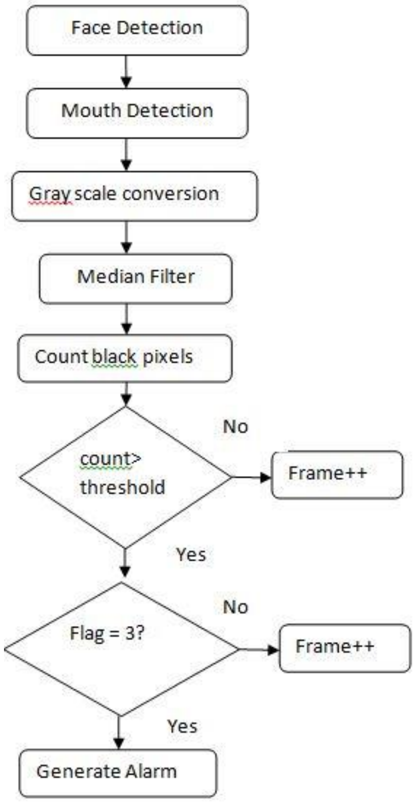


***Fig.4:-** Flow Chart for calculating black pixels in the mouth region*

Implementation Results
The results obtained by applying the algorithms discussed in pervious sections are shown for each modules. Implementation is done using Matlab Tool. We collected our own dataset for working with the proposed work.

Image Acquisition

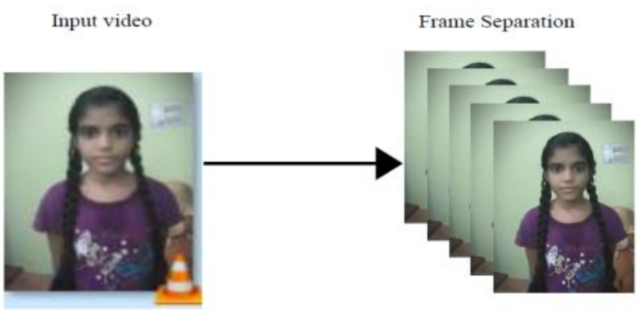


***Fig.5:-**Frame separation from video file*

Figure 5 depicts the frame separation from the video file. The video file is read using the *videoReader()* method. The frame rate is taken as one frame per second. Each and every frame is processed. Thus the sequence of images is separated from the video and processed.

Face Detection

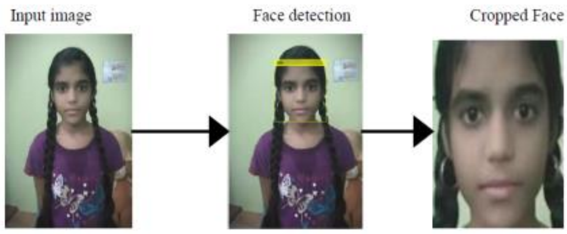


***Fig.6:**-Face detection from input image*

Figure 6 depicts the face detection from the given image. The face is detected using the *faceDetector()* object. The FrontalFaceCART parameter is used to detect the face. The face is detected using the coordinate points. The boundary box draws a rectangle line in the face region. The face is annotated and the face is detected. The detected face is cropped using *imcrop()* function. The face is resized using *imresize()* function and the images are fitted into the window.

Blinking Rate Detection

Eye Detection from the Face

To detect blinking rate of the eyes, the eye region is detected from the face. The blinking rate is normally 12-19/min. The frequency lesser than this range indicates the drowsiness state.

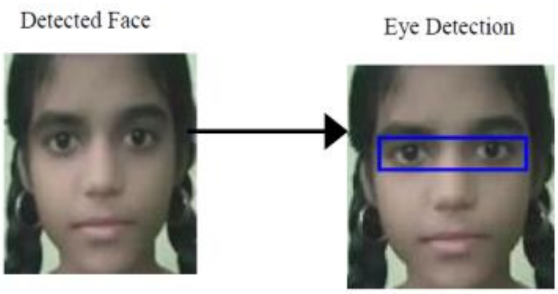


***Fig.7:**-Open eye detection from input face*

Figure 7 depicts open eye detection for the given image. The eye region is detected using the EyeDetect object. The cascading classifier 'EyePairBig' is used to detect the eye in pairs. The eye region is annotated.

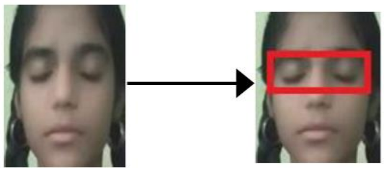


***Fig.8:**-Closed eye detection from input face*

Figure 8 depicts the closed eye detection from the given image. The eye region is detected using the *EyeDetect* object. The cascading classifier 'EyePairBig' is used to detect the eye in pairs. The eye region is annotated.

