

# Team members

**R.Lathika**

**K.Kaviya**

**S.Rajadurai**

**P.Mutharasan**

## IoT-Based Smart Alert System for Drowsy Driver Detection

### 1. Introduction

Driver fatigue has been the main issue for countless mishaps due to tiredness, tedious road condition, and unfavorable climate situations [1]. Every year, the National Highway Traffic Safety Administration (NHTSA) and World Health Organisation (WHO) have reported that approximately 1.35 million people die due to vehicle crashes across the world. Generally, road accidents mostly occur due to inadequate way of driving [2]. These situations arise if the driver is addicted to alcohol or in drowsiness [3]. The maximum types of lethal accidents are recognised as a severe factor of tiredness of the driver. When drivers fall asleep, the control over the vehicle is lost [4]. There is a need to design smart or intelligent vehicle system through advanced technology [5]. This paper implements a mechanism to alert the driver on the condition of drowsiness or daydreaming. A camera monitors the driver's eye blinking, eye closure, face detection, head posture, etc. with face land-

mark algorithm and Euclidean distance in the behavioral-based approach. These characteristics help to measure driver fatigue and instantly alert him with the help of voice speaker and forwarding an e-mail to a person (owner of vehicle) who can make him conscious [6]. An e-mail is being transmitted to a destination using IoT module, which relies on wireless transmission [7, 8]. But, the proposed system is being integrated by a credit card-sized computer known as Raspberry Pi3 and Pi camera which can trace an eye movement [9] thereby monitoring intensity of collision effects that happen at the time of accident and alerting the emergency ward of the hospitals or owners that are nearby to the accident spot along with GPS location of the accident [10, 11].

*Problem Statement.* An IoT-based system is designed to avoid countless mishaps due to drowsy drivers' behavioral and psychological changes by focusing on driver's eye movements. In addition to monitoring the intensity of the collisions impacts during road accidents, it is also

required to keep records of the location for taking supportive action.

*Contribution to This Work.* The drowsiness of driver and impact of collision monitoring or alert system is constructed using IoT technology along with Raspberry Pi. For finding the fatigue or sleepiness of driver, a Pi camera can be used during driving. Apart from it, the vehicle needs to be well mounted by crash sensor and FSR sensor for detecting the extremity of collision. When the drowsiness is detected, the driver is alerted by voice speaker and a mail sent to the vehicle owner. Similarly, suppose any sudden collision happens due to drowsiness. In that case, the data collected from the sensors and the alert message are messaged to the nearer hospitals nearby the prone location from Google Maps link where the mishap has happened.

The rest of the paper has been organised as follows: in Section 2, the literature review is included. Section 3 describes about components which are used for designing the proposed system. Section 4 illustrates the proposed system with a block diagram along with a brief explanation of the methodology with a flowchart in Section 5. The result analysis is discussed in Section 6. Finally, a conclusion of the paper with its future work is detailed in Section 7.

## 2. Literature Review

Drowsiness of driver can be determined with different aspects using vehicle-based, psychological, and behavioral measurements implemented through different predictive algorithms as discussed in the following sections.

*Face and Eye Detection by Machine Learning (ML) and Deep Learning (DL) Algorithms.* Jabbar et al. [2] proposed Convolutional Neural Network (CNN) technique of the ML algorithm to detect microsleep and drowsiness. In this paper, detection of driver's facial landmarks can be achieved through a camera that is then passed to this CNN algorithm to properly identify drowsiness. Here, the experimental classification of eye detection is performed through various data sets like without glasses and with glasses in day or night vision. So, it works for effective drowsiness detection with high precision with android modules. The algorithm of Deep CNN was used to detect eye blink and its state recognition as provided by Sanyal and Chakrabarty [12]. Saleh et al. [13] developed an algorithm of LSTM and Recurrent Neural Networks (RNN) to classify driver's behaviors through sensors. Ed-Doughmi et al. [14] analyzed the driver's behaviors through the RNN algorithm. It specially focuses on construction of real-time fatigue detection to prevent roadside accidents. This system formulates a number of drivers' faces, which works on multilayered 3D CNN models to identify drowsy drivers and provide 92 percentage acceptance rate.

*FPGA-Based Drowsiness Detection System.* A low-intrusive drowsiness detection system using field-programmable gate array (FPGA) has been designed by Vitabile et al. [15]. This system focuses on bright pupils of eyes which are detected by IR sensor light

source embedded in a vehicle. Due to this visual effect, the retinas

Wireless Communications and Mobile Computing identified up to 90%, which helps to find drivers' eyes for analyzing drowsiness through a number of frames for avoiding serious mishaps. Navaneethan et al. [16] implemented a real-time system to track human eyes using cyclone II FPGA.

*Eye Recognition System Based on Wavelet Network Algorithm.* Jemai et al. [17] introduced a technique for drowsy warning system using wavelet networking. That network tracks eyes with the help of classifying algorithms like Wavelet Network Classifier (WNC) that relies on Fast Wavelet Transform (FWT), which specifically leads to binary way decision (conscious or not). The physiological aspects are heart beat rate and electrocardiogram that are repeatedly extracted through wavelet transformation with regression technique for fatigue detection, designed by Babaeian et al. [18]. This principle worked on heart rate data classification through wavelet network which can find an average way of drowsiness alert system.

*Fatigue Detection Using Vehicle State (Steering Wheel) Algorithm.* Arefnezhad et al. [19] proposed a noninterfering drowsy detection system based on vehicle steering data using neurofuzzy system with support vector machine and particle swarm optimization algorithm. Mutya et al. [20] established a system to resolve the problem of drowsiness using steering wheel algorithm. It is basically based on image-formed or pictorial-based steering movement and the CNN algorithm for proper classification of drowsiness, which can also reduce false drowsy detection rates.

*Drowsy Alert System Designed Using Electroencephalography (EEG), Electrocardiography (ECG), Electrooculogram (EOG), and Electromyogram (EMG) Algorithm.* Budak et al.

[21] designed a drowsy detection system through EEG technique which is designed with various components like AlexNet method, VGGNet method, and wavelet transform algorithm. This process effectively analyses the state of sleepiness using the brain indicator signal (EEG), camera, and sensors that are activated with the help of machine learning method to alert drowsy driver. Hayawi and Waleed [22] proposed a method to observe drowsiness through signal of Heart Rate Variability (HRV) which is obtained using EEG sensors. Hayawi and Waleed [23] established an intrusive method for measuring eyeball movement using EOG technique to construct a fatigue alert system that is also embedded with an Arduino controller board with  $K$  Nearest Neighbors (KNN) classifier to improve the percentage of accuracy. Song et al. [24] proposed a system to identify the fatigue of driver through the movement of muscular skin of eyes which is processed using EMG sensors with the help of a human machine interface. Similarly, the closure of eyelids and muscle part movements are also observed through the EMG sensors signals that function with the help of ESP8266 to provide or monitor the drowsy data on the Internet, which is designed by Artanto et al. [25]. Ma et al. [26] designed a driving fatigue detection system by measuring the EEG signals. It provided a robust platform for detecting

drowsiness which is based on a deep learning process to find the accuracy of fatigue through EEG signals. But the deep learning process is structured through a principal component analysis network (PCANet) that preprocesses EEG data to create accuracy of detection. This process was tested in small sample size and offline mode, but it violates the accuracy in a large population of samples in real-time situations. Due to that reason, the IoT module is used to test online or offline in large sample sizes. Ma et al. [27] proposed an efficient application for the detection of driver fatigue through facial expression. Here, the facial movement is observed by deep learning of multiblock local binary patterns (MB-LBP) and AdaBoost classifier. But it is also used to accurately and quickly detect drowsiness with the help of a fuzzy inference system. When the driver wears a glass, then the accuracy of detection is decreased. So IoT modules are used to make it more intelligent and to improve accuracy level of fatigue detection.

