IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 2 January 2023, accepted 4 February 2023, date of publication 10 February 2023, date of current version 15 February 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3244008

RESEARCH ARTICLE

IoT-Based Non-Intrusive Automated Driver Drowsiness Monitoring Framework for Logistics and Public Transport Applications to Enhance Road Safety

M. ADIL KHAN, TAHIR NAWAZ, UMAR S. KHAN, (Member, IEEE), AMIR HAMZA, AND NASIR RASHID
Department of Mechatronics Engineering, National University of Sciences and Technology, Islamabad 44000, Pakistan
National Centre of Robotics and Automation (NCRA), National University of Sciences and Technology, Islamabad 44000, Pakistan

Corresponding author: M. Adil Khan (m.adilkhan@ieee.org)

This work was supported by the Higher Education Commission of Pakistan and the National Centre of Robotics and Automation under Grant DF 1009-0031.

ABSTRACT The exponential growth in road accidents has led to a need for continuous driver monitoring to enhance road safety. Existing techniques rely on vehicle sensor-based and behavior analysis-based approaches, where the behavior analysis-based approaches are generally considered more desirable as they enable reliable detection of a more elaborate set of driver behaviors. They are categorized as intrusive and non-intrusive approaches. Unlike intrusive approaches that generally rely on constant direct human contact with sensors (physiological signals) and are sensitive to artifacts, non-intrusive approaches offer a more effective behavior monitoring using computer vision-based techniques. This paper proposes an end-to-end non-intrusive IoT-based automated framework to monitor driver behaviors, designed specifically for logistic and public transport applications. It consists of an embedded system, edge computing and cloud computing modules, and a mobile phone application, in an attempt to provide a holistic unified solution for drowsiness detection, monitoring, as well as evaluation of drivers. Drowsiness detection is based on detecting sleeping, yawning, and distraction behaviors using an image processing-based technique. To minimize the effects of latency, throughput, and packet losses, edge computing is performed using commercial off-the-shelf embedded boards. Moreover, a cloud-hosted real-time database for remote monitoring on interactive Android mobile application has been set up, where admin can add multiple drivers to get drowsiness notifications along with other useful related information for driver evaluation. An extensive experimental testing has been performed, obtaining encouraging results. An overall accuracy of 96% is achieved along with an enhanced robustness, portability, and usability of the proposed framework.

INDEX TERMS ADAS, driver drowsiness, road safety, driver monitoring, IoT, edge computing, mobile application, road accidents, cloud computing.

I. INTRODUCTION

The importance of driver drowsiness detection has inevitably led to an enhanced focus of research in this domain. According to the World Health Organization, more than 1.3 million people die every year in road accidents making it the seventh leading cause of death globally among young people aged between 15 and 29 years [1]. The main underlying reasons for these fatalities include careless driving, drowsiness, health issues, lack of road safety awareness, inconsistent and improper law enforcement, and sleepiness [2], [3]. According to statistics by the National Highway

The associate editor coordinating the review of this manuscript and approving it for publication was Rodrigo S. Couto.

Traffic Safety Administration (NHTSA), in 2017, 91,000 police-reported vehicle crashes involved drowsy drivers [4]. In addition to drowsiness, distracted driving is also risky, claiming 3,142 lives in 2020 [5]. It is particularly critical for drivers of buses and heavy trucks, as they may have to work over a prolonged durations during the peak drowsiness periods (i.e., 2:00 am. to 6:00 am. and 2:00 pm. to 4:00 pm) and under monotonous or boredom working conditions [6], [7], which can lead to accidents. Therefore, the need remains for an intelligent system that can efficiently and effectively detect drowsiness and fatigue [8].

Existing driver monitoring methods are broadly categorized as vehicle sensor-based and behavior analysis-based approaches. These approaches are based on the use of either deep learning [9], traditional machine learning [10], or simple thresholding [11] for driver monitoring. Vehicle sensor-based approaches [12], [13] rely on the use of onboard sensors to primarily detect drivers' aggressive behaviors (e.g., over-speeding, aggressive braking, abrupt lane changing, etc.) without explicitly detecting drowsiness behavior. Behavioral analysis-based approaches [14], [15], [16], [17] are human-centric [18] and are primarily aimed at analysis of driver behaviors. They can be divided into intrusive and non-intrusive approaches. Intrusive approaches rely on the analysis of physiological signals [14], [15] to monitor vital signs (e.g., electrocardiogram (ECG) signals, electroencephalogram (EEG) signals, heart rate, oxygen saturation, respiration, etc.). Unlike vehicle sensor-based methods, intrusive approaches are preferable as they enable a more reliable driver health monitoring and drowsiness detection (from ECG or EEG signals). However, being intrusive in nature, they require a constant and reliable direct human contact with sensors that are normally mounted on the steering wheel (e.g. in case of ECG signals) or rely on placing sensors on the driver's head directly to record brain activity (e.g., in case of EEG signals) [19]. Intrusive approaches are highly sensitive to internal or external artifacts such as vibration, sound, or power line interference, circuit noise, loose contact, and movements that can interrupt the signal. Differently, non-intrusive approaches [16], [17] rely mostly on the use of computer vision-based methods. Unlike intrusive approaches, non-intrusive methods have lesser constraints as they do not require direct human contact and are not sensitive to onboard artifacts plus they enable the identification of a more elaborate set of driver behaviors such as head movement, eye movement, mouth openness, and facial expressions, thus offering a more effective driver monitoring (including also drowsiness detection). Moreover, they have been demonstrated to achieve better performance, and are therefore generally considered to be more desirable [20].

To this end, this paper proposes a solution that uses a behavior analysis-based non-intrusive approach. Unlike existing related non-intrusive approaches that largely focused on presenting driver drowsiness detection methods only without explicitly offering a holistic IoT-based remote monitoring solution, the proposed framework offers an end-to-end unified solution based on the use of IoT infrastructure, which covers automated drowsiness detection, remote monitoring, as well as evaluation at the back-end. The proposed framework is particularly suited for longer route public and logistics transportation, where a driver is expected to drive over extended period of time. The effectiveness of the proposed framework has been demonstrated in terms of performance accuracy, robustness, portability, and usability on two different commercial off-the-shelf embedded boards.

This paper is organized as follows. Section II reviews existing related work that is categorized into thresholding-based, traditional machine learning-based, and deep learning-based techniques for driver monitoring. Section III describes the proposed system design and implementation, which is followed by results and analysis in section IV, and the conclusion in Section V.

II. RELATED WORK

Several approaches have been proposed that can be broadly categorized as follows: thresholding-based approaches, traditional machine learning-based approaches, and deep learning-based approaches. Next, we provide a review of works belonging to each of these categories.

A. THRESHOLDING-BASED TECHNIQUES

In [21], a threshold-based model relying on image processing was proposed by using a smartphone camera, which used eye openness, mouth openness, head pitch angle, and head yaw angle as parameters. It utilized data from the gyro, accelerometer sensor, mobile camera, and GPS. However, the limitation of this method was high battery consumption (smartphone) due to the use of a live camera feed for a long time. Another study [22] proposed a computer vision-based system relying on applying thresholding on the eye blink time. It used a Viola-Jones algorithm to detect faces in camera images, followed by a classification of open and closed eyes based on the Eye Aspect Ratio (EAR) and a multilayer perceptron (MLP) neural network. The results of both methods were 84% and 97% respectively. In [23], a thresholding-based model was proposed based on eye blink and yawn counts. The face is detected using SVM and HoG detectors. The facial landmarks are detected using the pre-trained Dlib library. Then thresholding is done on the eyes and mouth to detect drowsiness. The combined feature showed 100% accuracy in the results. Another threshold-based model [24] proposes link rate and yawn counts. Where fatigue detection is done by multi-feature weighted sum for fatigue state recognition in which different weights were assigned to different factors. In [25], a threshold-based system is proposed that monitors heart rate and blood oxygenation, as measured by an electrocardiograph and oximeter integrated into the steering wheel. An inertial unit was also used to study the driving pattern of a driver. They used sensors ECG sensor, pulse oximeter sensor for PPG, and IMU sensor to detect motion changes due to steering rotation.

