"IoT-Based Vehicle Monitoring and Driver Assistance System Framework for Safety and Smart Fleet Management"

Article *in* International Journal of Integrated Engineering · April 2023

DOI: 10.30880/ijie.2023.15.01.035

Bhuvendhraa Rudrusamy
Heriot-Watt University

© Universiti Tun Hussein Onn Malaysia Publisher's Office

IJIE

http://penerbit.uthm.edu.my/ojs/index.php/ijie
ISSN : 2229-838X e-ISSN : 2600-7916

The International Journal of Integrated Engineering

IoT-Based Vehicle Monitoring and Driver Assistance System Framework for Safety and Smart Fleet Management

Bhuvendhraa Rudrusamy[1*], Hock Chye Teoh[1], Jia Yew Pang[1], Tou Hong Lee[2], Sung Choong Chai[3]

[1]School of Engineering and Physical Sciences,
Heriot-Watt University Malaysia, 62200 Putrajaya, MALAYSIA

[2]TMS-Lite Sdn Bhd, Sungai Ara,
11900 Bayan Lepas, Pulau Pinang, MALAYSIA

[3]CPeEnergy Sdn Bhd,
Damansara Utama, 47400 Petaling Jaya, Selangor, MALAYSIA

*Corresponding Author

DOI: https://doi.org/10.30880/ijie.2023.15.01.035
Received 30 October 2022; Accepted 25 January 2023; Available online 31 March 2023

Abstract: Curbing road accidents have always been one of the utmost priority of nations worldwide. In Malaysia, the Traffic Investigation and Enforcement Department reported that Malaysia's total number of road accidents have increased from 373,071 to 533,875 in the last decade. One of the significant causes of road accidents is the driver's behaviors. However, to regulate drivers' behavior by the enforcement team or fleet operators is challenging, especially for heavy vehicles. In our research, we have proposed the Internet of Things (IoT) scalability framework and its' emerging technologies to monitor and alert driver's behavioral and driving patterns to reduce road accidents. To prove this work, we have implemented a lane tracking, and iris detection algorithm, to monitor and alert the driver's behavior when the vehicle sways away from the lane, and to detect if the driver is feeling drowsy. We implemented electronic devices such as cameras, a global positioning system module, a global system communication module, and a microcontroller as the hardware for an intelligent system in the vehicle. We also applied face recognition for person identification using the same in-vehicle camera and recorded the working duration for authentication and operation health monitoring. With the GPS module, we monitored and alerted against permissible vehicle's speed accordingly. We integrated IoT on the system for the fleet centre to monitor and alert the driver's behavioral activities in real-time through the user access portal. We have validated it successfully on Malaysian roads. The outcome of this pilot project ensures the safety of drivers, public road users, and passengers. The impact of this framework leads to a new regulation by the government agencies towards merit and demerit system, real-time fleet monitoring of intelligent transportation systems, and socio-economy such as cheaper health premiums. The big data can be used to predict the driver's behavioral in the future.

Keywords: Accident, drowsiness detection, fleet management, internet of things, intelligent transportation systems, lane tracking

1. Introduction

Several studies show that human behavior is one of the most critical factors contributing to road accidents, which includes speeding, not following road signs, and disregarding safety. According to Keall et al. [1], human behavior

*Corresponding author: b.rudrusamy@hw.ac.uk

contributes to about 95% to road accidents. The common causes of road accidents are bad eyesight, slow human reaction, speeding, overtaking, negligence of vehicle maintenance, incorrect application of driving aids, driver's mental state and physical fitness, abuse of alcohol and drugs, fatigue, lack of education and training, the influence of religion believes, vehicle ownership, and use of charms as protection by drivers [2]. In addition to these factors, age and gender are also closely related to road accidents, according to González-Sánchez et al. [3]. On the other hand, Machata [4] categorized the behavior risk model as problems experienced by road users in perceiving, accepting and controlling risk. Although the law is heavily enforced, the drivers have yet to adhere to the regulations [5]. Therefore, regulating driver's behavior is somewhat complicated, and if it can be monitored and alerted to the driver or fleet management in real-time, road mishaps can be avoided.

In this work, we have proposed the Internet of Things (IoT) and emerging technologies framework to monitor and alert the driver in reducing road accidents in real-time. We implemented few exemplars of major cause of accidents due to human error categorized as (i) direct cause, and (ii) human condition/states [6], which are lane tracking and iris detection algorithms as vehicle driving assistance systems in reducing the mishaps, respectively. Furthermore, we used Global Positioning System (GPS) to monitor the location and alert the vehicle's speed as an added safety feature. Additionally, we adopted face recognition to authenticate and permit the driver's service, especially for fleet or transportation-based service operations. Overall, if any related risk is found, the driver and the fleet centre will be alerted using this framework in real-time. The advantage of this framework is monitoring human behavior at all times, which the typical road transport regulation is unable to enforce. Such preventive measures to avoid accidents related to human error and condition/states benefit both road users and fleet operators, especially heavy vehicles. Surely, this framework can be expanded to include more sensors in assisting drivers in reducing road accidents for sustainable smart cities.

2. Related Works

Many safety features related to human behavior were proposed in the past, and we have discussed the related intelligent transportation system (ITS) works in this paper. One of the direct causes of accident features is lane tracking. The smart lane tracking system has been tremendously developed in recent years and introduced as part of the advanced driver assistance system (ADAS) found in new vehicles these days. However, keeping the driver/vehicle on the lane is limitedly implemented and yet to mature due to many challenges. The primary technology used in ADAS is a vision-based lane tracking system. Zhou & Wang [7] compiled recent techniques used for lane tracking and highlighted possible solutions for shortcomings. Feniche & Mazri [8] also shared the recent development of state-of-the-art techniques that helps to improve the lane detection system. Chen et al. [9] emphasize one of the key techniques for better lane tracking by improving lane segmentation. The research community continues to address the challenges such as harsh weather, varying road conditions, and signage, especially in urban areas.

Another aspect of road accidents caused by the human condition/state is drowsiness. Many parameters were used to identify drowsiness, and such early detection reduces the risk of road accidents. Previously, head movement was explored [10], [11] to identify fatigue signs that could/may lead to drowsiness. The research community continues to explore other prominent parameters that are related to drowsiness, such as bio-signal (e.g. EEG, PPG, GSR) and vehicular pattern (e.g. steering wheel movement) [12]. Kamran et al. [13] comprehensively summarised the possible physiological cause of drowsiness, the related symptoms for crashes, and physiological strategies related to mishaps due to drowsiness. For this safety reason, sensors such as cameras, steering wheel-, breaking-, and pressure-based sensors are installed in the car to detect drowsiness. However, the research community has yet to address the shortcomings in this area [14]-[17].

Speeding has always been a direct contributing factor to road accidents. Pawłowski et al. [18] reported that 45% of drivers tend to speed on national roads, which increases the risk of fatality for drivers and other road users. Implementing speed display signs (SDSs) in the attempt to comply with the speed limit was found only effective for a limited distance in the short term [19]. Inappropriate speeding for traffic condition in urban area will cause vulnerability for other road users too. There are many legislative steps for speeding offenders, including fines, suspension of driving license, and legal punishment. Although such countermeasures have been taken, they are not significant in addressing driver's behaviors [20] and are unsustainable measures.

