Urban Traffic Monitoring and Modeling System: An IoT Solution for Enhancing Road Safety

Rateb Jabbar*†, Mohammed Shinoy*, Mohamed Kharbeche*,
Khalifa Al-Khalifa*, Moez Krichen‡, Kamel Barkaoui†

*Qatar Transportation and Traffic Safety Center, Qatar University, Qatar
{rateb.jabbar,m.shinoy,mkharbec,alkhalifa}@qu.edu.qa

†Cedric Lab, Computer Science Department,Conservatoire National des Arts et Meteirs, France
{jabbar.rateb.auditeur,kamel.barkaoui}@cnam.fr

‡ReDCAD Laboratory, National School of Engineers of Sfax, University of Sfax, Tunisia
moez.krichen@redcad.org

Abstract—**Qatar expects more than a million visitors during the 2022 World Cup, which will pose significant challenges. The high number of people will likely cause a rise in road traffic congestion, vehicle crashes, injuries and deaths. To tackle this problem, Naturalistic Driver Behavior can be utilised which will collect and analyze data to estimate the current Qatar traffic system, including traffic data infrastructure, safety planning, and engineering practices and standards. In this paper, an IoT-based solution to facilitate such a study in Qatar is proposed. Different data points from a driver are collected and recorded in an unobtrusive manner, such as trip data, GPS coordinates, compass heading, minimum, average, and maximum speed and his driving behavior, including driver's drowsiness level. Analysis of these data points will help in prediction of crashes and road infrastructure improvements to reduce such events. It will also be used for drivers' risk assessment and to detect extreme road user behaviors. A framework that will help to visualize and manage this data is also proposed, along with a Deep Learning-based application that detects drowsy driving behavior that netted an 82% accuracy.**

Index Terms—**Internet of Things, Android, Drowsiness Detection, Driver Behavior Analysis, Deep Learning.**

I. INTRODUCTION

According to the Global status report [1] on road safety 2018 from the World Health Organization, Road crashes and related forms of crashes is the eighth leading cause of death globally. According to the crash data published in Qatar for 2017 [2], the total number of road crashes in Qatar is around 240,333 [2] and negligence and reckless driving come as one of the first leading causes. It is worth to mention that 97.5% of these crashes are minor.

Qatar expects more than a million visitors during the 2022 World Cup, which will pose significant challenges. The high number of people will likely cause a rise in road traffic congestion and safety concerns. To tackle this problem, a system that incorporates Naturalistic Driving Behavior (NDB) can be utilized to collect and analyze data that help to understand the current Qatar traffic system and provide suggestions for better infrastructure, safety planning, and engineering interventions.

In order to understand driver behavior, NDB Study is used to investigate the causes of crashes and the typical daily driving behavior. There is no accepted definition of Naturalistic Driving Study. However, as defined by [3], it is in general considered as an unobtrusive observation method for investigating daily driving behavior of drivers in a non-experimental and natural setting (Schagen et al., 2012 [4]; Dingus et al., 2014 [5]; Ingrid et al., 2009 [6];). In this kind of study, drivers are not given instructions on how, where, and when to drive their vehicles. Accordingly, researchers can observe the natural interaction of drivers with traffic, road-way, and other vehicles during daily driving activities. Due to the improvements in storage capacities and data collection technology, researchers are able to carry out such studies on a large scale. In these studies, the standard Data Acquisition System (DAS) generally consists of video cameras, forward radars, lane tracking system, vehicle network, data storage system, eye-tracking system, Geographic Positioning System (GPS), vehicle network information, and accelerometers. To perform a naturalistic driving study, it is necessary to have a significant amount of data. As an example, in two years, SHRP2 [7] NDS obtained 2 petabytes (2 million gigabytes) data from over 3,000 drivers.

Furthermore, Machine-to-Machine (M2M) [8] is defined as the communication between mobile devices, actuators, smart sensors, embedded processors, and computers without or with limited human intervention. M2M is applied in areas such as city automation, safety and security, transportation management, e-health, smart grid, and smart power. These devices are interconnected and known as the Internet of Things (IoT). The IoT enables intelligent monitoring, reporting, and control of different areas of our daily lives. IoT implementations are on the rise due to the advancements in internet speed with higher bandwidth and smaller telecommunication equipment.