*Fatigue and Collision Alert System Using Smart Technique.* Chen et al. [28] implemented a smart glass to detect fatigue. The rear light of the vehicle is automatically flashed with a message being sent using the IoT module or cloud environment. Kinage and Patil [29] proposed a system to detect the drowsiness using eye blinking sensors and any accidents or collisions that happened; then, the vibration sensor was integrated with heart rate measurement sensor for forwarding alert message to the authorized user. So, it is also attached to the GPS and GSM device for tracking the location and transmission of message. Siva Reddy and Kumari [30] introduced a system to control cause of unconditional mishaps using Arduino board with sensors which operated through camera. But, it is an efficient system with less estimation cost for construction of it. Jang and Ahn

[31] implemented a system to detect an alcohol addict and drowsy drivers through sensors, where these elements are integrated with the Raspberry Pi controller module. So, the IoT modules are also used to send messages for any abnormal driver activities, which are properly invigilated with the help of a webcam (image processing) and controller unit. A new process has been developed for regular vigilance of facial detection and eye blink state, which predicts the driver's drowsiness. In addition to extra sensors, voice recognition application and machine learning methods are used to enhance the process of alert [32].

In the existing system, the fatigue of the driver is calculated through the eye or facial movements, deep learning, FPGA-based, ECG or EEG or EOG, vehicle steering movement, etc. But the implementation of the IoT-based technique helps to smartly control the various issues of driver drowsiness by the automatic buzzing of alarm, easily tracing the mishap location, and warning to the owner by sending emails or messages.

### 3. Component Description

The proposed system is made up of the following primary components.

*Pi Camera Module V2.* This Raspberry Pi Camera module v2 can be used instead of the original camera module, to capture high-definition video with still picture with the help of Sony IMX219 8-megapixel sensor [33, 34]. It operates with 1080p30, 720p60, and VGA90 video modes which connect to a CSI port via a 15 cm ribbon cable on the Raspberry Pi module that is depicted as in Figure 1(a).

*Raspberry Pi3 Model B Module.* It is a credit card-sized computer which is embedded with Quad Core 1.2 GHz Broadcom BCM2837 64-bit CPU, 1 GB RAM, and other parts. Raspbian OS is integrated in it, and its microprocessor is designed for Windows operating system [34] that is shown in Figure 1(b).

*Speaker.* It is an audio- or voice-generating device which can convert electromagnetic waves into sound [35]. If the driver's drowsiness is detected, then a voltage is supplied as an alert to generate regular programmed voice sound [36], which is depicted as in Figure 1(c).

*Crash Sensor.* It is an electronic equipment that provides safety and detects an impact through vibrations. Due to drowsiness, any type of crash may happen; then, it detects the impact of collision [37] as shown in Figure 2(a).

*Force Sensitive Resistor Sensor (FSR).* Special types of resistor known as force sensitive resistor, whose resistance may differ, according to pressure or force on it. So, it is made by a thick polymer sheet which is operated in different environments [38]. Due to good shocking resistance, it is integrated in this system as depicted in Figure 2(b).

*GPS Module.* This module is used to track the vehicle's location, if any serious collision occurs with a drowsy driver. It is the main component to detect the latitude and longitude of any area on the Earth with date and time from satellite. In this system, the mishap location is easily traced and the location is forwarded using GPS [39] as depicted in Figure 2(c).

## 4. Proposed System

The proposed system here is designed to minimise the occurrence of countless mishaps due to the drowsy driver. Nowadays, fatigue of driver causes road accidents every now and then across the world. So, these activities should be required to automatically handle an implementation of smart alert system or vigilance in a vehicle which is an objective of this system. To analyze different behavioral or visual-based attitudes of the driver, face movement and eye blink are measured to study the state of the driver. Here, eye blink is mainly focused to detect drowsiness of the driver. The threshold value of an EAR lies above 0.25 without any effect of exhaustion. When a driver automatically shuts down, then the threshold value of EAR falls below the given range. A threshold value of drowsy eye blink sample represents the number of video frames of the driver's closed eyes. If the consecutive count-

Wireless Communications and Mobile Computing  
ing frames increase above the range of the threshold value,  
then the drowsiness of the driver is detected. Here, a  
Pi

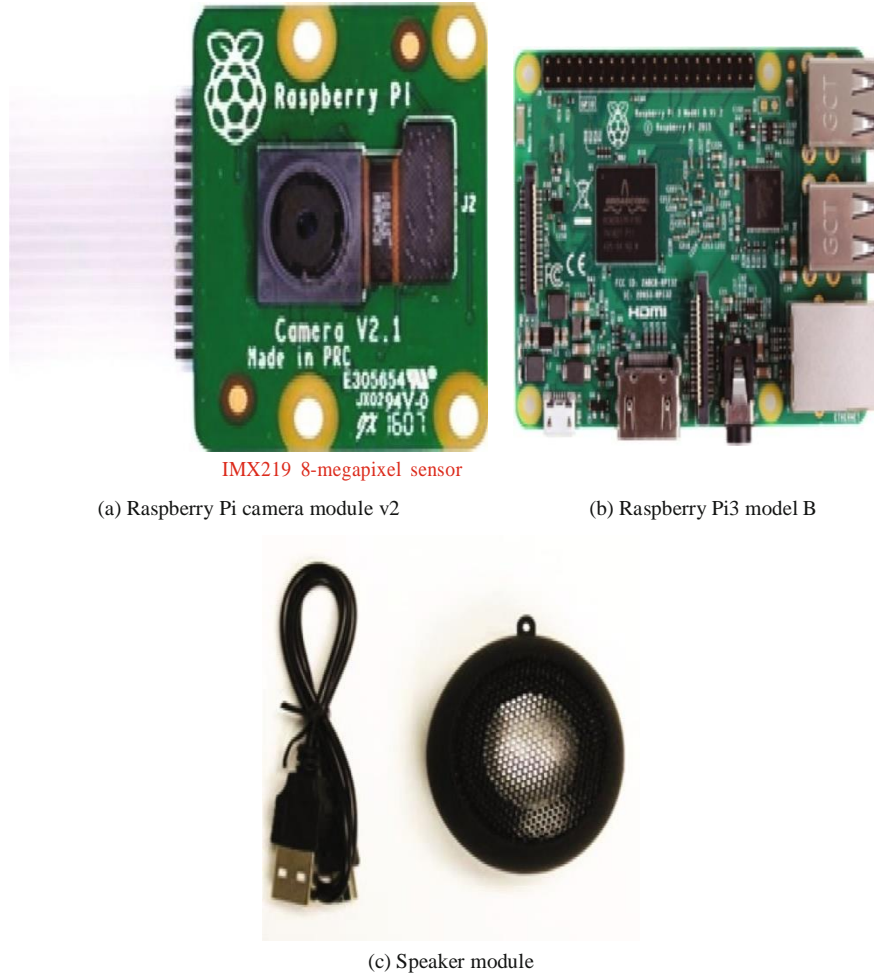


Figure 1: Basic requirement with IoT devices.