For operation safety of the fleet management, person identification enforces vehicle access as scheduled in operation while monitoring the total driving hours to avoid fatigue. One could consider biometrics technology such as face recognition for driver's authentication. Jiang [21] and Alsrehin & Al-Taamneh [22] compared the traditional and intelligent face recognition methods suitable for such applications. Depending on the complexity of the face recognition application, Anagha & Ram [23] shared possible solutions to address some of the shortcomings. However, the face identification or authentication application for ITS in sustainable smart cities [24] may not be the same as in high-security access areas such as banking services. Li et al. [25] shared evaluation standards, including face databases, that can be considered for developing face recognition technology for ITS.

Rojas et al. [26] proposed monitoring, controlling, and scheduling method for a fleet management system. Four prototype vehicles installed with GPRS and GSM technology were tested to monitor vehicle speed and validate the proposed fleet management system integration using LoRa. Many other IoT-based fleet management systems were proposed [27]-[29], but none provides end-to-end monitoring and alerts/communication between driver and fleet centre in real-time for safety aspects. This investigation emphasizes holistic ITS solution for sustainable smart cities, especially for safe transportation services. In this work, we proposed (i) lane-tracking for traffic safety, (ii) drowsiness detection to monitor driver's condition, (iii) face recognition for driver's authentication and monitoring working hours, and (iv) speeding and alert system. We validated it to show an end-to-end alert system if any related risk is found, establishing drivers' obligation to adhere to fleet compliances and traffic regulations.

3. Overall Framework

This work adopted an IoT-based framework to monitor and assist the driver's safety for sustainable ITS in smart cities, as shown in Fig. 1. To establish this framework, we implemented four major edge devices as ITS examplar: lane tracking, drowsiness detection, face recognition-based authentication and tracking, and speeding. As a result, the proposed transportation ecosystem increases the awareness of the employee (driver) and employer (transportation industry) to ensure the safety of the client (passenger) or other road users is preserved.

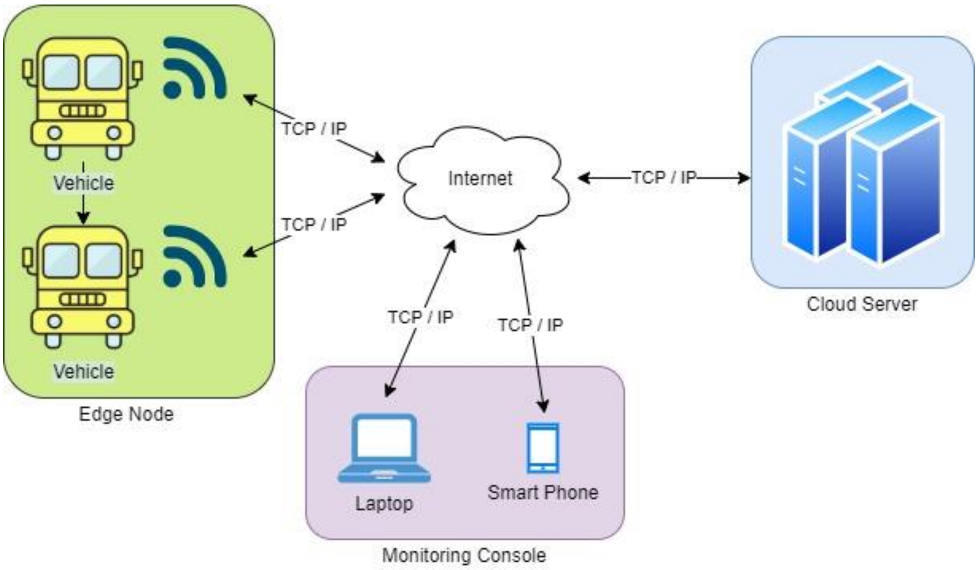


Fig. 1 - An overview of vehicle monitoring and assistance framework

3.1 Hardware Framework

The system consists of cameras, a microcontroller, local storage, electronic control unit (ECU) interface, liquid crystal display (LCD), speaker, global positioning system (GPS), global mobile communication (GSM), data server, and web server, as shown in Fig. 2. The real-time data was obtained from camera devices installed in the vehicle for vehicle lane assistance, monitoring the awareness of the driver and face recognition for authentication. In contrast, the speed data was obtained from both GPS and ECU.

The microcontroller acts as the master device for the edge node, while the rest of the units act as slave devices. The inputs for the microcontroller are the images from a stereo camera, visual camera, GPS, GSM, and ECU unit. These modules capture the data and feed it to the microcontroller for data processing. The local storage is used as a workspace for the microcontroller and keeps temporary data. The GSM is used for communication between the vehicle and fleet centre via the cloud. The LCD (with touchscreen input) in the edge node displays the status, alerts the driver for acknowledgements when necessary. The speaker is used to assist this process if needed. The communication module is password protected and used for programming and diagnosing the framework ensuring only the authorized person can update or modify the algorithm. In addition to that, any history of access and updates can be tracked and stored in the data centre.

3.2 Software Framework

The overview of the proposed software architecture is shown in Fig. 3**Error! Reference source not found.**. The continuous monitoring and alert edge nodes of the framework, namely (i) speeding, (ii) drowsiness detection, and (iii) lane tracking, use modular-based software architecture where each sub-subsystem is processed or executed in parallel. The software also consists of scoreboards that capture the number of times the driver breaks the regulation, namely (i) in-vehicle scoreboard and (ii) fleet scoreboard. For example, referring to Fig. 3**Error! Reference source not found.**, the speed of the bus is monitored during the journey. If the vehicle exceeds the speed limit, the system alerts the driver. The driver is expected to reduce the speed. Although the vehicle scoreboard captures the number of alerts the driver exceeds the limit, it may be considered appropriate in some scenarios. For example, the driver can exceed the speed limit when overtaking a vehicle. However, if the driver exceeds the speed limit for some time, in addition to the vehicle alert, this sub-system also alerts the fleet centre via the cloud. It returns with an auto-generated warning to the driver. The driver is expected to acknowledge the auto-generated warning and reduce the vehicle speed via the LCD touchscreen. In the case of no response from the driver, the fleet centre must be alerted for manual operator assistance. The fleet operator sends a warning to the driver via an LCD monitor until the driver obeys the speed limit. The fleet scoreboard captures the number of times the driver exceeds the speed limit and is reminded by the operator to follow the speed limit. The fleet operator should consider serious action to be taken against such drivers. These two scoreboards can also be used for data analytics to study the behavior of the driver. A similar approach is implemented for driver's awareness and lane tracking sub-system. The novelty of the proposed method is the end-to-end (or closed-loop) monitoring and alert system, where both vehicle driver and fleet management are well informed on the status of the transportation ecosystem, specifically on the safety aspect.