Cropping the Eye Region

Figure 9 shows the cropping of open and closed eye regions from the face. The eye region is cropped using the *imcrop()* function. The cropped image is then used for processing.

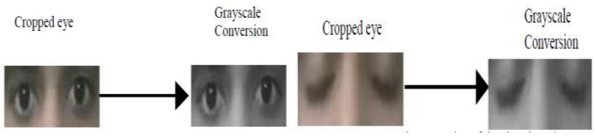
(a) (b)

Fig.9:- *Cropped eye region; (a) open eye state (b) closed eye state*

Binarizing Cropped Eye Region

Figure 10 shows the gray scale conversion for the given image. The gray scale conversion is applied using the *rgb2gray()* function.

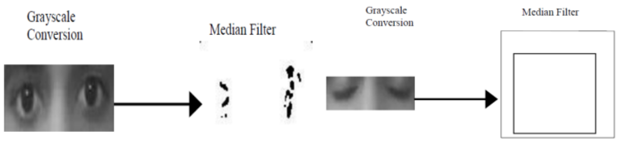
.



(a) (b)

Fig.10:-*Gray scale conversion; (a) open eye state (b) closed eye state*

Applying Median Filter



(a) (b)

Fig.11:- *Applying Median Filter; (a) open eye state (b) closed eye state*

Figure 11 (a) depicts applying median filtering technique for open eye state. Here the black pixel count is greater than 20. So it is correctly identified as open state. Figure 11 (b) depicts applying median filtering technique for closed eye state. Here the black pixel count in the detected image is lesser than 20. So it is considered as closed state of the eyes. If this condition occurs for three consecutive frames, then it is considered as drowsy condition and the alarm is sounded.

Detecting Redness of Eyes

The redness of the eyes are detected using the red pixels count in the iris region. Based on the red pixel count a threshold is set. Here the threshold is fixed to be 15, 00,000. Figure 12 shows redness detection. Here the red pixel count in the detected image is greater than 15 lakh pixels. Since it exceeds the threshold it is considered as one of the drowsy condition and the alarm is sounded.

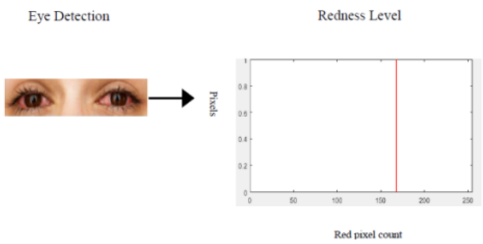


Fig.12:-*Redness level detection from the eye region*

Yawning State Detection

To detect the yawning state of the mouth, the mouth region is detected from the face same as that of eye region detection. The mouth region is detected using the *MouthDetect* object. The cascading classifier 'Mouth' is used to detect the mouth region. The mouth region is then annotated. The mouth region is also cropped using the *imcrop()* function. The cropped image is then used for processing. The gray scale conversion is applied using *rgb2gray()* function.

Figure 13 shows the effect of applying median filter to the detected open and closed mouth regions. In closed mouth state, the black pixel count is lesser than 7000. So it is considered as closed state of the mouth. In open mouth state the count of black pixel is greater than 7000. So it is considered as open state of the mouth. If this condition occurs for three consecutive frames, then it is considered as drowsy condition and the alarm is sounded.

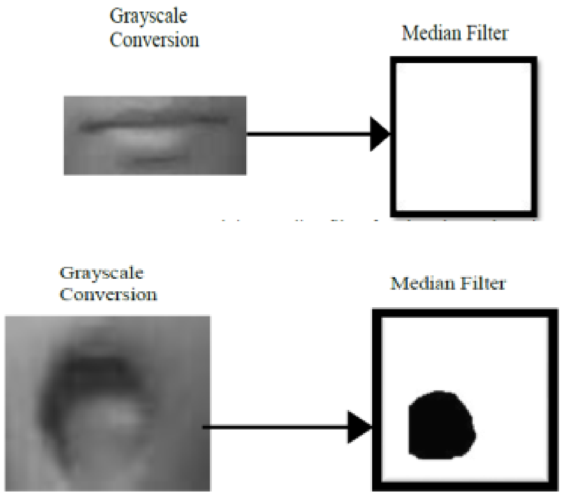


***Fig.13:-** Applying Median Filter; (a) open mouth state (b) closed mouth state*

Performance Comparison

We collected our own dataset from 100 students for blink rate detection and yawning detection. For redness detection we tested our proposed work with red eye images from Google. The proposed work is compared with results of [7,15] for yawning and eye closure detection which is shown in Table 1.