camera is used to regularly record the total movement of an eye through which the threshold value of an EAR is calculated. A counter is also included in it for counting occurrence of frames. Suppose that it exceeded above a range of 30. In that case, a voice is activated by a speaker and a mail is automatically sent to an authorized person of the vehicle which is generally processed at the time of drowsiness detection. The described modules work properly through Raspberry Pi3 which is programmed in Python programming language. Figure 3 depicts a test scenario of the proposed system.

The mail is received by an authorized one, who can alert the sleepy driver by ringing to him, if that drive is still not awake after turning on the voice alert message in the speaker. Thus, this process is successfully performed to detect a driver's drowsiness and detect the collision impacts due to braking of the vehicle through crash sensor and Force Sensitive Sensor (FSS). Due to the occurrence of the collision, the data collected from sensors and the location data are sent to an authorized person (owner) or any near hospitals with the help of a GPS module (Google Maps link).

## 5. Proposed Methodology

When the Pi camera model V2 is successfully integrated with Raspberry Pi3, it continuously records each movement of the driver's face. This proposed work specially focuses on behavioral measures of the driver with severity measurement of collision in following sections. The EAR is accurately calculated due to the use of Raspberry Pi3 model B and Pi camera modules to make a persistent recording of face landmarks that are localized through facial landmark points. But the Raspberry Pi3 model B and Pi camera modules are securely processed due to the operating system of the controller and predictable secure shell (SSH) keys. The use of SSH host keys provides secure network communications and helps to prevent unauthorized communications or file transfers. The IoT-based application is being developed through the integration of some IoT modules like wireless sensors, GPS tracker, Pi camera, and smart code for detecting drowsiness of the driver. So the above modules are properly integrated with the Raspberry Pi controller module that intelligently controls and smartly warns a drowsy driver. The successful



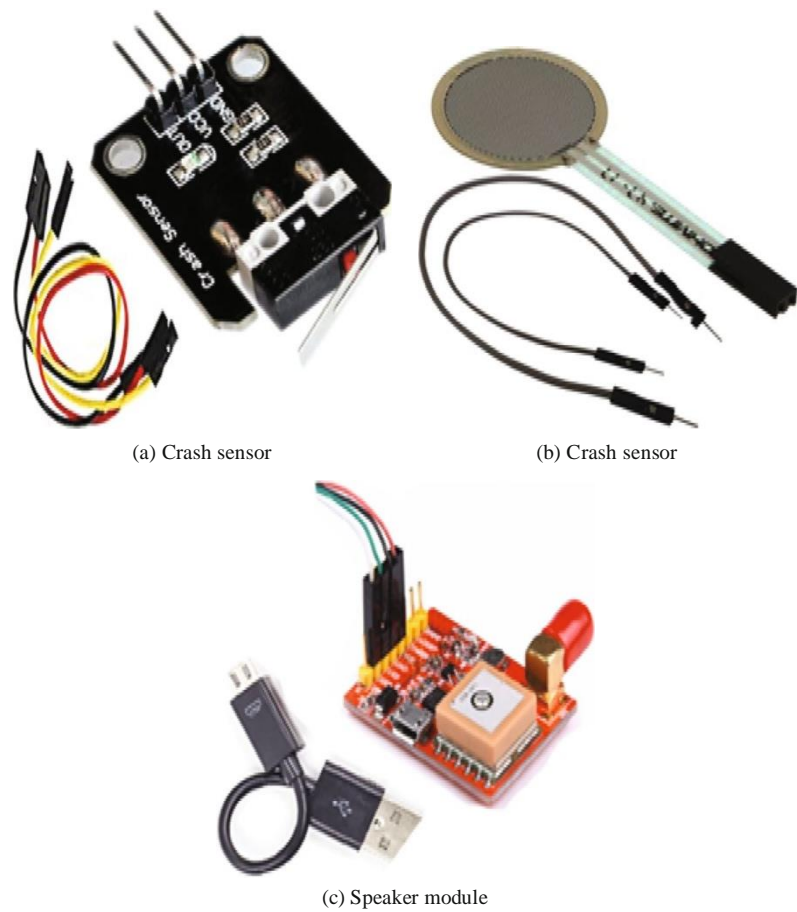


Figure 2: Basic sensor device requirement.

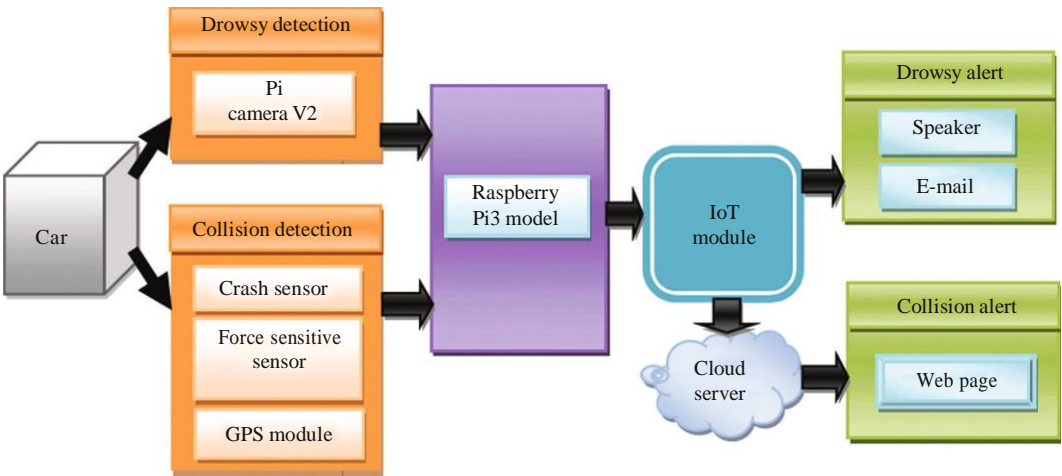


Figure 3: The proposed system: drowsy detection to drowsy alert.

integration of IoT modules is robustly used to prevent the cause of mishaps and also warns the drowsy driver to avoid careless driving. The Internet of Things (IoT) is helping to manage various real-time complexities like handling complex sensing environments and also provides a very flexible platform to control multiple connectivities.