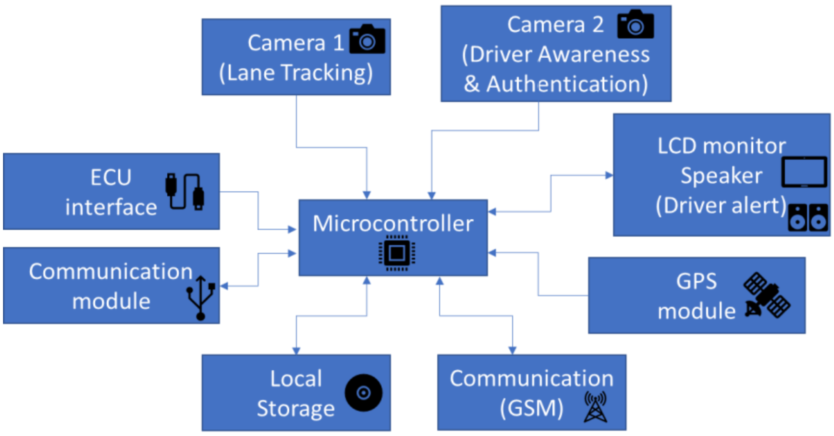


Fig. 2 - The hardware framework

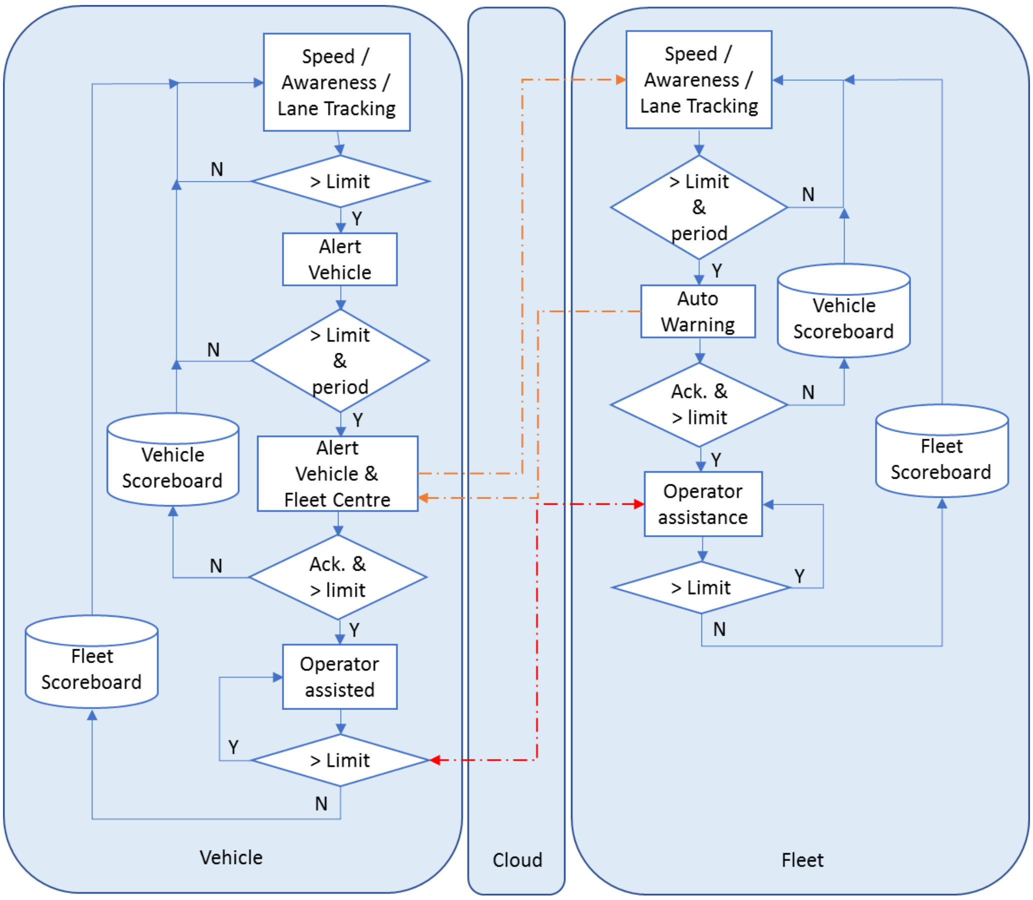


Fig. 3 - The software framework

4. Software Architecture

Typically, three different implementations can be considered for software architecture that is (i) bare metal, (ii) virtual machine (VM), and (iii) container. Bare metal would offer the best performance but is the least flexible for system migration, updates and upgrades. The VM has been a popular choice because hardware can be seen as a separate layer from the software. The VM image can be easily copied to another VM environment, and a new duplicate instance can be obtained. However, VM does not share resources and would compete with other VM for resources. It is configured to acquire a certain amount of resources from the operating system and does not release it for sharing even if it is not utilized. Similar concept to VM, a container is an improvement over VM on the resource sharing aspect. Unlike VM, it would not request resources when it is not utilized. Also, the container can be developed individually and ease migrating into the final production environment. Among the container technologies, we adopted Docker [30], as shown in Fig. 4. It is an open-source and free for use, although it does offer an enterprise version.

The advantage of using Docker technology is how quickly an environment can be replicated once appropriately configured. This technology is known as the Docker hub. Docker hub is a repository of resources made available for the developer community, open-source projects, and independent software vendors (ISV) to build and distribute their code in containers as Docker images. These Docker images can be stored and made available in the Docker hub hosted on the internet. However, only one private image can be stored under the free subscription, and an unlimited number of public images. It is not necessary to store these images in the hub to allow community sharing. Images can be stored privately in local storage (e.g. hard drive) and ~~be~~ shared like a file using a Docker tool. The image may consist of a pre-configured operating system, libraries, application, system configurations, and data. These are just a few key reasons to implement Docker technology in this work.

A user can log in to a server that has Docker installed (docker host). The user then retrieves the images from the Docker hub (docker registry) available on the internet at https://hub.docker.com/. The containers seen in the docker host are the instances of the running Docker image. The same image can be invoked multiple times concurrently. In addition, these instances can be configured to share the resources. Docker also supports more complex configurations as it is scalable.

Using the container approach, we implemented the microservices architecture as shown in Fig. 4. The bottom stack is the operating system. The Docker is installed as a layer above the operating system. This allows one to create "containerd", a term used in Docker for a container runtime instance. In this work, there are three containerds as follows:

(i) Containerd labelled "1" consists of an HTTP web server, a proxy for the webserver and web services. As all network traffic from outside of the Docker environment enters through this containerd, a local virtual private network is created for an additional layer of security. Other container IPs shall not be exposed unnecessarily.

(ii) Containerd labelled "2" consists of the actual web application. This web application provides the main landing page and access to other pages. The actual web service process also resides on this server.

(iii) Containerd labelled "3" consists of the database to store all the related information, including the latest latitude and longitude used to track all the vehicles.

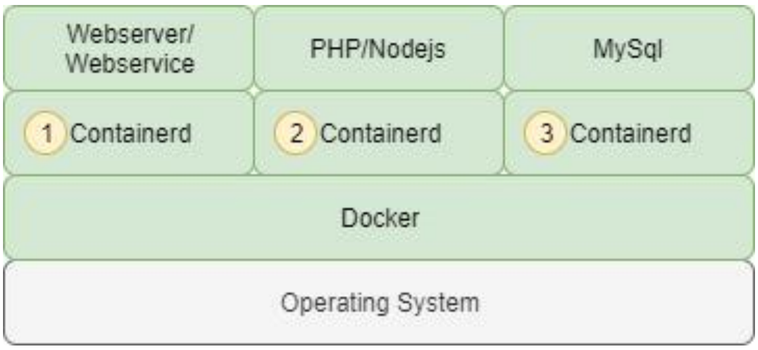


Fig. 4 - The software architecture

5. Microservices Architecture

The microservices architecture is most suitable for cloud environments. It can be easily scaled accordingly to demands. The processes have the flexibility of being created using a different set of languages, libraries and operating systems and putting them all together to form their own stack. Usually, the stack is kept as lean as possible, and only relevant library components shall be included, making it as lightweight as possible.