In this context, our proposed solution includes a system that captures driver behavior using mobile phone technologies. The system is able to capture the driver's GPS data and analyze facial image data to predict drowsiness using deep learning techniques such as CNN to classify drowsy driving. Furthermore, this solution also enables the collection of data about every vehicle trip which facilitate the NDB studies.

The rest of this paper consists of five sections. First, an overview of the developed system is described in section II. Following that, the experimental results and analysis are discussed in section III. Next, in section IV, summary and main conclusions are presented. Finally, in Section V, Future work is presented.
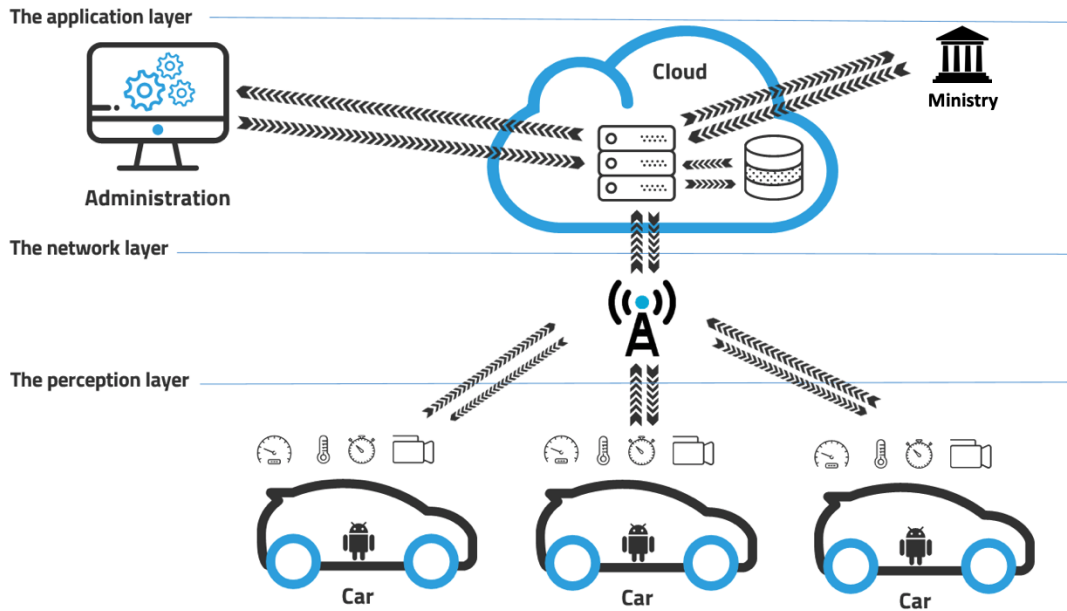
Fig. 1. Architecture of the proposed Internet of things solution.

II. METHODOLOGY

The architecture of our solution consists of three layers as outlined below: perception, network, and application layers:

i. The perception layer consists of multiple sensors that are designed to sense and collect information about the surroundings, including physical parameters and identification of nearby smart objects.

ii. The network layer is in charge of connecting sensors to other servers, network devices, and smart things, as well as transmission and processing of sensor data.

iii. The application layer delivers application-specific services to the user. The architecture of the proposed solution is shown in figure .1.

A. The perception layer

In the perception layer, an Android application performs two functions: Firstly, a driver drowsiness detection which analyses the facial data and alerts the user regarding drowsy driving behavior. The second function is to collect data about the trip of the car

1) Driver drowsiness detection: The goal of the driver drowsiness detection system is to prevent crashes caused by drowsiness. This is an important step that will enhance the existing Advanced Driver Assistance Systems (ADAS) in a car. The role of ADAS is to improve safety and ensure a satisfactory driving experience for the driver.

In recent years, there has been accelerated advancement in machine learning, particularly in the field of deep learning. This advancement along with the boom in the usage of embedded smart devices has resulted in increased data collection and connectivity. This has contributed to building efficient solutions to improve the current driver drowsiness detection system.

