DRIVER DROWSINESS USING OPENCV

A PROJECT REPORT
Submitted By

Divyansh Varshney (2000290140043)
Aman Jain (2000290140016)
Vaibhav Srivastava (2000290140129)

Submitted in partial fulfillment of the Requirements for the Degree of

MASTER OF COMPUTER APPLICATIONS

Under the Supervision of
Prof. (Dr.) Amit Goyal
(Associate Professor)

Submitted to

DEPARTMENT OF COMPUTER APPLICATIONS
KIET Group of Institutions, Ghaziabad
Uttar Pradesh-201206

(2022)

CERTIFICATE

Certified that **Vaibhav Srivastava (200029014005814), Divyansh Varshney (200029014005728) and Aman Jain (200029014005701)** have carried out the project work having " Driver Drowsiness Detection using openCV"for Master of Computer Applications from Dr. A.P.J. Abdul Kalam Technical University (AKTU) (formerly UPTU), Technical University, Lucknow under my supervision. The project report embodies original work, and studies are carried out by the student himself / herself and the contents of the project report do not form the basis for the award of any other degree to the candidate or to anybody else from this or any other University/Institution.

Date:

Vaibhav Srivastava (200029014005814)
Divyansh Varshney (200029014005728)
Aman Jain (200029014005701)

This is to certify that the above statement made by the candidate is correct to the best of my knowledge.

Date:

Prof. (Dr.) Amit Goyal
(Associate Professor)
Department of Computer Applications
KIET Group of Institutions, Ghaziabad

Signature of Internal Examiner

Signature of External Examiner

Dr. Ajay Shrivastava
Head, Department of Computer Applications
KIET Group of Institutions, Ghaziabad

ABSTRACT

Number of peoples dying due to road accidents are increasing day by day. Drowsiness and sleepiness of drivers are amongst the significant causes of road accidents. Studies have suggested that around 17% of all road accidents are fatigue related. However, initial stages of fatigue and drowsiness can be detected before a critical situation arises which may cause an accident. A direct way of measuring driver fatigue is measuring the drowsiness. Most of the methods used are vehicle based or Physiological based. Some methods are intrusive and distract the driver, some require expensive sensors manually. The proposed methodology will be a one of the approach to identify the drowsiness and alert the driver by which at least few of the accidents can be avoided.

Drowsiness detection is a safety technology that can prevent accidents that are caused by drivers who fell asleep while driving. Therefore, the proposed paper attempted to address the drowsiness issue. The proposed approach to combine Mouth Aspect Ratio, PRECLOS and Eye Aspect Ratio. The system will monitor the user eyes using a camera and by developing an algorithm we can detect symptoms of user fatigue early enough to avoid the user from sleeping. In this project we use OpenCV which is basically used for face detection followed by 68 points of facial landmark identification.

So, this project is aimed towards developing a prototype of drowsiness detection system which detects the eye closure of user or driver and give warning to them upon detecting drowsiness. Driver face is captured, and eye retina detection are done, and blinking values are calculated then threshold values are set. The mean eye landmarks’ distance is used to differentiate between the open eye and closed eye. To spot the sleepiness of the driver, firstly the face and then the eye of the driver are correctly detected. If the eye is closed, then the duration of time for the closed state is considered to determine the drowsiness condition. If the duration is high, for giving warning to the driver an alarming system is attached. One more aspect we are using, and it is yawning technique. To detect yawning we are calculating the mouth aspect ratio. By calculations we can detect drowsiness. So, we are using to calculate the drowsiness and to notify the alarm to the driver.

Keywords—Driver drowsiness, face detection, OpenCV, Perclos.

ACKNOWLEDGEMENTS

Success in life is never attained single handedly. My deepest gratitude goes to my thesis supervisor, **Prof Amit Goyal** for his guidance, help and encouragement throughout my research work. Their enlightening ideas, comments, and suggestions.

Words are not enough to express my gratitude to **Dr. Ajay Kumar Shrivastava**, Professor and Head, Department of Computer Applications, for his insightful comments and administrative help at various occasions.

Fortunately, I have many understanding friends, who have helped me a lot on many critical conditions.

Finally, my sincere thanks go to my family members and all those who have directly and indirectly provided me moral support and other kind of help. Without their support, completion of this work would not have been possible in time. They keep my life filled with enjoyment and happiness.

Vaibhav Srivastava

Divyansh Varshney

Aman Jain

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

The following are the list of abbreviations:

NHTSA	National Highway Traffic Safety Administration
HMM	Hidden Markov Model
EAR	Eye Aspect Ratio
ECR	Eye Closure Ratio
NREM	Non-rapid eye movement sleep
REM	Rapid eye movement sleep
KSS	Karolinska Sleepiness Scale
VLP	variation of lane position
SWM	Steering Wheel Movement
SDLP	Standard Deviation of Lane Position
ECG	electrocardiogram
EEG	electroencephalogram
EOG	Electrooculogram
EMG	electromyogram
PERCLOS	percentage of eyelid closure

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

1.1 INTRODUCTION

Drowsiness is one of the important causes of fatigue accidents which may produce several damage, injury, and deaths. Fatigue and micro sleep at the driving controls are often the root cause of serious accidents. This automatically becoming a major concern around globe and most of the countries are making great effort on how to detect drowsiness during driving. The drivers may be tired due to continuous driving for a long period.

Growth in roads infrastructures, advancements in vehicles and development of road safety laws are intended to reduce road accidents. However according to a report published by WHO, death count of 1.25 million people per annum on road accidents has not reduced [1]. Changes in our lifestyle resulted in increased number of traffic accidents due to driver's drowsiness or sleepiness [2]. National Highway Traffic Safety Administration (NHTSA) estimates that approximately 25% of police reported accidents involves driver fatigue [3]. Although some countries have imposed restrictions on the number of hours a driver can drive at a stretch, but it is still not enough to solve this problem as its implementation is very difficult and costly.

In the growing nations like India, with advancement in the technology of transportation and uplift of vehicle numbers on the roads, there is increase in road accidents. Hence the proposed work is to give alertness by means of voice alert system or alarm to the driver when he is inattentive to drive and shift from normal alert mode to non-alert. In this system real time data is collected by video camera, this data gives information about driving condition of the driver which acts as input to controller. Based on the input appropriate measures are taken by the controller to alert the driver.

The development of technologies for detecting or preventing drowsiness at the wheel is a major challenge in the field of accident-avoidance systems. Because of the hazard that drowsiness presents on the road, methods need to be developed for counteracting its affects. The aim of this project is to develop a prototype drowsiness detection system. The focus will be placed on designing a system that will accurately monitor the open or closed state of the driver's eyes in real-time.

When driver drives for more than normal period for human then excessive fatigue is caused and also results in tiredness which drives the driver to sleepy condition or loss of consciousness. Drowsiness is a complex phenomenon which states that there is a decrease in alerts and conscious levels of the driver. Though there is no direct measure to detect the drowsiness, but several indirect methods can be used for this purpose.

The purpose of the research paper is to alert drivers so that they can be cautioned to pull over and stop driving to take some time in a drowsy state. This paper proposed a method by tracking eye position of drivers or users. Once the position of the eyes is located, the system is designed to determine whether the eyes are opened or closed and detect fatigue.

1.2 LITERATURE REVIEW

As of now there has been an extensive amount of work done on drowsiness detection. But here we specify only a few important and relevant literature works.

B.N in 2016, has proposed a method that detect the face using Haar feature-based cascade classifiers. Initially, the algorithm needs a lot of positive images (images of faces) and negative images (images without faces) to train the classifier that will detect the object. So along with the Haar feature-based classifiers, cascaded Adaboost classifier is exploited to recognize the face region then the compensated image is segmented into numbers of rectangle areas, at any position and scale within the original image. Due to the difference of facial feature, Haar-like feature is efficient for real-time face detection. These can be calculated according to the difference of sum of pixel values within rectangle area and during the process the Adaboostalgorithm will allow all the face samples and it will discard the non-face samples of images.

Amna Rahman in 2015, has proposed a method to detect the drowsiness by using Eye state detection with Eye blinking strategy. In this method first, the image is converted to gray scale and the corners are detected using Harris corner detection algorithm which will detect the corner at both side and at down curve of eye lid. After tracing the points then it will make a straight-line between the upper two points and locates the mid-point by calculation of the line, and it connects the mid-point with the lower point. Now for each image it will perform the same procedure and it calculates the distance 'd' from the mid-point to the lower point to determine the eye state. Finally, the decision for the eye state is made based on distance 'd' calculated. If the distance is zero or is close to zero, the eye state is classified as "closed" otherwise the eye state is identified as "open". They have also invoked intervals or time to know that the person is feeling drowsy or not. This is done by the average blink duration of a person is 100-400milliseconds (i.e. 0.1-0.4 of a second).

In June, 2014, Eyosiyas et.al. Proposed „Drowsiness Detection through HMM based Dynamic detection‟. According to the proposed approach, facial expressions analysed and sleepiness is detected using Hidden Markov Model (HMM) based dynamic modelling. They used a virtual driving environment to enforce the algorithm. The proposed technique's effectiveness was confirmed by experimental findings. Thus, with reference to the literature work we have a proposed a system that detects driver's drowsiness using EAR and ECR or PERCLOS which are detailed in the following section.

In June 2012, A. Cheng et.al. „Driver Drowsiness Recognition Based on Computer Vision' was the title of the paper. Image processing and eye tracking is used to recognize non-intrusive sleepiness. Due to the changes in illumination and driver posture a strong eye detection algorithm is given to handle the problems caused by the change. Percentage of eyelid closure, maximum closure duration, blink frequency, average opening level of eyes, and closing of eyes are the six measures that are calculated. 86% of accuracy is shown in the video-based drowsiness recognition.

In June,2010, Bin Yang et.al under simulated or observed conditions, "Drowsiness is referenced for the Driver State of eye and Head pose for Driver Alertness" is found. Physical measurements are based on the results of the most recent in-vehicle eye monitoring system. These metrics are assessed statistically and using a classification methodology based on a large database of 90 hours of real-world driving time.

In 2008, HONG Su et.al. "Squares Regression-Based Fusion Model for Predicting Drowsiness" was defined. It is suggested that partial least squares information fusion technique for modelling driver drowsiness with multiple eyelid movement features (PLSR) with that to deal with the matter of robust co-linear relations among protective fold movement options and, thus, predicting the tendency of the sleepiness.

1.2.1 Related Work :

i.) Real-Time Driver Drowsiness Detection System Using Eye Aspect Ratio and Eye Closure Ratio(Sukrit Mehta, Sharad Dadhich, Sahil Gumber, Arpita Jadhav Bhatt)[12]:

The system is implemented on Android. By recording the video and detect driver's face in every frame using image processing techniques. The system is capable of detecting facial landmarks, computes Eye Aspect Ratio (EAR) and Eye Closure Ratio (ECR) to detect driver's drowsiness based on adaptive thresholding. This research uses some common techniques:

- facial landmark of driver using Dlib library
- Compute EAR value from facial landmark

The proposed application then captures the real-time images of the driver. Images are captured every time the application receives a response from the server. The process goes on until the driver stops the ride.

Advantages –

- When the drivers are used to strenuous workload and drive continuously for long distances.
- The proposed system works with the collected data sets under different conditions.
- The system gives best case accuracy of 84% for random forest classifier.

Limitations:

- This system has best case accuracy is 84%.
- This system fails when the driver is taking yawn by occluding face.

ii.) Drowsy Driver Identification Using Eye Blink detection(Raees Ahmad et al, / (IJCSIT) International Journal of Computer Science and Information Technologies) [8]:

The web camera will cover the facial expression and head movement of the driver. For simulation purpose saved video as well as online video are considered here. Web camera record the head movement as well as facial expression. Recorded video are converted into frames, each frame then processes one by one. The normal eye blinking rate is vary from 12-19 per minute. The frequency less than this normal range indicates the drowsy condition of a person/driver.

Advantages –

- Instead of calculating blinking rate, calculation of average drowsiness.
- For eye blinking, detected eye is equated with zero, which indicates closed eye. Whereas nonzero value is considered as fully open /partially open eye.
- Percentage drowsiness is calculated as the ratio of number of closed eyes found to number of frames.

Limitations –

- The system is only dependent on eye aspect ratio, not on Perclos.

- This system does not work when driver has covered his face.

1.3 PROBLEM DEFINITION

The factors causing accidents are: -

- **Speeding:** Majorly in highways truck drivers ignore the speed limit. Speed kills and travelling above the speed limit is an easy way to cause accident.
- **Drunk Driving:** When the driver is drunk, he loses the ability to focus and function properly, hence it is dangerous when operating a vehicle. This is one of the most contributing factors of accident.
- **Reckless Driving:** Improper driving as in speeding up the vehicle or changing lanes too quickly or tailgating and many more can cause reckless driving. Reckless driving is when you're operating vehicle with deliberate disregard.
- **Night Driving**: Driving in daylight can be hazardous, but driving at night nearly doubles the risk of accident. When you can't see what's ahead you don't know what to anticipate as you drive towards it.
- **Driving Under the Influence of Drug:** Drugs, both legal and illegal can impair your ability to fully function as a driver. Mind clearance and control over the body is need else it may cause accident.
- **Drowsy Driving:** Driver fatigue isn't talked about a lot, but how well we can expect anyone to drive when they are having trouble staying awake. Most of the car accidents caused by drowsy driving occur at night.

1.3.1 Drowsy - The term "drowsy" is synonymous with sleepy, which simply means an inclination to fall asleep. The stages of sleep can be categorized as awake, non-rapid eye movement sleep (NREM), and rapid eye movement sleep (REM). The second stage, NREM, can be subdivided into the following three stages [15]:

- Stage I: transition from awake to asleep (drowsy)
- Stage II: light sleep
- Stages III: deep sleep

In order to analyze driver drowsiness, researchers have mostly studied Stage I, which is the drowsiness phase. The crashes that occur due to driver drowsiness have a number of characteristics [16]:

- Occur late at night (0:00 am–7:00 am) or during mid-afternoon (2:00 pm–4:00 pm)
- Involve a single vehicle running off the road
- Occur on high-speed roadways
- Driver is often alone
- Driver is often a young male, 16 to 25 years old
- No skid marks or indication of braking

In relation to these characteristics, the Southwest England and the Midlands Police databases use the following criteria to identify accidents that are caused by drowsiness [14]:

- Blood alcohol level below the legal driving limit
- Vehicle ran off the road or onto the back of another vehicle
- No sign of brakes being applied
- Vehicle has no mechanical defect
- Good weather conditions and clear visibility
- Elimination of “speeding” or “driving too close to the vehicle in front” as potential causes
- The police officer at the scene suspects sleepiness as the primary cause

1.3.2 Causes of drowsiness - While alcohol and some medications may contribute to sleepiness individually, sleep restriction, sleep disturbance, and circadian factors are the key triggers of sleepiness and drowsiness in people without sleep problems. A limited sleep period tends to impact alertness more negatively. While sleep needs differ among people, sleep is healthy for 8 hours every 24 hours, and performance improvement requires about 7–9 hours. Sleeping for less than four hours a night is seen to affect efficiency in surveillance activities with experimental data. Acute lack of sleep, even one night’s sleep deprivation, contributes to drained situations.

Sleep is an ongoing cycle, and the appropriate amount of time in bed does not suggest ample rest. Disruption of sleep and division causes inadequate sleep, which may impact working conditions adversely. Sleep variability may have internal and external sources, as is the case for sleep regulation. The major internal factor is sickness, namely, sleep disturbances, not being managed. Exteriors may disrupt and decrease sleep quality and quantity distractions such as loudness, noise, movements, and lighting, a disturbed partner, or job-related duties (e.g., staff on call).

Statistics derived using these criteria cannot account fully for accidents caused by drowsiness because of the complexity involved; therefore, accidents that can be attributed to driver drowsiness may be more devastating than the statistics reveal. Hence, in order to avoid these types of accidents, it is necessary to derive effective measures to detect driver drowsiness and alert the driver.

