Noname manuscript No.
(will be inserted by the editor)

A Novel Approach to Enhance Safety on Drowsy Driving in Self-Driving Car

Md. Motaharul Islam[1] · Ibna Kowsar[2] · Mashfiq Shahriar Zaman[2] · Md. Fahmidur Rahman Sakib[2] · Nazmus Saquib[2] · Syed Md. Shamsul Alam[2]

Received: date / Accepted: date

Abstract Drowsy driving centric accidents are increasing at a frightening rate. Needless to say that the state-of-the-art technologies only have competencies in detecting drowsiness and alerting the drowsy driver. Existing methods have some remarkable hindrances in the domain of handling the distressed situation. Therefore these methodologies are ineffective to take additional safety measures if the driver is not proficient enough to operate the vehicle even though an alarm is given. Consequently, after evaluating the existing methodologies and the growth of autonomous vehicles, we have proposed an innovative approach that detects driver drowsiness in real-time. Our suggested model can locate a nearest available safe parking space and reach the parking location after initiating the autonomous driving mode to ensure safety. The proposed methodology has achieved an accuracy of 98%.

Keywords Driver Drowsiness · Safe Parking Space · Autonomous Vehicle · Yawning · Gaze Detection · Eye Aspect Ratio

1 Introduction

In recent years, driver drowsiness has created tremendous problems in the field of transportation, human health and safety. A recent study [1] shows that human health and security have been greatly suffered due to driver drowsiness. The real number of accidents due to driver drowsiness is utterly complicated to determine. A report published from National Highway Traffic Safety Administration describes that about 91,000 accidents occurred in the year of 2017. An estimation depending on the crashes reported to police in the USA appears that drowsiness provoked more than 800 casualties. According to the estimation of the National Sleep Foundation [2], about half of the drivers in the USA have driven cars while

Computer Science and Engineering
United International University[1]
BRAC University[2]
Dhaka, bangladesh
Corresponding author: motaharul@cse.uiu.ac.bd

feeling drowsy. Moreover, more than 20% people have admitted to fall asleep while driving. There are many reasons behind drowsy driving. The report [3] narrates that at late night, early in the morning and in the middle afternoon most of the accidents occur. The behavior of the driver is also responsible, as the drivers who feel drowsy are not responsive enough to press the brake to avoid collision. Moreover, it describes that 1 out of every 10 car crash happens as a result of drowsy driving [4]. Though distracted driving and the disobeying of traffic rules are some of the reasons for road accidents each year, drowsy driving is also a significant factor of road accidents.

The initial process of sleepiness may be defined as drowsy. Additionally, being awake, non-rapid eye movement sleep (NREM) and rapid eye movement sleep (REM) are considered as the three steps of sleep. The NREM has also three more steps. The first step is the tendency of falling asleep which is known as drowsy. Most of the drivers fall into a micro or deep sleep at this stage [5]. The modern cars do not have the capabilities to take additional safety step due to the deficiencies of safety procedures. If the drivers remain unable to operate the vehicle thus giving an alarm, it is very necessary to design a system to ensure more safety in a situation while the driver lacks concentration and competence to operate the vehicle.

In this regard, we have introduced a model which can ensure safety by undertaking the operation of driving from the driver to reach the nearest safe parking space (SPS) when the drowsiness has been detected. The preeminent goal of this proposed model is to spot facial points to extract data based on the eye features and yawning. Moreover, our model feeds the fetched data to an ensemble method that we have used to get the most optimized and fastest decision on drowsiness. After that it gives an alarm if the model detects drowsiness. Despite of giving the alarm, if consecutive 24 frames of closed eyelids get detected, the proposed model initiates the autonomous functionality to reach the SPS.
The main contributions of this paper are stated as below:

- We have proposed an algorithmic approach for handling drowsy driving that can be embedded in an autonomous car for enhancing safety.
- We have also developed a system which can extract yawning status and eye features such as amplitude, eye aspect ratio, gaze, eye closeness frequency.
- Our developed ensemble model for detecting drowsiness contains only four classifiers having an accuracy of 98%.
- Along with the proposed drowsiness detection model, our system can find the nearest available parking spot by implementing a proposed SPS finding algorithm.
- Additionally, we have built a prototype trial car using Raspberry Pi for moving autonomously to the nearest available SPS for parking.

The rest of the paper has been organized as follows. Section 2 discusses the literature reviews on drowsiness detection and autonomous vehicles. The detailed system architecture including proposed algorithms has been presented in section 3. In section 4, we have articulated the implementation of the prototype model and section 5 shows the performance evaluation. The limitations and future works have been represented in section 6 and finally section 7 concludes our paper.

2 Literature Reviews

It goes without saying that driver drowsiness has created tremendous problems in the field of transportation, human health, and safety. Therefore, researchers proposed many approaches to reduce the causalities. A related study [6] proposed a system named DriCare, which has the capability of detecting drowsiness by using a recognition technique for facial landmark regions based on 68 facial key points. Moreover, the model implements a non-contact methodology by using a vehicle-attached camera. The authors have introduced an algorithm named multiple convolution neural networks-KCF which can track the face of the driver in the vehicle. Subsequently, in paper [7] authors have proposed a real-time drowsiness detection algorithm that can detect fatigue by taking consideration of the individual persons data processing. To improve the accuracy of the artificial feature extraction, the authors have created a deep cascaded convolutional neural network.

The authors of the paper [8] illustrate a model that has the capability of detecting the range of drowsiness from an initial level to an intense level. The authors have proposed a posture sensitivity index for measuring the initial level of drowsiness. Furthermore, to detect various levels of sleepiness some drowsiness indices have been taken into consideration such as vehicular, blink, posture, and physiological index. In the paper [9], the authors have proposed a multitasking convolutional neural network model for detecting driver drowsiness after processing data from the eye and mouth. In the described model [10], information from both the eye and the mouth has been classified simultaneously in the same model. The authors have discussed a drowsiness detection and alarming system after merging data which has been collected from the eye and the yawning status by maintaining RGB-D cameras.

However, the existing systems only focus on detecting drowsiness and do not consider any posterior functionality to ensure safety. Moreover, the time-complexity of detecting drowsiness need to be minimized. And to get high accuracy for detecting drowsiness using various factors could cause time complexity. Therefore, an equilibrium should be maintained between the accuracy and the time for the detection procedure.