The IoT module is a very reliable way of capturing images of the drowsiness of the driver as well as sending an alert message to the owner for awareness.

*Drowsy Eyes and Face Detection.* The step by step methods for detection of drowsy drivers in the scenario are

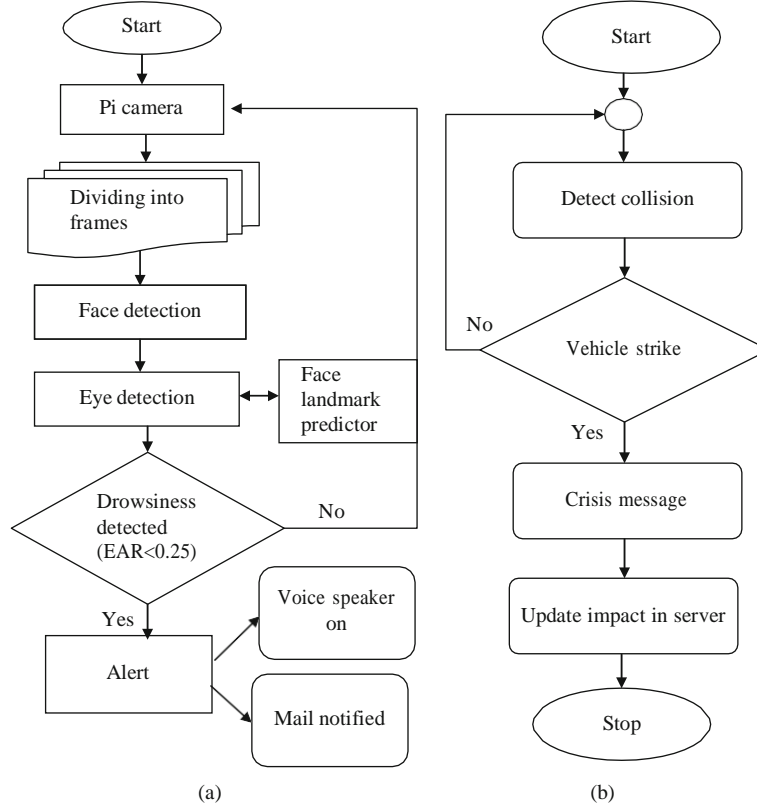


Figure 4: A flowchart of the (a) drowsy detection system and (b) collision detection system.

previously explained in Figure 4, and the various steps are as follows:

- (i) Step 1: video recording
- (ii) Step 2: face detection
- (iii) Step 3: eye detection
- (iv) Step 4: drowsiness detection (combination of steps 2 and 3)

The Raspberry Pi3 camera is an 8-megapixel video camera which captures continuous video stream in good quality. Recorded video stream is converted into a number of frames which are forwarded to face detection step. The face detection is analyzed by face landmark algorithm, which can detect eye, mouth, and nose from facial appearance. Actually, this detection technique works with Python-based library package like OpenCv. OpenCv is used for real-time image processing which is implemented by computer vision algorithm. When facial features are successfully detected, the next step, like eye detection of drowsy drivers, is possible to focus through facial landmark predictor algorithms. So, it can convert that image frame format to gray scale level, where detected eye areas are traced by six coordinates as depicted in Figure 5. Now, it is needed to calculate EAR which measures the distance between vertical and horizontal eye landmark points by using a Euclidean distance (ED) method as in

Equation (1). The calculation of the distance of the eyelid section is made as

$$ED = \sqrt{\sum_{n=1}^6 (X_i - Y_i)^2} \quad (1)$$

where  $ED$  is denoted as a Euclidean distance between  $X_i$  and  $Y_i$ ; these are two Cartesian coordinate points. So it is represented in Python program which is explained below:

- (i)  $A = \text{dist.euclidean}(\text{eye}[2], \text{eye}[6])$
- (ii)  $B = \text{dist.euclidean}(\text{eye}[3], \text{eye}[5])$
- (iii)  $C = \text{dist.euclidean}(\text{eye}[1], \text{eye}[4])$

From the above statement,  $\text{dist}$  is an object of distance package that belongs to *scipy* library file which is invoked a Euclidean  $\sqrt{X_i^2 + Y_i^2}$ . Here, there are two coordinate points of an eye landmark. The variables ( $A$ ,  $B$ , and  $C$ ) are used for calculation of the EAR value of an eye. In order to find the EAR by using detected eye landmark coordinate values from every video frames, it is being computed as

$$EAR = \frac{kx_2 - x_6k + kx_3 - x_5k}{2kx_1 - x_4k} \quad (2)$$



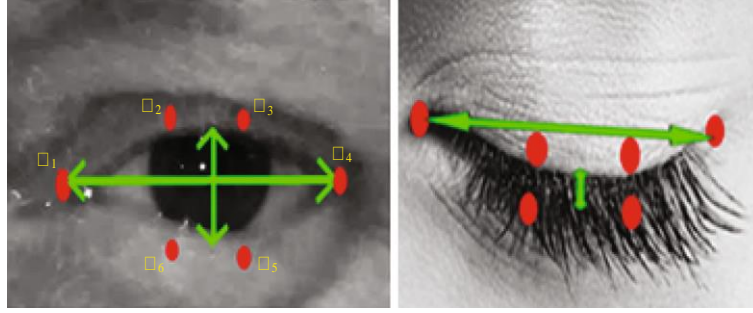


Figure 5: Eye landmarks (6 coordinates) of opening and closing eye.

where  $x_1$ ,  $x_2$ ,  $x_3$ ,  $x_4$ ,  $x_5$ , and  $x_6$  are 6 landmark coordinates which are shown in Figure 2(b). The EAR value of the left and right eyes is averaged during synchronous eye blink. The threshold value of EAR remains constant when both eyes continue to be open, where values will be randomly changed during eye blink. The threshold range of EAR is above 0.25 that means the driver's eyes are unlatched. If any drowsiness of the driver is detected from video frames, it happens due to drop of its threshold value below 0.25. When the number of frames is more than 30, then the voice speaker is turned on and a mail is sent. So, the Raspberry Pi3 is programmed to operate a speaker by using a software which can convert text-to-speech (TTS) conversion technique. TTS in speaker is more reliable for a sleepy driver to be alerted than other sound. All the statements of the speaker are coded as follows:

The above code is having a text message like wake up; it will be converted to speech by *espeak* synthesizer and operating system (os) module in Python.

The Raspberry Pi also forwards a mail at the time of fatigue detection to the driver to be more vigilant to avoid a number of mishaps. The mails are transferred through Simple mail Transfer Protocol (SMTP) that is accessed as a predefined library *smtplib* in Python. The SMTP client session object is created by using *smtplib* module to send mail source to destination. Here, the MIME protocol is used to define a multipart section of a mail like subject, to and from parts. The mail is securely transferred from drowsy driver unit to an authorized person (owner of vehicle or any) through Transport Layer Security (TLS) and server port number 587 for <http://smtp.gmail.com> server, which is shown in program section Figure 6.