Nishiyama listed the following six conditions required for detecting drowsiness in in-vehicle systems:

- Do not interfere with the driver’s safe driving environment;
- Can be equipped in a vehicle and withstand hard operating environments;
- Can detect driver’s drowsiness in real-time;
- Have a wide detection range from shallow to deep sleep;
- Consider all drivers are detectable targets;
- Have low cost and high scalability than other applications.

1.4 SCOPE

The Scope of the Drowsy Driver Detection framework in the Drowsy Driver Detection framework diminishes the vehicle street mishap and furthermore this framework utilized for security reason for a driver. Scope of Drowsy Driver Detection system is Reduce accident and Security purpose of the driver.

CHAPTER 2

2.1 DROWSINESS

Drowsiness is defined as a decreased level of awareness portrayed by sleepiness and trouble in staying alarm but the person awakes with simple excitement by stimuli. It might be caused by an absence of rest, medicine, substance misuse, or a cerebral issue. It is mostly the result of fatigue which can be both mental and physical. Physical fatigue, or muscle weariness, is the temporary physical failure of a muscle to perform ideally. Mental fatigue is a temporary failure to keep up ideal psychological execution. The onset of mental exhaustion amid any intellectual action is progressive, and relies on an individual's psychological capacity, furthermore upon different elements, for example, lack of sleep and general well-being. Mental exhaustion has additionally been appeared to diminish physical performance. It can show as sleepiness, dormancy, or coordinated consideration weakness. In the past years according to available data driver sleepiness has gotten to be one of the real reasons for street mishaps prompting demise and extreme physical injuries and loss of economy. A driver who falls asleep is in an edge of losing control over the vehicle prompting crash with other vehicle or stationary bodies. Keeping in mind to stop or reduce the number of accidents to a great extent the condition of sleepiness of the driver should be observed continuously.

2.2 METHODS FOR MEASURING DROWSINESS

Researchers have used various methods to measure driver drowsiness. This section provides a review of the four most widely used methods, among which the first method is measured either verbally or through questionnaire and the remaining three by means of various sensors. Different techniques have been reported for the detection of driver's drowsiness [4]. Here is a below table will summarize all the techniques:

Category	**Measurement**	**Characteristics**
Subjective	Through questioners by professional	A well-defined reference developed by expert
Psychological	EEG, ECG, EoG etc. [5]	Measurement through sensors attached with driver
Vehicular	Steering wheel movement, Acceleration, Lateral distance, etc.	Measurement with sensors attached to the vehicle
Visual	Eye blinking per minute, Yawning, Head pose, Head motion, etc.	Computer vision-based measurement of the driver's behaviour

Table 2.1 Summarize all techniques

2.2.1 Subjective Measure: - Subjective measures that evaluate the level of drowsiness are based on the driver's personal estimation and many tools have been used to translate this rating to a measure of driver drowsiness. The most used drowsiness scale is the Karolinska Sleepiness Scale (KSS), a nine-point scale that has verbal anchors for each step, as shown in Table 2.2[18]. Hu et al. measured the KSS ratings of drivers every 5 min and used it as a reference to the EoG signal collected [17].Portouli et al. evaluated EEG data by confirming driver drowsiness through both a questionnaire and a licensed medical practitioner [19]. Some researchers compared the self-determined KSS, which was recorded every 2 min during the driving task, with the variation of lane position (VLP) and found that these measures were not in agreement [20]. Ingre et al. determined a relationship between the eye blink duration and the KSS collected every 5 min during the driving task [21].

Researchers have determined that major lane departures, high eye blink duration and drowsiness-related physiological signals are prevalent for KSS ratings between 5 and 9 [21].However, the subjective rating does not fully coincide with vehicle-based, physiological and behavioral measures.

Because the level of drowsiness is measured approximately every 5 min, sudden variations cannot be detected using subjective measures. Another limitation to using subjective ratings is that the self-introspection alerts the driver, thereby reducing their drowsiness level. In addition, it is difficult to obtain drowsiness feedback from a driver in a real driving situation. Therefore, while subjective ratings are useful in determining drowsiness in a simulated environment, the remaining measures may be better suited for the detection of drowsiness in a real environment.

Rating	**Verbal descriptions**
1	Extremely alert
2	Very alert
3	Alert
4	Fairly alert
5	Neither alert nor sleepy

6	Some signs of sleepiness
7	Sleepy, but no effort to keep alert
8	Sleepy, some effort to keep alert
9	Very sleepy, great effort to keep alert, fighting sleep

Table 2.2 Karolinska sleepiness scale (KSS).

2.2.2 Vehicle Based Measures: - Another method to measure driver drowsiness involves vehicle-based measurements. In most cases, these measurements are determined in a simulated environment by placing sensors on various vehicle components, including the steering wheel and the acceleration pedal; the signals sent by the sensors are then analyzed to determine the level of drowsiness. Liu *et al.*[22] published a review on current vehicle-based measures. Some researchers found that sleep deprivation can result in a larger variability in the driving speed [23]. However, the two most commonly used vehicle-based measures are the steering wheel movement and the standard deviation of lane position.

Steering Wheel Movement (SWM) is measured using steering angle sensor and it is a widely used vehicle-based measure for detecting the level of driver drowsiness [23]. Using an angle sensor mounted on the steering column, the driver's steering behavior is measured. When drowsy, the number of micro-corrections on the steering wheel reduces compared to normal driving [24]. Fairclough and Graham found that sleep deprived drivers made fewer steering wheel reversals than normal drivers [23]. To eliminate the effect of lane changes, the researchers considered only small steering wheel movements (between 0.5° and 5°), which are needed to adjust the lateral position within the lane [18]. Hence, based on small SWMs, it is possible to determine the drowsiness state of the driver and thus provide an alert if needed. In a simulated environment, light side winds that pushed the car to the right side of the road were added along a curved road in order to create variations in the lateral position and force the drivers to make corrective SWMs [25]. Car companies, such as Nissan and Renault, have adopted SWMs but it works in very limited situations [26]. This is because they can function reliably only at particular environments and are too dependent on the geometric characteristics of the road and to a lesser extent on the kinetic characteristics of the vehicle [26].

Standard Deviation of Lane Position (SDLP) is another measure through which the level of driver drowsiness can be evaluated [21]. In a simulated environment, the software itself gives the SDLP and in case of field experiments the position of lane is tracked using an external camera. Ingre *et al.* conducted an experiment to derive numerical statistics based on SDLP and found that, as KSS ratings increased, SDLP (meters) also increased [21]. For example, KSS ratings of 1, 5, 8, and 9 corresponded to SDLP measurements of 0.19, 0.26, 0.36 and 0.47, respectively. The SDLP was calculated based on the average of 20 participants; however, with some drivers, the SDLP did not exceeded 0.25 m even for a KSS rating of 9. In the above experiment by performing correlation analysis on a subject-to-subject basis significant difference is noted. Another limitation of SDLP is that it is purely dependent on external factors like road marking, climatic and lighting conditions. In summary, many studies have determined that vehicle-based measures are a poor predictor of performance error risk due to drowsiness. Moreover, vehicular-based metrics are not specific to drowsiness. SDLP can also be caused by any type of impaired driving, including driving under the influence of alcohol or other drugs, especially depressants.

2.2.3 Psychological Measures: - As drivers become drowsy, their head begins to sway and the vehicle may wander away from the center of the lane. The previously described vehicle-based and vision based measures become apparent only after the driver starts to sleep, which is often too late to prevent an accident.

However, physiological signals start to change in earlier stages of drowsiness. Hence, physiological signals are more suitable to detect drowsiness with few false positives; making it possible to alert a drowsy driver in a timely manner and thereby prevent many road accidents.

Many researchers have considered the following physiological signals to detect drowsiness: electrocardiogram (ECG), electromyogram (EMG), electroencephalogram (EEG) and electro-oculogram (EoG) (Table 2.2.3). Some researchers have used the EoG signal to identify driver drowsiness through eye movements [17].The electric potential difference between the cornea and the retina generates an electrical field that reflects the orientation of the eyes; this electrical field is the measured EoG signal. Researchers have investigated horizontal eye movement by placing a disposable Ag-Cl electrode on the outer corner of each eye and a third electrode at the center of the forehead for reference [17].The electrodes were placed as specified so that the parameters - Rapid eye movements (REM) and Slow Eye Movements (SEM) which occur when a subject is awake and drowsy respectively, can be detected easily [27].

Monitoring Heart Rate: An ECG sensor can be installed in the steering wheel of a car to monitor a driver's pulse, which gives a sign of the driver's level of fatigue indirectly giving the state of drowsiness. Additionally the ECG sensor can be introduced in the back of the seat.

Monitoring Brain Waves: Special caps embedded with electrodes measures the brain waves to identify fatigue in drivers and report results in real time. Then each brain waves can be classified accordingly to identify drowsiness.

Monitoring muscle fatigue: As muscle fatigue is directly related to drowsiness. We know during fatigue the pressure on the steering wheel reduces and response of several muscle drastically reduces hence it can be measured by installation of pressure sensors at steering wheel or by measuring the muscle response with applied stimuli to detect the fatigue.

Monitoring eye movements: Invasive measurement of eye movement and eye closure can be done by using electro occulogram but it will be very uncomfortable for the driver to deal with. Though this method gives the most accurate results regarding drowsiness. But it requires placement of several electrodes to be placed on head, chest and face which is not at all a convenient and annoying for a driver. Also they need to be very carefully placed on respective places for perfect result.

Sensors	**Preprocessing**	**Feature Extraction**	**Classification**	**Classification accuracy (%)**
EEG, ECG, EoG	Optimal Wavelet Packet, Fuzzy Wavelet Packet	The Fuzzy MI-based Wavelet-Packet Algorithm	LDA, LIBLINEAR, KNN, SVM	95–97%
ECG	Band Pass Filter	Fast Fourier Transform (FFT)	Neural Network	90%
EEG	Independent Component Analysis Decomposition	Fast Fourier Transform	Self-organizing Neural Fuzzy Inference Network	96.7%
EEG, EMG	Band Pass Filter & Visual Inspection	Discrete Wavelet Transform (DWT)	Artificial Neural Network (ANN) Back Propogation Algorithm (Awake, Drowsy, Sleep)	98–99%
EEG	Low pass filter 32 Hz	512 point Fast Fourier Transform with 448 point overlap	Mahalanobis distance	88.7%
EoG, EMG	Filtering & Thresholding	Neighborhood search	SVM	90%

EEG, EoG, EMG	Low pass pre Filter and Visual Inspection	Discrete Wavelet Transform	ANN	97–98%
EEG	Least mean square algorithm and Visual Inspection	Wavelet packet analysis with Daubechies 10 as mother wavelet	Hidden Markov Model	84%

Table 2.3 List of previous works on driver drowsiness detection using physiological signals.

The heart rate (HR) also varies significantly between the different stages of drowsiness, such as alertness and fatigue [28]. Therefore, heart rate, which can be easily determined by the ECG signal, can also be used to detect drowsiness. Others have measured drowsiness using Heart Rate Variability (HRV), in which the low (LF) and high (HF) frequencies fall in the range of 0.04–0.15 Hz and 0.14–0.4 Hz, respectively [29]. HRV is a measure of the beat-to-beat (R-R Intervals) changes in the heart rate. The ratio of LF to HF in the ECG decreases progressively as the driver progresses from an awake to a drowsy state [29].

The Electroencephalogram (EEG) is the physiological signal most commonly used to measure drowsiness. The EEG signal has various frequency bands, including the delta band (0.5–4 Hz), which corresponds to sleep activity, the theta band (4–8 Hz), which is related to drowsiness, the alpha band (8–13 Hz), which represents relaxation and creativity, and the beta band (13–25 Hz), which corresponds to alertness [30]. A decrease in the power changes in the alpha frequency band and an increase in the theta frequency band indicates drowsiness. Akin *et al.* observed that the success rate of using a combination of EEG and EMG signals to detect drowsiness is higher than using either signal alone [30].

The measurement of raw physiological signals is always prone to noise and artifacts due to the movement that is involved with driving. Hence, in order to eliminate noise, various preprocessing techniques, such as low pass filter, digital differentiators, have been used. In general, an effective digital filtering technique would remove the unwanted artifacts in an optimal manner [31]. A number of statistical features are then extracted from the processed signal using various feature extraction techniques, including Discrete Wavelet Transform (DWT) and Fast Fourier Transform (FFT) [30]. The extracted features are then classified using Artificial Neural Networks (ANN), Support Vector Machines (SVM), Linear Discriminant Analysis (LDA), or other similar methods [17].

EEG: - An electroencephalogram (EEG) is a test that detects abnormalities in your brain waves, or in the electrical activity of your brain. During the procedure, electrodes consisting of small metal discs with thin wires are pasted onto your scalp. The electrodes detect tiny electrical charges that result from the activity of your brain cells. The charges are amplified and appear as a graph on a computer screen, or as a recording that may be printed out on paper. Your healthcare provider then interprets the reading. During an EEG, your healthcare provider typically evaluates about 100 pages, or computer screens, of activity. He or she pays special attention to the basic waveform, but also examines brief bursts of energy and responses to stimuli, such as flashing lights. Evoked potential studies are related procedures that also may be done. These studies measure electrical activity in your brain in response to stimulation of sight, sound, or touch.

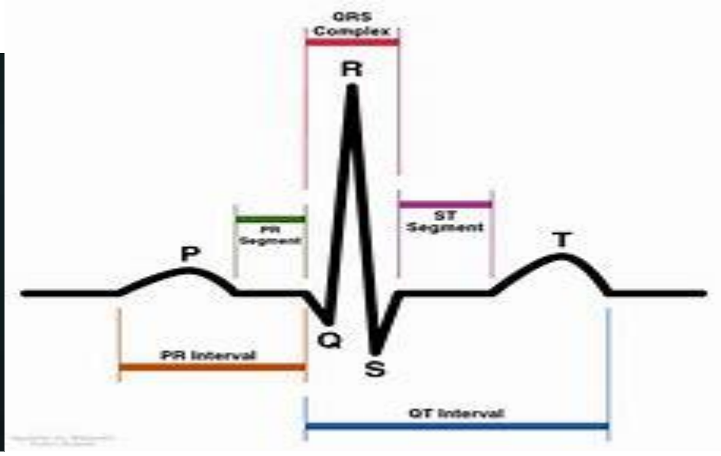


Figure 2.1.a EEG **Figure 2.1.b ECG**

ECG: - An electrocardiogram records the electrical signals in the heart. It's a common and painless test used to quickly detect heart problems and monitor the heart's health. An electrocardiogram — also called ECG or EKG — is often done in a health care provider's office, a clinic or a hospital room. ECG machines are standard equipment in operating rooms and ambulances. Some personal devices, such as smartwatches, offer ECG monitoring. To obtain a standard ECG graph, a patient is connected to the machine with three electrical leads, one to each wrist and to the left ankle, that continuously monitor the heart activity and functioning. The human heart produces an electrical impulse passing through our heart, it generates an electrical impulse by itself. To trace the electrical activity of the heart, special kinds of machines are used and they are known as ECG recording machines or electrocardiograms. The process of developing ECG is called electrocardiography. A typical machine uses few electrodes (sensors) and a central unit. Today's modern-day ECG machines have analog to digital converters which convert the analog inputs to digital inputs. Such digital signals are either displayed on the screen or printed on paper.

EOG: - The electrooculogram (EOG) measures the cornea-positive standing potential relative to the back of the eye. By attaching skin electrodes outside the eye near the lateral and medial canthus, the potential can be measured by having the patient move the eyes horizontally a set distance. The voltage becomes smaller in the dark, reaching its lowest potential after 8-12min, the so-called dark trough. When the lights are turned on, the potential rises, reaching a peak by about 10min. When the size of the light peak is compared to the dark trough, the normal ratio should be near 2:1. A light peak:dark trough ratio of less than 1.7 is considered abnormal. The origin of electrooculographic potentials is the pigment epithelium of the retina interacting with the midretina. The light rise of the potential requires both a normal pigment epithelium and normal midretinal function. The most common use of the electrooculogram is to confirm Best disease. Best disease is identified by the appearance of an egg-yellow fundus and can be confirmed by recording both an electroretinogram (ERG) and electrooculogram (EOG). The ERG will be normal and the EOG will be abnormal. The EOG is also used for tracking eye movement.