In [11], the authors have mentioned that in the future self-driving cars may play a significant role in the field of transportation by accomplishing more comfort and safety in the whole process of driving. The self-driving cars have been built in a certain way so that it can deal with human-centric vehicles and can perform tasks without human interaction. In [12,13], authors show the application of smart parking system by using cloud computing. The authors in [14,15] illustrates how driver behavior from lane departure can be observed. Therefore, the self-driving cars can use the safety features such as the ability to avoid collisions, detect the road signs, automated parking and most importantly autonomous movement capability to a certain location after selecting the place.

To the best of our knowledge, the recent papers mostly considered drowsiness detection along with alarm systems. Furthermore, recent researches have been estimated that the popularity of autonomous vehicles will be increasing in the near future. Also, we have been motivated by these notions and influenced by the [16] paper which focuses on the detection by taking input data only from the eye and yawning status. As a result, we make an effort to improve the safety by proposing a model which has the functionality of extracting data from the eye and mouth region to predict drowsiness more accurately. Lastly, depending on the prediction and further monitoring, the model can take control of the vehicle to reach the SPS for ensuring safety.

3 System Architecture

Our proposed system architecture shown in Fig. 1 represents the whole procedure into two main parts having respective sub-parts. The first part has focused on detecting drowsiness while the second part handles the autonomous operation until the vehicle reaches the nearest SPS.

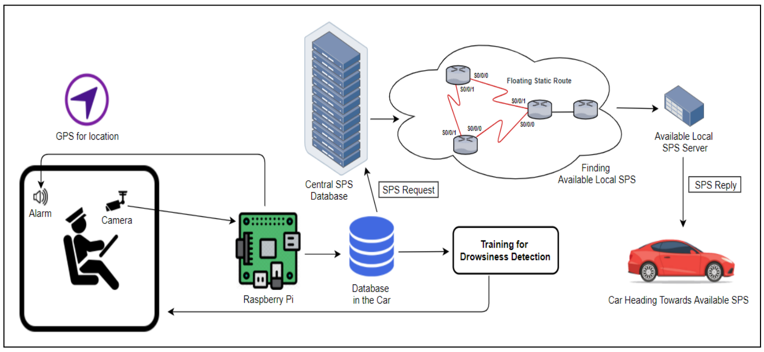


Fig. 1: Architecture of the proposed model

3.1 Facial Feature Extraction

Initially to extract facial features, we have used facial landmarks depending on some parameters such as the structure of the model, the facial appearance and lastly the facial shape. The facial shape can be classified into three major categories: the holistic methods, the constrained local model methods and the regression-based methods. In the proposed model, we have used the cascaded regression-based supervised descent method (SDM) which learns the descent direction with regression [17,18]. Furthermore, it is simplified as the cascaded regression model has been implemented with linear regression function which can predict the landmark location updates from shape indexed local appearance as shown in Fig. 2.

The objective of SDM is to find a sequence of descent directions and to estimate the location updates of δx which has been derived by using Newton's method as shown in Eq. 1-6.

$$f(x_0 + \delta x) = f(x_0) + J_f(x_0)^T \delta x + \frac{1}{2}(\delta x)^T H(x_0)\delta x \qquad (1)$$

$$\delta x = -H_f(x_{n-1})^{-1} J_f(x_{n-1}) = -2H_f(x_{n-1})^{-1} J_\phi^T(\phi_{n-1} - \phi_n) \qquad (2)$$

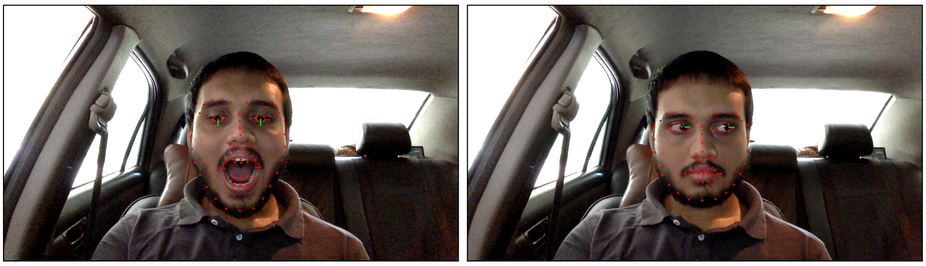


Fig. 2: Spotting landmarks and extracting facial features

Measuring δx, requires calculation extensive methods as it needs to compute the Jacobian(J_f) and Hessian(H_f) matrix for each update of x_0. Therefore, a supervised descent method is used to learn the descent direction with a regression method. Subsequently, with linear regression function it is converted as the cascaded regression that can predict location updates.

$$R_{n-1} = -2H_f(x_{n-)})^{-1}J_\phi^T \tag{3}$$

$$b = 2H_f(x_{n-1})^{-1}J_\phi^T\phi(I(X^k)) \tag{4}$$

$$\delta x = R_{n-1}\phi_{n-1}(I(x_{n-1})) + b \tag{5}$$

Here R is the descent direction which is estimated by learning a linear regression between δx and $\delta\phi$. Also, b is the bias which is used to find the location updates δx and finally, the minimized feature distance equation can be written as:

$$\delta x = argmin(f(x_0 + \delta x)) = argmin||(\phi(I(x^k))) - \phi(I(x_0 + \delta x))||^2 \tag{6}$$

3.2 Drowsiness Detection(DD)

The system detects drowsiness after getting processed data from the embedded webcam which has been used for real time monitoring on the driver. At first, the system marks face from frames using 68 facial landmark points and further detects the eyes and mouth region. Based on the spotted facial points, some features from eye region such as eye aspect ratio, amplitude, blinking rate, gaze direction and yawning status have been extracted and also with the help of other parameters as shown in Table 1. are used to detect drowsiness procedure more accurately.

3.2.1 Feature selection and analysis of eye region

The calculation initializes by detecting 6 points from each of the eyes and measures the contour area around the eyes as shown in the following Fig. 3. The width and height of each eye poses a close association-ship and based on the relation a real time eye blink detection procedure has been implemented by deriving an equation named as the Eye Aspect Ratio (EAR)[19] which has been presented in Eq. 7.

$$EAR = \frac{|x_2 - x_6| + |x_3 - x_5|}{2|x_1 - x_4|} \tag{7}$$

Here x_1, x_2 ... x_6 are the 2D landmark points located on the eye region visualized in Fig. 3. The processed data from EAR remains mostly constant when the eyes are open and gets close to zero while the eyes are closed. Since eye blinking is performed by both eyes synchronously, the mean is taken.