**Collision Detection.** The collision alert system is generally constructed by a combination of crash sensor and force sensitive register for providing data about collision impact [40]. So, the data related to impacts of collision are collected to measure the severity of mishaps due to drowsiness or unconscious driving. The cause of severity is monitored through following stages:

- (i) Stage 1: Python script
- (ii) Stage 2: local database
- (iii) Stage 3: update server (web page)

```

1  while alarm_status: do
2      print('call')
3      s = 'espeak' '+' wake up '+'
4      os.system(s)
5  end

```

Code 1

The IoT module or server forwards detected data to program script which is based on Python. Here, Python script is specially preferred for reliable communication with connected sensors. When any crashes of vehicle occur, then the respective sensors generate information which detects the range of severity as well as the location of collision that are recorded in database which is shown in Figure 4. The location of accident is fetched from Google Maps link with longitude, latitude, time, and date that are accessed through the Raspberry Pi module. The longitude and latitude of a location are specially calculated by using the haversine technique to measure the distance of earth surface between two points which is represented as in Equations (3) to (5):

$$x = \sin^2 \frac{\Delta\phi}{2} + \cos \phi_1 * \cos \phi_2 * \sin^2 \frac{\Delta\lambda}{2}, \quad (3)$$

$$y = 2 * x \tan 2 \frac{\phi_1 - \phi_2}{2}, \quad (4)$$

$$z = \text{Rad} * y, \quad (5)$$

where  $\phi$  is denoted as latitude,  $\lambda$  is denoted as longitude, and earth's radius is denoted as Rad. Here, the mean radius is represented as 6,371 km.

The GPS is automatically tracked by using this formula in Python script which perfectly puts the values of latitude and longitude of the location where the accident occurred. So, the tracking script is implemented by using a reserved library package, i.e., *pynmea* which is used for *NMEA 0183* protocol. This protocol provides an interface to parse date, time, altitude, longitude, latitude, and location pin code with Google Maps links, which is programmatically represented as follows:

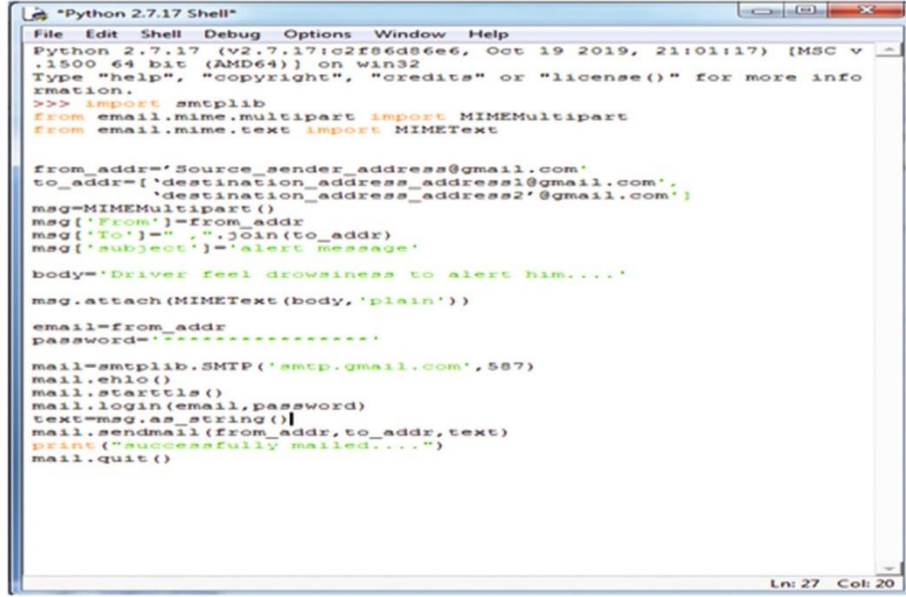


Figure 6: Source Code for sending mail to Server/Owner.

```
from pynmea import nmea
import serial, time, sys, threading, datetime, shutil;
message = pynmea.parse "$($GPGGA, 172245.25, 20.4377,
S, 85.8997, E, 2.09, 3.7, 110.00, M, -36.9,M,00013*7D")
```

Code 2

Here, line 3 is denoted as a combination of some parameters which are listed in Table 1.

This indicates the serial library for the port number of the GPS module; *shutil library* was used for copying *nmea* file to other text file in script, and other libraries are used for some additional activities in the system. These parameters are parsed through *pynmea library* object to create plot location of vehicle accident due to drowsiness or unconscious mind. After these parameters are retrieved from mishap location, they are forwarded to the database (*MySQL, Oracle*, etc.) for remote visualisation by everybody or authorized one (owner or hospital authority). So, the database is updated by fetched data; then, it is displayed in the web page of official websites which belong to stages 2 and 3 in this detection system. Simultaneously, an emergency message (URL link of collision web portal) is sent to a nearer hospital or owner of the vehicle for taking steps to recover from that situation. Figure 7 represents the collision impact with map program section.

## 6. Result Discussion and Performance Analysis

This section provides a successful experimental consequence which is obtained from the proposed method at the time of driving. The drowsiness is calculated through the observation of EAR of the driver. This process provides the status of eyes if they are open or closed thereby providing data about collision impact on the website.

TABLE 1: Parameter of pynmea package.

Parameter	Value
Date time	17-22:45:25
Latitude	20.4377
Longitude	85.8997
Altitude	110.00
Station_ID	00013*7D
GEO_SetupID	M

*Prediction of Drowsiness.* When the EAR value is greater than 0.25, it indicates the eyes are open. This test shows that the driver has not fallen into drowsiness that is graphically represented in Figure 8 and detected the face is not recognized as a drowsy one which is shown in Figure 9. Similarly, the drowsiness of the driver is detected due to the EAR value being less than 0.25 as graphically plotted in Figure 8 and finally, a drowsy face is detected as shown in Figure 9. The EAR values are frequently changed due to movements of the eyelids as shown in above figure. When the drowsiness is found, then the driver is alerted with repeated voice sound and an email message is forwarded to the owner or related authority. Here, speech speaker is implemented instead of buzzer for more vigilance; if it fails, then the owner will provide any warnings after receiving the message from its mail as represented in Figure 10.

*Impact of Collision.* Due to the drowsiness or unintentional driving of a driver, it causes a road accident which is a serious problem. So there is a need to find the status of severity through sensors as well as the most important thing which is to trace the location of accident for urgent hospital service.

So, the emergency message is also provided to a nearer hospital authority or owner to take the right precaution. That

```
Python 2.7.17 Shell
File Edit Shell Debug Options Window Help
Python 2.7.17 (v2.7.17:02f8ed5ee6, Oct 19 2019, 21:01:17) [MSC v.1500 64 bit (AMD64)] on win32
Type "help()", "copyright()", "credits()" or "license()" for more information.
>>> from pyntv import nmea
import matplotlib.pyplot as plt
import serial, time, sys, threading, datetime, shutil

TRX = -105.1621
TRY = 40.0868
BLK = -105.2898
BLV = 40.001
BAUDRATE = 4800
lat_input = 0
long_input = 0
def altitude(i):
    global alt, i
    f1 = open('temp.txt', 'w')
    f1.truncate()
    shutil.copyfile('nmea.txt', 'temp.txt')
    f1.close()
    f1 = open('temp.txt', 'r')
    cxy = 0
    for line in f1:
        if (line[4] == 'G'):
            if (len(line) > 50):
                gpqga = nmea.GPQGA()
                gpqga.parse(line)
                alt = gpqga.antenna_altitude
                i += 1
                print(i)
                print(alt)
                plt.scatter(x=[i], y=[float(alt)], s=1, c='r')
    finally:
        f1.close()
    i = 0
plt.ylabel('meters')
plt.xlabel('counts')
```

Figure 7: Collision impact with map program section.