These applications are usually loosely coupled through web services. Hence, any new requirements can be easily adopted into its ecosystem rather than changing to an existing application, which is the case for traditional monolith systems.

The applications that make up the system as a whole are viewed as services. These services can be exposed to the outside world or remain visible only within their ecosystem. The obsolete services can easily shred away when the functionalities and purposes of each application are kept as simple as possible.

Another aspect of microservices architecture is that testing activities are made so much easier. This ecosystem greatly encourages continuous integration and continuous delivery (CI/CD) because each application can be easily switched in and out to a newer version or rolled back to an older one. Automation is one of the key ingredients to satisfy CI/CD.

Other forms of tests can also easily endure, such as load and stress testing. Simulating the packets needed for a web service is much easier than simulating data entry from the user interface (UI).

Web services monitoring tools can easily observe these services, providing valuable information for fine performance tuning and data usage control. Performance fine-tuning improves system efficiency and resource utilization. In addition, monitoring data usage can help identify data leakage or system abuse, which would normally go undetected as these applications already have the necessary credentials and privileges to access the resources and data.

Web services management tools can be placed into the ecosystem to prevent abuse and quota usage. This tool also allows the creation of users, groups and roles and assigns privileges to create, read, update and delete (CRUD) accordingly. Hence, a limitation on a transaction per second can be imposed to safeguard the application's responsiveness to a specific system user and other data policy configurations. Therefore, data can be tracked and monetized to bill external users or internal application system users.

6. Sensors as Intelligent Transportation System

One can integrate as many sensors to make up vehicle intelligence. Further connecting it to IoT enables smart ITS in sustainable smart cities. Nonetheless, the challenges in deploying more sensors in ITS must be addressed to ensure it is fully cooperative and operational on an IoT-based framework [31], [32]. In this framework, a few key sensors were deployed on the prototype for proof of concept in increasing the driver's awareness, increasing the safety of the road users, and efficiently managing a sustainable fleet centre.

6.1 Speeding Monitoring, Tracking, and Alert System

All road users must observe the speed limit at all times. On some occasions, speed limits differ on the type of vehicle. For example, a car and a bus are regulated at 110- and 90 km/h on the national road, respectively, and in general, many road signs are meant for car users. Therefore, it is possible for misinterpretation by bus drivers in particular. Similar regulations are enforced for various roads categorized as expressway, federal-, state- and municipal roads in Malaysia. Having this system in heavy vehicles would be an added benefit.

In this work, we use a GPS module to sense the location and speed of the vehicle. The shared edge microcontroller processes the GPS data before transmitting it to the cloud using GSM. The data server verifies the vehicle's location and speed against the road speed limit and generates an alert if it exceeds the limits in real-time. The edge device receives the alert and displays it to the driver via an LCD monitor installed in the vehicle. In addition to speed estimation using GPS, the vehicle speed can also be obtained from the ECU interface. This speed information is useful in slow response from the GPS signal, which rarely happens and usually for a short period of time only. As all the information is processed using the microcontroller, a temporary storage system backs up the data when the GSM signal becomes unavailable. The transmission returns when the GSM signal is reconnected and the data is synced back to the cloud (data server).

6.2 Lane Tracking and Alert System

The lane assistance system ensures the vehicle is kept to the lane and alerts the driver when the vehicle sways from the host lane. The overview of this algorithm is shown in **Error! Reference source not found.**. This sub-system has a camera with a shared edge microcontroller and local storage to capture the front view of the road video images in real-time. In general, the microcontroller processes the video images using a pre-image processing technique, removing the unnecessary objects in the image.

The RGB colour image frames are converted to HLS colour space and grayscale to observe the white and yellow lane and removes the unrelated background. The road feature is extracted using Gaussian filtering and Canny edge detection techniques. The possible road lines are detected using Hough Transform [33] and is averaged out to determine the road line. Road lines are further filtered using the time smoothing technique when applied on multiple image frames; hence, the road lane is tracked. The lane drifting or weaving can be detected using simple programming, and it alerts the driver if frequent or drastic changes are detected. The lane monitoring and alert mechanism are shown in Fig. 3. Hence, both the driver and fleet centre are well informed about this closed-loop monitoring system.

In the absence of lane marking (road features), it is possible to estimate the lane using road modelling approximately. For example, Yeo et al. [34] proposed Gaussian-based road modelling with a thermal camera for lane estimation. Although this method helps in such event, the driver should decide based on his intuition for safe driving in case of uncertainty. Nevertheless, many other techniques can be added to our proposed modular framework to improve vehicle safety.

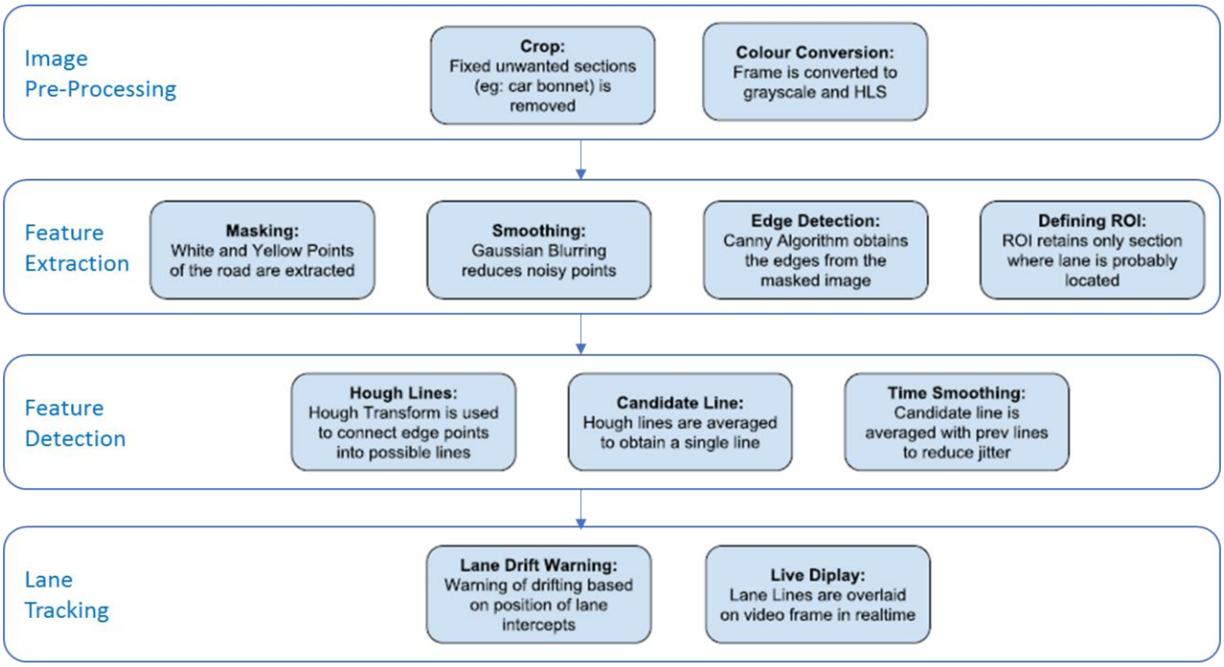


Fig. 5 - Lane tracking

6.3 Drowsiness Monitoring and Detection System

The lack of focus of vehicle drivers may contribute to road mishaps, and one of the causes is sleepy drivers [35]. Simple prevention, such as short breaks, may reduce the risk of mishaps. However, the earliest detection (when the driver gets drowsy) is required to avoid road mishaps. In this work, we will monitor the awareness of the driver and alert the driver when necessary.