Our work aims to develop a drowsiness detection system with improved efficiency based on CNN as a classifier on Android platforms.

a) Literature review: Deep learning has become a widely used method for resolving challenging classification problems where conventional machine learning methods have lower accuracy. CNN's are used for tasks related to machine vision such as image classification, image segmentation, object detection [9] [10], etc. This technology has also been used by researchers to detect drowsy driving behavior too.

Vijayan et al. [11] used 3 different CNN models to form a feature fused architecture to detect drowsiness which resulted in a 78% accuracy. Dwivedi et al. [12] used shallow CNNs for detecting drowsy drivers with a 78% accuracy rate. Liu, Weihuang, et al. [13] used a two-stream network along with multi-facial features to detect drowsiness on the NTHU-DDD dataset. The two-stream network was able to combine static and dynamic image information, this implementation even accounts for improving the lighting conditions of the images by using gamma correction on the images.

Miguel et al. [14] have implemented a low-cost drowsy detection system and it was able to achieve an accuracy of 72%. This implementation shows promises as they were able to develop the system for an Android application. In our related work [15], we were able to attain an 81% accuracy on average using an MLP model with facial landmarks while keeping the size of the model at only 100KB.

In this paper, our contribution is a CNN algorithm for drowsiness detection and a framework that can work with the mobile application to collect this data for naturalistic driving behaviour studies.

b) Dataset and Preprocessing: In this study, the National Tsing Hua University Driver Drowsiness Detection dataset is used [16]. From this dataset, 22 participants of varied ethnic diversity are selected, a sample of which is given in figure 2. Out of this, 18 people were used in training data and 4 of them were used in the testing. All subjects were recorded in different simulated driving scenarios under night and day-lighting conditions. The participants enacted different driving behaviors such as dozing off, head nods, slow blink rate, yawning while simulating a natural driving scenario. The videos are taken using an infrared (IR) camera that has a resolution of 640 X 480 and 30 frames per second.

c) Classifier model preparation: A deep learning model is utilized here to detect the driver's drowsy status. Individual

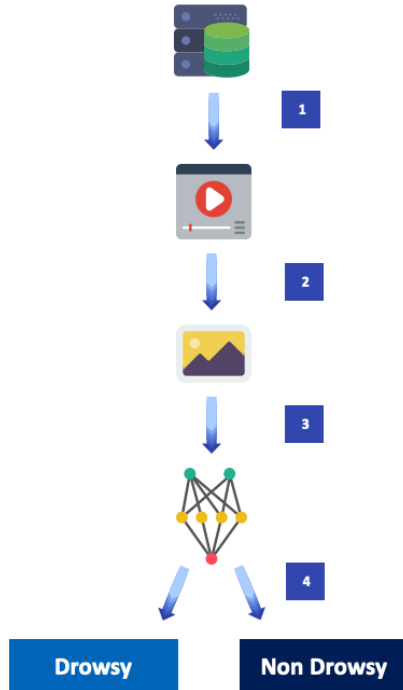Fig. 2. Sample of the pictures from the selected dataset

Fig. 3. Pre-prossessing and training steps

Algorithm 1 Real-Time Driver Drowsiness Detection Algorithm

Input: images of drivers and labels
Output: Learned CNN model
 1: Input (1X128X128)
 2: Convolution 64(3X3)
 3: Leaky ReLU ($\alpha = 0.1$)
 4: Max Pooling (2X2)
 5: Dropout (0.25)
 6: Convolution 128(3X3)
 7: Leaky ReLU ($\alpha = 0.1$)
 8: Max Pooling (2 X2)
 9: Dropout (0.25)
10: Convolution 128(3X3)
11: Leaky ReLU ($\alpha = 0.1$)
12: Max Pooling (2X2)
13: Dropout (0.25)
14: Flatten
15: Dense (128 , activation='linear')
16: Leaky ReLU ($\alpha = 0.1$)
17: Dropout (0.5)
18: Softmax Output

algorithm is saved onto the Android device to be used by the mobile application.

2) Vehicle Data Collection System: The vehicle data collection system is designed to collect trip data information i.e start time, end time, distance, and the minimum, maximum, and average speed, acceleration, distance, and GPS position every 15 seconds. It also collects information from the magnetometer in the phone to measure the rotational velocity along the Roll, Pitch and Yaw axes.