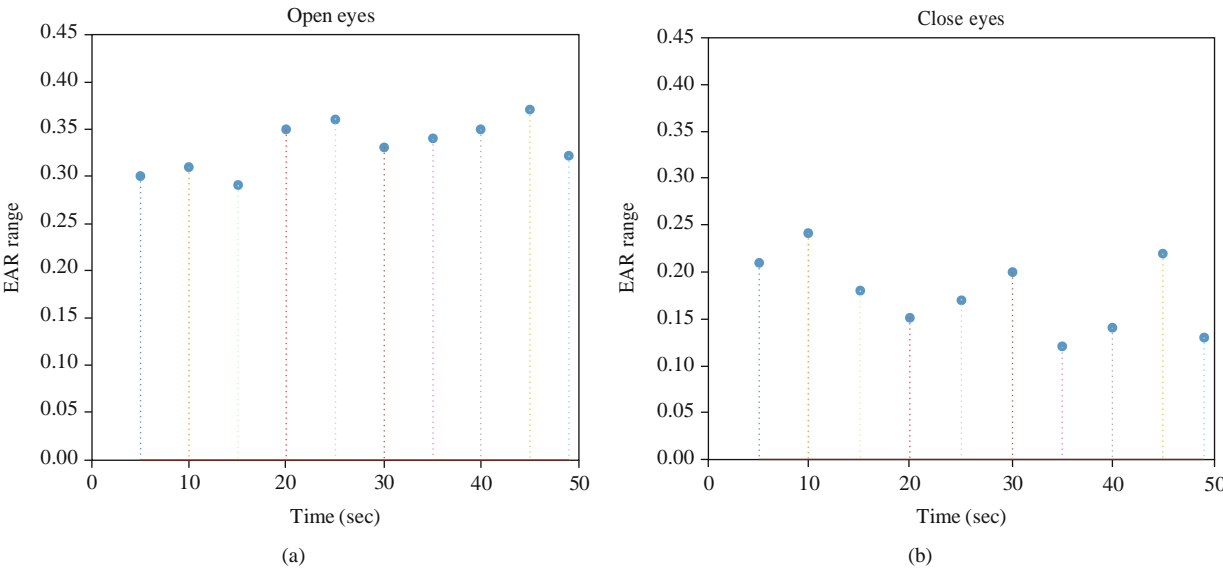


Figure 8: EAR mapped: (a) when eyes are opened; (b) when eyes are closed.

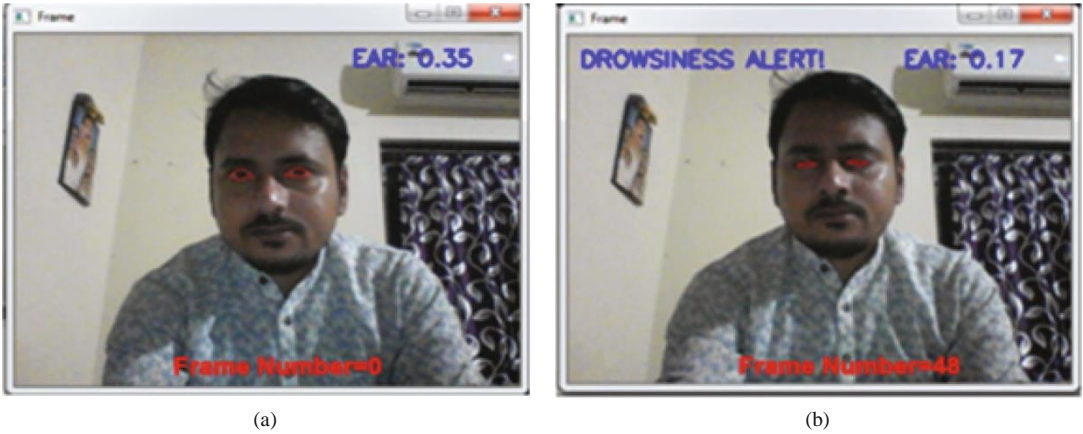


Figure 9: Face detection: (a) not a drowsy face; (b) drowsy face.

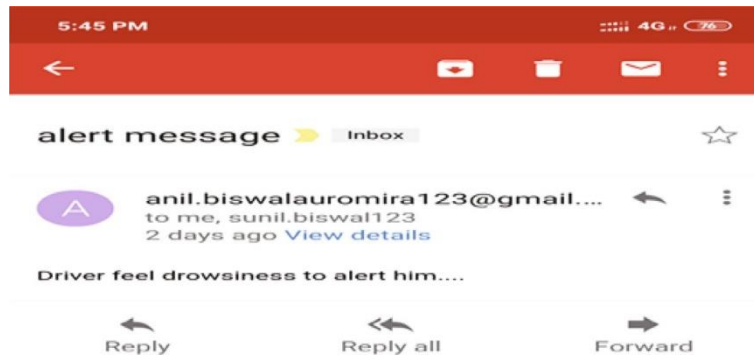


Figure 10: Alert email received on mobile.

impact_severity	latitude	logitude	time	location_pincode	google_maps_pos
high	20.4377	85.8997	2020-04-17 22:45:25	753013	https://www.google.com/maps/@20.4377008,85.899782,15z
medium	20.4377	85.8997	2020-04-17 22:45:25	753013	https://www.google.com/maps/@20.4377008,85.89981,15z
medium	20.4377	85.8997	2020-04-17 22:45:25	753013	https://www.google.com/maps/@20.4377008,85.89981,15z
high	20.4377	85.8997	2020-04-17 22:45:25	753013	https://www.google.com/maps/@20.4377008,85.89981456,15z

Figure 11: Collision impact measuring system generated data on the web page.