This sub-system consists of an in-vehicle camera with a shared edge microcontroller and local storage to capture the video images of the driver in real-time. The captured images are then fed to the microcontroller and processed using facial feature detection. The face of a driver in the image is detected using the Histogram of Oriented Gradients (HOG) [36] and Linear Support Vector Machine (SVM) [37]. Only eye-related landmarks will be used to detect drivers' awareness. It is done by computing the aspect ratio of the eye in relation to the number of video frames to determine if the driver is in a drowsy state. Hence, the awareness of the driver is determined. Also, since the data is made available via the cloud, the control fleet centre is kept informed of the driver's status. Hence, both the driver and fleet centre are well informed with this closed-loop monitoring system.

Although the eye feature-based drowsiness detection method becomes limited when the driver uses sunglasses, since both hardware and framework are made modular, other drowsiness detection techniques such as head movement, yawning, heart/pulse rate, etc. can be added to the system to improve the performance [15]-[17].

6.4 Driver Identification and Recognition System

Driver identification and monitoring system were implemented using the face recognition method to achieve two objectives, (i) personal identification and (ii) monitoring driving hours. This sub-system shares the in-vehicle camera, edge microcontroller and local storage. During the initial state (when handling the vehicle), the driver images were captured and fed to the microcontroller for authentication. As the position of the in-vehicle camera may not be aligned with the driver seating, a cascaded convolution neural network (CNN) proposed by Cai et al. [38] was adopted. The pose variation face recognition was achieved by training the CNN over various face poses. Upon enabling the face recognition system, the edge device acquires a list of registered vehicles with fleet and the legal drivers' face biometrics from the cloud to the edge device. The edge device compared it with the list for authentication. The driver is given access to operate the vehicle once identified. The system records the driver's duration of working hours on the cloud at at the end of the service. By keeping such record, the fleet centre can track the access and monitor the operation's health, avoiding any illegal driving and long working hours, leading to fatigue and alert can be generated when exceeding the fleet operation policies.

7. Design, Results and Discussion of Prototype Experiment

The edge devices were developed and validated in the lab independently. We then integrated and implemented it on a vehicle and validated them in real-time at Cyberjaya, Malaysia.

The face recognition algorithm was validated on the local lab-based image database and resulted in satisfactory identification. It was then implemented on the edge device in the vehicle to validate the framework. The system can identify and register the driver's information and vehicle identity in real-time before operating the vehicle. Otherwise, the framework alerts both driver and fleet centre on an unauthorized operation. Further action by the fleet centre can be explored in such cases. Also, the total driving hours were recorded on the cloud at the end of the operation. Policies on driver's working hours and duration can be determined before implementing them on the framework to monitor automatically.

There were issues with streaming images in real-time across different modules, each carrying out different processing needs. An event-based routine was introduced to communicate and interrupt other processes during runtime to optimize the sharing, as shown in Fig. 6 and Fig. 7.

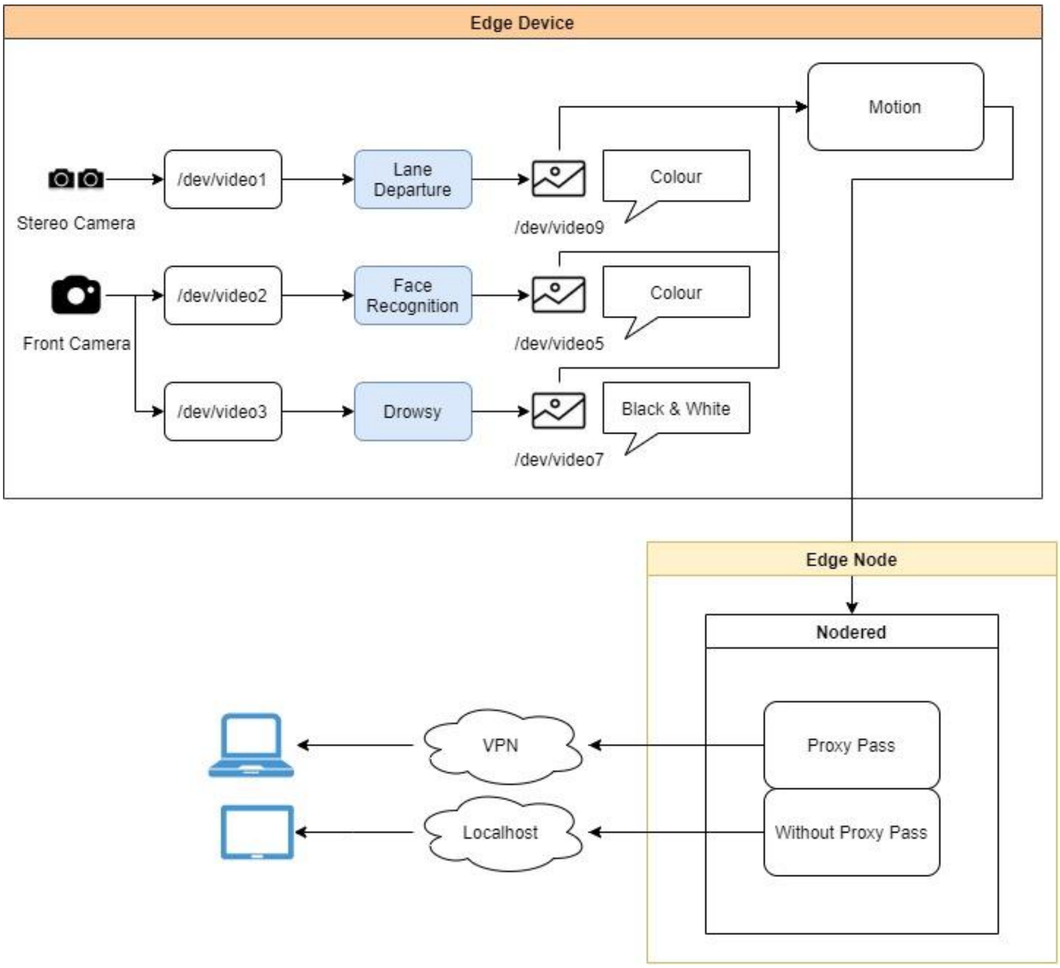


Fig. 6 - Optimization for real-time image sharing

The initial GPS experiment on the vehicle location tracking system was inaccurate. However, with careful calibration of both GPS and Google Maps, the vehicle could be located at a 1.1 mm distance accuracy in real-time. In addition, other data such as date, time, and speed were recorded on the map and published on the webserver in real-time, as illustrated in Fig. 8 (b).

The lane tracking algorithm was successfully validated in ~~a~~ lab-based experiment, as shown in **Error! Reference source not found.**. The algorithm's parameters were optimized on local image database images to ensure they suit the local environment before implementing it in the vehicle. It is then validated in the real world, as shown in (e) and (f) of **Error! Reference source not found.**. Similarly, the drowsiness system was successfully validated in the lab-based experiment before implementing it in the vehicle for real-world application. **Error! Reference source not found.** shows an example of eye detection. Both lane tracking and drowsiness detection systems were able to respond according to the software architecture of the framework in real-time.