B. TRADITIONAL MACHINE LEARNING-BASED TECHNIQUES

Traditional machine learning-based techniques have also been used to detect drowsiness. In [26], the author proposed a driver monitoring approach based on image processing and machine learning. It makes use of the HAAR cascade algorithm for face detection. The images are taken from a camera. It detects eye openness using the EAR formula, where the vehicle information is collected from the On-Board Diagnostics-II such as vehicle velocity, location, engine revolutions, and throttle position. After getting data, the state of the driver is detected. Another work [27] introduced a system that detects drowsiness and classifies fatigue levels based on face detection, head-shoulder detection, eye detection, eye openness estimation, and fatigue levels classification. Head and shoulder were detected using the Histogram of oriented gradients (HoG) and an SVM classifier whereas face detection is done using OpenCV. For oblique view faces, an additional algorithm based on Haar features and Adaboost classifiers are used. Eye detection is done using their custom eye detectors. Moreover, the two eye detectors (CV-ED and I2R-ED) were fused with an OpenCV-based eye detector to detect eye openness. The results showed that there was an error of 0.033 in the detection of the right eye and 0.03 for the left eye. But the drawback of this system was that the computation time was rather high (0.19 seconds on a quad-core computer). The study [28] demonstrates the use of several machine learning algorithms for the detection of the ECG signal of the driver. The algorithms included unsupervised models (K-means clustering and Gaussian Mixture model) and supervised models (Support Vector Machine (SVM), Linear Discriminant Analysis, and Convolutional Neural Network (CNN)). The results showed that CNN produced the highest accuracy (>99%) but at the expense of a higher computational cost. On the other hand, SVM produced a good trade-off in terms of a low computational cost and an encouraging accuracy of 94%. In [29], a comparison of three classification methods, including random forest, gradient boosting, and logistic regression are presented in terms of the classification of driver distraction behavior. The results showed that the gradient-boosting algorithm achieved the best performance. In [30], a hybrid technique is proposed by using eye and body motion features, where body and eye motion detection is performed by frame differencing and parameterized using HoG descriptors. Then, SVM is trained and tested on a hybrid feature (body and eyes) to detect drowsiness, achieving an accuracy of 90%.

C. DEEP LEARNING-BASED MODEL

Various deep learning models have been proposed for diver drowsiness detection. In [31], proposed a Deep Neural Network (DNN) based on multilayer perception classifiers. The system detects facial features from images and delivers them to the trained model to identify the state of drowsiness on an Android application. The model achieved an accuracy of more than 80%. However, a better accuracy of 94.39% using DNN was achieved in [32] which outperformed some previous works focusing on night scenarios in which the driver is not wearing glasses. In [33] and [34], the Convolutional Neural Network (CNN) based techniques were used in which facial features were extracted using the Dlib library and then passed onto the CNN model to identify whether the driver is drowsy or not. Furthermore, a hybrid deep learning-based model was proposed in [35], in which long short-term memory (LSTM) and modified Inception V3 were used. It used a modified Inception V3 to prevent over-fitting of the training data. When compared to CNN, the aforementioned hybrid model performed better with an accuracy of 91.36%. Moreover, in [36], a deep convolutional neural network model is proposed based on bio-signals to detect the aggressive behavior of the driver. Their experimental results show the proposed model achieves a validation accuracy of 73.02% and 79.15% on two different datasets. The results show the applicability of the proposed system based on bio-signals to detect the aggressive behavior of the driver.

D. DISCUSSION

Existing works do not provide holistic end-to-end IoT-based automated solutions for driver drowsiness detection, remote monitoring, as well as evaluation (statistical analysis in terms of frequency of driver drowsiness in the past) for enhanced road safety. In [37], the author proposed an IoT framework that triggers an alert and warning message, when fatigue was detected along with the impact of a collision and location information. But the system did not provide a complete solution in terms of effective drowsiness detection, monitoring, and evaluation, which is particularly important for the logistic or public transport application where a vehicle is to be driven for extended periods of time on longer routes. The proposed system aims to provide a complete unified IoT-based solution to enable transport companies to monitor their driver's drowsiness behavior through an interactive Android mobile application as shown in **Fig. 1**. The system triggers an in-vehicle alert on detecting drowsiness or distraction and captures images of the aforementioned behavior along with the timestamp information, the current location for real-time tracking, and route information through a cloud-hosted real-time database that is synchronized with the admin nodes, where operators could analyze drivers' behavior remotely along with automated driver drowsy behavior analysis. In the next section, we describe the design and implementation of the proposed system.

III. SYSTEM DESIGN AND IMPLEMENTATION

This section describes the design and implementation of the proposed system as shown in **Fig. 1**. Broadly, the system consists of four major blocks, namely the embedded

TABLE 1. Layered architecture of the proposed system.

Blocks	Layers			Implementation
User Interface	**Application Layer**			**Mobile Application**
	Authentication	Drivers status	Drowsiness notification	Android
Cloud Computing	**Storage Layer**			**Cloud Database**
	Authentication	Real-time database	Storage	Firebase
	Network Layer			**IEEE Standard**
	Wi-Fi			IEEE 802.11b
Edge Computing	**Processing Layers**			**Libraries**
	Facial landmarks detection	Eyes, mouth aspect ratio, ear-to-nose Euclidean distance	Thresholding, trigger alert, establishing an internet connection	Dlib Open CV Threshold
	Pre-Processing Layer			
	RGB to greyscale	ROI detection		
Embedded System	**Sensor Layer**			**Sensor**
	Camera, Buzzer, Push Button			RGB camera Sound sensor Raspberry Pi 4 Jetson Nano
	Embedded Board			

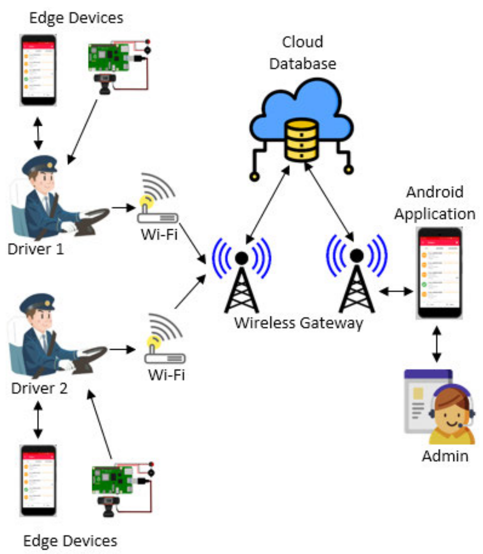


FIGURE 1. An abstract depiction of the proposed system.

system, edge computing, cloud computing, and user interface (see **Table 1**). Each block contains layer(s), which are interconnected and perform a set of functionalities for drowsiness detection, monitoring, and evaluation of driver. Under the embedded system block, we have separately tested and analyzed the computational performance of two different commercial off-the-shelf boards, i.e., a Raspberry pi 4 with 8GB RAM and an Nvidia Jetson Nano with 4GB RAM. Under the edge computing block, the extraction of facial features is performed on the live video stream at the sensor edge for detecting driver's eyes, mouth, and jaw lines, to trigger an onboard alert if and when a driver is found drowsy. Furthermore, it also sends alert notifications to the relevant authorities along with the visual evidence ('drowsy' images) and other data such as driver name, real-time location, time stamp, route, vehicle number, and an automated drowsiness score for driver evaluation, so that the admin can view these details on their mobile application remotely. Each block is described in **Table 1**.

A. EMBEDDED SYSTEM

The embedded system block consists of a sensor layer that contains an embedded board, an RGB camera, a buzzer, and a push button. As for the choice of embedded board, to demonstrate the effectiveness of the proposed framework, we here tested with Raspberry Pi and Jetson Nano, which are commercial off-shelf open-source boards and commonly used in related applications [38], [39]. The framework is however flexible to the use of other embedded boards as well. Raspberry pi has an onboard BCM43438 Wi-Fi/Bluetooth combo kit. It supports 2.4GHz 802.11n wireless communication. Jetson Nano has a Wi-Fi card based on an Intel 8265AC chip and LAN port that acts as a gateway for internet connectivity and allows wireless communication for data streaming to a cloud database. Moreover, the USB camera, buzzer, and push button can be connected via GPIO pins and USB port, respectively, as shown in **Fig. 2**. The specifications of both embedded boards are as follows:

Raspberry Pi 4 Model B is based on Broadcom BCM2711, Quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz processer with 8 GB RAM, standard 40 pin GPIO header, and built-in Wi-Fi, 2.4 GHz, and 5.0 GHz IEEE 802.11ac, 2 USB 3.0 ports; 2 USB 2.0 ports [40]. With a micro SD card slot for loading the operating system and data storage.

Jetson Nano specifications include GPU with NVIDIA Maxwell architecture with 128 NVIDIA CUDA®cores, CPU: Quad-core ARM Cortex-A57 MPCore processor. 4 GB 64-bit LPDDR4, 1600MHz, 4x USB 3.0, USB 2.0 Micro-B, GPIO, I2C, I2S, SPI, UART, Gigabit Ethernet, M.2 Key E [41]. This board gives much better computational performance for computer vision-related applications the circuit diagram is shown in **Fig. 3**.