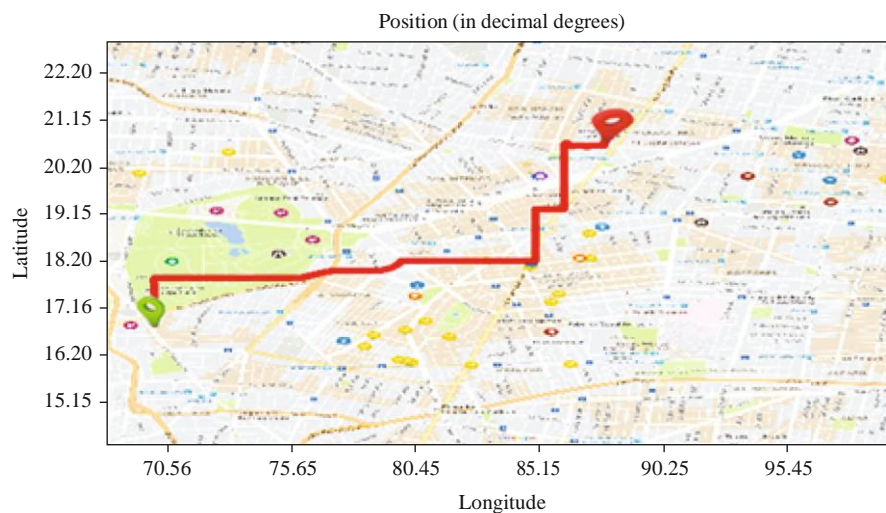


Figure 12: Map locates the place of collision.

message is interpreted from a specific web portal (web page) with brief data about the mishap which is shown in Figure 11. Here, this web page is formed using *Glassfish* local server (JSP page) and it also facilitates to manage multiple accidents happening nearly at the same time period. Whenever the collision location place is traced from the web portal, then it is specifically located in a map link as shown in Figure 12. This map locates the source or traced location

in a green bar as well as destination or severity location in a red bar.

*Performance Analysis.* This system experimented on ten different persons with samples on various conditions like using glasses, instead of glasses as an alternative view. So, the performance is measured on different levels such as eye accuracy and drowsiness accuracy detection. So, the blinking



TABLE 2: Performance analysis of the proposed system in view of 500-frame test in various conditions.

Face condition	TED	FED	TAD	FAD	TMD	FMD	Eye accuracy detection	Drowsy accuracy detection
Without glasses	498	2	497	3	495	5	99.6%	99.2%
With glasses	489	1	485	15	486	14	97.8%	97.1%
Alternative view	478	2	474	26	477	23	95.6%	95.1%
Finally	1465	5	1456	44	1458	42	97.7%	97.1%

eyes and drowsiness were detected on 500 frames which are used to observe accuracy from video streams. The overall analysis was performed on frames with TED (Truly Eye Detect), FED (Falsely Eye Detect), TAD (Truly Alarm Detect), FAD (Falsely Alarm Detect), TMD (Truly Mail Detect), and FMD (Falsely Mail Detect).

From Table 2, the performance of eye accuracy and drowsiness can be accurately calculated and are measured by given data set frames on various situations. We consider Eye Accuracy Detection as EAD and Drowsy Accuracy Detection as DAD.

$$EAD = \frac{TED}{\delta TED + FED}, \quad \delta 6b$$

$$DAD = \frac{\delta TAD + TMD}{\delta TAD + FAD + TMD + FMD} : \quad \delta 7b$$

From the overall analysis of tested result, it can be concluded that using face landmarks does not change under any conditions and its performance is more accurate than the cascaded method. But, this process takes a little bit more time to load at deep night vision than the cascaded technique. Thus, this proposed system has provided an efficient or successful drowsy detection using the facial landmark method as well as another interface for detection of collision due to drowsiness or unconscious state of driving. However, it is tested on 10 distinct people with various situations and detection of drowsiness warned through speaking alarm with an attached mail sent to the owner or authority whereas the random data set forwarded to this system to check its functionality and it performed in 97.1% accuracy. Due to this reason, this system can be more useful for monitoring the fatigue of the driver and impact of collisions.

## 7. Conclusion

This research provides a robust method for detecting drowsiness of drivers and collision impact (severity) system in the present time. This method generally combines two different systems in one integrated system. But, the existing techniques are based on psychological or vehicle-based approach to detect drowsiness of drivers and also, the severity of collision is separately measured, but such technique is highly intruding as well as fully turns on the physical environment. So, the proposed system is used to construct a nonintruding technique for measuring drowsiness of the driver with severity of collision due to braking or mishap. This system's main components are the

Raspberry Pi3 model B module and Pi camera module that are used for persistent recording of face landmarks that are localized through facial landmark points then to calculate EAR. However, if the calculated EAR value increases from the threshold range, then the eyes are kept open and no change in the state of system occurs. Similarly, if the EAR value falls from the threshold range, then the system urgently alerts using speech speaker and warning e-mail to the authority (owner) for extra supportive alertness to the driver. In addition, measurement of collision severity (impact) is made through implementation of sensors with the GPS module to properly track the location of accident thereby alerting the nearer medical service centre to serve emergency diagnosis.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work was partially supported by the Ministry of Science and Technology and National Applied Research Laboratories of Taiwan under Grant MOST 109-2218-E-009-010 and NARL-ISIM-109-006.

## References

- [1] E. Vural, M. Cetin, A. Ercil, G. Littlewort, M. Bartlett, and J. Movellan, "Drowsy driver detection through facial movement analysis," *International Workshop on Human-Computer Interaction*, vol. 4796, 2007.
- [2] R. Jabbar, M. Shinoy, M. Kharbeche, K. Al-Khalifa, M. Krichen, and K. Barkaoui, "Driver drowsiness detection model using convolutional neural networks techniques for android application," in *2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT)*, IEEE, 2020.
- [3] G. Turan and S. Gupta, "Road accidents prevention system using drivers drowsiness detection," *International Journal of Advanced Research in Computer Engineering Technology*, 2013.
- [4] S. Gupta and E. Garima, "Road accident prevention system using driver's drowsiness detection by combining eye closure and yawning," *International Journal of Research*, pp. 839–842, 2014.
- [5] R. Malekian, A. F. Kavishe, B. T. Maharaj, P. K. Gupta, G. Singh, and H. Waschefort, "Smart vehicle navigation system using hidden markov model and RFID technology," *Wireless Personal Communications*, vol. 90, no. 4, pp. 1717–1742, 2016.