As illustrated in figure.4, the Android application is composed of four main pages. The first page serves for logging in by using a username and password. Following the authentication, the user can start a new trip or access the information about the last trips.

If the user chooses a new trip, the application will start recording information about the driver's behavior, trip, and send the collected data via a web service to the database server. The current time and speed, the GPS position, duration of the trip, GPS precision, compass heading, minimum, average, and maximum speed, a rotational velocity along the Roll, Pitch and Yaw axes, and acceleration are also displayed. Video is captured using the front-facing camera and this is fed to the Machine Learning algorithm for drowsy driving detection.

B. The network layer

The network layer establishes the connection between the servers and transmits and processes the sensor data. The application can use either Wi-Fi or mobile internet (3G/3G+/4G) to send the data to the server. This process uses a hybrid system to gather and store data locally before transmitting them to the server. This technique is very effective when the internet connection is poor or unstable.

C. The application layer

The application layer delivers application-specific services to the end-user. It sends the obtained data to the web services

frames in the video to carry out the prediction were used.

The creation of the model involves four steps as illustrated in figure 3.

- Step 1– Selection of videos from the NTHU Database: The subjects selected for the model training needs to be of different ethnicities. Out of the selected participants, 18 subjects were part of the training dataset and 4 subjects were used in the evaluation dataset.
- Step 2 – Converting video into frames of images: The videos are converted into a series of images and each frame is represented as drowsy or non- drowsy.
- Step 3 – Training the algorithm: The series of images are the input to the algorithm, the algorithm is a CNN with three hidden layers as explained in Algorithm 1. During the training phase, the model learns about the drowsiness levels from the training dataset. The training will continue until the preferred level of accuracy is achieved for the model on the training data.
- Step 4 – Predictions and Saved Model: With the trained model, predictions can be made. This
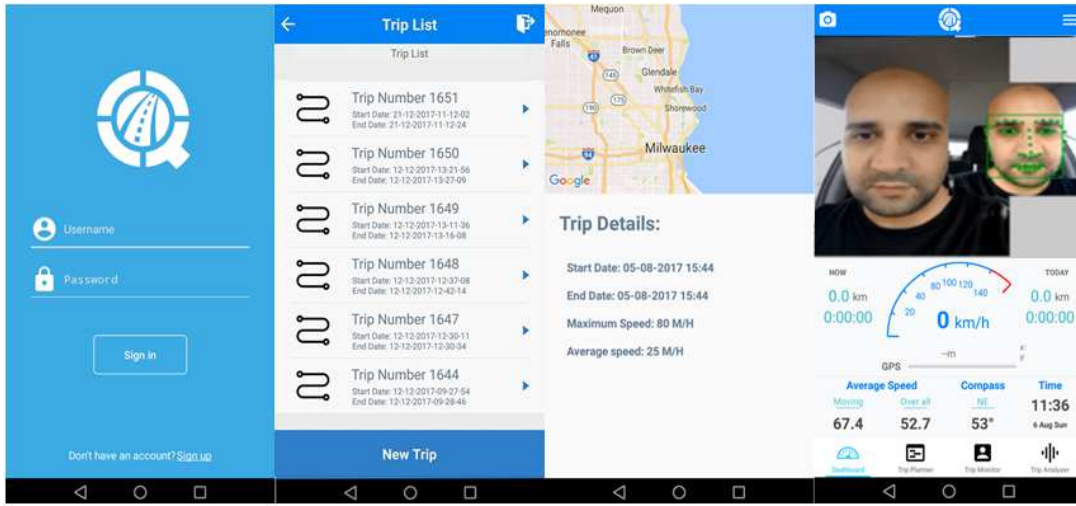
Fig. 4. Screenshot of the four main pages in the Android application.

for processing and analysis before showing them to the end-user. The web service is a component of the application layer. In addition to collecting data from the devices, it can also use sources such as the general traffic directory of the ministry to obtain information about crashes. This web service stores data in the database server and performs the analysis. User data can also be viewed by the end-user through this web interface. Windows Communication Foundation by Microsoft is used to implement the web service, based on the REST architecture and JSON format. As a result, the system can be easily set up to interact with any other embedded system.