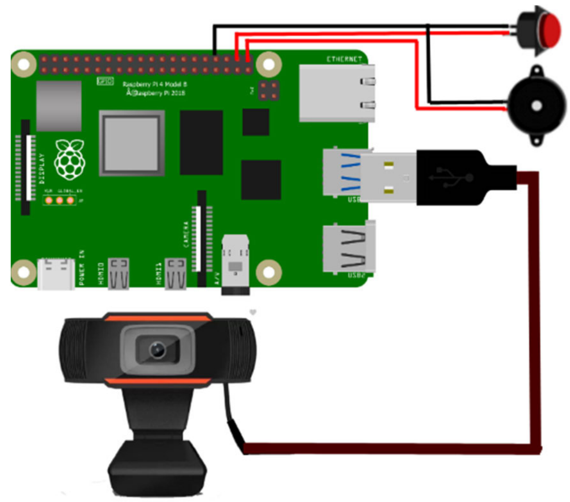


FIGURE 2. Circuit diagram of Raspberry pi 4.

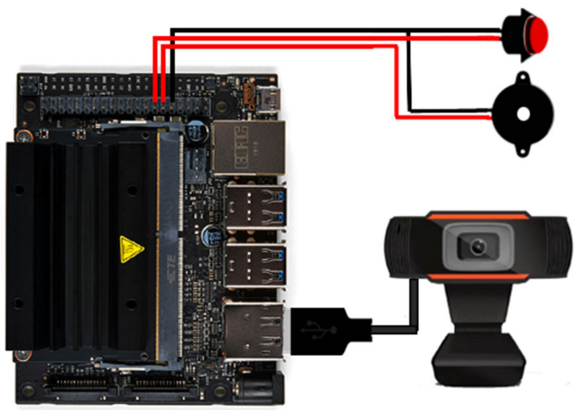


FIGURE 3. Circuit diagram of Nvidia Jetson Nano.

B. EDGE COMPUTING

Edge computing is followed by an embedded block that consists of pre-processing and processing layers. This block capture a live video stream to pass frames for further processing that includes, RGB into a greyscale image, ROI detection, and facial landmarks detection. To determine the drowsiness, the suggested work implements a pre-trained Dlib library for the detection of facial landmarks. The 68 facial landmarks (preset indexed landmarks) aid in shape prediction to distinguish between different facial parts such as the eyebrows, mouth area, jaw points, etc., as shown in **Fig. 4**. Based on facial landmarks, we have calculated Eye Aspect Ratio (EAR), Mouth Aspect Ratio (MAR), and Ear to Nose Euclidean Distance (ENED) for drowsiness and distraction detection as follows:

1) EYE ASPECT RATIO (EAR)

Drowsiness can be detected from the eyes based on blink pattern and blink duration. Human adults blink on average 12 times per minute, where the duration of one blink is approximately 1/3 seconds [43]. Dlib library gives us 6 points of an eye as shown in **Fig. 5**. From these points, we calculate the Eye Aspect Ratio (EAR) [44] Refer to "**(1)**". Based on the threshold value, we declare whether either eye is open

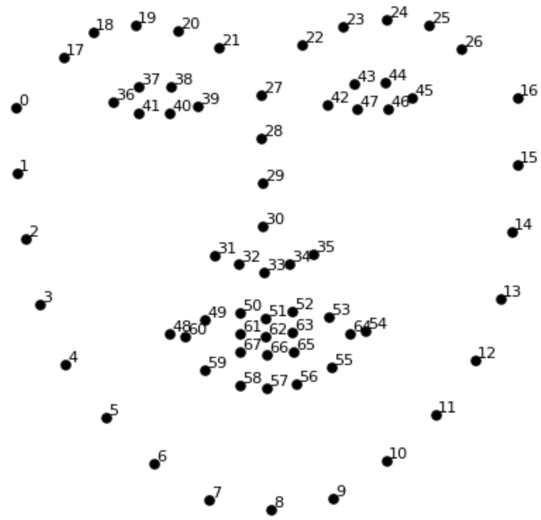


FIGURE 4. Representation of 68 facial landmark coordinates. Image taken courtesy [42].

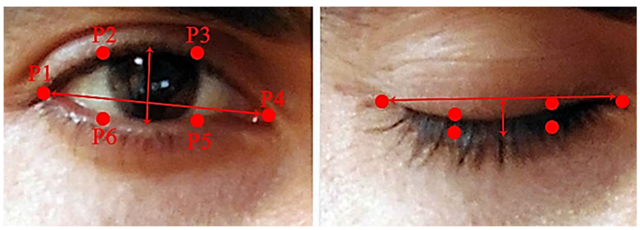


FIGURE 5. Eye Aspect Ratio to detect open/closed eyes.

or closed as in "**(2)**". As used in [45], if EAR > 30 is considered as open eyes, whereas if EAR < 0.20 for more than 2 seconds, it is considered as closed eye, thus triggering an alert as well as sending a notification to the admin through the cloud database along with a picture, time stamp, and real-time location from driver's mobile. Moreover, the admin can visualize driver data and trace the real-time location if needed. Drowsiness data is stored in a cloud database and can be fetched by the admin to evaluate drivers to enhance road safety.

$$\text{EAR} = \frac{||P2 - P6|| + ||P3 - P5||}{2||P1 - P4||} \tag{1}$$

$$f(x) = \begin{cases} x, & x > 0.30;\ open \\ x, & x < 0.20\ and\ t > 2sec;\ close \end{cases} \tag{2}$$

2) MOUTH ASPECT RATIO (MAR)

Yawning is an early sign of a person getting possibly drowsy. Frequent yawning enhances the possibility of drowsiness. To detect yawning, **Equation (3)** is used in which 8 points of lips from 68 facial landmarks, as shown in **Fig. 6**, are computed to compare with the threshold value as given in "**(4)**", where MAR is the Mouth Aspect Ratio of horizontal (e.g. p1, p5) and vertical points (e.g. p2, p3, p4, p6, p7, p8) of mouth. Based on the aspect ratio, we can analyze the openness and closeness of the mouth. If MAR ≥ 0.60 [46] and time is > 3 sec is considered yawning and if MAR < 0.42,

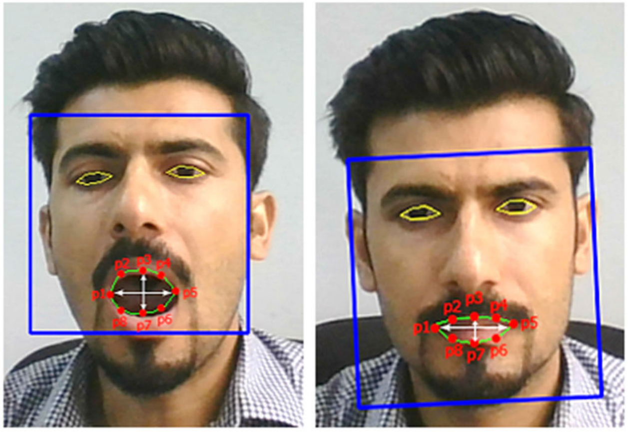


FIGURE 6. **Mouth Aspect Ratio (MAR) to detect yawning.**

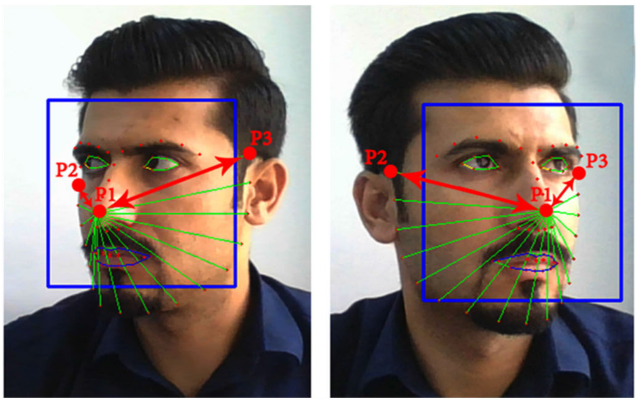


FIGURE 7. **Ear to Nose Euclidean Distance (ENED) to distraction.**

it is considered normal behavior. Between these values, the person might be talking, or if MAR exceeds 0.60 but the time duration is less than 3 seconds it is also considered as 'talking'.

$$\text{MAR} = \frac{||P2 - P8|| + ||P3 - P7|| + ||P4 - P6||}{2||P1 - P5||} \tag{3}$$

$$f(y) = \begin{cases} y, & y \geq 0.60 \text{ and } t > 3sec; \text{ yawn} \\ y, & y < 0.42; \text{ no yawn} \end{cases} \tag{4}$$

3) EAR TO NOSE EUCLIDEAN DISTANCE (ENED)

A related important behavior to detect is 'distraction', as it could potentially cause an accident. It normally involves the driver looking sideways, i.e. to the left or right side, for a certain duration of time due to (or without) being drowsy. Here, we define 'distraction' as looking sideways for more than 3 seconds, which could be considered as a reasonably longer enough time to stay distracted while driving. To detect this behavior, we use p1, p2, and p3 points from the 68 facial landmarks (see **Fig. 7**). It specifically involves first calculating the Euclidean distance between p1 & p2 (denoted as *d1*) and between p1 & p3 (denoted as *d2*) using (**5**). The underlying motivation is that when a driver turns the face to the left or right side (i.e. distracted), the absolute difference: $D = |d1 - d2|$, is expected to significantly increase