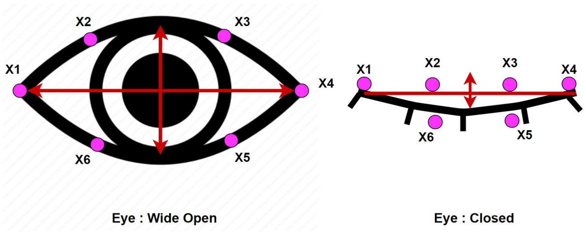


Fig. 3: Detected co-ordinates of eye region

Table 1: System Parameters

Symbol	Description
$\boldsymbol{x_0}$	Initial estimated location
$\boldsymbol{\delta_x}$	Update on landmark location
$\boldsymbol{J_f}$	Jacobian matrices of f
$\boldsymbol{H_f}$	Hessian matrices of f
$\boldsymbol{R_{n-1}}$	Descent direction
$\boldsymbol{\phi_n}$	n^{th} Newton step
EAR	Eye Aspect Ratio
LP	Left pupils coordinate values
RP	Right pupils coordinate values
LP_Center	Left pupils center coordinate
RP_Center	Right pupils center coordinate
LAR	Lip Aspect Ratio

The amplitude of the eye decreases when a person feels drowsy. Therefore, we calculated amplitude as illustrated in Eq. 8.

$$Amplitude = \frac{|x_2 - x_6| + |x_3 - x_5|}{2} \tag{8}$$

Here the equation calculates distance between the upper eye points and lower eye points and finds the average distance which has been named as Average Amplitude. It has been used to compare the processed data with the defined threshold. If a person feels drowsy, the eye blinking rate increases and the blink time decreases. Further, these measurements have been used to examine the blink features, from where the threshold can be defined. Also, another feature has been analyzed by our algorithm where it finds the coordinate of the pupil as shown in Eq. 9, 10. Also, it calculates the *Vertical* and *Horizontal* ratio of the pupil and based on that the gaze direction gets updated in real time [20].

$$Horizontal\ [LP, RP] = \left[\frac{LP[x]}{LP\_Center[0] \times 2 - 10}, \frac{RP[x]}{RP\_Center[0] \times 2 - 10}\right] \quad (9)$$

$$Vertical\ [LP, RP] = \left[\frac{LP[y]}{LP\_Center[1] \times 2 - 10}, \frac{RP[y]}{RP\_Center[1] \times 2 - 10}\right] \quad (10)$$

Here, LP represents left pupil, RP represents right pupil, X and Y are the x and y coordinates respectively. LP_Center and RP_Center represent the pupils location. Also, the horizontal and vertical positions of the pupils of the eyes are represented by 0 and 1 respectively. Finally, by calculating the *Vertical* and *Horizontal* ratio of the pupil the gaze direction gets updated in real time.

3.2.2 Feature Selection and Analysis of Mouth Region

After spotting a total number of 18 landmark points on the driver's mouth, we have extracted the 8 landmark points of the inner lips region. According to [21,22], a yawn on average lasts for 4-7 seconds for a human being. So, if the mouth remains open for consecutive 4 seconds or 120 frames for our implemented camera, the system will detect yawning. The following Eq. 11,12 shows the process of selecting mouth region [23] from the facial points.

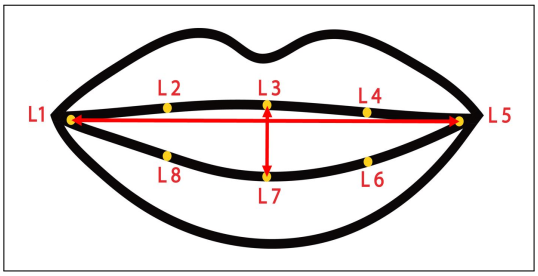


Fig. 4: Detected Lip Points

From Fig. 4, it can be seen that for calculating Lip Asepect Ratio (LAR), we need only 4 points from the detected 8 points. To illustrate, the difference between *L3* and *L7* defines the maximum height of the mouth region and the difference between *L1* and *L5* defines the maximum width. So, LAR can be calculated by dividing the middle lip points distance value by the side lip points distance value as shown in Eq. 11. The system will detect the driver is yawning while being drowsy if the LAR value exceeds the threshold which is 0.5 for consecutive 120 frames. This threshold has been set to greater than 0.5 because while the mouth is fully open, the value of the ratio is 1 and here, above 0.5 defines the mouth is mostly open.

$$LAR = \frac{|L3 - L7|}{|L1 - L5|} \quad (11)$$

3.2.3 Determining Drowsiness

Algorithm 1 is the proposed drowsiness detection algorithm which takes the detected facial landmark points as input and outputs the drowsiness status. Initially, it extracts facial features to detect gaze, calculate amplitude, EAR, frequency. These features will be extracted for 1 min and after that all these values will be stored in a dataset for further making decisions on drowsiness detection process. After 1 min, the system starts checking yawning status by using Eq. 11, and based on that condition and the threshold yawning count will be updated. Moreover, to predict drowsiness status, these extracted data will be sent to the ML classifier. If the classifier detects drowsiness it will give an alarm and initiate a counter for eye closeness. Also, the implemented camera will continuously extract data and measure the EAR value by using the Eq. 7. If the EAR value is less than 0.25 in a frame, the eye closeness count will be increased. Otherwise, if the EAR value is greater than 0.25 meaning the driver is not drowsy or eye closeness count is exact to 24, the system will break the continuous while loop and execute the next instruction. Now, if the eye closeness count is greater than 24, the eye closeness count is set to 0 and also the system will activate autopilot mode to move towards the SPS for parking. Otherwise, if the eye closeness count is less than 24, it will recursively execute the drowsiness detection function.

3.3 Safe Parking Space Functionality

The SPS functionalities have been presented in Fig. 5 which initiates after detecting drowsiness and again if the measurement of the consecutive eye closeness frame count is higher than 24. Then the rest of the functionalities described below will be started.