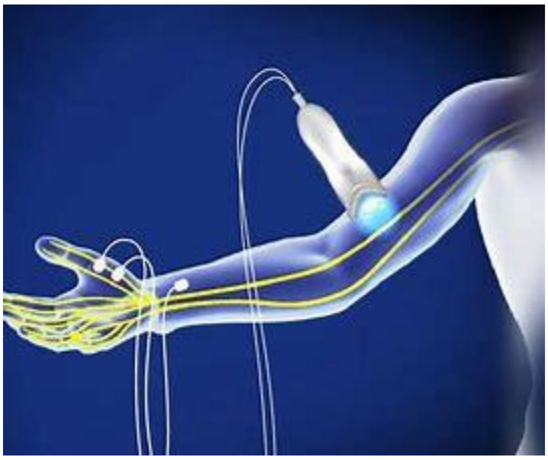


Figure 2.1.c EOG **Figure 2.1.d EMG**

EMG: - Electromyography (EMG) is a diagnostic procedure that evaluates the health condition of muscles and the nerve cells that control them. These nerve cells are known as motor neurons. They transmit electrical signals that cause muscles to contract and relax. An EMG translates these signals into graphs or numbers, helping doctors to make a diagnosis. A doctor will usually order an EMG when someone is showing symptoms of a muscle or nerve disorder. These symptoms may include tingling, numbness, or unexplained weakness in the limbs. EMG results can help the doctor diagnose muscle disorders, nerve disorders, and disorders affecting the connection between nerves and muscles.

An EMG uses tiny devices called electrodes to translate these signals into graphs, sounds or numerical values that are then interpreted by a specialist. During a needle EMG, a needle electrode inserted directly into a muscle records the electrical activity in that muscle. A nerve conduction study, another part of an EMG, uses electrode stickers applied to the skin (surface electrodes) to measure the speed and strength of signals traveling between two or more points. Some doctors may refer to electromyography as an electrodiagnostic exam.

2.2.4 Visual Measures: - A drowsy person displays a number of characteristic facial movements, including rapid and constant blinking, nodding or swinging their head, and frequent yawning. Computerized, non-intrusive, behavioral approaches are widely used for determining the drowsiness level of drivers by measuring their abnormal behaviors. Most of the published studies on using behavioral approaches to determine drowsiness, focus on blinking. PERCLOS (which is the percentage of eyelid closure over the pupil over time, reflecting slow eyelid closures, or “droops”, rather than blinks) has been analyzed in many studies [32]. This measurement has been found to be a reliable measure to predict drowsiness and has been used in commercial products such as Seeing Machines [33] and Lexus [34]. Some researchers used multiple facial actions, including inner brow rise, outer brow rise, lip stretch, jaw drop and eye blink, to detect drowsiness. However, research on using other behavioral measures, such as yawning [35] and head or eye position orientation [36], to determine the level of drowsiness is ongoing.

Eye Blinking Technique: - This technique detects the level of drowsiness and sleepiness by calculating the eye blinking rate and eye closure duration. The reason is that when a driver felt drowsy, his/her eye blinking rate and gaze between eyelids are different from normal situations. Using eye blinking rate drowsiness detection methodology is presented in Fig. 2.1 where eye-blinking classification is carried out through computer vision.

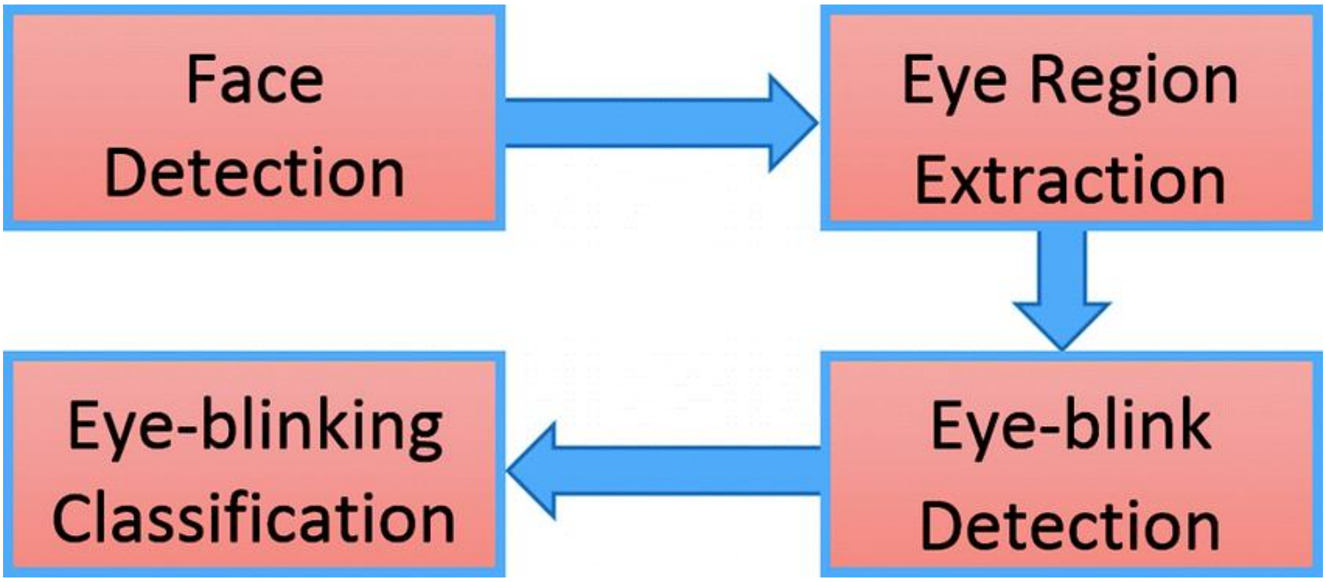


Figure 2.2 Computer vision based eye blinking classification

Ahmad and Borole monitor the position of irises and eye states in the technique [6]. They placed a camera at a suitable place in the vehicle and acquire video. Then by applying computer vision techniques to sequentially localize face, eyes and eyelids positions to measure ratio of closure.

PERCLOS (Percentage of Eye Closure) means percentage of time eye closed[6] in a given period. To sense the level of drowsiness PERCLOS is a well-known parameter. Yan et al. [7] consider that a human blinks once every 5 s on average that is 12 times blinks per minute. They tried gray-scale conversion and template matching for extracting data. In addition, another variable (Sleep Counter) is maintained which counts the number of times the EAR value is less than threshold value. Variable (Total Counter) stores the total count of responses from the server side and is used to calculate the ECR (Eye Closure Ratio). It is defined as the ratio of Sleep Counter and Total Counter value and was computed.

$$\mathbf{ECR} = \frac{\textbf{Sleep Counter}}{\textbf{Total Counter}} \qquad \textbf{(2.1)}$$

The value of ECR was calculated for every 15 consecutive frames (captured from camera) by Yan et al. As soon as the frame number reaches to 16, the value of total counter becomes one and sleep counter becomes zero. Whenever the ECR value exceeds the threshold value which is set to 0.5, then a notification is generated to indicate drowsy state of the driver.

Sensor used	Drowsiness Measure	Detection techniques	Feature Extraction	Classification	Positive Detection rate

CCD micro camera with Infra-Red Illuminator	Pupil	Ada-boost	Red eye effect, Texture detection method	Ratio of eye-height and eye-width	92%
Camera and Infra-Red Illuminator	PERCLOS, eye closure duration, blink frequency, and 3 other	Two Kalman filters for pupil detection	Modification of the algebraic distance algorithm for conics Approximation & Finite State Machine	Fuzzy Classifier	Close to 100%
CCD camera	Yawning	Gravity-center template and grey projection	Gabor wavelets	LDA	91.97%
Digital Video camera	Facial action	Gabor filter	Wavelet Decomposition	SVM	96%
Fire wire camera and webcam	Eye Closure Duration & Freq of eye closure	Hough Transform	Discrete Wavelet Transform	Neural Classifier	95%
Camera	Multi Scale dynamic features	Gabor filter	Local Binary Pattern	Ada boost	98.33%
IR Camera	Eye State	Gabor filter	Condensation algorithm	SVM	93%
Simple Camera	Eye blink	Cascaded Classifiers Algorithm detects face and Diamond searching lgorithm to trace the face	Duration of eyelid closure, No. of continuous blinks, Frequency of eye blink	Region Mark Algorithm	98%
Camera with IR Illuminator	PERCLOS	Haar Algorithm to detect face	Unscented Kalman filter algorithm	SVM	99%

Table 2.4 List of previous works on driver drowsiness detection using visual signals.

Category	**Advantages**	**Limitations**
Subjective	Simple, no sensors and no equipment	Not possible in real time
Psychological	Reliable, accurate, early detection	Intrusive, expensive
Vehicular	Non-intrusive, Ease of use, small sensors and moderate processing	Unreliable, late detection
Visual	Non-intrusive, Ease of use, proven hardware	Lighting conditions, background, tough threshold setting

Table 2.5 Advantages and limitations of different techniques.

2.3 DISCUSSION AND RECOMMENDATIONS

2.3.1 Comparison of Simulated and Real Driving Conditions: - It is not advisable to force a drowsy driver to drive on roads. Consequently, many experiments have been conducted in simulated environments and the results of the experiments are then elaborately studied. Dinges *et al.* presented various challenges involved in real time drowsiness detection. The subjective self-assessment of drowsiness can only be obtained from subjects in simulated environments. In real conditions, it is unfeasible to obtain this information without significantly distracting the driver from their primary task. Some researchers have conducted experiments to confirm the validity of simulated driving environments. For example, Blana *et al.* observed that the mean lateral displacement of the vehicle from the center of the roadway, obtained in real and simulated environments is statistically different for speeds higher than 70 km/h. This finding implies that real-road drivers feel less safe at higher speeds and, as a result, increase their lateral distance. The drivers in a simulated environment, however, did not appear to perceive this risk [37]. Most experiments using behavioral measures are conducted in a simulated environment and the results indicate that it is a reliable method to detect drowsiness. However, in real driving conditions, the results might be significantly different because a moving vehicle can present challenges such as variations in lighting, change in background and vibration noise, not to mention the use of sunglasses, caps, *etc*. Philip *et al.* compared drowsiness in simulated and real conditions and concluded that it can be equally studied in both environments but the reaction time and the sleepiness self-evaluation are more affected in a simulated environment which provides a more monotonous task. Engstorm *et al.* observed that the physiological workload and steering activity was higher in a real environment. This result can be interpreted as an indication of increased effort, which seems reasonable given the higher actual risk in real traffic [38]. Hence, while developing a drowsiness detection system, the simulated environment should be as close to a replica of the real environment as possible.

2.3.2 Hybrid Measures: - Each method used for detecting drowsiness has its own advantages and limitations. Vehicle-based measures are useful in measuring drowsiness when a lack of vigilance affects vehicle control or deviation. However, in some cases, there was no impact on vehicle-based parameters when the driver was drowsy [21], which makes a vehicle-based drowsiness detection system unreliable. Behavioral measures are an efficient way to detect drowsiness and some real-time products have been developed [39]. However, when evaluating the available real-time detection systems, Lawrence *et al.* observed that different illumination conditions affect the reliability and accuracy of the measurements [39]. Physiological measures are reliable and accurate because they provide the true internal state of the driver; however, their intrusive nature has to be resolved. Among all physiological parameters investigated, ECG can be measured in a less intrusive manner. EEG signals require a number of electrodes to be placed on the scalp and the electrodes used for measuring EoG signals are placed near the eye which can hinder driving. Non-obtrusive physiological sensors to estimate the drowsiness of drivers are expected to become feasible in the near future. The advantages of physiological measures and the increasing availability of non-intrusive measurement equipment make it beneficial to combine physiological signals with behavioral and vehicle-based measures. A sample drowsiness detection system developed by combining ECG signals, standard deviation of lane position and facial images is shown in figure 2.2.

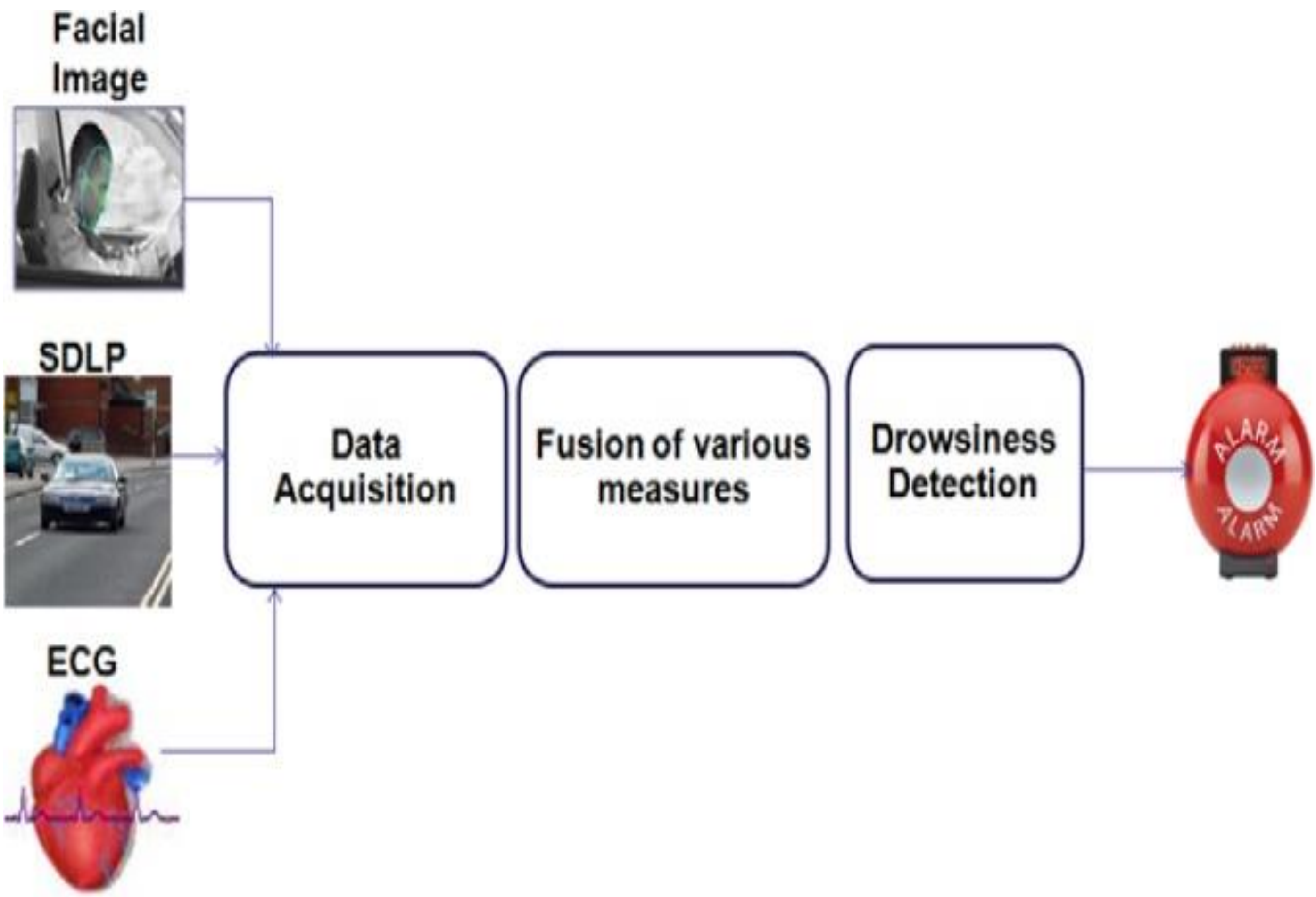


Figure 2.3 A sample hybrid drowsiness detection system using multiple sensors.