- [6] D. Mollicone, K. Kan, C. Mott et al., "Predicting performance and safety based on driver fatigue," *Accident Analysis Prevention*, vol. 126, pp. 142–145, 2019.
- [7] D. Singh, B. Pati, C. R. Panigrahi, and S. Swagatika, "Security issues in iot and their countermeasures in smart city applications," *Advanced Computing and Intelligent Engineering*, pp. 301–313, 2020.
- [8] M. Miranda, A. Villanueva, M. J. Buo, R. Merabite, S. P. Perez, and J. M. Rodriguez, "Portable prevention and monitoring of driver's drowsiness focuses to eyelid movement using internet of things," in *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, pp. 1–5, IEEE, 2018.
- [9] O. Khunpisuth, T. Chotchinasri, V. Koschakosai, and N. Hnoohom, "Driver drowsiness detection using eye-closeness detection," in *2016 12th International Conference on Signal-Image Technology & Internet-Based Systems (SITIS)*, pp. 661–668, IEEE, 2016.
- [10] S. Abraham, T. Luciya Joji, and D. Yuvaraj, "Enhancing vehicle safety with drowsiness detection and collision avoidance," *International Journal of Pure and Applied Mathematics*, pages, pp. 2295–2310, 2018.
- [11] M. Y. Hossain and F. P. George, "IOT based real-time drowsy driving detection system for the prevention of road accidents," in *2018 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, pp. 190–195, Bangkok, 2018.
- [12] R. Sanyal and K. Chakrabarty, "Two stream deep convolutional neural network for eye state recognition and blink detection," in *2019 3rd International Conference on Electronics, Materials Engineering & Nano-Technology (IEMENTech)*, pp. 1–8, Kolkata, India, 2019.
- [13] K. Saleh, M. Hossny, and S. Nahavandi, "Driving behavior classification based on sensor data fusion using lstm recurrent neural networks," in *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1–7, IEEE, 2017.
- [14] Y. Ed-Doughmi, N. Idrissi, and Y. Hbali, "Real-time system for driver fatigue detection based on a recurrent neuronal network," *Journal of Imaging*, vol. 6, no. 3, p. 8, 2020.
- [15] S. Vitabile, A. De Paola, and F. Sorbello, "A real-time non-intrusive fpga-based drowsiness detection system," *Journal of Ambient Intelligence and Humanized Computing*, vol. 2, no. 4, pp. 251–262, 2011.
- [16] S. Navaneethan, N. Nandhagopal, and V. Nivedita, "An FPGA-based real-time human eye pupil detection system using E2V smart camera," *Journal of Computational and Theoretical Nanoscience*, vol. 16, no. 2, pp. 649–654, 2019.
- [17] O. Jemai, I. Teyeb, and T. Bouchrika, "A novel approach for drowsy driver detection using eyes recognition system based on wavelet network," *International Journal of Recent Contributions from Engineering, Science & IT*, vol. 1, no. 1, pp. 46–52, 2013.
- [18] M. Babaeian, K. A. Francis, K. Dajani, and M. Mozumdar, "Real-time driver drowsiness detection using wavelet transform and ensemble logistic regression," *International Journal of Intelligent Transportation Systems Research*, vol. 17, no. 3, pp. 212–222, 2019.
- [19] S. Arefnezhad, S. Samiee, A. Eichberger, and A. Nahvi, "Driver drowsiness detection based on steering wheel data applying adaptive neuro-fuzzy feature selection," *Sensors*, vol. 19, no. 4, p. 943, 2019.
- [20] K. Mutya, J. Shah, A. D. McDonald, and J. Jefferson, "What are steering pictures are worth? using image-based steering features to detect drowsiness on rural roads," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 63, no. 1, pp. 2041–2045, 2019.
- [21] U. Budak, V. Bajaj, Y. Akbulut, O. Atila, and A. Sengur, "An effective hybrid model for EEG-based drowsiness detection," *IEEE Sensors Journal*, vol. 19, no. 17, pp. 7624–7631, 2019.
- [22] A. A. Hayawi and J. Waleed, "Driver's drowsiness monitoring and alarming auto-system based on EOG signals," in *2019 2nd International Conference on Engineering Technology and its Applications (IICETA)*, pp. 214–218, IEEE, 2019.
- [23] M. S. Song, S. G. Kang, K. T. Lee, and J. Kim, "Wireless, skin-mountable EMG sensor for human-machine interface application," *Micromachines*, vol. 10, no. 12, p. 879, 2019.
- [24] D. Artanto, M. P. Sulistyanto, I. D. Pranowo, and E. E. Pramesta, "Drowsiness detection system based on eye-closure using a low-cost EMG and ESP 8266," in *2017 2nd International conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, pp. 235–238, IEEE, 2017.
- [25] L. B. Chen, W. J. Chang, J. P. Su et al., "A wearable-glasses-based drowsiness-fatigue-detection system for improving road safety," in *2016 IEEE 5th Global Conference on Consumer Electronics*, p. 12, IEEE, 2016.
- [26] A. S. A. Bakar, G. K. Shan, G. L. Ta, and R. A. Karim, *Iot/eye Drowsiness Detection System by Using Intel Edison with GPS Navigation*, pp. 485–493, 2018.
- [27] Y. Ma, B. Chen, R. Li et al., "Driving fatigue detection from EEG using a modified PCANet method," *Computational Intelligence and Neuro-science*, vol. 2019, pp. 1–9, 2019.
- [28] S. M. Gowri, P. Anitha, D. Srivaishnavi, and M. Nithya, "Internet of things based accident detection system," in *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pp. 159–163, IEEE, 2019.
- [29] V. Kinage and P. Patil, "Iot based intelligent system for vehicle accident prevention and detection at real time," in *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pp. 409–413, Palladam, India, 2019.
- [30] R. V. Siva Reddy and P. A. Kumari, "Internet of things (IoT) based multilevel drunken driving detection and prevention system using Raspberry Pi 3," *International Journal of Computer Science and Information Security (IJCSIS)*, pp. 131–137, 2020.
- [31] S. W. Jang and B. Ahn, "Implementation of detection system for drowsy driving prevention using image recognition and iot," *Sustainability*, vol. 12, no. 7, p. 3037, 2020.
- [32] D. Singh and B. K. Pattanayak, "Markovian model analysis for energy harvesting nodes in a modified opportunistic routing protocol," *International Journal of Electronics*, vol. 107, no. 12, pp. 1963–1984, 2020.
- [33] S. S. Kulkarni, A. D. Harale, and A. V. Thakur, "Image processing for driver's safety and vehicle control using raspberry Pi and webcam," in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, pp. 1288–1291, Chennai, 2017.
- [34] Y. Zhang, F. Weninger, and B. Schuller, "Cross-domain classification of drowsiness in speech: the case of alcohol intoxication and sleep deprivation," in *Interspeech 2017*, pp. 3152–3156, IEEE, 2017.



- [35] D. Samanta, M. Shivamurthaiah, P. Kumar, D. Umesh, and K. Siddalingappa, "Wireless sensor networks model for monitoring system based on iot," *Solid State Technology*, 2020.
- [36] R. Gurunath, M. Agarwal, A. Nandi, and D. Samanta, "An overview: security issue in IoT network," in *2018 2nd International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2018 2nd International Conference on*, IEEE, 2018.
- [37] V. Pavitra, P. V. Rao, M. D. Gagana, and D. Samanta, *Internet of Things (iot): An Assessment*, 2015.
- [38] A. Dasgupta, B. Kabi, A. George, S. L. Happy, and A. Routray, *Cross-Domain Classification of Drowsiness in Speech: The Case of Alcohol Intoxication and Sleep Deprivation*, 2015.
- [39] S. R. Bhagyashree, T. S. Singh, J. Kiran, and L. S. Padmini, "Vehicle speed warning system and wildlife detection systems to avoid wildlife-vehicle collisions," *Emerging Research in Electronics, Computer Science and Technology*, pp. 961–968, 2019.
- [40] D. Dogan, S. Bogosyan, and T. Acarman, "Evaluation of driver stress level with survey, galvanic skin response sensor data, and force-sensing resistor data," *Advances in Mechanical Engineering*, vol. 11, no. 12, Article ID 168781401989155, 2019.