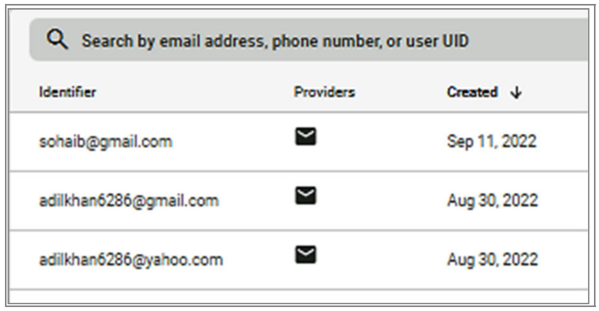


FIGURE 8. **Firebase user authentication data.**

as compared to when a driver is looking straight (i.e. not distracted) in which case D should approximately be zero. We therefore use the following criterion to define distraction: if $D \geq 90$ (set empirically) for more than 3 seconds, distraction is said to have occurred, as given in (6).

$$\text{ENED} = \sqrt{(x2 - x1)^2 + (y2 - y1)^2} \tag{5}$$

$$f(z) = \begin{cases} \text{Distraction}, D \geq 90 \text{ and } t > 3sec \\ \text{No Distraction}, D < 90 \text{ and } t < 3sec \end{cases} \tag{6}$$

C. CLOUD COMPUTING

This block consists of a network layer and a storage layer. The network layer allows connecting edge devices with the Internet, which enables access to the cloud-hosted real-time database for storing media, real-time data, and authentication. It allows real-time data streaming and storage in the storage layer. Firebase cloud-hosted real-time database is used to achieve the aforementioned functionality in this layer. It is a hybrid cloud-hosted real-time database that can be configured with a cross-platform embedded system, mobile application, and websites, where data is stored and synchronized among all the nodes and remains available when an application goes offline [47]. Firebase provides various features but we used: (a) authentication to authenticate legitimate users; (b) a real-time database for real-time data streaming; and (c) storage to save media. Our database is synced with three nodes such as an embedded device, admin, and driver mobile applications. The data is stored in a real-time database which consists of JSON objects that normally consist of key and value pairs. The key is a unique identifier within the database and the value contains the data that is being stored. There might be a nested/tree structure but values are stored in key-value pairs and supported data types are; string, Boolean, long, double, list, etc.

Authentication is an important feature of the Android application that allows only legitimate users to access the application. A new user needs to register on the Android application and the registration data is stored on the Firebase database as shown in **Fig. 8**. After successful registration, the user needs to use the same credentials (email, password) to sign into the application, where input credentials are authenticated with Firebase authentication. Moreover, the user can recover a forgotten password through the email address he entered at the time of registration.

A **real-time database** plays an important role in real-time remote monitoring. The data on the firebase database keeps updating in real-time and synchronized with all the client nodes. The real-time database has two main nodes: the driver node and the admin node. The data in the driver node comes from the embedded board and driver mobile application that includes the driver's name, age, blood group, license number, location coordinates, vehicle number, contact, and an email sent from driver's mobile application, whereas the drowsy image, drowsiness counter, and time stamp sent from the edge device to driver node on cloud database as shown in **Fig. 9**. The data in the admin node comes from the driver node (mobile application and edge device) and admin's mobile application (sign-up data) that includes name, contact, email ID, terminal name, terminal ID, profile image, and assigned driver nodes as shown in **Fig. 10**. Profile pictures of the driver and admin are stored in the **storage directory** and we have created another directory in which the drowsy images history of every registered driver are stored along with the timestamped and facial features information labelled on image that can be visualized on a mobile application for driver evaluation as an additional evidence along with automated drowsiness score assigned on detecting drowsiness behaviour (specifically sleeping). Moreover, for real-time drowsiness images, the image Unique Identifier (UID) is added to the real-time database node that can be visualized by the admin on receiving a drowsiness alert.

D. USER INTERFACE

The user interface consists of the application layer that provides an interactive interface on a mobile application for both driver and admin. They can register themselves, sign in, and access required the functionalities, such as driver status, drowsiness status, alert notification, and real-time location. The Firebase real-time database is connected and synchronized with all the edge nodes that provide real-time data for multiple drivers.

Android operating system is the most popular in the world, with over 2.5 billion active users across 190 countries of the world [48]. The android application plays a significant role in the Internet of Things (IoT) based driver monitoring systems. It provides an interactive interface to drivers and admins for performing a set of functionalities. Moreover, it also shows the data that is sent from the embedded board to the admin. We have developed a native Android application with android studio IDE using XML language for front-end and Java language for back-end development. The application functionalities include authentication, real-time data processing, and notification alert. A unified application allows both driver and admin to register into an application as shown in **Fig. 11**. (a), (b), and (c). At the time of registration, the data is stored on Firebase (cloud-hosted real-time database) database, and by using the same credentials, drivers, and admin can sign into the application to perform their respective functionalities. The driver can start/stop a ride, select a route, and activate/deactivate their location for real-time location tracking by the admin, as shown in **Fig. 13**(a). The location feature is implemented by using background services. Even if a driver exits the Android application, still location service runs in the background and keeps updating location coordinates on the cloud database after every 4 seconds as shown in **Fig. 13**.(b), (c) for real-time location tracking by the admin. After signing in, the admin can search and add/remove drivers by using the specific ID (as a primary key) of the vehicle device. After adding a driver, the admin can visualize the real-time status of drowsy drivers along with other details on the main screen as shown in **Fig. 12**. (a), (b), and (c). Moreover, automated driver evaluation is implemented by assigning drowsiness points along with drowsy images as evidence, where the admin can check previous drowsiness status (drowsy images with associated data) for driver's evaluation.

FIGURE 9. Driver data structure on Firebase real-time database.

IV. RESULTS AND ANALYSIS

This section provides experimental validation of the proposed framework. First, we describe the dataset (Sec. IV-A), which is followed by testing and evaluation of the vision-based drowsiness detection (Sec. IV-B), android application (Sec. IV-C), alert system (Sec. IV-D), and computational performance assessment (Sec. IV-E).

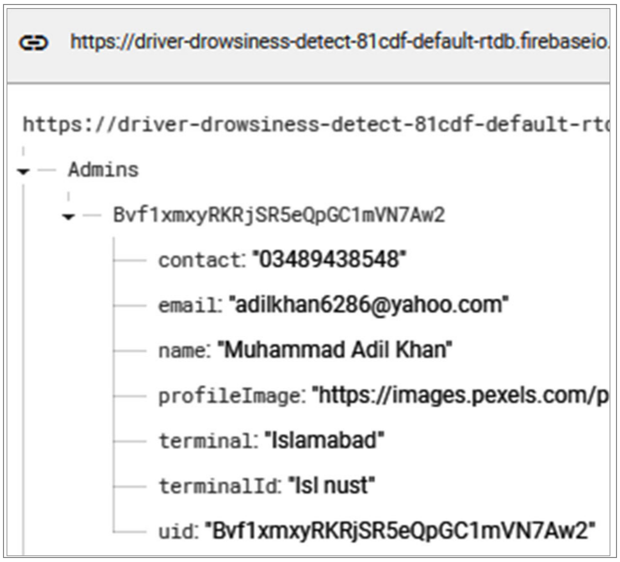


FIGURE 10. Admin data structure on Firebase real-time database.

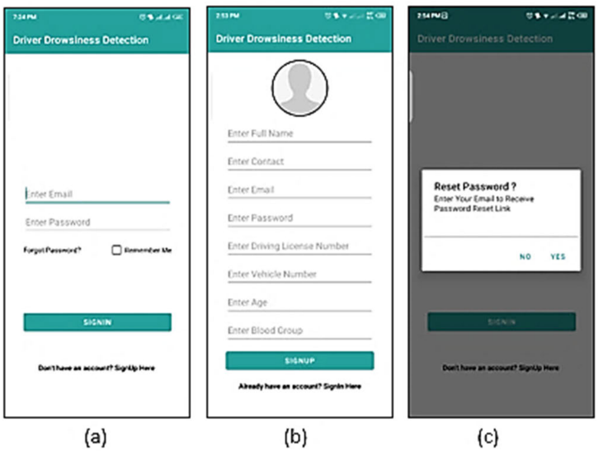


FIGURE 11. User Authentication - Login/ Signup/ forgot password.

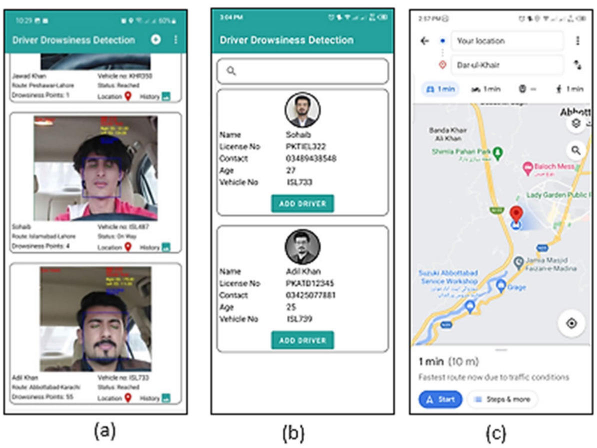


FIGURE 12. Admin panel to add drivers.