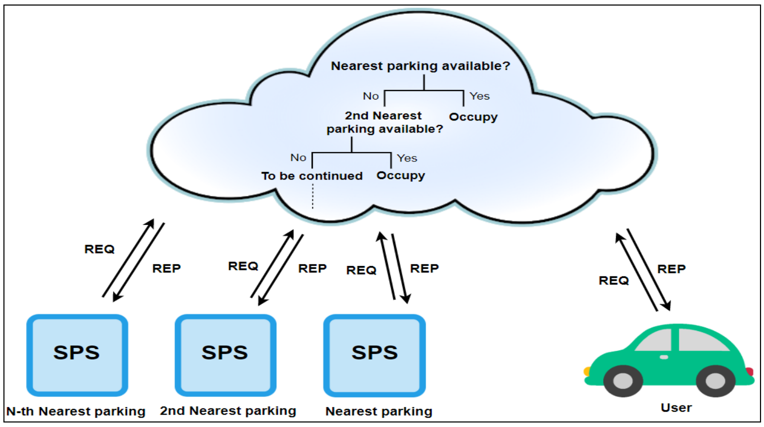


Fig. 5: Interfacing between central server, SPS and vehicle

Algorithm 1: Proposed algorithm on drowsy driving to intiate SPS

```
Input: detected_facial_landmark
Output: drowsiness_status
Drowsinss_Detection :
  while true do
      gaze ← gaze_detection(facial_feature)
      amplitude ← calculate_amplitude(facial_feature)
      ear ← calculate_ear(facial_feature)
      frequency ← calculate_frequency(facial_feature)
      Store these feature values to dataset
      if time > 1 min then
          while LAR ≥ 0.5 do
              frame_count++
              if frame_count > 120 then
                  Yawning_count++
              end if
          end while
          Send stored feature values to ML classifier to predict Drowsiness_status
          if Drowsiness_status is true then
              alarm()
              eye_closeness = 0
              while camera running do
                  if EAR in a frame < 25 then
                      eye_closeness++
                  else if EAR in a frame>0.25 or eye_closeness==24 then
                      break
                  end if
              end while
              if eye_closeness >24 then
                  eye_closeness = 0
                  activate_autopilot_mode()
                  SPS(Vehicle_info)
              else
                  Drowsinss_Detection()
              end if
          end if
      end if
  end while
```

3.3.1 Request to Central Server

In the first phase of initiated SPS functionalities, the model sends a notification message to the central server containing information about the detection of drowsiness of the driver, a live location of the vehicle and also information about the vehicle. Moreover, the message also contains a request to book a parking spot in the nearest available parking space.

3.3.2 Response from Central Server

Subsequently getting the distressed message from the vehicle, the central server will estimate the distance between the vehicle and a list of pre-installed SPS. Furthermore, after finalizing the nearest SPS, Central Server performs two tasks simultaneously; sends a booking request to the selected local SPS server for a parking spot and a confirmation message to the vehicle. Then the vehicle will move autonomously towards to the designated SPS.

Algorithm 2: Central Server Finding Nearest-SPS

```
Input: SPS_list and Vehicle_location
Output: Nearest_SPS_Location
Finding_SPS(SPS_list, Vehicle_location):
    dist[Vehicle_loc] = get_Vehicle_distance()
    for all SPS in SPS_list do
        SPS_Location = Booked_SPS_from_Server
        if SPS_Location is not Vehicle_location then
            dist[SPS_Location] = sets infinity
            append Chosen_SPS_Location to SPS_Dist_Calculation_List
        end if
    end for
    while SPS_Dist_Calculation_List is not empty do
        Nearest_SPS = SPS in SPS_List with min dist[Chosen_SPS_Location]
        append Chosen_SPS_Location to Nearest_SPS
        for all SPS in SPS_list do
            ALL_SPS_Location_asc.sort( reverse = False)
            return ALL_SPS_Location_asc
        end for
    end while
    for all SPS in ALL_SPS_Location_asc do
        if confirmation_message(ALL_SPS_Location_asc[0]) is positive then
            set ALL_SPS_Location_asc[0].distance to nearest_SPS
            break
        else
            for all SPS in ALL_SPS_Location_asc do
                send request to ALL_SPS_Location_asc[sps]
            end for
        end if
    end for
```

In Algorithm 2 , the SPS_list in central server, contains all the ids of the parking spaces. The calculated location of the SPSs will be saved in another list in ascending manner. The central server will send a booking request to the nearest calculated SPS in the list to discover whether the SPS is available for parking or not. If the confirmation is negative from the SPS, the central server sends another booking request to the second nearest SPS and the process continues until the central server finds a parking spot.

After the Central Server receives a positive message from a Local SPS server indicating a suitable parking space for the vehicle, the Central Server transmits a message to the vehicle consisting the information about location and space of the SPS and also a reservation message to the Local SPS server. After receiving the confirmation message from the Central SPS server, the vehicle will move towards to the location of the SPS.

4 Implementation of the Prototype Model

This section illustrates the whole implementation in details, how our proposed algorithm will handle the drowsy driving situation and operate autonomously to reach the SPS.

4.1 Drowsiness Prediction

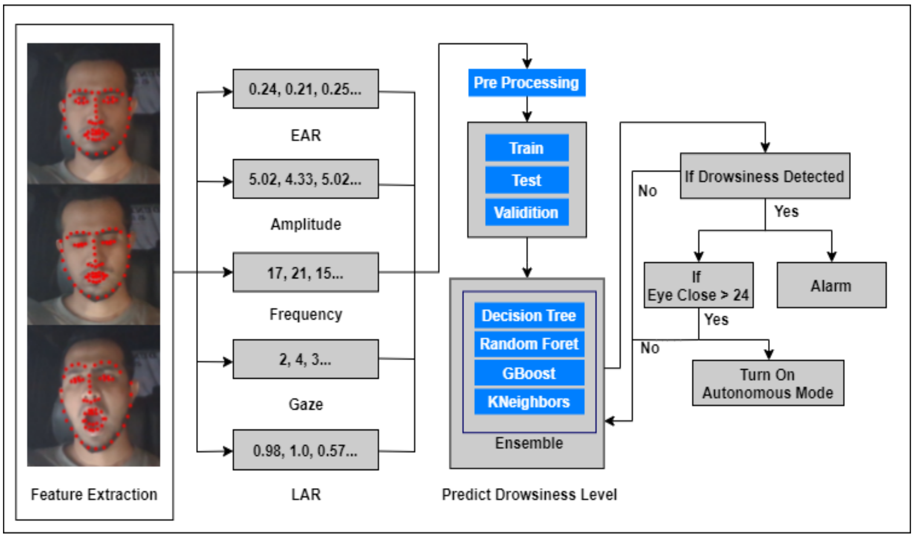


Fig. 6: Data processing and drowsiness prediction process

The implemented model uses webcam for extracting video frames as shown in Fig. 6. Initially, the landmark points will be detected and values from EAR, amplitude, frequency, yawning status and gaze will be computed. To evaluate the model, the stored data has been segmented into train, test and validation set. Further using live data, the trained ensemble model predict drowsiness. If the prediction is true, the model initiates the process of alarming. Simultaneously it observes, if the eyelids of the driver have been closed for than 24 consecutive frames.