***Table 1:-** Performance comparison for yawning and eye closure detection.*

Work	Normal	Yawning
Results of Saradadevi, [7] 2008	86%	81%
Proposed Work	94%	95%
Work	Normal	Eye closure
Results of Ahmad, 2015	90%	90%
Proposed Work	96%	96%

CONCLUSION AND FUTURE WORK

Drowsiness state of the drivers can be detected using both the physical and non-physical factors. This paper concentrates on the physical factors such as eye blinking rate, yawning state of the eyes and the redness level of the eyes. The video is converted into sequence of images by the image acquisition method.

Viola-Jones algorithm is used to detect the face, eye and mouth region. The eye and mouth region are extracted using the cascading classifiers.

The blinking rate of the eyes is detected by applying the median filter. By the median filter technique, the pupil region is extracted. In the detected pupils, black pixels are calculated and a threshold is set. If the detected pupils has the black pixel value exceeding the threshold value for three consecutive frames an alert will be given to the driver. The alert is given by means of an audio clipping region stating that they are no longer capable of handling the vehicle carefully. The yawning state is also detected using the median filter technique. Here the noisy areas are removed. The skinny areas are eliminated and the darker regions are taken. A threshold value is set .If the detected mouth has the black pixel value exceeding the threshold value for three consecutive frames alarm is sounded.

Redness of the eyes is also considered as a symptom for drowsiness. Color histogram technique is applied to detect the redness level. The red pixel value is calculated in the iris region. A threshold value is set. If the detected eye has a threshold value exceeding the threshold value an alarm will be sounded.

The proposed work is tested with existing works and proved to be better. The head posture change will make the detection of face harder. If the driver wears sunglasses, or covers his mouth using his hand while yawning then it will not able to detect the eye and mouth region. So in-painting technique can be applied in the future to overcome this problem.

ACKNOWLEDGEMENTS
We acknowledge the Management and the Principal of Mepco Schlenk Engineering College, Sivakasi for providing the dataset and resource support.

REFERENCES
1. Kaplan, S., Guvensan, M. A., Yavuz, A. G., & Karalurt, Y. (2015). Driver behavior analysis for safe driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, *16*(6), 3017-3032.
2. Chacon-Murguia, M. I., & Prieto-Resendiz, C. (2015). Detecting Driver Drowsiness: A survey of system designs and technology. *IEEE Consumer Electronics Magazine*, *4*(4), 107-119..
3. Singh, M., & Kaur, G. (2012). Drowsy detection on eye blink duration using algorithm. *International Journal of Emerging Technology and Advanced Engineering*, *2*(4), 363-365.
4. Sabet, M., Zoroofi, R. A., Sadeghniiat-Haghighi, K., & Sabbaghian, M. (2012, May). A new system for driver drowsiness and distraction detection. In *20th Iranian Conference on Electrical Engineering (ICEE2012)* (pp. 1247-1251). IEEE..
5. IDanisman, T., Bilasco, I. M., Djeraba, C., & Ihaddadene, N. (2010, October). Drowsy driver detection system using eye blink patterns. In *2010 International Conference on Machine and Web Intelligence* (pp. 230-233). IEEE.
6. Azim, T., Jaffar, M. A., & Mirza, A. M. (2009, December). Automatic fatigue detection of drivers through pupil detection and yawning analysis. In *2009 fourth international conference on innovative computing, information and control (ICICIC)* (pp. 441-445). IEEE..
7. Saradadevi, M., & Bajaj, P. (2008). Driver fatigue detection using mouth and yawning analysis. *International journal of Computer science and*

network security, *8*(6), 183-188.

8. Ishikawa, T. (2004). Passive driver gaze tracking with active appearance models.
9. Ishikawa, T. (2004). Passive driver gaze tracking with active appearance models..
10. Bagci, A. M., Ansari, R., Khokhar, A., & Cetin, E. (2004, August). Eye tracking using Markov models. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* (Vol. 3, pp. 818-821). IEEE..
11. Wang, T., & Shi, P. (2005, May). Yawning detection for determining driver drowsiness. In *Proceedings of 2005 IEEE International Workshop on VLSI Design and Video Technology, 2005.* (pp. 373-376). IEEE.
12. Ayoob, E. M., Grace, R., & Steinfeld, A. M. (2005). Driver-vehicle-interface (DVI) development of a drowsy driver detection and warning system for commercial vehicles. *Tech. Rep. CMU-RI-TR-05-46*..
13. Jahn, G., Oehme, A., Krems, J. F., & Gelau, C. (2005). Peripheral detection as a workload measure in driving: Effects of traffic complexity and route guidance system use in a driving study. *Transportation Research Part F: Traffic Psychology and Behaviour*, *8*(3), 255-275..
14. Baker, S. (2004). Real-time non-rigid driver head tracking for driver mental state estimation..
15. Ahmad, R., & Borole, J. N. (2015). Drowsy driver identification using eye blink detection. *IJISET-International Journal of Computer Science and Information Technologies*, *6*(1), 270-274.