A. DATASET

A challenge that we came across is the lack of an already available standardized benchmark dataset for the problem under consideration. To this end, we created an evaluation dataset that contains 50 subjects in total, out of which the data for 25 subjects have been collected by us and the data for the remaining 25 subjects have been obtained from a public online source [49]. The dataset contains samples for four behavior classes that are under consideration in this study: Active, Eyes Closed, Yawning, and Distraction. For the Active and Yawning classes, we used 50 samples – one from each of the 50 subjects; for the Eyes closed class, we used 35 samples from 35 subjects; and for the Distraction class, we employed 50 samples from 25 subjects. **Table 2** provides a summary of the dataset. The choice of subjects was made taking into account diversity in appearance such as presence/absence of beard and moustaches on the face, varying hair lengths, both gender types (33 males and 17 females), presence/absence of face spectacles, head uncovered and covered with a scarf, and different races. As for the camera setup, images have been captured with varying viewpoint angles, camera-subject distances, and illumination settings, and have different image resolutions. The effective camera-subject distance range is between 30cm to 60cm (see **Fig. 14**),with the camera placed in front of the driver ($\pm 20^{o}$ on either side) on the dashboard. Note that an informed consent has been obtained from the subjects regarding the usage of the data for this study.

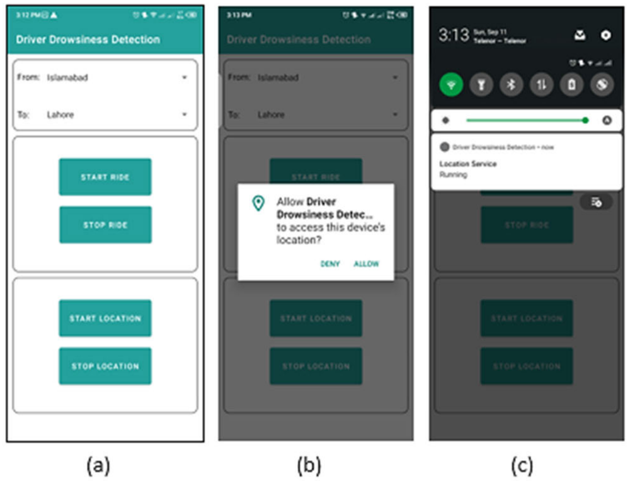


FIGURE 13. Driver app to start a ride and background location service.

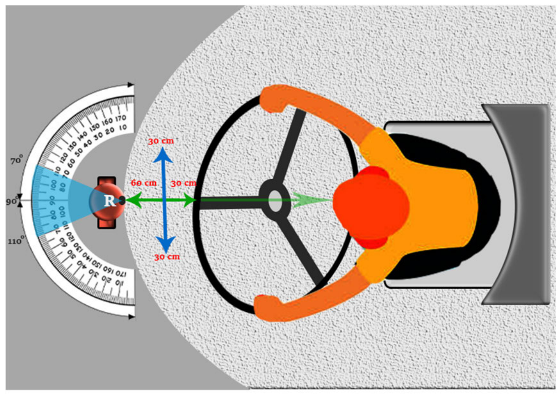


FIGURE 14. Visualization of the camera setup within a vehicle.

TABLE 2. Specifications of the dataset used. Key. CSD: variation in camera-subject distance, A: variation in viewpoint angle of the camera, L: varying illumination, IR: varying image resolution, OPD: online public dataset, OD: own dataset.

Class	Sample	Subject	Challenges	Source
Active	50	50	CSD, A, L, IR	OPD, OD
Eyes closed	35	35	CSD, L, IR	OPD, OD
Yawning	50	50	CSD, A, L, IR	OPD, OD
Distracted	50	25	CSD, L, IR	OPD, OD

TABLE 3. Performance evaluation for the four behavior classes in terms of the precision, recall, F1-score, and accuracy measures.

Class	Precision	Recall	F1- score	Accuracy
Active	0.89	0.98	0.93	0.96
Yawning	1.00	0.96	0.98	0.96
Eyes Closed	0.94	0.97	0.95	0.96
Distraction	1.00	0.92	0.96	0.96

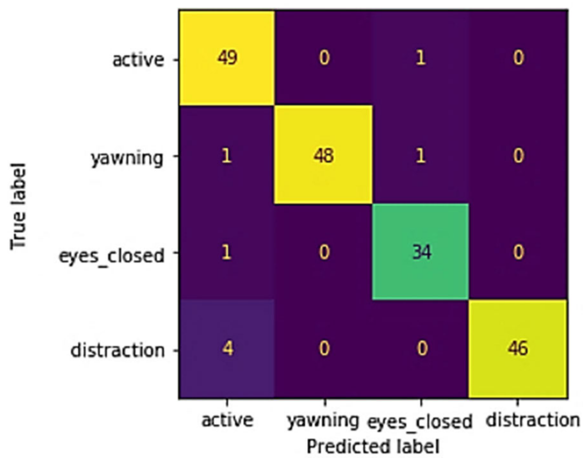


FIGURE 15. Performance evaluation for the four classes in terms of the Confusion Matrix.

B. VISION-BASED DROWSINESS DETECTION

This section describes the results of the detection of drowsiness in terms of classification performance evaluation on the four behavior classes: Active, Eyes Closed, Yawning, and Distraction. For a holistic performance assessment, we have employed Recall, F1-Score, and Accuracy measures. *Precision* provides the performance using true positives and false positives. *Recall* quantifies the performance in terms of true positives and false negatives. *F1-score* provides combined performance evaluation in terms of the harmonic mean of Precision and Recall values. *Accuracy* assesses the performance based on the use of true positives, false positives, false negatives, and true negatives. **Table 3** provides the results in terms of these measures for all four classes and **Fig. 15** shows the corresponding confusion matrix. The qualitative results on key sample images for the four classes are given in **Fig. 16-19**.

To assess the performance evaluation for the Active class, 50 samples from 50 unique subjects have been taken into account. The results show that Precision =0.89, Recall =0.98, F1-Score = 0.93, and Accuracy = 0.96 (**Table 3**). Here, the reason for a slightly lower precision is an incorrect classification of 1 Yawning sample as Active, 1 Eyes Closed sample as Active, and 4 Distraction samples as Active. It is also relevant to mention that there is one Active sample that is missed as Eyes closed (see **Fig. 20** (a)); hence the Recall = 0.98. The qualitative results for the Active class on some key sample images are given in **Fig. 16**.

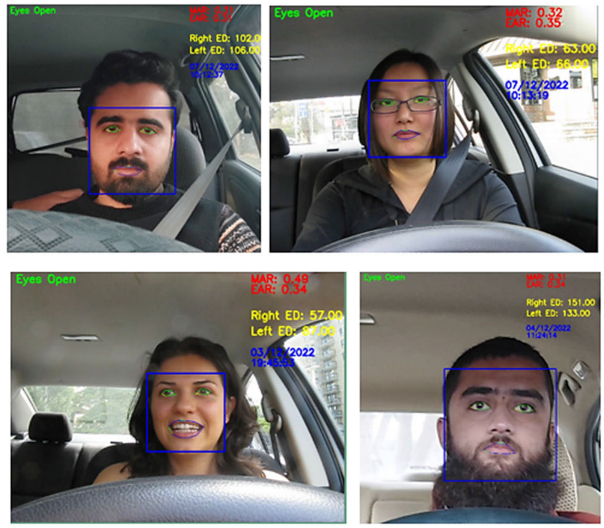


FIGURE 16. Qualitative results on key sample images for Active class.

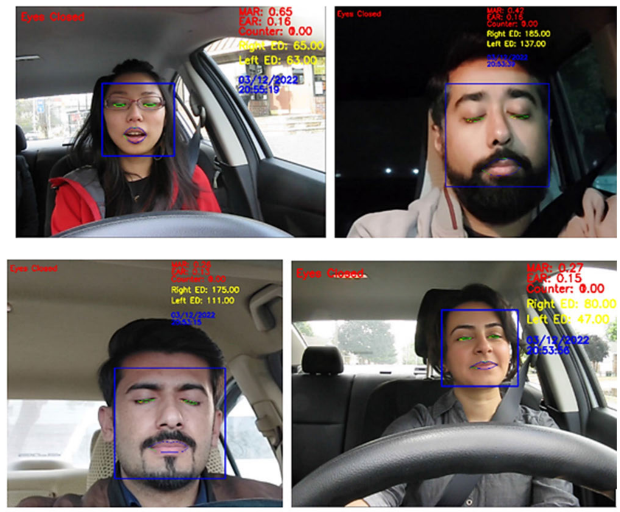


FIGURE 17. Qualitative results on key sample images for Eyes Closed class.