Few research studies are attempting to detect driver drowsiness by the fusion of different methods [40]. Cheng *et al.* combined behavioral measures and vehicle based measures and concluded that the reliability and accuracy of the hybrid method was significantly higher than those using single sensors [40]. Guosheng *et al.* used a mixture of subjective, behavioral (PERCLOS) and physiological measures (ECG, EEG) to detect drowsiness and found that this combination resulted in a significantly higher success rate than any individual metric. The average square error while removing physiological features were 1.2629, while the average square error for fusion was 0.5269.

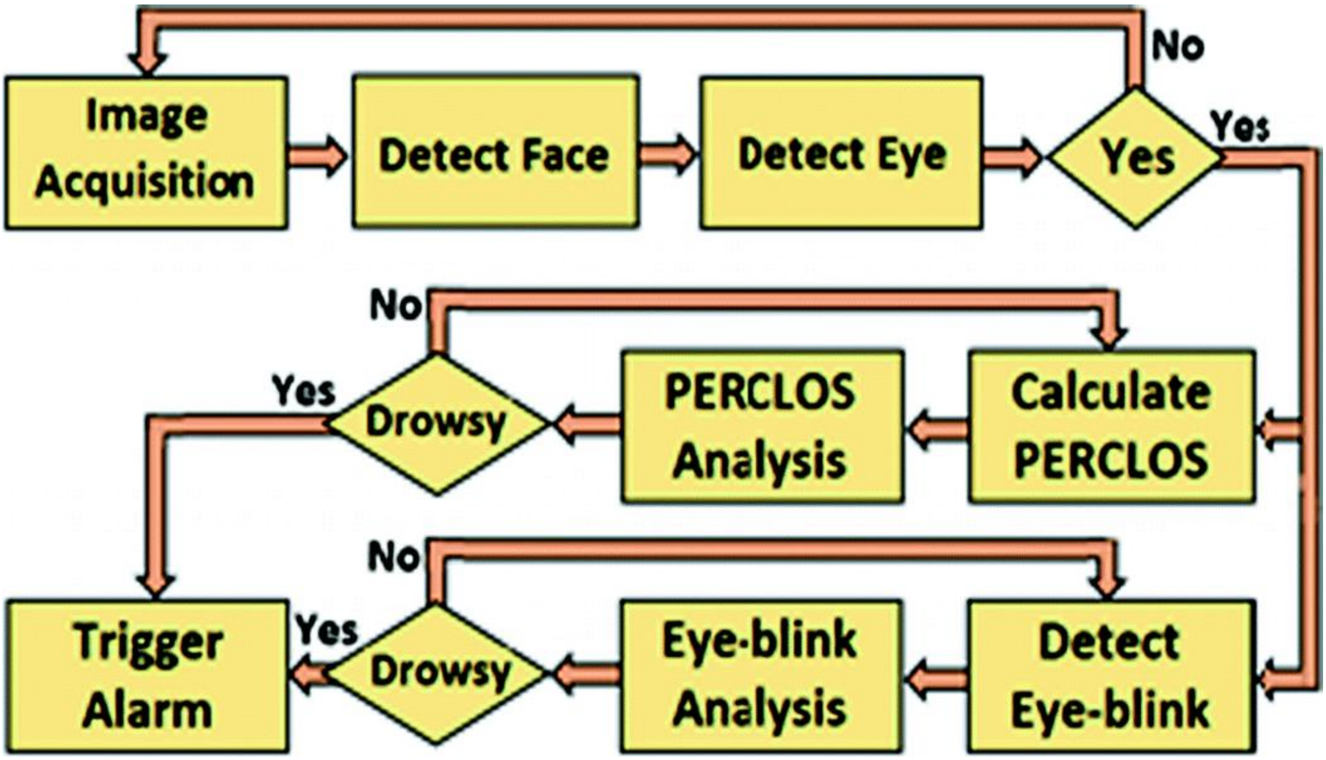


Figure 2.4 Proposed hybrid drowsiness detection using image processing

Although hybrid systems using different sensors have not been tested in a real environment, it would be interesting to investigate the ability to detect drowsiness using a combination of physiological signals with other measurements.

CHAPTER 3

3.1 REQUIREMENT SPECIFICATIONS

This involves both type of requirements i.e., software and hardware. In this section hardware requirements and software requirements for project is specified.

3.1.1 Hardware Requirement Specifications: -

i. Laptop with basic hardware.
ii. Webcam

3.1.2 Software Requirement Specifications: -

i. Jupiter Lab
ii. Any Operating System
iii. Libraries used –
  a. Numpy
  b. Scipy
  c. Playsound
  d. Dlib
  e. Imutils
  f. OpenCV
iv. Technology - Python
v. IDE – Visual Code

3.2 REQUIREMENT ANALYSIS

In this section the above specified requirements are analysed. We have an overview of all the requirements.

3.2.1 Jupiter Lab: - Jupyter Lab is the latest web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. A modular design invites extensions to expand and enrich functionality. The Jupyter Notebook is the original web application for creating and sharing computational documents. It offers a simple, streamlined, document-centric experience. Jupyter (formerly IPython Notebook) is an open-source project that lets you easily combine Markdown text and executable Python source code on one canvas called a notebook. We have used Visual Studio Code which supports working with Jupyter Notebooks natively, and through Python code files. This topic covers the native support available for Jupyter Notebooks and demonstrates how to:

- Create, open, and save Jupyter Notebooks

- Work with Jupyter code cells
- View, inspect, and filter variables using the Variable Explorer and Data Viewer
- Connect to a remote Jupyter server
- Debug a Jupyter Notebook

To select an environment, use the Python: Select Interpreter command from the Command Palette (Ctrl+Shift+P). Once the appropriate environment is activated, you can create and open a Jupyter Notebook, connect to a remote Jupyter server for running code cells, and export a Jupyter Notebook as a Python file. Create a new .ipynb file in your workspace.

3.2.2 Python: Python is a ground-breaking present day PC programming language. It bears a few likenesses to FORTRAN, one of the soonest programming dialects, however it is substantially more remarkable than FORTRAN. Python permits you to utilize factors without proclaiming them (i.e., it decides types verifiably), and it depends on space as a control structure. You are not compelled to characterize classes in Python (in contrast to Java) however you are allowed to do so when advantageous. Python was created by Guido van Rossum, what's more, it is free programming. Free as in free beer, in that you can acquire Python without going through any cash. However, Python is likewise free in other significant manners, for instance you are allowed to duplicate it the same number of times as you like, and allowed to examine the source code, and make changes to it. There is an overall development behind free programming, started in 1983 by Richard Stallman. This report centres around learning Python to do numerical computations. We expect the per user has some information on fundamental science, however we make an effort not to accept any past introduction to PC programming, albeit whatever presentation would absolutely be useful. Python is a decent decision for numerical figuring's, since we can compose code rapidly, test it effectively, and its language structure is like the manner in which scientific thoughts are communicated in the scientific writing. By learning Python, you will likewise be learning a significant device utilized by many web engineers.

Installation and Documentation- Mac OS X or Linux, then Python should already be installed on your computer by default. If not, you can download the most recent adaptation by visiting the Python landing page, at http://www.python.org where you will likewise discover heaps of documentation and other valuable data. Windows clients can likewise download Python at this site.

Features of Python:

i. Simple

Python is a straightforward and moderate language. Perusing a decent Python program feels practically like understanding English, albeit exceptionally exacting English! This pseudo-code nature of Python is probably the best quality. It permits you to focus on the answer for the issue as opposed to the language itself.

ii. Easy to Learn

As you will see, Python is very simple to begin with. Python has a phenomenally basic grammar, as of now referenced.

iii. Free and Open Source

Python is a case of FLOSS (Free/LibrÃ© and Open-Source Software). In basic terms, you can unreservedly circulate duplicates of this product, read its source code, make changes to it, and use bits of it in new free projects. FLOSS depends on the idea of a network which shares information. This is one reason why Python is so acceptable. It has been made and is continually improved by a network who simply need to see a superior Python.

iv. High-level Language

At the point when you compose programs in Python, you never need to make a fuss over the low-level subtleties, for example, dealing with the memory utilized by your program, and so on.

v. Object Oriented

Python bolsters method arranged programming just as article situated programming. In strategy situated dialects, the program is worked around systems or capacities which are only reusable bits of projects. In object-situated dialects, the program is worked around objects which join information and usefulness. Python has an extremely ground-breaking yet short-sighted method of doing OOP, particularly when contrasted with large dialects like C++ or Java.

vi. Extensive Libraries

The Python Standard Library is immense for sure. It can assist you with doing different things including normal articulations, documentation age, unit testing, stringing, databases, internet browsers, CGI, FTP, email, XML, XML- RPC, HTML, WAV documents, cryptography, GUI (graphical UIs), and other framework subordinate stuff.

3.2.3 NumPy: NumPy is the basic bundle for logical processing in Python. It is a Python library that gives a multidimensional cluster object, different determined items, (for example, conceal exhibits and frameworks), and a variety of schedules for quick procedure on exhibits, including scientific, legitimate, shape control, arranging, choosing, I/O, discrete Fourier changes, essential straight polynomial math, fundamental measurable tasks, arbitrary recreation and significantly more. At the centre of the NumPy bundle, is the ND cluster object. This embodies n-dimensional varieties of homogeneous information types, with numerous activities being acted in gathered code for execution. There are a few significant contrasts between NumPy exhibits and the standard Python successions:

i. NumPy exhibits have a fixed size at creation, not at all like Python records (which can develop progressively). Changing the size of and array will make another exhibit and erase the first.

ii. The components in a NumPy exhibit are totally required to be of similar information type, and in this way will be a similar size in memory. The special case: one can have varieties of (Python, including NumPy) objects, consequently taking into account varieties of various measured components.

iii. NumPy clusters encourage progressed scientific and different sorts of procedure on enormous quantities of information. Ordinarily, such tasks are executed more effectively and with less code than is conceivable utilizing Python's worked in successions. A developing plenty of logical and numerical Python-based bundles are utilizing NumPy clusters; however these normally bolster Python- arrangement input, they convert such contribution to NumPy exhibits preceding handling, and they frequently yield NumPy clusters. At the end of the day, so as to proficiently utilize a lot(maybe even a large portion) of the present logical/scientific Python-based programming, simply realizing how to utilize Python's worked in succession types is inadequate – one likewise has to realize how to utilize NumPy exhibits.

3.2.4 OpenCV: OpenCV [OpenCV] is an open-source computer vision library available from http://SourceForge.net/projects/opencvlibrary. OpenCV was designed for computational efficiency and having a high focus on real-time image detection. OpenCV is coded with optimized C and can take work with multicore processors. If we desire more automatic optimization using Intel architectures [Intel], you can buy Intel's Integrated Performance Primitives (IPP) libraries [IPP]. These consist of low-level routines in various algorithmic areas which are optimized. OpenCV automatically uses the IPP library, at runtime if that library is installed. 13 One of OpenCV's goals is to provide a simple-to-use computer vision infrastructure which helps people to build highly sophisticated vision applications fast. The OpenCV library, containing over 500 functions, spans many areas in vision. Because computer vision and machine learning often goes hand-in-hand, OpenCV also has a complete, general-purpose, Machine Learning Library (MLL). This sub library is focused on statistical pattern recognition and clustering. The MLL is very useful for the vision functions that are the basis of OpenCV's usefulness but is general enough to be used for any machine learning problem.

Computer vision is the transforming of data from a still, or video camera into either a representation or a new decision. All such transformations are performed to achieve a particular goal. A computer obtains a grid of numbers from a camera or from the disk, and that's that. Usually, there is no built-in pattern recognition or automatic control of focus and aperture, no cross-associations with years of experience. For the most part, vision systems are still fairly naïve.

The Origin of OpenCV came out of an Intel Research initiative meant to advance CPU-intensive applications. Toward this end, Intel launched various projects that included real-time ray tracing and also 3D display walls. One of the programmers working for Intel at the time was visiting universities. He noticed that a few top university groups, like the MIT Media Lab, used to have well-developed as well as internally open computer vision infrastructures—code that was passed from one student to another and which gave each subsequent student a valuable foundation while developing his own vision application. Instead of having to reinvent the basic functions from beginning, a new student may start by adding to that which came before.

OpenCV Structure and Content - OpenCV can be broadly structured into five primary components, four of which are shown in the figure. The CV component contains mainly the basic image processing and higher-level computer vision algorithms; MLL the machine learning library includes many statistical classifiers as well as clustering tools. HighGUI component contains I/O routines with functions for storing, loading video & images, while CXCore contains all the basic data structures and content.

Why OpenCV?

Specific - OpenCV was designed for image processing. Every function and data structure has been designed with an Image Processing application in mind. Meanwhile, Matlab, is quite generic. You can get almost everything in the world by means of toolboxes. It may be financial toolboxes or specialized DNA toolboxes.

Speedy - Matlab is just way too slow. Matlab itself was built upon Java. Also Java was built upon C. So when we run a Matlab program, our computer gets busy trying to interpret and compile all that complicated Matlab code. Then it is turned into Java, and finally executes the code. If we use C/C++, we don't waste all that time. We directly provide machine language code to the computer, and it gets executed. So ultimately we get more image processing, and not more interpreting. After doing some real time image processing with both Matlab and OpenCV, we usually got very low speeds, a maximum of about 4-5 frames being processed per second with Matlab. With OpenCV however, we get actual real time processing at around 30 frames being processed per second. Sure we pay the price for speed – a more cryptic language to deal with, but it's definitely worth it. We can do a lot more, like perform some really complex mathematics on images using C and still get away with good enough speeds for your application.

Efficient - Matlab uses just way too much system resources. With OpenCV, we can get away with as little as 10mb RAM for a real-time application. Although with today's computers, the RAM factor isn't a big thing to be worried about. However, our drowsiness detection system is to be used inside a car in a way that is non-intrusive and small; so a low processing requirement is vital. Thus, we can see how OpenCV is a better choice than Matlab for a real-time drowsiness detection system.

3.2.5 Dlib: Dlib is a modern C++ toolkit containing machine learning algorithms and tools for creating complex software in C++ to solve real world problems. It is used in both industry and academia in a wide range of domains including robotics, embedded devices, mobile phones, and large high performance computing environments. Dlib's open-source licensing allows you to use it in any application, free of charge. Dlib offers a wide range of functionality across a number of machine learning sectors, including classification and regression, numerical algorithms such as quadratic program solvers, an array of image processing tools, and diverse networking functionality, among many other facets. From a computer vision perspective, Dlib has a number of state-of-the-art implementations, including:

- Facial landmark detection
- Correlation tracking
- Deep metric learning

3.2.6 Imutils: A series of convenience functions to make basic image processing functions such as translation, rotation, resizing, skeletonization, displaying Matplotlib images, sorting contours, detecting edges, and much more easier with OpenCV and both Python 2.7 and Python 3.

3.2.7 Scipy: Scientists and researchers are likely to gather an enormous amount of information and data, which are scientific and technical, from their exploration, experimentation, and analysis. Dealing with such a huge amount of data becomes a hindrance to them. That is, calculating and computing large data manually is not an easy task. Hence, they make use of supercomputers and Data Science for the purpose of faster computing and accurate outcomes. Another simpler way to deal with scientific and technical computing of data is by making use of one of the Python libraries which is solely built for this purpose. It is referred to as Python SciPy**.** SciPy in Python is an open-source library used for solving mathematical, scientific, engineering, and technical problems. It allows users to manipulate the data and visualize the data using a wide range of high-level Python commands. SciPy is built on the Python NumPy extension. SciPy is also pronounced as "Sigh Pi." It provides many user-friendly and effective numerical functions for numerical integration and optimization.

The SciPy library supports integration, gradient optimization, special functions, ordinary differential equation solvers, parallel programming tools, and many more. We can say that SciPy implementation exists in every complex numerical computation.

History - Python was expanded in the 1990s to include an array type for numerical computing called numeric. This numeric package was replaced by NumPy (blend of Numeric and NumArray) in 2006. There was a growing number of extension module and developers were interested to create a complete environment for scientific and technical computing. Travis Oliphant, Eric Jones, and Pearu Peterson merged code they had written and called the new package SciPy. The newly created package provided a standard collection of common numerical operation on the top of NumPy.