An average human blink lasts for 100 milliseconds to 400 milliseconds and [24] states that if the driver is drowsy, the eyes of the driver remain closed for more or equal to 80 percent of time in a minute. Here, The frame count 24 has been set because it is 80 percent of 30 frames. As our implemented camera captures 30 frames per second, after detecting drowsiness if the eyes are being closed for more than 24 consecutive frames the model activates autopilot mode.

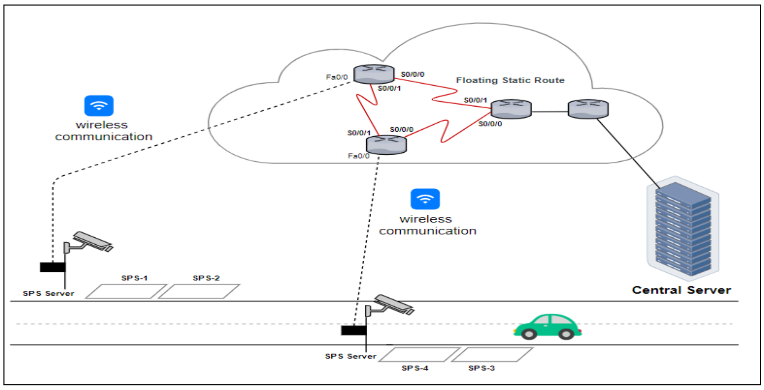


Fig. 7: Illustrates the network architecure of SPS

4.2 Network Topology between Car and SPS

Fig. 7 illustrates an ideal network architecture. The network communication between the SPS is represented by router to router communication where each router exhibits a single SPS. Because of hop to hop connection all SPS can interact.

4.3 Implemented Trial Car

After getting a parking acceptance message from the Central SPS Server, our trained autonomous experimental vehicle begins steering and reaches a predefined point that we have evaluated as the SPS.

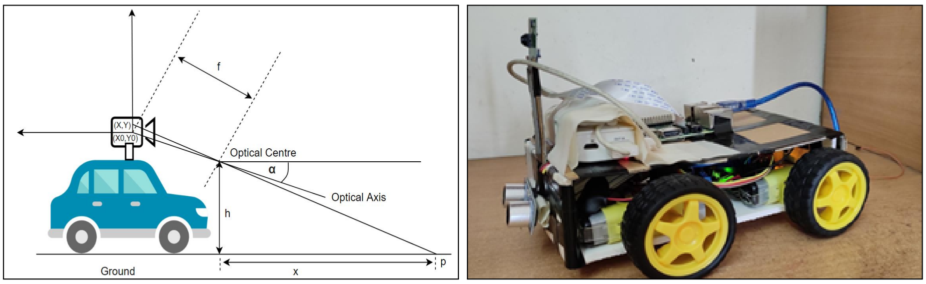


Fig. 8: Prototype trial car of the proposed model

To construct the framework of the autonomous test car shown in Fig. 8, a Raspberry Pi (model B+) attached with a Pi camera and a HC-SR04 sensor appended with an Arduino Uno have been used. Furthermore, for functioning autonomously, data has been fetched after conducting a few manual drives and video data frame has been collected using Pi camera and fed to the MLP model for training. And it provides prediction of steering wheel angle and measures the distance via monocular vision whether to move left, right, forward or backward.

The implemented MLP classifier has been illustrated in Fig. 9 which has been used for training our RC trial autonomous car. The classifier has 38,400 nodes in the input layer because the dimension of the input image is 320 × 120. The number of nodes in the hidden layers are 1024 and 32 respectively which are chosen arbitrarily. Lastly, the output layer has 4 nodes indicating the directions which are left, right, forward and backward.

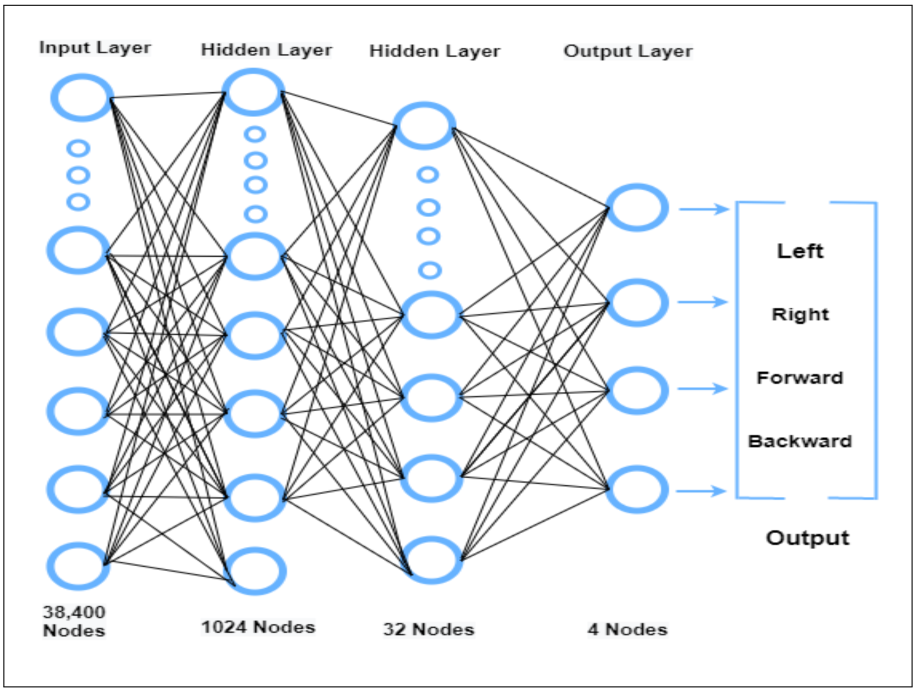


Fig. 9: MLP Architecture for RC autonomous car

5 Experiments

5.1 Dataset

Our dataset contains 24 subjects with 4 different scenarios(wearing_glasses_day, bare_face_day, wearing_glasses_night, bare_face_night). The subjects have different gender, age and eye size. For safety, we build the dataset by asking the driver to make driving operations on the simulator, including the awake driving and the drowsy driving. Here, the training set contains a total number of 40 videos of 20 subjects and the rest of the 4 videos from 4 subjects are used for validation purposes. The sequences for each subject including yawning and slow blink rate, downward gaze direction, lower amplitude of eye and non-drowsy actions are recorded for about 2 minute long and all the videos are captured in 30 frames per second. The ranges of feature values in our dataset has been shown in Table 2.