For Eyes closed class, contains a total of 35 samples from 35 unique subjects are used, out of which 34 samples are predicted correctly (true positives), one Eyes Closed sample missed as Active (**Fig. 20**(b)), one Active sample incorrectly classified as Eyes Closed (see **Fig. 20** (a)), and one Yawning sample incorrectly classified as Eyes Closed (see **Fig. 20** (c)); hence, Precision = 0.94, Recall = 0.97, and F1-Score = 0.95. The qualitative results for the Eyes Closed class are shown in **Fig. 17**.

For the Yawning class, 50 samples from 50 unique subjects are considered for evaluation. Recall = 0.96, as there are 48 true positives and 2 false negatives – one Yawning

FIGURE 18. Qualitative results on key sample images for Yawning class.

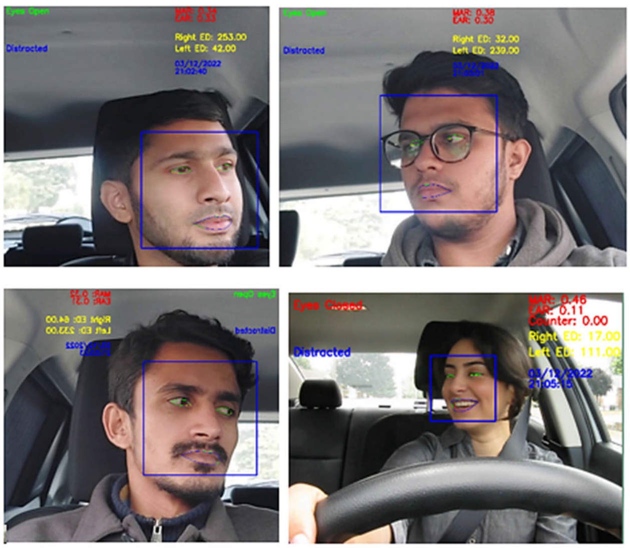


FIGURE 19. Qualitative results on key sample images for Distraction class.

sample missed as Active and another one missed as Eyes Closed (**Fig. 20** (c)). Precision = 1.00, as there are zero false positives. **Fig. 18** shows the qualitative results on key sample images for Yawning class.

For the Distraction class, we used 50 samples from 25 subjects. A Recall = 0.92 is achieved as there are 46 true positives and 4 false negatives (e.g. see a sample false negative in **Fig. 20**(d), in which a Distraction sample is missed as Active). We noticed that false negatives occurred for this class due to unreliable detection of facial landmarks (p1, p2, p3 points as defined earlier), which could be caused either when the viewpoint angle of the camera is significantly large (thus hiding right or left side of face) or due to reflection caused by spectacles. Precision = 1.00, as there are no false positives. Some key sample images for Distraction class are shown in **Fig. 19**.

(a) (b) (c) (d)

FIGURE 20. Misclassified cases for Eyes Closed class (a), Active class (b), Yawning class (c), and Distraction class (d).

C. ANDROID APPLICATION BLACK-BOX TESTING

Black-box testing is performed to test the functionalities of the developed Android application. The application provides a set of functionalities for both drivers and admin. Both users are synchronized by a Firebase cloud-hosted real-time database. For admin, we tested the application for every functionality several times such as authentication (registration, login, forgot password, remember the password), driver search, add and remove option, notification alert, drowsiness image history, drowsiness points, real-time location, and real-time drowsiness status.

For drivers, we tested authentication, route selection, ride start/stop, and location background service activate/deactivate functionality that runs in the background to keep driver locations updated in a cloud database. All the functionalities mentioned in **Table 4** passed black-box testing.

D. ALERT SYSTEM

The alert mechanism is categorized into onboard alert (for a driver) and remote mobile notification alert (for an admin). Where on-board alert triggers when any of the three facial actions (eyes closed, yawning, and distraction) are detected as abnormal to get the driver's attention back to avoid an accident. If a driver falls asleep (i.e. his/her eyes are continuously closed for more than 3 seconds), an alert notification is received on the admin mobile application along with the driver's drowsiness images, timestamp, and location for tracking. We tested the alert system for all three scenarios (eyes closed, yawning, and distraction) on different subjects,

TABLE 4. Android application Black-box testing.

User	Features	Working	Not working
Admin	Authentication	YES	--
	Search, add, and remove a driver	YES	--
	Receive drowsiness notification alert	YES	--
	Driver's real-time drowsiness status	YES	--
	Driver real-time location tracking	YES	--
	Drowsiness points	YES	--
	History of driver's drowsiness	YES	--
Driver	Authentication	YES	--
	Start/ stop ride	YES	--
	Select route	YES	--
	Activate/ deactivate location	YES	--

TABLE 5. On-board and remote alert testing.

Test Case	On-board Alert	Mobile Alert (admin)
Yawning	Yes	--
Eyes Closed	Yes	Yes
Distraction	Yes	--

TABLE 6. Computational performance comparison on two embedded boards.

Board	Scenario 1	Scenario 2	Scenario 3
Raspberry pi 4	0.27 sec	0.37 sec	4.73 sec
NVIDIA Jetson Nano	0.16 sec	0.23 sec	3.07 sec

and acquired a 100% success rate for both on-board and mobile application alert notifications as shown in **Table 5**.

E. COMPUTATIONAL PERFORMANCE

The computational performance of the proposed system is tested with two different commercial-off-the-shelf boards based on the computational time for three different scenarios defined as follows. Scenario 1 refers to the case when a driver is not present in the vehicle; hence, no need for the detection of facial landmarks. Scenario 2 refers to the case when a driver is present; hence, a need for the detection of facial landmarks. Scenario 3 involves drowsy detection (as a result of occurrence of sleeping behavior) that involves cloud database communication as well. The computational performance (see **Table 6**) is discussed next.

The overall script processing time on raspberry pi takes 0.27 sec while the script is running with no face detection. It takes 0.37 sec on detecting a person, which calculates EAR, MAR, and ENED. And it takes overall 4.73 seconds to detect drowsiness, capture an image to store in a local drive, and send that image to the cloud database.

The overall python script processing time on NVIDIA Jetson Nano is 0.16 without person detection, and 0.23 seconds on detecting a person in which it calculates EAR, MAR, and ENED. It took overall 3.07 seconds to detect drowsiness, capture an image to store in a local drive, and send that image to the firebase cloud database.

Based on our analysis, it is evident that Jetson Nano is preferable as it shows a better computational performance with even 4GB RAM. This is apparently because it has an additional capability of GPU for fast processing. An even better performance is expected to be achieved by optimizing code and embedded boards further. It is relevant to mention that, since the underlying face recognition algorithm (and the code) stays the same, the algorithmic performance accuracy of Dlib library does not change on both embedded boards.

V. CONCLUSION

This paper presented an end-to-end unified framework to robustly perform driver drowsiness detection, remote monitoring, and evaluation. The system consists of four key building blocks, including an embedded platform, edge computing, cloud computing, and an interactive user interface for both driver and admin to perform a set of functionalities. This work relied on the use of computer vision techniques to detect the drowsiness of drivers, aimed primarily for a use case in which a driver is driving a public or logistic transport for longer routes. Unlike existing works, the proposed system offers a more holistic and effective end-to-end framework, covering drowsiness detection, remote monitoring, as well as automated evaluation by using IoT infrastructure. We performed an extensive performance evaluation for different aspects of the proposed framework with promising results.

The system has obtained a very encouraging performance for drowsiness detection in terms of achieving promising scores for Precision, Recall, F1-score, and Accuracy measures for the four defined classes: Active, Yawning, Eyes Closed, and Distraction. We identified a limitation of the proposed framework in the form of the misclassifications that have been obtained in some sample images of the dataset containing reflections caused due to spectacles worn by subjects (drivers), which is an aspect that could be investigated further as part of future work.

The experimental testing has also been performed for assessing computational performance separately using Raspberry pi 4 and Nvidia Jetson Nano. Nvidia Jetson achieved a better computational performance that can be further improved by optimizing code and board configuration.

Future work could also focus on devising a hybrid approach by fusing the physiological (ECG, heart rate, etc.) and visual cues within a framework involving the use of deep learning, to achieve enhanced robustness.

ACKNOWLEDGMENT

The authors would like to thank the Higher Education Commission of Pakistan and the National Center of Robotics and Automation (NCRA) for providing the funding to perform this research.

REFERENCES

[1] *Road Traffic Injuries*. Accessed: Sep. 22, 2022. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-Injuries

[2] J. J. Rolison, "Identifying the causes of road traffic collisions: Using police officers' expertise to improve the reporting of contributory factors data," *Accident Anal. Prevention*, vol. 135, Feb. 2020, Art. no. 105390, doi: 10.1016/J.AAP.2019.105390.