Why use SciPy?

- SciPy contains varieties of sub packages which help to solve the most common issue related to Scientific Computation.
- SciPy package in Python is the most used Scientific library only second to GNU Scientific Library for C/C++ or MATLAB's.
- Easy to use and understand as well as fast computational power.
- It can operate on an array of NumPy library.

3.2.8 Play Sound:

- Play sound in Python using some of the most popular audio libraries.
- The playsound module contains only a single function named playsound().
- It requires one argument: the path to the file with the sound we have to play. It can be a local file, or a URL.
- There's an optional second argument, block, which is set to True by default. We can set it to False for making the function run asynchronously.
- It works with both WAV and MP3 files.
- pip install playsound

CHAPTER 4

4.1 PROPOSED METHODOLOGY

4.1.1 Block Diagram- Sleepiness of an individual can be estimated by the all- encompassing timeframe for which his/her eyes are in shut state. In our framework, essential consideration is given to the quicker recognition and preparing of information. The quantity of edges for which eyes are shut is observed. On the off chance that the quantity of edges surpasses a specific worth, at that point an admonition message is created on the presentation indicating that the driver is feeling sluggish. In our calculation, first the image is obtained by the webcam for handling. At that point we utilize the dataset to distinguish the countenances in every individual edge. On the off chance that no face is distinguished, at that point another edge is gained. In the event that a face is identified, at that point an area of enthusiasm for set apart inside the face. This locale of intrigue contains the eyes. Characterizing a district of intrigue essentially lessens the computational prerequisites of the framework. After that the eyes are distinguished from the district of enthusiasm by utilizing dataset Euclidian spaces point.

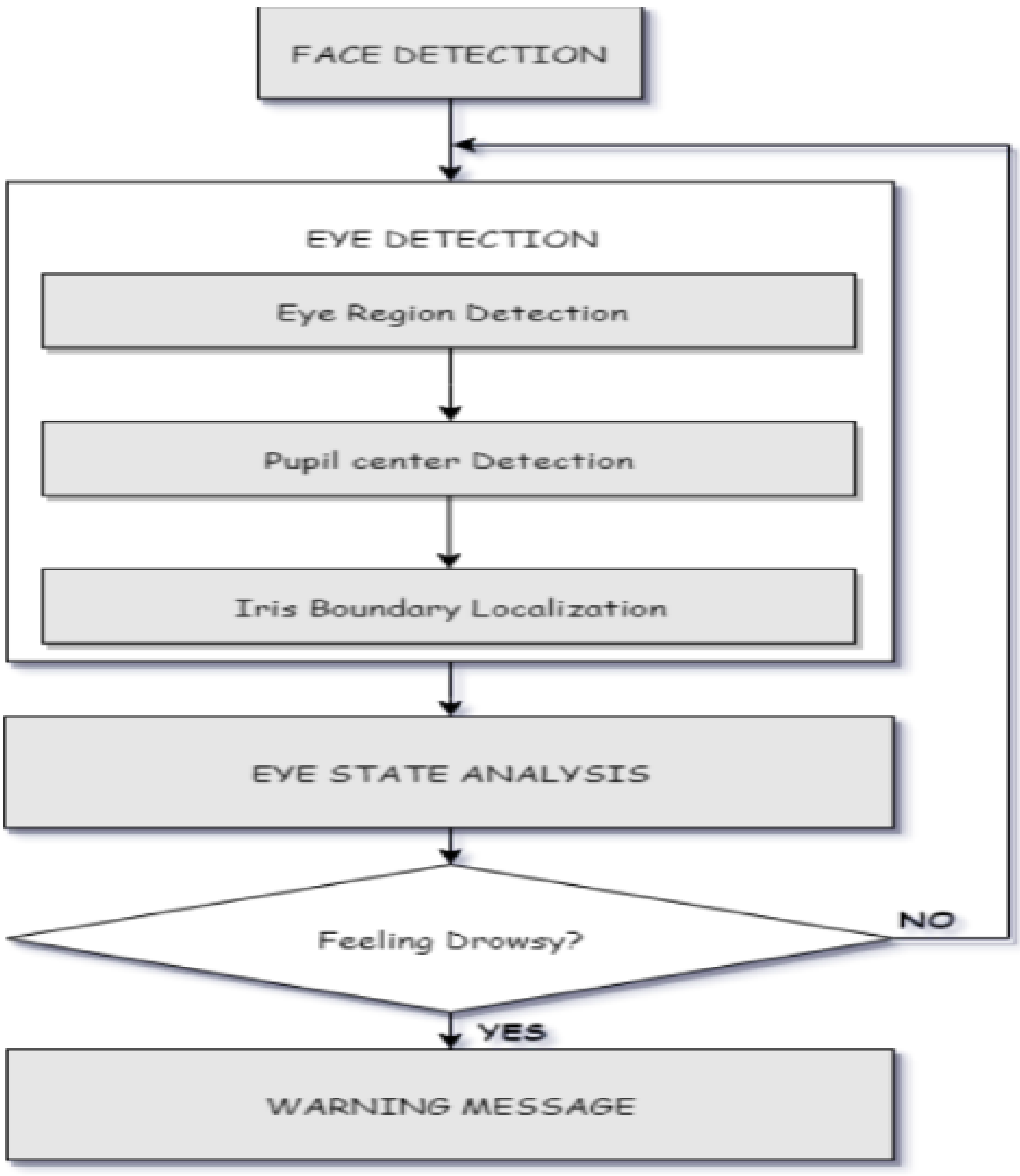


Figure 4.1 Block Diagram

4.1.2 Image Capture: Utilizing a web camera introduced inside the automobile we can get the picture of the driver. Despite the fact that the camera creates a video clip, we have to apply the developed algorithm on each edge of the video stream. In this the video is captured using webcam and each frames are extracted and are processed for face detection. Using image processing techniques on these images driver face is detected using webcam with periodic eye blinking, eye closing, yawning and head bending.

4.1.3 Dividing into Frames: We are dealing with real time situation where video is recorded and has to be processed. But the processing or application of algorithm can be done only on an image. Hence the captured video has to be divided into frames for analyzing.

4.1.4 Face Detection: In this stage we identify the area containing the essence of the driver. A predetermined calculation is for location of face in each edge. By face discovery we implies that finding the face in a casing or as such discovering area of facial characters through a kind of innovation with the utilization of PC. The casing might be any irregular edge. Just facial related structures or highlights are distinguished, and all others sorts of articles like structures, tree, bodies are overlooked. We realize that face is likewise a sort of item. So we can think about location of face as a specific instance of item discovery. In this sort of item kind of class recognition, we attempt to know where the articles in the intrigue picture are found and what is their size which may has a place with a specific class. Crafted by calculation that is made for face location is generally focused on finding the front side of the face. In any case, the calculations that are grown as of late spotlight on increasingly broad cases. For our case it might be face in the inclined position or some other bit of the appearances and furthermore it finds the chance of various countenances. Which implies the turn pivot concerning the current eyewitness from the reference of face in a specific? Or on the other hand regardless of whether there is vertical revolution plane then additionally it can understand the reason. In new kind of calculation, it is viewed as that the image or video is a variable which implies that diverse condition in them like tone complexity may change its fluctuation. The measure of light may likewise influence.

The facial landmark detector included in the dlib library is an implementation of the One Millisecond Face Alignment with an Ensemble of Regression Trees paper by Kazemi and Sullivan (2014).This method starts by using:

- **i.** A training set of labelled facial landmarks on an image. These images are manually labelled, specifying specific (x, y) -coordinates of regions surrounding each facial structure.
- **ii.** Priors, of more specifically, the probability on distance between pairs of input pixels. The pre-trained facial landmark detector inside the dlib library is used to estimate the location of 68 (x, y)-coordinates that map to facial structures on the face.

We can detect and access both the eye region by the following facial landmark index show below

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

4.1.5 Eye Detection: In the system we have used facial landmark prediction for eye detection. Facial landmarks are used to localize and represent salient regions of the face, such as:

• Eyes

•Eyebrows

• Nose

•Mouth

•Jawline

Facial landmarks have been successfully applied to face alignment, head pose estimation, face swapping, blink detection and much more. In the context of facial landmarks, our goal is detecting important facial structures on the face using shape prediction methods. Detecting facial landmarks is therefore a twostep process:

•Localize the face in the image.

•Detect the key facial structures on the face ROI.

Localize the face in the image: The face image is localized by Haar feature-based cascade classifiers which was discussed in the first step of our algorithm i.e. face detection. Detect the key facial structures on the face ROI: There are a variety of facial landmark detectors, but all methods essentially try to localize and label the following facial regions:

•Mouth

•Right eyebrow

•Left eyebrow

•Right eye

•Left eye

• Nose

Poor contrast of eyes generally creates a lot of problems in its detection. After successful detection of face eye needs to be detected for further processing. In our method eye is the decision parameter for finding the state of driver. Though detection of eye does not look complex, but the actual process is quite hectic. In this case it performs the detection of eye in the specified region with the use of feature detection.

4.1.6 Recognition of Eye State: The eye area can be estimated from optical flow, by sparse tracking or by frame-to-frame intensity differencing and adaptive thresholding. And Finally, a decision is made whether the eyes are or are not covered by eyelids. A different approach is to infer the state of the eye opening from a single image, as e.g. by correlation matching with open and closed eye templates, a heuristic horizontal or vertical image intensity projection over the eye region, a parametric model fitting to find the eyelids, or active shape models. A major drawback of the previous approaches is that they usually implicitly impose too strong requirements on the setup, in the sense of a relative face-camera pose (head orientation), image resolution, illumination, motion dynamics, etc. Especially the heuristic methods that use raw image intensity are likely to be very sensitive despite their real-time performance.

4.1.7 Eye State Determination: Finally, the decision for the eye state is made based on EAR calculated in the previous step. If the distance is zero or is close to zero, the eye state is classified as "closed" otherwise the eye state is identified as "open".

4.1.8 Digital Image Processing: Digital image processing is the utilization of PC calculations to perform image processing on digital images. In the field of digital sign processing, there are numerous favourable circumstances of digital image processing with contrast with simple image processing. There is a lot more extensive scope of calculations to be applied to the info information. It can maintain a strategic distance from issues like the development of commotion and sign distortion during processing. Since images are characterized more than two measurements, even applied on account of more measurements likewise, digital image processing might be displayed as multidimensional frameworks.

Digital image processing licenses the utilization of considerably more troublesome calculations. Digital image processing can offer both increasingly entangled execution at straightforward assignments, and the usage of strategies which would be unimaginable by simple methods. Digital image processing is the main pragmatic innovation for grouping, design acknowledgment, projection, include extraction and multi-scale signal investigation Some strategies which are utilized in digital image processing names as pixlation, head segments examination, direct sifting, anisotropic dispersion, wavelets, neural systems, free segment investigation, shrouded markov model, self- sorting out guides and fractional differential condition. There is important to talk about on Digital Image processing. since all through entire philosophy this method just is utilized.

An image might be characterized as a two- dimensional capacity, f(x,y), where x and y are spatial directions, and the abundancy of f at any pair of directions (x,y) is known as the power or dark degree of the picture by then. When x.y, and the sufficiency estimations of f are generally limited, discrete amounts, the picture is called as an advanced picture. Advanced picture handling alludes to preparing computerized pictures utilizing a computerized PC. A propelled picture is made out of a set number of parts, all of which has a particular zone and worth. These parts are implied as picture segments, picture segments, pels, and pixels. Pixel is the term utilized most broadly to signify the components of a computerized picture. Vision is the most developed of among all detects, so clearly pictures take an interest the absolute most imperative job in human recognition. Be that as it may, in contrast to people, who are constrained to the visual band of electromagnetic range, imaging machines spread nearly the whole electromagnetic range. It has run from gamma to radio waves. They can work likewise on pictures created by sources that people don't as a rule relate with pictures. These incorporate electron microscopy, ultrasound, and PC created pictures. Thus, it is effortlessly observed that the computerized picture preparing incorporates a huge, tremendous and wide differed field of uses.

4.1.9 Overall Scenario of Implementation: To defeat the difficult we thought of the arrangement executed as picture preparing. To perform picture handling, OpenCV and dlib open source libraries are utilized. Python is utilized as a language to actualize the thought. A Webcam camera is utilized to ceaselessly follow the facial milestone and development of eyes and lips of the driver. This task for the most part focuses on the tourist spots of lips and eyes of the driver. For discovery of sluggishness, tourist spots of eyes are followed constantly. Pictures are caught utilizing the camera at fix outline pace of 20fps. These pictures are passed to picture handling module which performs face milestone identification to identify interruption and tiredness of driver. In the event that the driver is seen as diverted, at that point a voice (sound) alert and is given and a message is shown on the screen. Following use cases are shrouded in this venture.

- **i.** If eyes of drivers are shut for a limit timeframe, then it is viewed as that driver is feeling drowsy and comparing sound caution is utilized to make the driver mindful.
- **ii.** If the mouth of driver stays open for the specific timeframe, then it is viewed as that driver is yawning and comparing recommendation are given to the driver to conquer sleepiness.
- **iii.** If driver don't keep eyes out and about then it is watched utilizing facial milestones, and the relating alert is utilized to make the driver mindful.

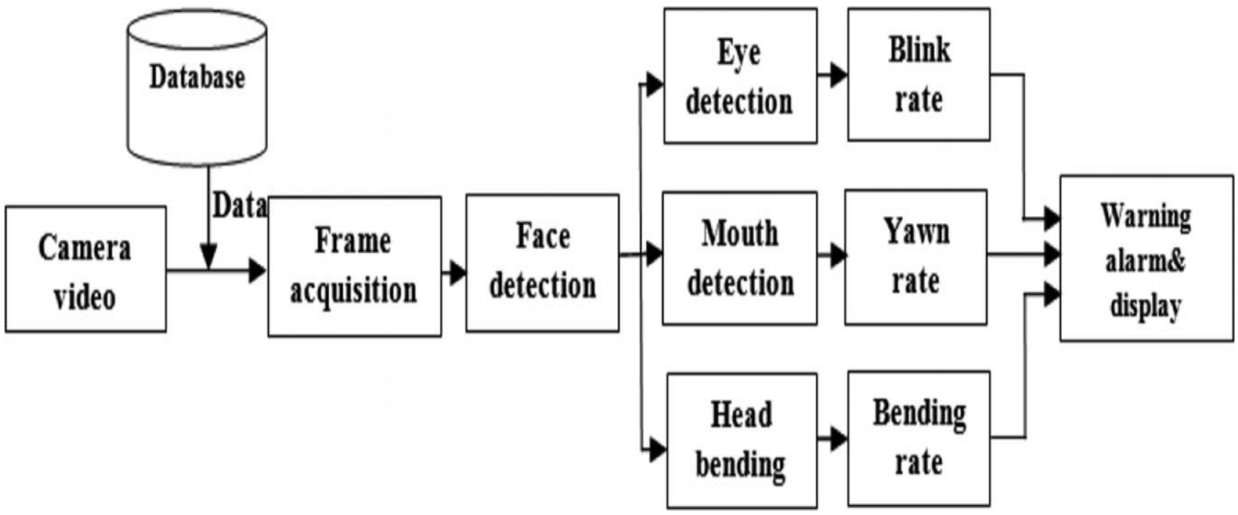


Figure 4.2 Flow Diagram of the System

4.2 OUR PROPOSED SYSTEM

The proposed method consist of a computer vision system that can automatically detect driver drowsiness in a real-time video stream and then ring an alarm and alert the driver when appears to be drowsy [9], [10].

4.2.1 Facial Landmark Detection: - Facial landmark is a computer-based function for automatically dealing with detecting distinctive features in human faces. In this we need to find different facial features like the regions of the eyes, mouth and head. To extract the facial landmarks of drivers, Dlib library was imported which uses a pre-trained face detector. The basic idea of this technique is to locate 68 specific points on face such as corners of the mouth, along the eyebrows, on the eyes, and so forth.