Table 2: The ranges of feature values in dataset

Parameter	Range
Gaze	0-5
EAR	0.01 - 0.339
Left EAR	0.01 - 0.313
Right EAR	0.01-0.341
Amplitude	0.25 - 8.25
Frequency	7-15 /min
LAR	0 - 1

Here, we have converted the categorical feature of gaze to 0 - 5. The categorical features of gaze directions are looking right, up, down, lower right and lower left respectively. Rest of the values are stored as they were extracted from facial point.

5.2 System Evaluation

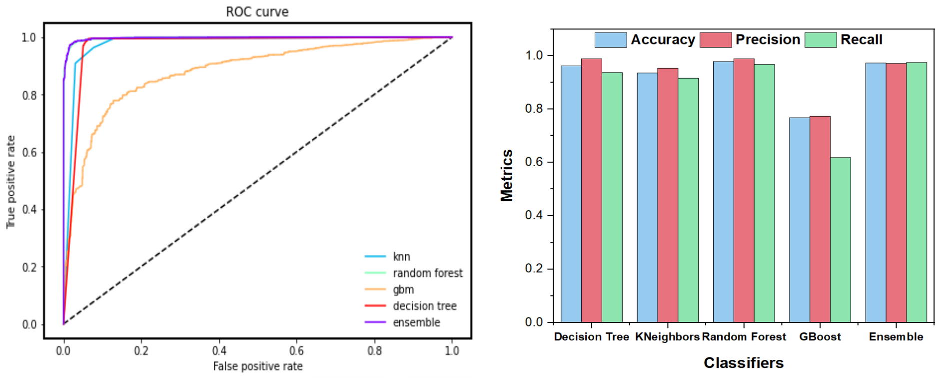
The analysis of the prototype model's drowsiness prediction performance of various classifiers, the feature correlations and the succeeding impacts of implementation has been demonstrated below.

Table 3: Evaluation results of drowsiness prediction

Classifier	RMSE	Accuracy	ROC Score
Decision Tree	0.036	0.963	0.970
Random Forest	0.021	0.978	0.970
KNeighbors	0.064	0.935	0.970
GBoost	0.129	0.895	0.784
Ensemble	0.188	0.980	0.980

The table 3 shows the accuracy, the RMSE and the ROC score of all the four implemented classifiers in our proposed ensemble model. Here the ensemble model performs comparatively better than the other classifiers in all three factors of the analysis process.

The graphical representation in Fig. 10 shows the performance of drowsiness detection classification models. From the curve areas of Fig 10(a) we can easily observe that the ensemble method has achieved the highest possible ROC score which is 98%. After that, KNeighbors, random forest and descision tree have achieved the ROC score of 97%. Lastly, the gradient boosting method(GBM) has acquired the ROC score of 78% Moreover, in [25] authors have shown that there is a relationship between ROC score and precision-recall (PR). They have proved that if a curve dominates its ROC space, it will proportionally dominate the PR space. So, we



(a) Receiver operating characteristic curves (b) Accuracy, precision and recall metrics

Fig. 10: Performance evaluation of trained classifiers

can say that a model with higher ROC score can perform more precisely because it has higher precision-recall score. Therefore, it can be undoubtedly perceived that the ensemble method has outperformed all the remaining classifiers.

The above line graph in Fig. 10(b) shows the normalized form of accuracy, precision and recall score in the range of 0 to 1. To begin with, accuracy is considered to be the most important measurement of performance and from Eq. 12 it can be seen that it is the ratio of correctly predicted observation to the total observation. Here, from Eq. 13 it is apparent that precision score can be derived from the ratio of correctly predicted positive observations to the total predicted positive observations. Then, recall can be measured from the ratio of correctly predicted positive observations to the all observations in actual class as shown in Eq. 14. Therefore, after considering the performance evaluation functions we can see that ensemble method surpasses all other classifiers acquiring the score of 0.895, 0.886 and 0.998 in terms of accuracy, precision and recall accordingly.

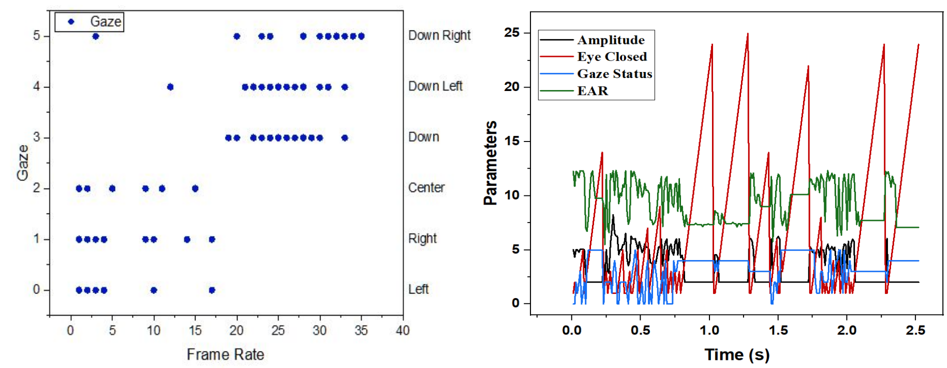
$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{12}$$

$$Precision = \frac{TP}{TP + FP} \tag{13}$$

$$Recall = \frac{TP}{TP + FN} \tag{14}$$

Here, TP is considered as true positive, TN is considered as true negative, FP is false positve and lastly FN is false negative.