Further improvements were made to the webserver to support the increasing number of devices as more rigorous testing occurred. A layer of in-memory data store was added to the architecture stack. Any data meant for the server will be first directed to this layer, and other services consuming it will be fetched from it. It acts as a runtime cache and dramatically reduces the roundtrip time to fetch data from the database.

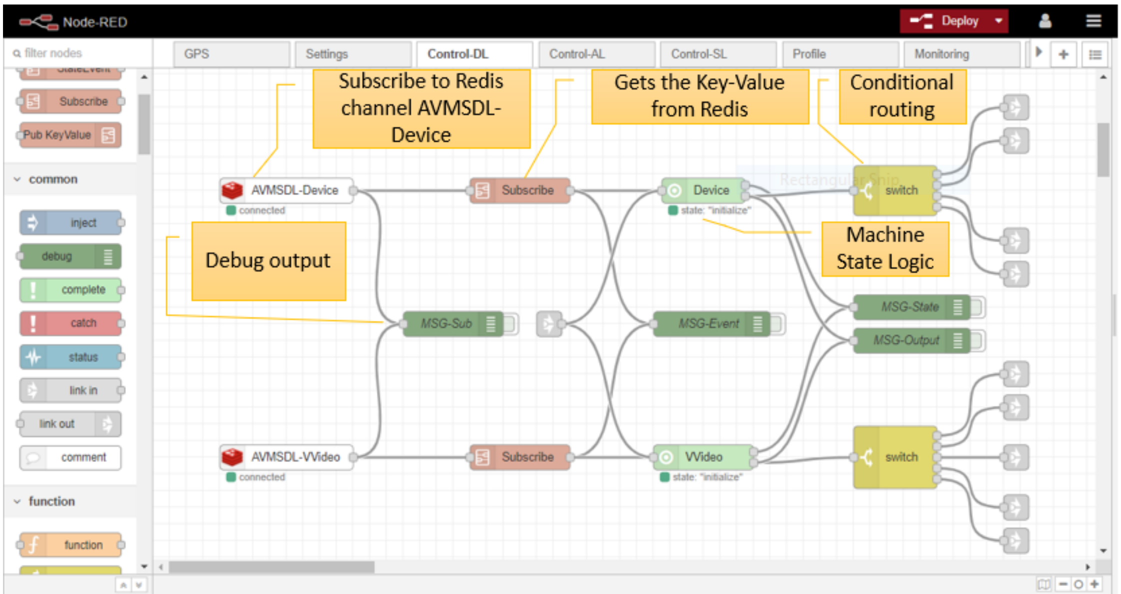
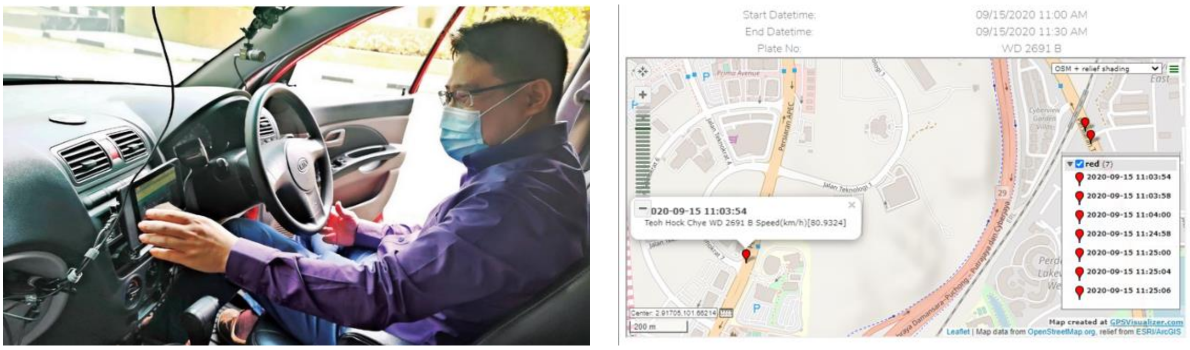


Fig. 7 - Use of machine state logic for event-based process



(a) (b)

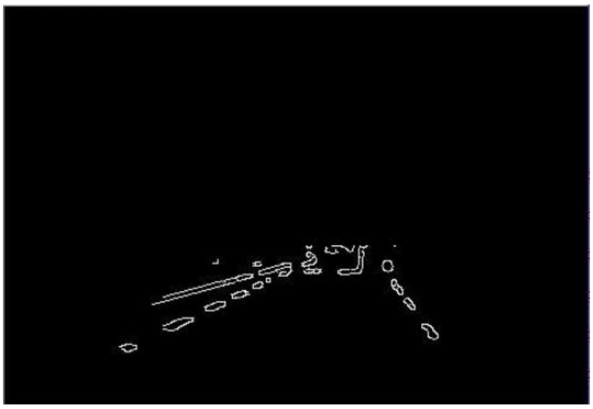
Fig. 8 - Real-time driver awareness monitoring and alert system (a) the authorization process using face recognition before starting the journey. Photo courtesy by theedgemarket, and; (b) real-time report vehicle obtained via the cloud

The web server was accessed on a secured web page with the proposed IoT-based framework. The vehicle's lane tracking activities were monitored and shared on the fleet management system. Besides that, the alert generated by the lane tracking system and drowsiness detection systems were captured on the scoreboard for future management review and enforcement. When necessary, we showed that the fleet operator could reach out to the vehicle in real-time via the framework to intervene with the driver's behavior. We also demonstrated when there was internet interruption on the GSM edge module, the vehicle activity can be stored on the edge storage device and transmitted when the connection is back to normal.

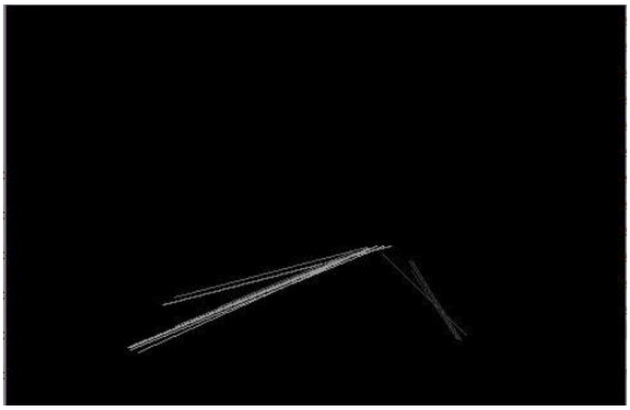
As for the framework, we implemented Docker technology on a standard microserver framework ensuring the user and its function can be expanded easily without compromising end-to-end data encryption for security. Only authorized personnel from the fleet centre can access UI via the login page. An admin dashboard with information such as the number of users, drivers and vehicles were displayed for quick admin access. Detailed tabs such as route report, speeding report, etc., were made available for analysis. The fleet manager could choose to enable the application, device-, and service layer for admin control, as shown in **Error! Reference source not found.**. The summary of the experiment results are shown in **Error! Reference source not found.**.

The bi-directional alert at the vehicle (edge device) is an essential part of the ITS. The alerts must responds efficiently when the number of edge devices grows significantly. Therefore, simple UI at the edge device is crucial in maintaining the comfort of the driver. Similarly, as the number of drivers increases in the management system, the alerts must be managed profoundly to avoid mishaps due to slow response by the fleet centre. Therefore, the alerts need to be categorized based on the priority before broadcasting or responding to the other end.