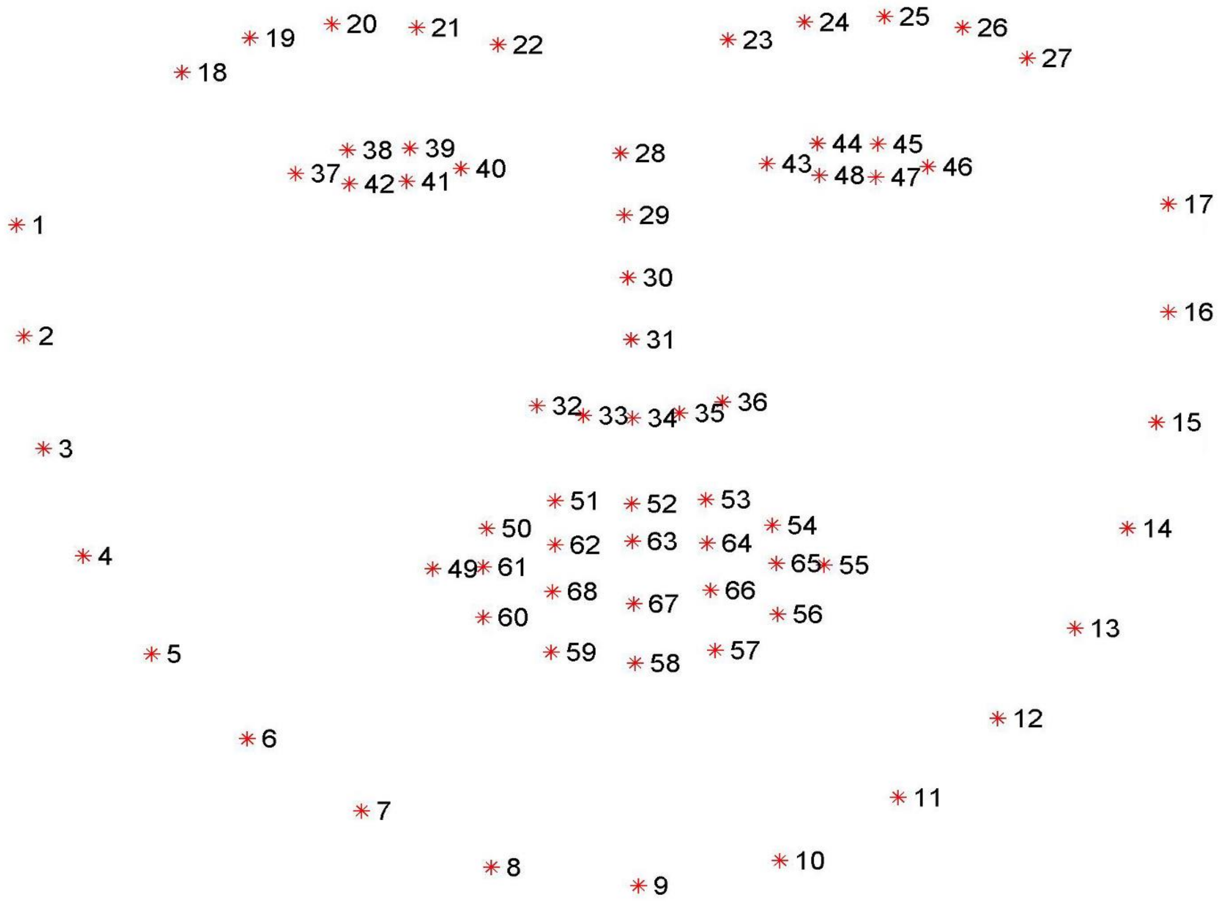


Figure 4.3 Facial Landmark Diagram

4.2.2 Eye Aspect Ratio Detection: - The eye aspect ratio [11]is an efficient tool to determine whether the eye is closed or not. In this method, each eye is represented by 6 points, starting at the left corner of the eye, and going clockwise around the remainder of the eye region (P1 to P6).

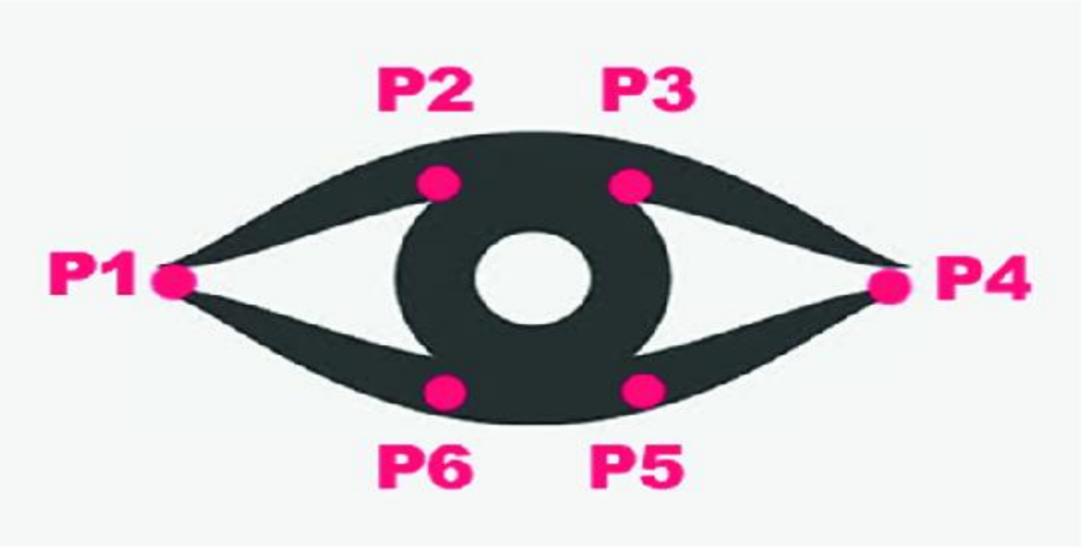


Figure 4.4 The Six facial landmark points associated with the eye

EAR is defined as the ratio of height and width of the eye and numerator denotes the height of the eye and the denominator denotes the width of the eye. The numerator calculates the distance between the upper eyelid and the lower eyelid. The denominator represents the horizontal distance of the eye.

$$\text{EAR} = \frac{(|\mathbf{p2}-\mathbf{p6}|+|\mathbf{p3}-\mathbf{p5}|)}{\mathbf{2}*|\mathbf{p1}-\mathbf{p4}|} \qquad \textbf{(4.1)}$$

After capturing the facial landmark points, EAR value computed by the server is now received to the driver and compared with the threshold value which was earlier set to be 0.25 as this value is taken from T. Soukupova and J. Cech,2016[13]. If the value is less than the threshold then the counter value is incremented, else the counter value is set back to zero. If the counter value reaches to three, an alarm is triggered.

4.2.3 Yawning Technique: - One of an important sign of fatigue is yawn. It is assume with a large vertical mouth opening compared. As compared to speaking, mouth is widely open in yawning process. Bhandari et al. detected yawn by face and then mouth tracking [8]. If M.A.R. was greater than 0.6, it would be declared as a yawning mouth. After yawn detection, the system alarms the driver. The MAR is calculated using the eight points bounding the inner lip portion of the faces detected by following the formulae –

$$\textbf{MAR} = \frac{|\mathbf{p2}-\mathbf{p8}|+|\mathbf{p3}-\mathbf{p7}|+|\mathbf{p4}-\mathbf{p6}|}{\mathbf{2}*|\mathbf{p1}-\mathbf{p5}|} \qquad \textbf{(4.2)}$$

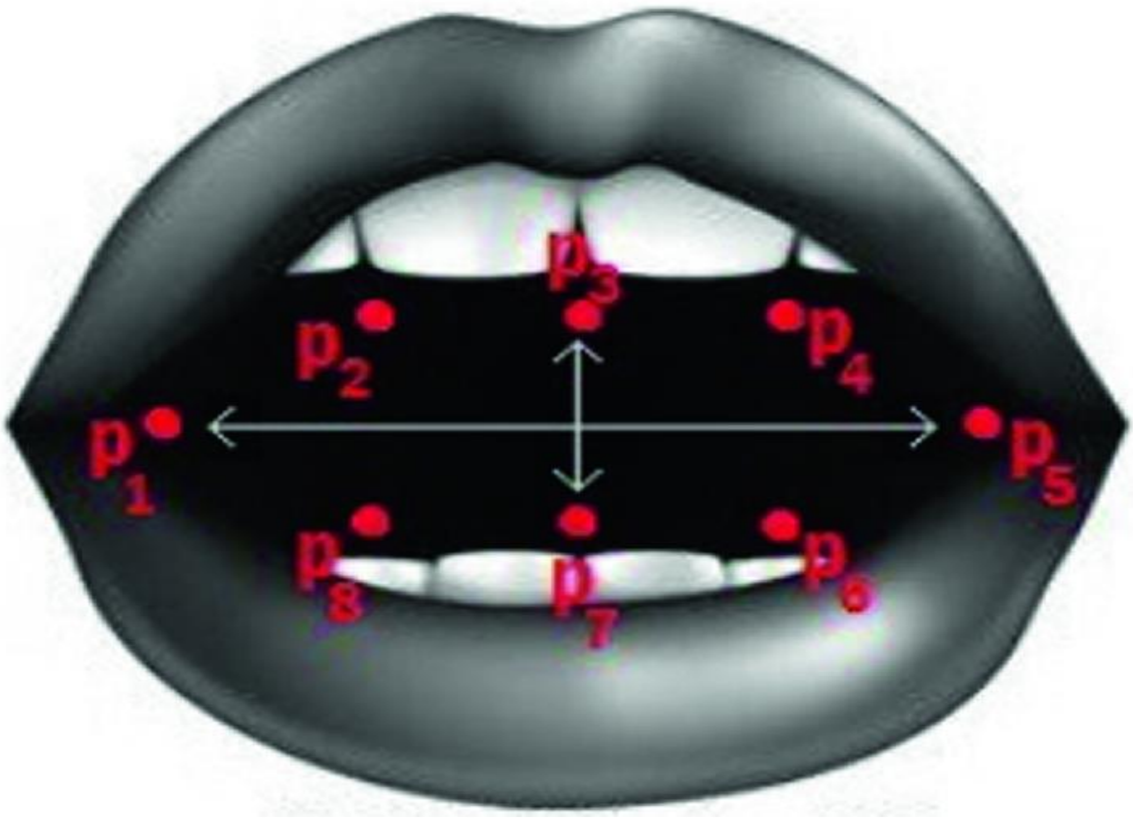


Figure 4.5 Facial landmark points associated with the mouth

Part	Landmark Points
Left Eye	[37-42]
Right Eye	[43-48]
Mouth	[49-68]

Table 4.1 Eye and Mouth Co-ordinates in facial landmark

Algorithm:

1. Start.
2. Take image as input from a camera.
3. Detect the face in the image and create a Region of Interest (ROI).
4. Detect the eyes from ROI.
5. Calculate the EAR value.
6. Compare EAR and threshold value.
7. Calculate MAR and compare with threshold value.
8. Either MAR > threshold or Boolean value turns True then Alarm is triggered.
9. Alert the driver by providing some sound.
10. Stop.

4.2.4 Source Code:

We’ll need the SciPy package so we can compute the Euclidean distance between facial landmarks # # points in the eye aspect ratio calculation.

#We’ll also need the imutils package, my series of computer vision and image processing functions to # make working with OpenCV easier.

#We’ll also import the Thread class so we can play our alarm in a separate thread from the main thread #to ensure our script doesn’t pause execution while the alarm sounds.

#In order to actually play our WAV/MP3 alarm, we need the playsound library, a pure Python, cross-# #platform implementation for playing simple sounds.

#To detect and localize facial landmarks we’ll need the dlib library

```
from scipy.spatial import distance as dist

from imutils.video import VideoStream

from imutils import face_utils

from threading import Thread

import numpy as np

import playsound

import argparse

import imutils

import time

import dlib

import cv2

import winsound

################################################################################
```

#Next, we need to define our sound_alarm function which accepts a path to an audio file residing on #disk and then plays the file:

```
def sound_alarm(E:\music\ab.mp3):

    # play an alarm sound

    playsound.playsound(E:\music\ab.mp3)

##########################################################################
```

#We also need to define the eye_aspect_ratio function which is used to compute the ratio of distances #between the vertical eye landmarks and the distances between the horizontal eye landmarks.

#The return value of the eye aspect ratio will be approximately constant when the eye is open. The #value will then rapid decrease towards zero during a blink.

#If the eye is closed, the eye aspect ratio will again remain approximately constant, but will be much #smaller than the ratio when the eye is open.

#Next, let's parse our command line arguments

```
def eye_aspect_ratio(eye):

    # compute the euclidean distances between the two sets of

    # vertical eye landmarks (x, y)-coordinates

    A = dist.euclidean(eye[1], eye[5])

    B = dist.euclidean(eye[2], eye[4])

    # compute the euclidean distance between the horizontal

    # eye landmark (x, y)-coordinates

    C = dist.euclidean(eye[0], eye[3])

    # compute the eye aspect ratio

    ear = (A + B) / (2.0 * C)
```

```
    # return the eye aspect ratio

    return ear

def get_landmarks(im):

    rects = detector(im, 1)

    if len(rects) > 1:

        return "error"

    if len(rects) == 0:

        return "error"

    return np.matrix([[p.x, p.y] for p in predictor(im, rects[0]).parts()])

# live landmark detection

def annotate_landmarks(im, landmarks):

    im = im.copy()

    for idx, point in enumerate(landmarks):

        pos = (point[0, 0], point[0, 1])

        cv2.putText(im, str(idx), pos,

                    fontFace=cv2.FONT_HERSHEY_SCRIPT_SIMPLEX,

                    fontScale=0.4,

                    color=(0, 0, 255))

        cv2.circle(im, pos, 3, color=(0, 255, 255))

    return im

def mouth_aspect_ratio(mouth):
```

```
        # compute the euclidean distances between the two sets of
        # vertical mouth landmarks (x, y)-coordinates
        A = dist.euclidean(mouth[2], mouth[9]) # 51, 59
        B = dist.euclidean(mouth[4], mouth[7]) # 53, 57
        # compute the euclidean distance between the horizontal
        # mouth landmark (x, y)-coordinates
        C = dist.euclidean(mouth[0], mouth[6]) # 49, 55
        # compute the mouth aspect ratio
        mar = (A + B) / (2.0 * C)
        # return the mouth aspect ratio
        return mar
def mouth_open(image):
    landmarks = get_landmarks(image)
       if landmarks == "error":
        return image, 0
        image_with_landmarks = annotate_landmarks(image, landmarks)
    mouth_distance = mouth_aspect_ratio(landmarks)
         return image_with_landmarks,mouth_distance
#################################################################################
#Now that our command line arguments have been parsed, we need to define a few important variables:
# (i) EYE_AR_THRESH = 0.25
```

(ii) EYE_AR_CONSEC_FRAMES = 16

#Similarly we define variable for yawning:

#(i)yawns = 0

#(ii)yawn_thresh=4

#(iii)yawn_status = False

#The dlib library ships with a Histogram of Oriented Gradients-based face detector along with a facial #landmark predictor — we instantiate both of these in the following code block.

#Therefore, to extract the eye regions from a set of facial landmarks, we simply need to know the #correct array slice indexes.

#Using these indexes, we'll easily be able to extract the eye regions via an array slice.
#dlib's face detector to find and locate the face(s) in the image.

#For each of the detected faces, we apply dlib's facial landmark detector and convert #the result #to a NumPy array.

#Using NumPy array slicing we can extract the (x, y)-coordinates of the left and right eye, #respectively.