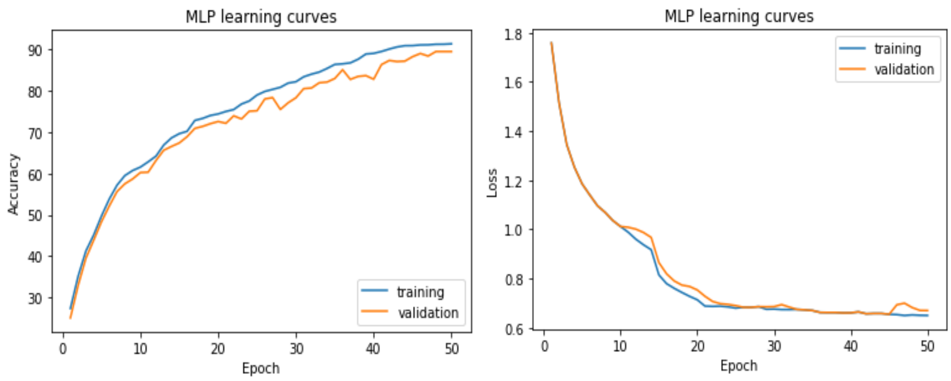
The Fig. 11(a) illustrates the gaze direction over the frame sequence when a person is drowsy.From the graphical data it can be seen that the gaze direction of the left, right and center has lower consecutive frequency than down, down left and down right. It can be noticed that the gaze of a drowsy person remains mostly downward. Also, the left downward and right downward gaze direction indicate the variation of the face calibration while being sleepy. Therefore, the gaze directions data clustering can be used to deduce if a person is sleepy or not.



(a) Drowsiness analysis from gaze clusters (b) Extracted feature representation

Fig. 11: Analysis of extracted facial features

The Fig. 11(b) represents normalized eyes feature comparison over a period of time. Moreover, it can be easily interpreted from the figure that the amplitude, gaze, frequency and EAR remain steady when the eye closeness rate is increasing or the closeness rate is at its maximum level. Subsequently, a stable result for a prolonged time (i.e 24 frame > threshold) supports the idea of being drowsy or incapability to operate the vehicle.



(a) Accuracy rate of MLP (b) Loss function of MLP

Fig. 12: Performance visualization of prototype car

From the graphical illustration of Fig. 12 (a), it can be seen that the path finding accuracy has increased significantly with the increasing of epochs. After 50 epochs, the model has shown accuracy over 90 percent. Also, it can be seen that the model performed quite well as the training and validation curves are close to each other. From the Fig. 12 (b), it can be seen that the loss function value has reduced with the increasing of the epochs. After completing 50 epochs, the loss functions for both training and validation processes have become less than 0.7.

Table 4: Comparison with similar systems architecture

No.	Ref	Method and Classifier	System Description	Accuracy
1	[29]	AlexNet,VGG-FaceNet, FlowImageNet	Deep robust architecture	73.06%
2	[30]	Viola-Jones algorithm	Android based DL method	81 %
3	[31]	Minimized Network Structure	DL based Real-time detection	89.5 %
4	[32]	MC-KCF algorithm	Yawning, blinking and eye closure duration	92 %
5	[33]	Viola Jones algorithm and PERCLOS	Analyzed lighting conditions	95 %
6	[34]	Funnel-structured cascade algorithm	PERCLOS based detection	95.50 %
7	[35]	Wavelet Network Classifier (WNC)	Focused on eye closure duration	97 %
8	-	Ensemble	Proposed ensemble model	98%

We have compared the drowsiness detection accuracy of our proposed ensemble model with several existing models and it has shown in Table 4. From the table, it can be seen that our developed ensemble method has performed relatively well.

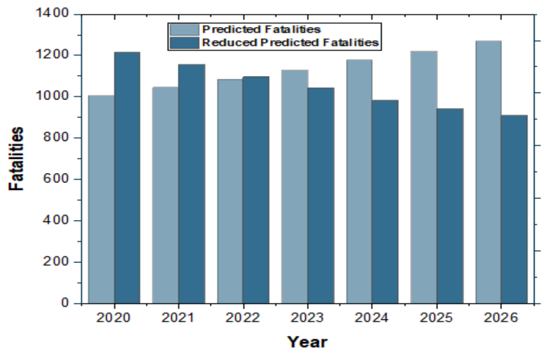


Fig. 13: Predicted reduced fatalities rate

The adaptation of the proposed model can significantly lower the casualty rate which has been depicted in the Fig. 13. As the autonomous vehicle will play an important role in the field of transportation, the implementation of our system will reduce the rate of fatalities to a great extent. Here, we have predicted the accident rate from 2020 to 2026 based on the analysis [26, 27] of past ten years. Another study [28] shows that the sales of the autonomous car will be expanded about 2-5 % within next decade. Hence, using the relationship of the predicted accident rate with the sales and fleet we calculated the expected value and the variance σ^2 over the year. This evaluation graph shows the effect of our proposed method in future. Therefore, we can say that, embedding our model in the future autonomous vehicles will have positive effect and we hope to ensure safe driving.

6 Limitations and Future Works

The main limitation of research works related to this field is the lack of available autonomous car on road. Therefore, real-time data acquisition process is challenging in this area. In future, along with facial features, extracted data from the steering wheel, body posture and temperature can be used to ensure more accurate decisions about drowsiness detection before turning the car into autonomous mode.

7 Conclusions

In this paper, we have proposed an automated approach for the autonomous vehicles to ensure safety by reaching at the safe parking space after detecting drowsiness of the driver thus giving alarm. The drowsiness detection accuracy of our proposed method is about 98%. Moreover, the system has the capability of sending a message to the central server to book a parking spot in the nearest available SPS. In this regard, we have developed a trial model car by using Raspberry Pi and trained the car by using neural network to build an autonomous trail vehicle which is used for finding the path to the nearest SPS. Currently, the functionalities of autonomous vehicles have improved significantly. We have mainly focused on the drowsiness detection and self parking system for enhancing safety that can be implemented as an embedded system in the existing autonomous vehicles. Here, we have achieved 91% accuracy to find paths using a trial autonomous car for demonstrating the feasibility of our proposed system. By utilizing the model we strongly believe, the fatalities due to drowsy driving can be reduced to a great extent.