(a)



(b)



(c)

(d)

(e)

(f)

Fig. 9 - An example outcome of the lane tracking algorithm on a lab environment; (a) masking white and yellow line; (b) background removal; (c) hough transformation; (d) the lane tracking; (e) an example of lane tracking on real-world daylight experiment, and; (f) an example of lane tracking on real-world nightlight experiment

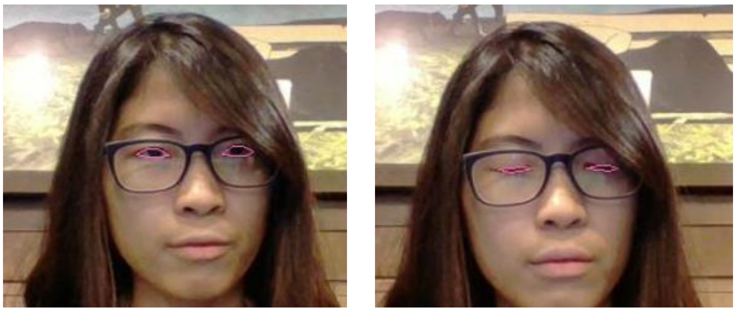
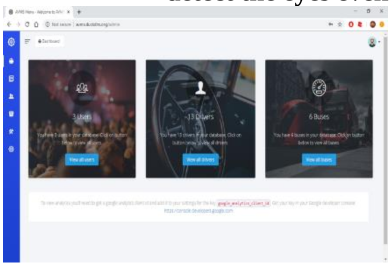
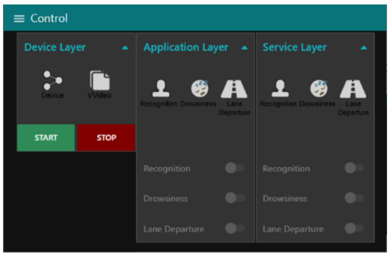
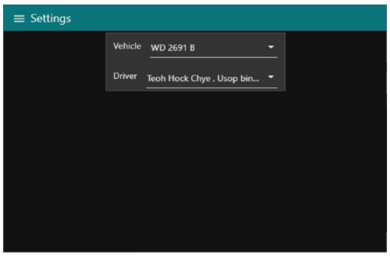
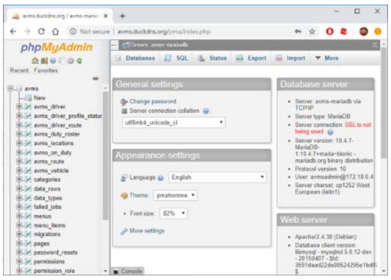


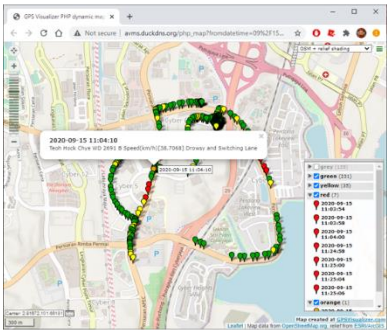
Fig. 10 - Driver awareness monitoring and alert system (a) an example of an input image. The system was able to detect the eyes even with spectacles, and; (b) the system detects and alerts the user


(a)


(b)


(c)


(d)


(e)


(f)

Fig. 11 - Framework implementation (a) cloud server UI-admin dashboard; (b) edge node UI control panel; (c) edge node UI profile configuration; (d) cloud server UI-admin database; (e) an example of Cloud server UI speed report, and; (f) edge node UI profile login

Table 1 - Experimental results

Parameter Measurement	Results / Accuracy
Vehicle Identification	Car registration plate (Embedded in the Edge device with vehicle ID)
Driver Name	Face recognition on cloud at 2 second interval
Vehicle Location (Longitude/Latitude)	8th place decimal or 1.1 mm accuracy
Cloud Tracking / Monitoring / Recognition	2 second intervals
Edge Node Tracking / Monitoring	7 frames per second
Vehicle Speed	0.1 km/h accuracy

We found that weather such as heavy rain can dramatically increase the risk of transmission losses. Although the data can be held and stored at both ends before reconnecting the transmission, it is still open to risk. Extreme temperature prompt to risk of edge device failure. Therefore, along with the framework, a standard operating procedure (SOP) needs to be defined for such rare situations to avoid panic.

As suggested earlier in the proposed framework, the number of services (user) can be increased by expanding the containers and nodes for a sustainable ITS framework. However, thorough validation must be conducted to ensure the edge node can support the increase of edge devices. In addition, the data traffic at the edge device must also be optimized for maximum data transfer efficiency.

Imagine a speeding camera/device stationed at-risk area to enforce the regulation, and the road user would probably slow down to avoid the penalty and speed off after that. With the proposed IoT-based framework, such regulation can always be efficiently monitored holistically. One can consider road transportation policymaking by collecting the data and the driver's scoreboard in the longer term. Expanding more safety-based smart sensors with ITS standardization increases the safety measure and provides better road transportation regulations for sustainable smart cities. The impact of this framework would reduce fatality and provide safety to all road users, saves national wealth in both implementation and operations, and, more importantly, create an efficient transportation system.

As far as verification and validation of the IoT system, we adhered to the standards (best practices), data privacy and security of the overall system during implementation. We also meet the individual layers and overall system's latency requirements, so it meets the real-time demands (scalability, connectivity, security, modularity). With the Docker technology, we found that automation, simulation, and analytics made iterative test execution, high load testing, and data monitoring easy.

Compared to related works [26]-[29] that focus on managing the fleet, our proposal focuses on a modular-based IoT fleet management system that can easily be adopted to enhance vehicle safety features and reduce road mishaps. Furthermore, the real-world experiments show that vehicle activities, especially the safety parameters, can be monitored and analyzed in real-time.

8. Conclusion

This paper presents a sustainable IoT-based framework for vehicle monitoring and driver assistance system in smart cities to reduce road mishaps using intelligent sensors. We proposed modular-based edge node expansion via a scalability microarchitecture framework that allows the seamless inclusion of intelligent edge devices to be connected, monitored, and controlled within the IoT-based fleet management system. We designed and implemented the cloud-based fleet management framework on Docker technology, where each edge node's application was containerized in an image. This avoids custom or proprietary communication protocol (standards), reduces the dependability of application environment, and enables the scalability at both edge node and cloud framework. In this investigation, we showed edge devices such as lane tracking, drowsiness detection, and driver identification/recognition can be easily implemented for real-time monitoring and alert system performed at 2 seconds intervals. We noted several challenges in making the edge devices reliable in a dynamic situation. A focused investigation such as GSM communications interruption and bad weather conditions will be investigated in future to improve the intelligence of the system. We hope that adopting our framework and investigating it in a complex environment spurs further research and nurture new solutions to many of these issues, including autonomous vehicle and vehicle-to-vehicle communication.

Acknowledgement

This project was funded by the Malaysian Technical Standard Forum Bhd (MTFSB) [MTFSB/GRANT2018/IOT/037] under the Malaysian Communications and Multimedia Commission (MCMC). The authors would also like to thank Cyberjaya Sdn Bhd, TMS Lite Sdn Bhd, and CPeEnergy Sdn Bhd for supporting this proof-of-concept project.

References

[1] Keall M. D., Frith W. J., & Patterson T. L. (2004). The influence of alcohol, age and number of passengers on the night-time risk of driver fatal injury in New Zealand. Accident Analysis & Prevention, 36, 49-61.