#Given the (x, y)-coordinates for both eyes, we then compute their eye aspect ratios.

two constants, one for the eye aspect ratio to indicate blink and then a second constant for the number of consecutive

frames the eye must be below the threshold for to set off the alarm

EYE_AR_THRESH = 0.25

EYE_AR_CONSEC_FRAMES = 16

similarly do the same for yawn counts

yawns = 0

yawn_thresh=4

yawn_status = False

```
# initialize the frame counter as well as a boolean used to

# indicate if the alarm is going off

COUNTER = 0

ALARM_ON = False

# initialize dlib's face detector (HOG-based) and then create

# the facial landmark predictor

print("[INFO] loading facial landmark predictor...")

detector = dlib.get_frontal_face_detector()

predictor = dlib.shape_predictor('shape_predictor_68_face_landmarks.dat')

# grab the indexes of the facial landmarks for the left and

# right eye, respectively

(lStart, lEnd) = face_utils.FACIAL_LANDMARKS_IDXS["left_eye"]

(rStart, rEnd) = face_utils.FACIAL_LANDMARKS_IDXS["right_eye"]

# start the video stream thread

print("[INFO] starting video stream thread...")

cap = cv2.VideoCapture(0)

# loop over frames from the video stream

while True:

  # grab the frame from the threaded video file stream, resize

  # it, and convert it to grayscale

  # channels)
```

```
ret, frame = cap.read()

frame = cv2.resize(frame, (0,0), fx=1.20, fy=1.20)

gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

# detect faces in the grayscale frame

rects = detector(gray, 0)

image_landmarks, mouth_distance = mouth_open(frame)

prev_yawn_status = yawn_status

# loop over the face detections

for rect in rects:

    # determine the facial landmarks for the face region, then

    # convert the facial landmark (x, y)-coordinates to a NumPy array

    shape = predictor(gray, rect)

    shape = face_utils.shape_to_np(shape)

    (mStart, mEnd) = (49, 68)

    # extract the left and right eye coordinates, then use the

    # coordinates to compute the eye aspect ratio for both eyes

    leftEye = shape[lStart:lEnd]

    rightEye = shape[rStart:rEnd]

    leftEAR = eye_aspect_ratio(leftEye)

    rightEAR = eye_aspect_ratio(rightEye)
```

```
# extract the mouth coordinates, then use the
# coordinates to compute the mouth aspect ratio
mouth = shape[mStart:mEnd]
mar = mouth_aspect_ratio(mouth)
# average the eye aspect ratio together for both eyes
ear = (leftEAR + rightEAR) / 2.0
# compute the convex hull for the left and right eye, then
# visualize each of the eyes
#EAR
leftEyeHull = cv2.convexHull(leftEye)
rightEyeHull = cv2.convexHull(rightEye)
cv2.drawContours(frame, [leftEyeHull], -1, (0, 255, 0), 1)
cv2.drawContours(frame, [rightEyeHull], -1, (0, 255, 0), 1)
# compute the convex hull for the mouth, then
# visualize the mouth
mouthHull = cv2.convexHull(mouth)
cv2.drawContours(frame, [mouthHull], -1, (0, 255, 0), 1)
cv2.putText(frame, "MAR: {:.2f}".format(mar), (30, 30),
                cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)


#----------------------------------------------------------------------------------------------------------
```

```
#MAR:
        if mar > 0.6:
            yawn_status = True
            output_text = " Yawn Count: " + str(yawns + 1)
            cv2.putText(frame, output_text, (0,350),cv2.FONT_HERSHEY_SIMPLEX, 1,(0,255,127),2)
        else:
            yawn_status = False
        if prev_yawn_status == True and yawn_status == False:
            yawns += 1
#------------------------------------------------------------------------------------------------------
        # check to see if the eye aspect ratio is below the blink
        # threshold, and if so, increment the blink frame counter
        if ear < EYE_AR_THRESH:
            COUNTER += 1
            # if the eyes were closed for a sufficient number of
            # then sound the alarm
            if COUNTER >= EYE_AR_CONSEC_FRAMES:
                # if the alarm is not on, turn it on
                if not ALARM_ON:
                    ALARM_ON = True
                    winsound.PlaySound("alarm.wav", winsound.SND_ASYNC | winsound.SND_ALIAS )
```

```
                # check to see if an alarm file was supplied,

                # and if so, start a thread to have the alarm

                # sound played in the background

            # draw an alarm on the frame

            cv2.putText(frame, "DROWSINESS ALERT!", (10, 30),cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

    # otherwise, the eye aspect ratio is not below the blink

    # threshold, so reset the counter and alarm

    else:

        COUNTER = 0

        ALARM_ON = False

    # do the same for yawns

    if yawns>=yawn_thresh:

        cv2.putText(frame, "'Please stop the Vehicle & take a Fresh Air'", (0,150),cv2.FONT_HERSHEY_COMPLEX, 1,(0,0,255),2)

    # draw the computed eye aspect ratio on the frame to help

    # with debugging and setting the correct eye aspect ratio

    # thresholds and frame counters

    cv2.putText(frame, "EAR: {:.2f}".format(ear), (300, 30),

        cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

  # show the frame

  cv2.imshow('Live Landmarks', image_landmarks )
```

```
    cv2.imshow("Frame", frame)

    key = cv2.waitKey(1) & 0xFF

    # if the `q` key was pressed, break from the loop and exit

    if key == ord("q"):

        break

# clean the window

cap.release()

cv2.destroyAllWindows()
```

4.2.5 Results:

I. Without Glass -

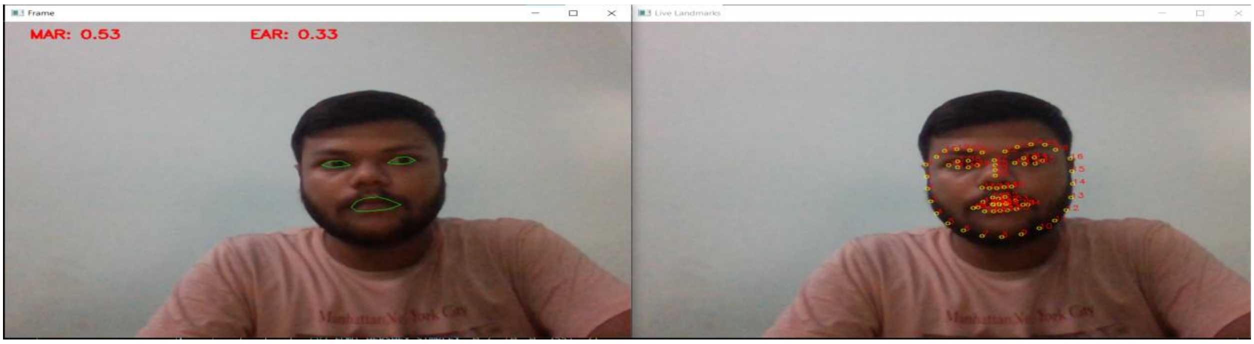


Figure 4.6 NO alert

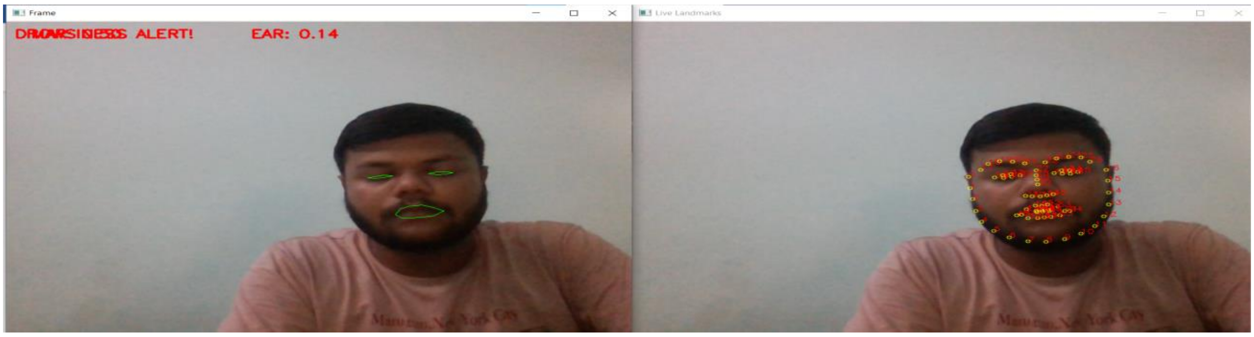


Figure 4.7 Alert due to Eyes are closing

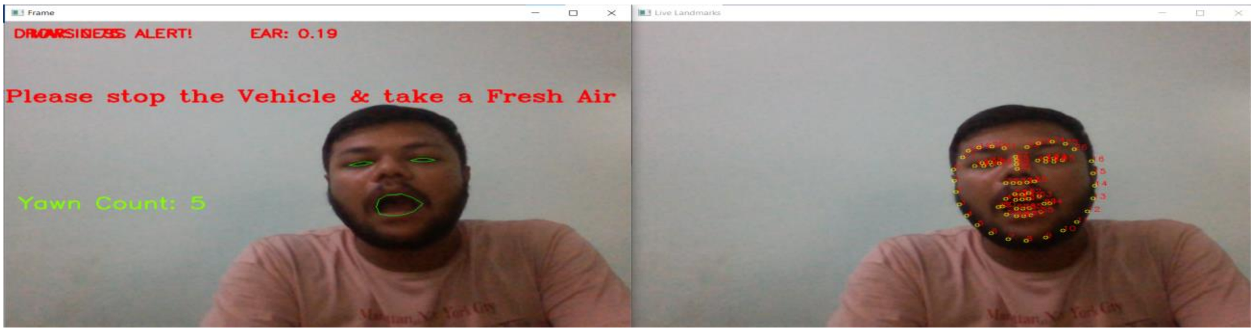


Figure 4.8 Alert due to Yawning

II. With Glass –

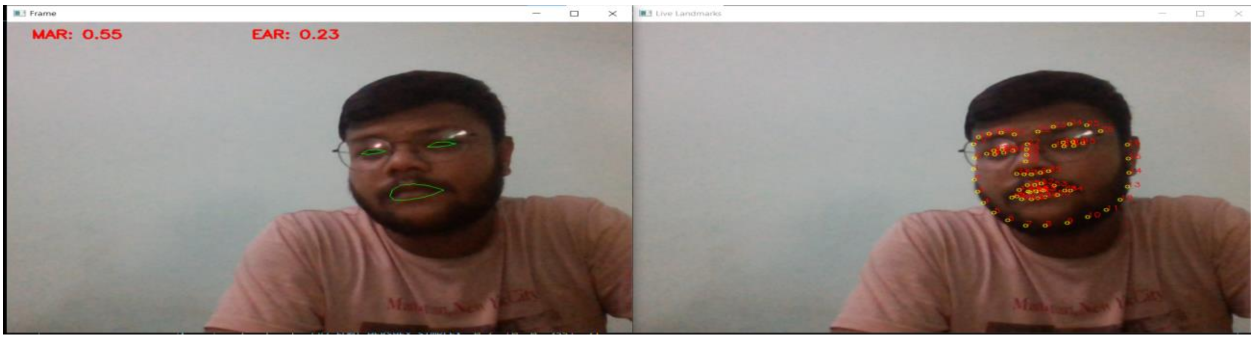


Figure 4.9 NO alert

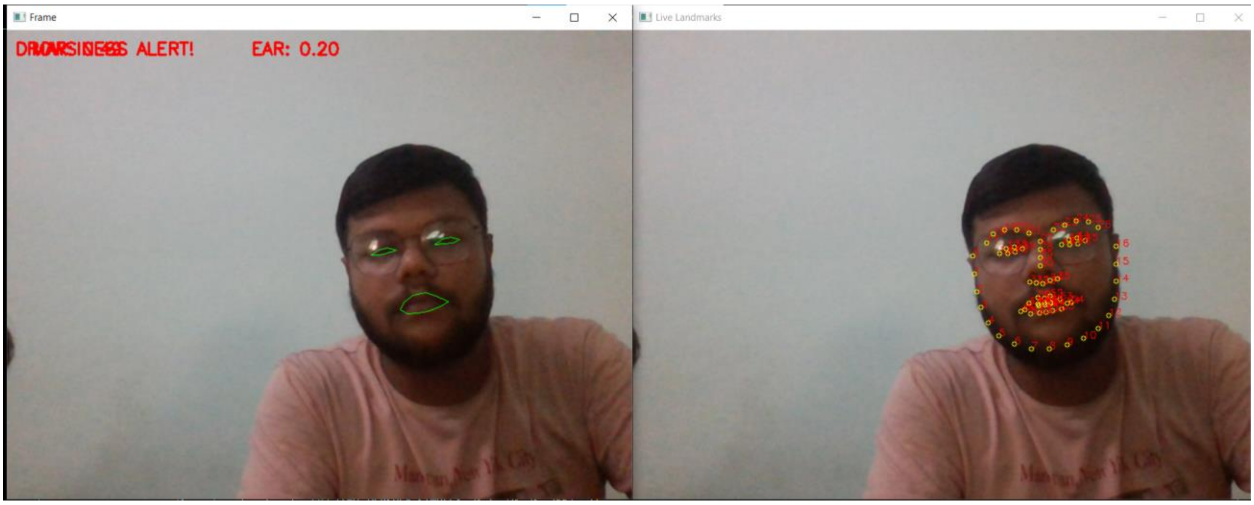


Figure 4.10 Alert due to Eyes are closing

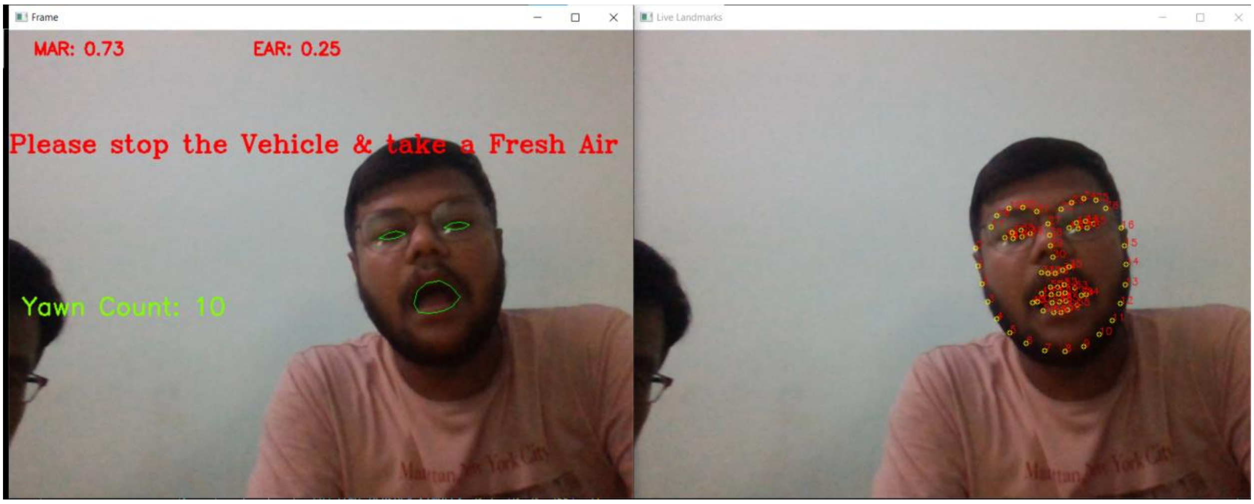
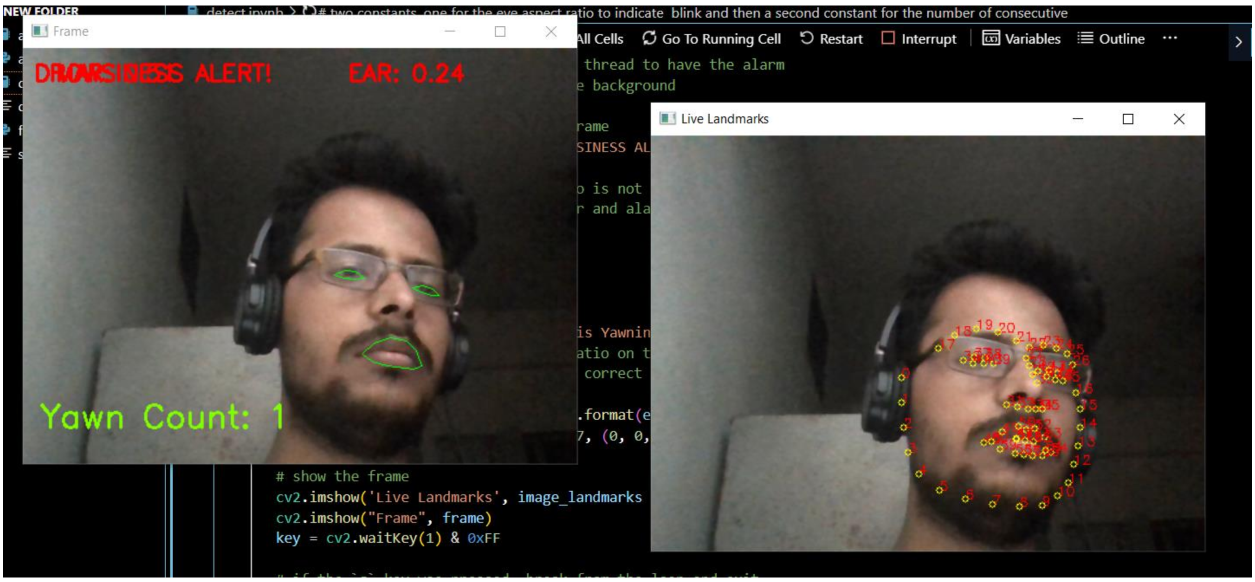


Figure 4.11 Alert due to Yawning



4.2.6 Research Outcome:

Driver's drowsiness is the prime aspect of grave street mishaps. So as to fulfil the client ever expanding interest, the degree of subjective weight on drivers is likewise expanded. Driver with recklessness level in understanding, detecting, and administrating the vehicle, accordingly represents a genuine risk to their own lives and other's lives. For this purpose, adapting procedures that observe the driver's degree of sleepiness and heeding the driver of any unreliable driving circumstances is crucial.