[3] A. Bener, E. Yildirim, T. Özkan, and T. Lajunen, "Driver sleepiness, fatigue, careless behavior and risk of motor vehicle crash and injury: Population based case and control study," *J. Traffic Transp. Eng.*, vol. 4, no. 5, pp. 496–502, Oct. 2017, doi: 10.1016/J.JTTE.2017.07.005.

[4] *Drowsy Driving: Avoid Falling Asleep Behind the Wheel | NHTSA*. Accessed: Sep. 22, 2022. [Online]. Available: https://www.nhtsa.gov/risky-driving/drowsy-driving

[5] *Distracted Driving Dangers and Statistics | NHTSA*. Accessed: Sep. 22, 2022. [Online]. Available: https://www.nhtsa.gov/risky-driving/distracted-driving

[6] A. Sahayadhas, K. Sundaraj, and M. Murugappan, "Detecting driver drowsiness based on sensors: A review," *Sensors*, vol. 12, no. 12, pp. 16937–16953, Dec. 2012, doi: 10.3390/S121216937.

[7] Y. Zhang and S. Zhu, "Study on the effect of driving time on fatigue of grassland road based on EEG," *J. Healthcare Eng.*, vol. 2021, pp. 1–9, Jul. 2021, doi: 10.1155/2021/9957828.

[8] L. B. Leng, L. B. Giin, and W.-Y. Chung, "Wearable driver drowsiness detection system based on biomedical and motion sensors," in *Proc. IEEE SENSORS*, Nov. 2015, pp. 1–4, doi: 10.1109/ICSENS.2015.7370355.

[9] M. F. F. M. Hanafi, M. S. F. M. Nasir, S. Wani, R. A. A. Abdulghafor, Y. Gulzar, and Y. Hamid, "A real time deep learning based driver monitoring system," *Int. J. Perceptive Cogn. Comput.*, vol. 7, no. 1, pp. 79–84, 2021.

[10] A. Ziryawulawo. (2021). *A Machine Learning Based Driver Monitoring System for the Kayoola EVS*. Accessed: Sep. 22, 2022. [Online]. Available: http://dissertations.mak.ac.ug/handle/20.500.12281/9026

[11] I. Gupta, N. Garg, A. Aggarwal, N. Nepalia, and B. Verma, "Real-time driver's drowsiness monitoring based on dynamically varying threshold," in *Proc. 11th Int. Conf. Contemp. Comput. (IC3)*, Aug. 2018, pp. 1–6, doi: 10.1109/IC3.2018.8530651.

[12] S. Lawoyin, D. Y. Fei, and O. Bai, "Accelerometer-based steering-wheel movement monitoring for drowsy-driving detection," *J. Automobile Eng.*, vol. 229, no. 2, pp. 163–173, Oct. 2015, doi: 10.1177/0954407014536148.

[13] L. Alam and M. M. Hoque, "Vision-based driver's attention monitoring system for smart vehicles," in *Proc. Int. Conf. Intell. Comput. Optim.*, in Advances in Intelligent Systems and Computing, vol. 866, 2019, pp. 196–209, doi: 10.1007/978-3-030-00979-3_20.

[14] R. Dhanapal and P. Visalakshi, "Real time health care monitoring system for driver community using adhoc sensor network," *J. Med. Imag. Health Informat.*, vol. 6, no. 3, pp. 811–815, Jun. 2016, doi: 10.1166/JMIHI.2016.1768.

[15] H. Rahman, S. Barua, and B. Shahina, "Intelligent driver monitoring based on physiological sensor signals: Application using camera," in *Proc. IEEE 18th Int. Conf. Intell. Transp. Syst.*, Sep. 2015, pp. 2637–2642, doi: 10.1109/ITSC.2015.424.

[16] M. Ngxande, J.-R. Tapamo, and M. Burke, "Driver drowsiness detection using behavioral measures and machine learning techniques: A review of state-of-art techniques," in *Proc. Pattern Recognit. Assoc. South Afr. Robot. Mechatronics (PRASA-RobMech)*, Nov. 2017, pp. 156–161, doi: 10.1109/ROBOMECH.2017.8261140.

[17] R. Li, H. Brand, A. Gopinath, S. Kamarajugadda, L. Yang, W. Wang, and B. Li, "Driver drowsiness behavior detection and analysis using vision-based multimodal features for driving safety," SAE Tech. Papers 2020-01-1211, Apr. 2020, doi: 10.4271/2020-01-1211.

[18] T. Nawaz, B. Rinner, and J. Ferryman, "User-centric, embedded vision-based human monitoring: A concept and a healthcare use case," in *Proc. ACM Int. Conf. Distrib. Smart Cameras (ICDSC)*, Paris, France, 2016, pp. 25–30, doi: 10.1145/2967413.2967422.

[19] M. Osmalina, N. Rahma, O. N. Rahma, and A. Rahmatillah, "Drowsiness analysis using common spatial pattern and extreme learning machine based on electroencephalogram signal," *J. Med. Signals Sensors*, vol. 9, no. 2, p. 130, 2019, doi: 10.4103/jmss.JMSS_54_18.

[20] M. R. Ullah, M. Aslam, M. I. Ullah, and M.-E. A. Maria, "Driver's drowsiness detection through computer vision: A review," in *Advances in Computational Intelligence*. Cham, Switzerland: Springer, 2018, pp. 272–281. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-030-02840-4_22#citeas, doi: 10.1007/978-3-030-02840-4_22.

[21] A. Kashevnik, I. Lashkov, and A. Gurtov, "Methodology and mobile application for driver behavior analysis and accident prevention," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 6, pp. 2427–2436, Jun. 2020, doi: 10.1109/TITS.2019.2918328.

[22] A. Revelo, R. Alvarez, and F. Grijalva, "Human drowsiness detection in real time, using computer vision," in *Proc. IEEE 4th Ecuador Tech. Chapters Meeting (ETCM)*, Nov. 2019, pp. 1–6, doi: 10.1109/ETCM48019.2019.9014884.

[23] H. Garg, "Drowsiness detection of a driver using conventional computer vision application," in *Proc. Int. Conf. Power Electron. IoT Appl. Renew. Energy Control (PARC)*, Feb. 2020, pp. 50–53, doi: 10.1109/PARC49193.2020.236556.

[24] J. Huang and Z. Lin, "Multi-feature fatigue driving detection based on computer vision," *J. Phys., Conf. Ser.*, vol. 1651, no. 1, Nov. 2020, Art. no. 012188, doi: 10.1088/1742-6596/1651/1/012188.

[25] B. Babusiak, A. Hajducik, S. Medvecky, M. Lukac, and J. Klarak, "Design of smart steering wheel for unobtrusive health and drowsiness monitoring," *Sensors*, vol. 21, no. 16, p. 5285, Aug. 2021, doi: 10.3390/s21165285.

[26] S. Shaily, S. Krishnan, S. Natarajan, and S. Periyasamy, "Smart driver monitoring system," *Multimedia Tools Appl.*, vol. 80, no. 17, pp. 25633–25648, Apr. 2021, doi: 10.1007/s11042-021-10877-1.

[27] B. Mandal, L. Li, G. S. Wang, and J. Lin, "Towards detection of bus driver fatigue based on robust visual analysis of eye state," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 3, pp. 545–557, Mar. 2017, doi: 10.1109/TITS.2016.2582900.

[28] M. McConnell, B. Schwerin, N. Podolsky, M. Lee, B. Richards, and S. So, "Classification of steering wheel contacts from electrocardiogram signals using machine learning," in *Proc. IEEE Int. Conf. Signal Process., Commun. Comput. (ICSPCC)*, Aug. 2020, pp. 1–6, doi: 10.1109/ICSPCC50002.2020.9259459.

[29] R. Ghandour, A. J. Potams, I. Boulkaibet, B. Neji, Z. A. Barakeh, and A. S. Karar, "Machine learning methods for driver behaviour classification," in *Proc. 4th Int. Conf. Bio-Eng. Smart Technol. (BioSMART)*, Dec. 2021, pp. 1–4, doi: 10.1109/BioSMART54244.2021.9677801.

[30] A. A. Sheikh and J. Mir, "Machine learning inspired vision-based drowsiness detection using eye and body motion features," in *Proc. 13th Int. Conf. Inf. Commun. Technol. Syst. (ICTS)*, Oct. 2021, pp. 146–150, doi: 10.1109/ICTS52701.2021.9608977.

[31] R. Jabbar, K. Al-Khalifa, M. Kharbeche, W. Alhajyaseen, M. Jafari, and S. Jiang, "Real-time driver drowsiness detection for Android application using deep neural networks techniques," *Proc. Comput. Sci.*, vol. 130, pp. 400–407, Jan. 2018, doi: 10.1016/j.procs.2018.04.060.