The Database server stores and queries the collected data. The database architecture used in the Database server consists of seven main tables. "Driver" table stores demographic information such as age and gender of the driver. The "Vehicle" section has information related to the driver's car. The road information is kept in the "Road" table. In the table "SegmentRoad," the focus is on collecting data related to the road segments, which encompass the infrastructural information about the road. The table "Crashes" consists of the data retrieved from the stakeholders about historical crashes. This table is related to the table "Segment Road" so that it is possible to compute the risk related to different segments, as they are variable. Lastly, the "Events" table records the data that the mobile records every 15 seconds. The system is flexible in a way that you may keep adding variables to each table as the need arises in order to facilitate the computation of new proposed analytical models. Microsoft SQL Technologies are used to build the database server to ensure adequate performance, flexibility ,and scalability when dealing with extensive data in a cost-effective manner.

The Web application is the interface the researchers use to interact and query the recorded data. The website displays demographic information about the driver. It also includes information about the vehicle such as the model and date of putting it into service.

By using Google Maps API, the website displays the tracked trip and the position of individual events as illustrated figure 5, as well as the details of all recorded events as shown in figure.6.

TABLE I
ACCURACY PER DRIVING SCENARIOS

Category	Accuracy
With glasses	85.548
Night Without glasses	83.20
Night With glasses	79.142
Without glasses	89.16
With sunglasses	78.115
All	82.9

III. EXPERIMENTAL RESULT

For our driver drowsiness detection system, subjects were recorded in five simulated driving scenarios - with glasses, without glasses, with sunglasses, night without glasses and night with glasses. Every driving scenario has two different states - sleepy and non-sleepy. As a result, a total of 200 videos were used.

The number of videos used for each category in the training dataset was 36. However, the number of extracted images varies. In total, there were 50,991 images in the scenario recorded in "night with glasses", 52,372 performed in "night without glasses", 106,992 in "with glasses", 107, 990 in "with sunglasses" and 108,380 in "without glasses". In the evaluation set, each category has 4 videos with the following number of extracted images: 32,922 images in "the night with glasses" scenario, 29,781 "the night without glasses" scenario, 37,357 in "with glasses" scenario, 28,214 in "with sunglasses" scenario, and 45,005 in "without glasses" scenario. In total, 599,894 images were extracted all from the 200 videos.

The training on the neural network model was performed by the computer with the following specifications: Intel Core i7-7500U, 16 GB RAM, NVIDIA GeForce GTX 670MX. Table I shows the results for the mentioned categories. According to the obtained results, the proposed solution attained on average for all scenarios an 83% accuracy rate, which when compared to Park et al. [17] exceeds it and comes shy of 2 percentage points when compared to Guo et al. [18] work.

On further inspection of the images that were detected and not detected, it was revealed that the most important facial feature for drowsiness classification in all situations
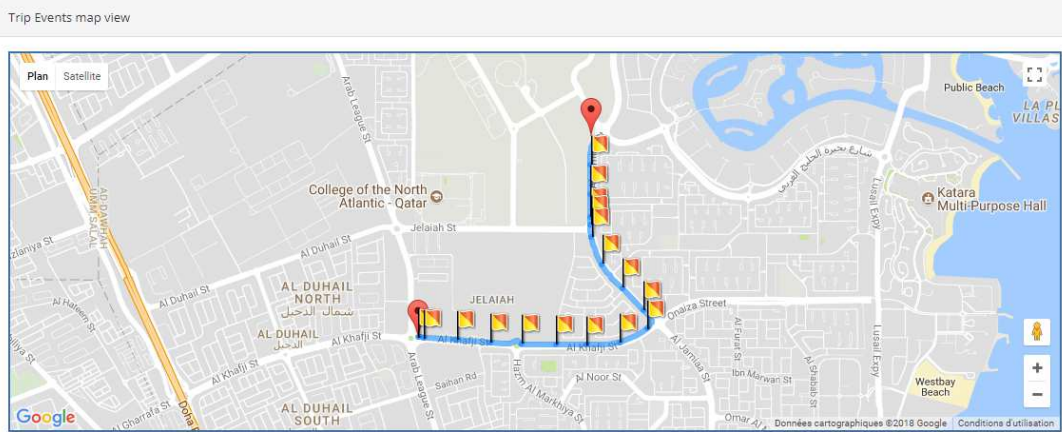
Trip Events map view

Fig. 5. Screenshot of a real trip displaying the tracked trip and the position of individual events.