References

1. (2019) Drowsy Driving. In: NHTSA. https://www.nhtsa.gov/risky-driving/drowsy-driving. Accessed 14 Jun 2020
2. (2020) On The Road. In: Drowsy Driving. https://www.nsc.org/road-safety/safety-topics/fatigued-driving. Accessed 14 Jun 2020
3. (2020) Drowsy Driving. In: Sleep Education. http://sleepeducation.org/sleep-topics/drowsy-driving. Accessed 14 Jun 2020
4. Beau PL (2018) Drowsy driving may be the cause of 1 out of every 10 auto crashes. In: CNBC. https://www.cnbc.com/2018/02/07/drowsy-driving-may-be-the-cause-of-1-out-of-every-10-auto-crashes.html. Accessed 14 Jun 2020
5. (2006) Sleep-Information about Sleep. In: National Institutes of Health. https://www.nih.gov/news-events/news-releases/nih-offers-new-comprehensive-guide-healthy-sleep. Accessed 14 Jun 2020
6. Deng W, Wu R(2019) "Real-Time Driver-Drowsiness Detection System Using Facial Features," in IEEE Access, vol. 7, pp. 118727-118738.
7. You F, Li X, Gong X, Wang H, Li H (2019) "A Real-time Driving Drowsiness Detection Algorithm With Individual Differences Consideration," in IEEE Access, vol. 7, pp. 179396-179408.
8. Sunagawa M, Shikii S, Nakai W, Mochizuki M, Kusukame K, Kitajima H (2020) "Comprehensive Drowsiness Level Detection Model Combining Multimodal Information," in IEEE Sensors Journal, vol. 20, no. 7, pp. 3709-3717.
9. Savaş BK, Becerikli Y (2020) "Real Time Driver Fatigue Detection System Based on Multi-Task ConNN," in IEEE Access, vol. 8, pp. 12491-12498.

10. Yazdi MZJ, Soryani M (2019) ”Driver Drowsiness Detection by Yawn Identification Based on Depth Information and Active Contour Model”, 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), Kannur,Kerala, India,pp. 1522-1526.
11. Straub J et al. (2019) ”An internetworked self-driving car system-of-systems”, 2017 12th System of Systems Engineering Conference (SoSE), Waikoloa, HI,pp. 1-6.
12. Hasan MO,Razoan K, Islam MM(2020) “Parking Recommender System using Q-Learning and Cloud Computing”, 2nd International Conference on Cyber Security and Computer Science.
13. Hasan MO, Islam MM et al. (2019) ”Smart Parking Model based on Internet of Things (IoT) and TensorFlow”, 7th International Conference on Smart Computing and Communications, Curtin University, Miri, Sarawak, Malaysia.
14. Arnob FA, Fuad MA, Nizam AT, Islam MM (2020) “A Novel Traffic System for Detecting Lane-Based Rule Violation”, Annals of Emerging Technologies in Computing, Vol. 4, No.
15. Arnob FA, Fuad MA, Nizam AT, Barua S, Choudhury AA, Islam MM (2020) “An Intelligent Traffic System for Detecting Lane Based Rule Violation”, International Conference on Advances in the Emerging Computing Technologies, Islamic University of Madinah, Madinah, Saudi Arabia.
16. Islam MM, Kowsar I, Zaman MS, Sakib FR, Saquib N (2020) ”An Algorithmic Approach to Driver Drowsiness Detection for Ensuring Safety in an Autonomous Car”, 2020 IEEE Region 10 Symposium (TENSYMP).
17. Xiong X, Torre FDL (2013) ”Supervised Descent Method and Its Applications to Face Alignment”, 2013 IEEE Conference on Computer Vision and Pattern Recognition, pp. 532-539.
18. Wu Y, Ji Q (2019) ”Facial Landmark Detection: A Literature Survey”, Int J Comput Vis 127, 115–142.
19. Owais S (2017) “Eye Blink Detection Algorithms: Details”, Details Hackaday.io, Available Online: https://hackaday.io/project/27552-blinktotext/log/68360-eye-blink-detection-algorithms. Accessed 14 June 2020.
20. Vicente F, Huang Z, Xiong X, Torre FDL, Zhang W, Levi D (2015) ”Driver Gaze Tracking and Eyes Off the Road Detection System”, IEEE Transactions on Intelligent Transportation Systems. 1-14.
21. Jacques B, “Yawning,” J. Neurol., Neurosurg. Psychiatry, vol. 21, no. 3, pp. 203–209.
22. Deng W, Wu R. Real-time driver-drowsiness detection system using facial features. IEEE Access. 2019 Aug 21;7:118727-38.
23. Abtahi S, Hariri B, Shirmohammadi S (2011) ”Driver drowsiness monitoring based on yawning detection”, IEEE International Instrumentation and Measurement Technology Conference, Binjiang, pp. 1-4.
24. Galarza EE, Egas FD, Silva FM, Velasco PM, Galarza ED.(2018) Real time driver drowsiness detection based on driver’s face image behavior using a system of human computer interaction implemented in a smartphone. InInternational Conference on Information Technology Systems.
25. Davis j, Goadrich M (2006) ”The relationship between Precision-Recall and ROC curves”, In: Proceedings of 23rd International Conference on Machine Learning - ICML.
26. (2018) Facts + Statistics: Drowsy driving. https://www.iii.org/fact-statistic/facts-statistics-drowsy-driving. Accessed 9 July 2020.
27. Covington T (2020). Drowsy Driving Statistics in 2020: The Zebra. https://www.thezebra.com/research/drowsy-driving-statistics/. Accessed 9 July 2020.
28. Litman T (2020). Autonomous Vehicle Implementation Predictions (pp. 1-45, Rep.). Victoria Transport Policy Institute. from https://www.vtpi.org/avip.pdf. Accessed 9 July 2020.
29. Park S, Pan F, Kang S, Yoo CD.(2016) Driver drowsiness detection system based on feature representation learning using various deep networks. InAsian Conference on Computer Vision 2016.
30. Jabbar R, Al-Khalifa K, Kharbeche M, Alhajyaseen W, Jafari M, Jiang S.(2018) Real-time driver drowsiness detection for android application using deep neural networks techniques. Procedia computer science.
31. Reddy B, Kim YH, Yun S, Seo C, Jang J.(2017) Real-time driver drowsiness detection for embedded system using model compression of deep neural networks. InProceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops(pp. 121-128).
32. Deng W, Wu R.(2019) Real-time driver-drowsiness detection system using facial features. IEEE Access.7:118727-38.

33. Nguyen TP, Chew MT, Demidenko S. (2015) Eye tracking system to detect driver drowsiness. In2015 6th International Conference on Automation, Robotics and Applications (ICARA).
34. Navastara DA, Putra WY, Fatichah C.(2020) Drowsiness Detection Based on Facial Landmark and Uniform Local Binary Pattern. InJournal of Physics: Conference Series (Vol. 1529, No. 5, p. 052015)
35. Teyeb I, Jemai O, Zaied M, Amar CB.(2014) A novel approach for drowsy driver detection using head posture estimation and eyes recognition system based on wavelet network. InIISA 2014, The 5th International Conference on Information, Intelligence, Systems and Applications.(pp. 379-384).