The major aim of this project is to develop a drowsiness detection system by monitoring the eyes; it is believed that the symptoms of driver fatigue can be detected early enough to avoid a car accident. In such a case when drowsiness is detected, a warning signal is issued to alert the driver. This detection system provides a noncontact technique for judging different levels of driver alertness and facilitates early detection of a decline in alertness during driving. In such a case when fatigue is detected, a warning signal is issued to alert the driver.

Various methods have been utilized to detect drowsiness. Some of these methods give high accuracy. Most of these methods used computer vision algorithm for detecting face from photo or videos, detecting gesture of closing eyes and detecting mouth. Computer vision algorithm gives high accuracy. After detecting face, eyes and mouth various algorithms are used on this to detect drowsiness and alert the driver. The driver gets alert when he feels drowsy by taking continuous yawning or eye is shutting down rapidly.

4.3 DRIVER DROWSINESS DETECTION APPLICATION DESIGN AND ITS ISSUES:

In this application, we watch the Driver Drowsiness ID, which is a vehicle prosperity development which foresees setbacks when the driver is getting worn out. Various assessments have prescribed that around 20% of all road incidents are exhaustion related, up to half on explicit boulevards. Driver fatigue is an essential factor in a large number of vehicle incidents. Driver carelessness might be the eventual outcome of nonattendance of sharpness when driving in light of driver drowsiness and redirection. In perspective on the getting of video from the camera, that is before the driver performs ceaseless planning of a moving toward video stream to deduce the driver's level of shortcoming in case the drowsiness is evaluated, by then it will give the alert by identifying the eyes.

The main explanation behind this concept was to use the retinal reflection as an approach to managing to find the eyes on the face, and starting their forward, using the nonattendance of this reflection as a framework for perceiving when the eyes closed. Applying the computation on the consecutive video housings may help in the figuring of the eye end period. The eye end period for sluggish drivers is more extended than standard flashing. So we will alert the driver when the eye closed distinguished. Nowadays, a consistently expanding number of reasons for a living require a whole deal obsession. Drivers must watch out for the road so that they can react to unforeseen events right away. Driver exhaustion, much of the time, transforms into a quick purpose behind some vehicle crashes. Therefore, there is a need to develop the structures that will perceive and tell a driver of her/him horrendous psycho-physical condition, which could inside and out reduce the amount of exhaustion related minor collision.

Nevertheless, the progression of such structures experiences various difficulties related to brisk and suitable affirmation of a driver's exhaustion symptoms. One of the specific possible results to complete driver laziness area structures is to use the vision-based system. This article presents the at present used driver drowsiness area structures. The particular pieces of using the vision structure to perceive driver lethargy discussed. A couple of assessments have conveyed various evaluations of the level of absence of rest as it relates to road incidents. Moreover, driver preoccupation or carelessness is another fundamental issue for safe driving.

4.4 Conclusions & Future Work:

Every year thousands of people died in road accidents. Mostly, the accident occurs due to the drowsiness of the driver. Drowsiness the of driver is a state in which the person is neither fully alert nor completely in sleep mode it is in between state. For reducing the frequency of road accidents, effective steps should be taken to reduce driver drowsiness. Such causes can be avoided by exploiting the advanced technology. The progress in the field of image processing and computer vision made it possible to detect the drowsiness of the driver by monitoring drowsiness visual symptoms. A reasonable number of road accidents are preventable with the use of computer vision-based drowsiness detection.

A real time driver drowsiness monitoring system has been proposed based on visual behavior. Here the features like mouth aspect ratio, eye closure ratio, eye aspect ratio and facial landmark detection is done. The paper is a comprehensive review on various methods to detect drowsiness with the focus on computer vision-based detection. It is intended that computer vision-based detection of the hybrid symptoms eliminate the drawbacks of previously used techniques. The proposed system is more reliable and dependable for driver drowsiness detection. It is non-intrusive in its nature. A reasonable number of road accidents are preventable with the use of computer vision-based drowsiness detection.

An accurate algorithm in real time eye tracking system has been challenging problem for computer vision. To prevent such dangerous and harmful situations, a real-time driver monitoring system is implemented using OpenCV, where the aspect ratios of eye is generated. With EAR i.e., Eye Aspect Ratio 0.25, the most common value for EAR in studies, the results show that the alarm is generated for the blinks. In this way an alarm can prevent such fatigue accidents and may save lives of the people. The facial landmarks are continuously scanned and captured using camera. These collected landmark frames are given to EAR algorithm to read the eye aspect ratio and to MAR algorithm to read the Mouth Aspect ratio for drowsiness detection. The proposed method sets a threshold and predicts that the driver is drowsy when the EAR ratio goes below the threshold or when the eye is kept close for a long duration and when the MAR ratio goes beyond the fixed threshold. Then the system instantly alerts the driver with the help of a sound system. The driver gets alerted when he feels drowsy.

The algorithms used in our proposed for face detection and recognition are efficient and of high accuracy. The main concept is calculating eye aspect ratio which is considered to be more accurate and reliable technique. And we use Mouth aspect ratio to provide more accuracy. With either of two enable to trigger the alarm the drowsiness is detected. Our proposed system can be used to monitor the driver's state and alert the driver, thereby reducing the number of road accident.

Our model is designed for detection of drowsy state of eye and give and alert signal or warning in the form of audio alarm. But the response of driver after being warned may not be enough to stop causing the accident meaning that if the driver is slow in responding towards the warning signal, then accident may occur. Hence to avoid this we can design and fit a motor driven system and synchronize it with the warning signal so that the vehicle will slow down after getting the warning signal automatically We have developed a system where we can detect the state of the eye and give warning signal. Now this is an adjustable mechanism which drives through the state of the driver.

4.5 REFERENCES :

1. Peden, M., Toroyan, T., Krug, E., Iaych, K.: The status of global road safety: the agenda for sustainable development encourages urgent action. J. Australas. Coll. Road Saf. 27, 37 (2016 Google Scholar)

2. Arvind, P.D., Jivaji, M.J., Romi, K., Kamble, P.: Accident informer and prevention system. Int. J. Eng. Sci. 7, 4772 (2017) Google Scholar

3. Ahmed, R., Emon, K.E.K., Hossain, M.F.: Robust driver fatigue recognition using image processing. In: 2014 International Conference on Informatics, Electronics & Vision (ICIEV), pp. 1–6. IEEE (2014)Google Scholar

4. Saini, V., Saini, R.: Driver drowsiness detection system and techniques: a review. Int. J. Comput. Sci. Inf. Technol. 5, 4245–4249 (2014)Google Scholar

5. Cona, F., Pizza, F., Provini, F., Magosso, E.: An improved algorithm for the automatic detection and characterization of slow eye movements. Med. Eng. Phys. 36, 954–961 (2014)CrossRef Google Scholar

6. Ahmad, R., Borole, J.: Drowsy driver identification using eye blink detection. IJISET-Int. J. Comput. Sci. Inf. Technol. 6, 270–274 (2015) Google Scholar

7. Yan, J.-J., Kuo, H.-H., Lin, Y.-F., Liao, T.-L.: Real-time driver drowsiness detection system based on PERCLOS and grayscale image processing. In: 2016 International Symposium on Computer, Consumer and Control (IS3C), pp. 243–246. IEEE (2016)Google Scholar

8. Bhandari, G., Durge, A., Bidwai, A., Aware, U.: Yawning analysis for driver drowsiness detection. Int. J. Eng. Res. Technol. 3, 502–505 (2014)Google Scholar

9. H. Dinh, E. Jovanov, R. Adhami Eye blink detection using intensity vertical projection International Multi-Conference on Engineering and Technological Innovation, IMETI (2012)Google Scholar

10. W.H. Lee, E.C. Lee, K.E. Park Blink detection robust to various facial poses J. Neurosci. Methods (2010)

Google Scholar

11. Soukupová, T., Cech, J.: Eye blink detection using facial landmarks. In: 21st Computer Vision Winter Workshop, Rimske Toplice, Slovenia (2016) Google Scholar

12. Sukrit Mehta, Sharad Dadhich, Sahil Gumber, Arpita Jadhav Bhatt;,Real-Time Driver Drowsiness Detection System Using Eye Aspect Ratio and Eye Closure Ratio Delivery
13. T. Soukupova and J. Cech, “Real-time eye blink detection using facial landmarks,” Computer Vision Winter Workshop (CVWW), 2016.
14. Liu C.C., Hosking S.G., Lenné M.G. Predicting driver drowsiness using vehicle measures: Recent insights and future challenges. *J. Saf. Res.* 2009;40:239–245. [PubMed] [Google Scholar]

15. Brodbeck V., Kuhn A., von Wegner F., Morzelewski A., Tagliazucchi E., Borisov S., Michel C.M., Laufs H. EEG microstates of wakefulness and NREM sleep. *NeuroImage.* 2012;62:2129–2139. [PubMed] [Google Scholar]
16. *Drowsy Driving and Automobile Crashes.* National Center on Sleep Disorder Research and the National Highway Traffic Safety Administration; Howe, TX, USA: 1998. [Google Scholar]
17. Hu S., Zheng G. Driver drowsiness detection with eyelid related parameters by support vector machine. *Exp. Syst. Appl.* 2009;36:7651–7658. [Google Scholar]
18. Otmani S., Pebayle T., Roge J., Muzet A. Effect of driving duration and partial sleep deprivation on subsequent alertness and performance of car drivers. *Physiol. Behav.* 2005;84:715–724. [PubMed] [Google Scholar]
19. Portouli E., Bekiaris E., Papakostopoulos V., Maglaveras N. On-road experiment for collecting driving behavioural data of sleepy drivers. *Somnology.* 2007;11:259–267. [Google Scholar]
20. Sommer D., Golz M., Trutschel U., Edwards D. *Agents and Artificial Intelligence.* Vol. 67. Springer; Berlin, Germany: 2010. Biosignal based discrimination between slight and strong driver hypovigilance by support-vector machines; pp. 177–187. [Google Scholar]
21. Ingre M., ÅKerstedt T., Peters B., Anund A., Kecklund G. Subjective sleepiness, simulated driving performance and blink duration: Examining individual differences. *J. Sleep Res.* 2006;15:47–53. [PubMed] [Google Scholar]
22. Liu C.C., Hosking S.G., Lenné M.G. Predicting driver drowsiness using vehicle measures: Recent insights and future challenges. *J. Saf. Res.* 2009;40:239–245. [PubMed] [Google Scholar]
23. Fairclough S.H., Graham R. Impairment of driving performance caused by sleep deprivation or alcohol: A comparative study. *J. Hum. Factors Ergon.* 1999;41:118–128. [PubMed] [Google Scholar]
24. Ruijia F., Guangyuan Z., Bo C. An on-Board System for Detecting Driver Drowsiness Based on Multi-Sensor Data Fusion Using Dempster-Shafer Theory. Proceedings of the International Conference on Networking, Sensing and Control; Okayama, Japan. 26–29 March 2009; pp. 897–902. [Google Scholar]
25. Thiffault P., Bergeron J. Monotony of road environment and driver fatigue: A simulator study. *Accid. Anal. Prevent.* 2003;35:381–391. [PubMed] [Google Scholar]
26. Vural E. Sabanci University; Istanbul, Turkey: 2009. Video Based Detection of Driver Fatigue. Ph.D. Thesis, [Google Scholar]
27. Lal S.K.L., Craig A. A critical review of the psychophysiology of driver fatigue. *Biol. Psychol.* 2001;55:173–194. [PubMed] [Google Scholar]
28. Miyaji M., Kawanaka H., Oguri K. Driver's Cognitive Distraction Detection Using Physiological Features by the Adaboost. Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems; St. Louis, MO, USA. 3–7 October 2009; pp. 1–6. [Google Scholar]
29. Patel M., Lal S.K.L., Kavanagh D., Rossiter P. Applying neural network analysis on heart rate variability data to assess driver fatigue. *Exp. Syst. Appl.* 2011;38:7235–7242. [Google Scholar]
30. Akin M., Kurt M., Sezgin N., Bayram M. Estimating vigilance level by using EEG and EMG signals. *Neural Comput. Appl.* 2008;17:227–236. [Google Scholar]
31. Michail E., Kokonozi A., Chouvarda I., Maglaveras N. EEG and HRV Markers of Sleepiness and Loss of Control during Car Driving. Proceedings of the 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society; Vancouver, BC, Canada. 20–25 August 2008; pp. 2566–2569. [PubMed] [Google Scholar]

32. McKinley R.A., McIntire L.K., Schmidt R., Repperger D.W., Caldwell J.A. Evaluation of eye metrics as a detector of fatigue. *Hum. Factors.* 2011;53:403–414. [PubMed] [Google Scholar]
33. Seeingmachines Driver State Sensor. Available online: http://www.seeingmachines.com/product/dss/ (accessed on 21 November 2012).
34. Lexus L.X. Driver Monitoring System. Available online: http://www.lexus.eu/range/ls/key-features/safety/safety-driver-monitoring-system.aspx (accessed on 21 November 2012).
35. Smith P., Shah M., Vitoria L.N. Determining driver visual attention with one camera. *IEEE Trans. Intell. Transport. Syst.* 2003;4:205–218. [Google Scholar]
36. Murphy-Chutorian E., Trivedi M.M. Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness. *IEEE Trans. Intell. Transp. Syst.* 2010;11:300–311. [Google Scholar]
37. Blana E., Golias J. Differences between vehicle lateral displacement on the road and in a fixed-base simulator. *Hum. Factors.* 2002;44:303–313. [PubMed] [Google Scholar]
38. Engström J., Johansson E., Östlund J. Effects of visual and cognitive load in real and simulated motorway driving. *Transport. Res. Traffic Psychol. Behav.* 2005;8:97–120. [Google Scholar]
39. Lawrence B., Stephen P., Howarth H. *An Evaluation of Emerging Driver Fatigue Detection Measures and Technologies.* Volpe National Transportation Systems Center Cambridge; Cambridge, UK: 2009. [Google Scholar]
40. Cheng B., Zhang W., Lin Y., Feng R., Zhang X. Driver drowsiness detection based on multisource information. *Hum. Factors Ergon. Manuf. Serv. Indust.* 2012;22:450–467. [Google Scholar]