[2] Wagenaar A. C., O'Malley P. M., & LaFond C. (2001). Lowered legal blood alcohol limits for young drivers: Effects on drinking, driving, and driving-after-drinking behaviors in 30 states. American Journal of Public Health, 91, 801-804.

[3] González-Sánchez G., Olmo-Sánchez M. I., Maeso-González E., Gutiérrez-Bedmar M., & García-Rodríguez A. (2021). Traffic Injury Risk Based on Mobility Patterns by Gender, Age, Mode of Transport and Type of Road. Sustainability, 13, 10112.

[4] Board A. R. S., (2017). Monitoring Road Safety in the EU: Towards a Comprehensive Set of Safety Performance Indicators. European Commission, Directorate General for Transport.

[5] Voas R. B., Tippetts A. S., & Fell J. C. (2003). Assessing the effectiveness of minimum legal drinking age and zero tolerance laws in the United States. Accident Analysis & Prevention, 35, 579-587.

[6] Treat J. R., Tumbas N. S., McDonald S. T., Shinar D., Hume R. D., Mayer R. E., et al. (1979). Tri-Level Study of the Causes of Traffic Accidents. National Highway Traffic Safety Administration, US Despartment of Transportation.

[7] Zhou H. & Wang H. (2017). Vision-based lane detection and tracking for driver assistance systems: A survey. 2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM), pp. 660-665.

[8] Feniche M. & Mazri T. (2019). Lane detection and tracking for intelligent vehicles: A survey. 2019 International Conference of Computer Science and Renewable Energies (ICCSRE), pp. 1-4.

[9] Chen W., Wang W., Wang K., Li Z., Li H., & Liu S. (2020). Lane departure warning systems and lane line detection methods based on image processing and semantic segmentation: A review. Journal of Traffic and Transportation Engineering (English Edition), 7, 748-774.

[10] Mittal A., Kumar K., Dhamija S., & Kaur M. (2016). Head movement-based driver drowsiness detection: A review of state-of-art techniques. 2016 IEEE International Conference on Engineering and Technology (ICETECH), pp. 903-908.

[11] Ngxande M., Tapamo J., & Burke M. (2017). Driver drowsiness detection using behavioral measures and machine learning techniques: A review of state-of-art techniques. 2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech), pp. 156-161.
[12] Pratama B. G., Ardiyanto I., & Adji T. B. (2017). A review on driver drowsiness based on image, bio-signal, and driver behavior. 3rd International Conference on Science and Technology - Computer (ICST), pp. 70-75.
[13] Kamran M. A., Mannan M. M. N., & Jeong M. Y. (2019). Drowsiness, fatigue and poor sleep's causes and detection: A Comprehensive study. IEEE Access, 7, 167172-167186.
[14] Laouz H., Ayad S., & Terrissa L. S. (2020). Literature review on driver's drowsiness and fatigue detection. 2020 International Conference on Intelligent Systems and Computer Vision (ISCV), pp. 1-7.
[15] Kumar M. (2020). Driver drowsiness detection techniques: A review. Advances in Mathematics: Scientific Journal, 9(6), 6.
[16] Gupta N. K., Bari A. K., Kumar S., Garg D., & Gupta K. (2021). Review paper on yawning detection prediction system for driver drowsiness. 5th International Conference on Trends in Electronics and Informatics (ICOEI), pp. 1-6.
[17] Awasthi A., Nand P., Verma M., & Astya R. (2021). Drowsiness detection using behavioral-centered technique-A Review. 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence), pp. 1008-1013.
[18] Pawłowski W., Goniewicz K., Schwebel D. C., Shen J., & Goniewicz M. (2019). Road traffic injuries in Poland: magnitude and risk factors. European Journal of Trauma and Emergency Surgery, 45, 815-820.
[19] Siev S. & Kliger D. (2019). Priming drivers by exposure to active and inactive speed displays. Journal of Transportation Safety and Security, 1-19.
[20] Gargoum S. A. & El-Basyouny K. (2018). Intervention analysis of the safety effects of a legislation targeting excessive speeding in Canada. International Journal of Injury Control and Safety Promotion, 25, 212-221.
[21] Jiang E. (2020). A review of the comparative studies on traditional and intelligent face recognition methods. 2020 International Conference on Computer Vision, Image and Deep Learning (CVIDL), pp. 11-15.
[22] Alsrehin N. & Al-Taamneh M. A. (2020). Face recognition techniques using statistical and artificial neural network: A comparative study. 3rd International Conference on Information and Computer Technologies (ICICT), pp. 154-159.
[23] Anagha K. & Ram A. R. (2020). Pose tolerant face recognition: A review. 2020 International Conference on Communication and Signal Processing (ICCSP), pp. 0147-0152.
[24] Praveen G. B. & Dakala J. (2020). Face recognition: Challenges and issues in smart city/environments. 2020 International Conference on COMmunication Systems & NETworkS (COMSNETS), pp. 791-793.
[25] Li L., Mu X., Li S., & Peng H. (2020). A review of face recognition technology. IEEE Access, 8, 139110-139120.
[26] Rojas B., Bolaños C., Salazar-Cabrera R., Ramírez-González G., Pachón de la Cruz Á., & Madrid Molina J. M. (2020). Fleet management and control system for medium-sized cities based in intelligent transportation systems: From review to proposal in a city. Electronics, 9, 1383.
[27] Hamadaqa E., Mars A., & Adi W. (2020). Physical security for fleet management systems. Cryptography, 4, 1.
[28] Hussain S., Mahmud U., & Yang S. (2021). Car e-talk: An IoT-enabled cloud-assisted smart fleet maintenance system. IEEE Internet of Things Journal, 8, 9484-9494.
[29] Xu G., Li M., Luo L., Chen C.-H., & Huang G. Q. (2019). Cloud-based fleet management for prefabrication transportation. Enterprise Information Systems, 13, 87-106.
[30] Docker Technology. https://www.docker.com/
[31] Guerrero-Ibáñez J., Zeadally S., & Contreras-Castillo J. (2018). Sensor technologies for intelligent transportation systems. Sensors (Basel, Switzerland), 18, 1212.
[32] Tewolde G. S. (2012). Sensor and network technology for intelligent transportation systems. IEEE International Conference on Electro/Information Technology, pp. 1-7.
[33] Duda R. O. & Hart P. E. (1972). Use of the Hough transformation to detect lines and curves in pictures. Communications of the ACM, 15, 11-15.
[34] Yeo B. C., Lim W. S., Lim H. S. (2016). Lane detection in the absence of lane markings for roadway surveillance with thermal vision. International Journal of Innovative Computing, Information and Control, 12, 677-688.
[35] Phillips R. O. & Sagberg F. (2013). Road accidents caused by sleepy drivers: Update of a Norwegian survey. Accident Analysis and Prevention, 50, 138-146.
[36] Dalal N. & Triggs B. (2005). Histograms of oriented gradients for human detection. 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), pp. 886-893.
[37] Cortes C. & Vapnik V. (1995). Support-vector networks. Machine Learning, 20, 273-297.
[38] Cai Z., Liu Q., Wang S., & Yang B. (2018). Joint Head pose estimation with multi-task cascaded convolutional networks for face alignment. 24th International Conference on Pattern Recognition (ICPR), pp. 495-500.