[32] M. S. Mahmoud, A. Jarndal, A. Alzghoul, H. Almahasneh, I. Alsyouf, and A. K. Hamid, "Driver drowsiness detection system using deep learning based on visual facial features," in *Proc. 14th Int. Conf. Develop. eSyst. Eng. (DeSE)*, Dec. 2021, pp. 453–458, doi: 10.1109/DESE54285.2021.9719409.

[33] 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 *Proc. IEEE Int. Conf. Informat., IoT, Enabling Technol. (ICIoT)*, Feb. 2020, pp. 237–242, doi: 10.1109/ICIoT48696.2020.9089484.

[34] M. Arslan and R. Abiyev, "Vision-based drowsiness detection system using convolutional neural networks," in *Proc. Int. Conf. Electr., Commun., Comput. Eng. (ICECCE)*, Jun. 2020, pp. 2–6, doi: 10.1109/ICECCE49384.2020.9179207.

[35] V. Kumar, S. Sharma, and R. Walia, "Driver drowsiness detection using modified deep learning architecture," *Evol. Intell.*, pp. 1–10, Jun. 2022. [Online]. Available: https://link.springer.com/article/10.1007/s12065-022-00743-w#article-info, doi: 10.1007/s12065-022-00743-w.

[36] A. Alamri, A. Gumaei, M. Al-Rakhami, M. M. Hassan, M. Alhussein, and G. Fortino, "An effective bio-signal-based driver behavior monitoring system using a generalized deep learning approach," *IEEE Access*, vol. 8, pp. 135037–135049, 2020, doi: 10.1109/ACCESS.2020.3011003.

[37] A. K. Biswal, D. Singh, B. K. Pattanayak, D. Samanta, and M.-H. Yang, "IoT-based smart alert system for drowsy driver detection," *Wireless Commun. Mobile Comput.*, vol. 2021, pp. 1–13, Mar. 2021, doi: 10.1155/2021/6627217.

[38] V. P. Kumar, P. Aravind, S. N. D. Pooja, S. Prathyush, S. AngelDeborah, and K. R. S. Chandran, "Driver assistance system using Raspberry Pi and Haar cascade classifiers," in *Proc. 5th Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, May 2021, pp. 1729–1735, doi: 10.1109/ICICCS51141.2021.9432361.

[39] S. Cass, "NVIDIA makes it easy to embed AI: The Jetson nano packs a lot of machine-learning power into DIY projects—[Hands on]," *IEEE Spectr.*, vol. 57, no. 7, pp. 14–16, Jul. 2020, doi: 10.1109/MSPEC.2020.9126102.

[40] *Raspberry Pi 4 Model B specifications—Raspberry Pi*. Accessed: Sep. 24, 2022. [Online]. Available: https://www.raspberrypi.com/products/raspberry-pi-4-model-b/specifications/

[41] *Jetson Nano | NVIDIA Developer*. Accessed: Sep. 24, 2022. [Online]. Available: https://developer.nvidia.com/embedded/jetson-nano

[42] B. Amos, B. Ludwiczuk, and M. Satyanarayanan. (2016). *OpenFace: A General-Purpose Face Recognition Library With Mobile Applications*. Accessed: Oct. 17, 2022. [Online]. Available: http://cmusatyalab.github.io/openface/

[43] K.-A. Kwon, R. J. Shipley, M. Edirisinghe, D. G. Ezra, G. Rose, S. M. Best, and R. E. Cameron, ''High-speed camera characterization of voluntary eye blinking kinematics,'' *J. Roy. Soc. Interface*, vol. 10, no. 85, Aug. 2013, Art. no. 20130227, doi: 10.1098/RSIF.2013.0227.

[44] C. Dewi, R.-C. Chen, X. Jiang, and H. Yu, ''Adjusting eye aspect ratio for strong eye blink detection based on facial landmarks,'' *PeerJ Comput. Sci.*, vol. 8, p. e943, Apr. 2022, doi: 10.7717/peerj-cs.943.

[45] T. Zhu, C. Zhang, T. Wu, Z. Ouyang, H. Li, X. Na, J. Liang, and W. Li, ''Research on a real-time driver fatigue detection algorithm based on facial video sequences,'' *Appl. Sci.*, vol. 12, no. 4, p. 2224, Feb. 2022, doi: 10.3390/APP12042224.

[46] T. V. N. S. R. S. Mounika, P. H. Phanindra, N. V. V. N. S. Charan, Y. K. K. Reddy, and S. Govindu, ''Driver drowsiness detection using eye aspect ratio (EAR), mouth aspect ratio (MAR), and driver distraction using head pose estimation,'' in *ICT Systems and Sustainability* (Lecture Notes in Networks and Systems), vol. 321. Singapore: Springer, 2022, pp. 619–627. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-16-5987-4_63, doi: 10.1007/978-981-16-5987-4_63.

[47] *Firebase Realtime Database*. Accessed: Sep. 30, 2022. [Online]. Available: https://firebase.google.com/docs/database

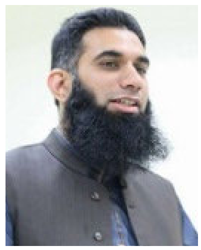
[48] *10 Years of Google Play and Our Commitment to a Thriving App Ecosystem in India*. Accessed: Sep. 30, 2022. [Online]. Available: https://blog.google/intl/en-in/products/platforms/10-years-of-google-play-and-our-commitment-to-a-thriving-app-ecosystem-in-india/

[49] *Driver Drowsiness Using Keras | Kaggle*. Accessed: Dec. 15, 2022. [Online]. Available: https://www.kaggle.com/code/adinishad/driver-drowsiness-using-keras/comments

M. ADIL KHAN received the bachelor's degree (Hons.) in software engineering from the Abbottabad University of Science and Technology, Pakistan, and the master's degree from the Department of Computer and Technology, Chang'an University, Xi'an, China. Currently, he is working as a Research Associate at the National Centre of Robotics and Automation (NCRA). His research interests include the Internet of Things (IoT), intelligent computing, and embedded system related studies. He has developed several innovative projects and won prestigious awards at national and international level.

TAHIR NAWAZ received the M.Sc. degree in computer vision and robotics under the Erasmus Mundus Scholarship, a joint master's program from Heriot-Watt University (U.K.), University of Girona (Spain), and University of Burgundy (France), and the Ph.D. degree with a specialization in computer vision, a joint Doctoral program under the highly prestigious Erasmus Mundus Fellowship from the Queen Mary University of London (U.K.) and Alpen-Adria University of Klagenfurt (Austria). He has a strong demonstrated track record of research and development in the areas of computer vision (visible/thermal imagery) and artificial intelligence, with more than 15 years of experience of working in academic and industrial sectors across multiple European countries in prestigious organizations. He is currently working as an Assistant Professor at the Department of Mechatronics Engineering, College of Electrical and Mechanical Engineering, National University of Sciences and Technology (NUST), Pakistan, where his core interests focus around multi-modal sensing techniques, particularly investigating cutting-edge technologies pertaining to automated video surveillance and autonomous vehicles. In 2005, he also represented Pakistan as a Team Leader in Asia–Pacific Broadcasting Union (ABU) Robocon Contest (an international robot competition), held at Beijing, China. He has published more than 25 papers in prestigious publication venues and has been involved in several international funded projects.

UMAR S. KHAN (Member, IEEE) received the bachelor's degree in mechatronics engineering from the National University of Sciences and Technology, Pakistan, in 2005, and the Ph.D. degree in electrical engineering from the University of Liverpool, U.K., in 2010. Currently, he is working as an Associate Professor at the Department of Mechatronics Engineering, National University of Sciences and Technology. He is also the Project Director of the National Centre of Robotics and Automation (NCRA). His research interests include embedded systems and image processing.



AMIR HAMZA received the bachelor's degree in mechatronics engineering from the National University of Sciences and Technology, Pakistan, the master's degree in mechanical and aerospace engineering from Seoul National University, South Korea, and the Ph.D. degree in mechanical engineering from the King Fahd University of Petroleum and Minerals, Saudi Arabia. He is currently an Associate Professor and the Head of the Department of Mechatronics Engineering, NUST College of Electrical and Mechanical Engineering. He is also serving as the Director of the Robot Maker Laboratory, established under the National Centre of Robotics and Automation (NCRA). His research interests include robotics, advanced manufacturing techniques, and materials engineering.

NASIR RASHID received the B.E. degree (Hons.) in mechanical engineering from the College of Electrical and Mechanical Engineering (EME), Islamabad, Pakistan, in 1993, and the M.S. and Ph.D. degrees in mechatronics engineering from the College of Electrical and Mechanical Engineering, National University of Sciences and Technology (NUST), Pakistan. His Ph.D. is in the field of non-invasive brain signal classification (biomedical engineering). He is currently working as an Associate Professor with the College of Electrical and Mechanical Engineering, NUST. His field of specialization is artificial intelligence with applications in biomedical engineering. He has vast experience as a Professional Engineer in Pakistan. He is a Lifetime Member of the Pakistan Engineering Council.