Drag a column header and drop it here to group by that column

ID	Timestamp	Lat	Long	Speed	Dista...	Acc-x	Acc-y	Acc-z	Roll-x	Yaw-y	PitchR-z	Note	Ima...
37507	16-10-2019-05-48-48												
37776	16-10-2019-05-50-35	25.354382470378...	51.49667511547988	45	0.20	-0.6464542	8.219545	-3.9050622	0.0778285...	0.13099578	-0.025073575		
37777	16-10-2019-05-49-34	25.36450145355348	51.4950845537416	39	0.15	-1.4916332	10.235525	1.4533248	0.13830417	0.0967873...	-0.03729088		
37778	16-10-2019-05-52-51	25.340079868532...	51.505891981609...	39	0.18	-0.696733...	8.868394	3.1580484	-0.056561...	-0.018666...	0.047008526		
37779	16-10-2019-05-53-57	25.329602207398...	51.505599858888771	22	0.15	-0.3519584	8.61939	3.0024207	0.14074764	0.0919004	0.0011936307		
37780	16-10-2019-05-51-35	25.346381951568	51.504992488028...	41	0.17	0.47167215	10.379181	2.7725701	-0.018077...	0.0473072...	-0.04156694		
37788	16-10-2019-05-54-43	25.329034006165...	51.5053388132745	12	0.02	-0.052674...	9.289786	-3.014392	-0.004027...	0.1102263...	-0.055616837		
37790	16-10-2019-05-53-27	25.3340411848466	51.505906163000894	39	0.17	-0.102953...	9.617802	2.2889266	0.0283484...	0.04547464	-0.006136752		
37791	16-10-2019-05-51-05	25.350428983979...	51.5009134575 31	43	0.20	-0.909824...	8.118986	-3.5171897	0.0564482...	-0.034548...	0.020130454		
37800	16-10-2019-05-55-43	25.32271573867297	51.510366362 38765	36	0.17	-1.0678465	9.002473	-1.3360053	-0.008303...	-0.163441...	0.08671477		
37808	16-10-2019-05-49-04	25.367945754964...	51.494725029 02323	19	0.03	-0.440546...	10.491713	-0.531529	-0.013801...	0.01737484	-0.03179309		
37809	16-10-2019-05-51-51	25.348303584781...	51.503250616504...	46	0.21	-0.225061...	8.961771	0.08140534	0.0393439...	0.09312213	-0.0721102		

|◀ ◀ **1** 2 ▶ ▶| 20 ▼   items per page   1 - 20 of 32 items

Fig. 6. Screenshot of a real trip displaying the data recorded for every event.

were the eyes. As a consequence, it is not surprising that wearing sunglasses decreases the efficiency of the model, as the algorithm does not take into consideration the driver's eyes. Another crucial criterion for better performance is the brightness of the image. Indeed, when the image is brighter, the error rate decreases by 6%.

IV. CONCLUSION

In this paper, we presented a platform based on the Internet of Things to examine and analyse the traffic in the State of Qatar. The system focuses on implementing a naturalistic driver behavior approach and includes three primary layers. The perception layer represented in the Android application that includes the detector system of the behavior of the driver and the vehicle data collection system in order to collect data about the trip and the vehicle. Second, the network layer establishes the connection between the smartphone and the cloud. In the third layer, the web server obtains and analyzes data collected from the sensors, the database server stores the collected data and the website displays the data and analysis. The platform has been tested using two mobile phones and relevant variables were recorded. Results confirm a good performance of the drowsiness detection since the average accuracy is more than 82% as shown in table I for all scenarios.

V. FUTURE WORK

The current IoT-based system can be improved in several scenarios by porting many functionalities of ADAS into a portable modular Android device in the context of the Internet of Vehicles. This application can make use of the rear camera on the mobile device and include information from the environment too. For example, data from traffic signals, road signals can also be identified to provide feedback to the driver. The ADAS will be able to detect pedestrians and other vehicles and warn the driver of a potential collision.

ACKNOWLEDGEMENT

REFERENCES

[1] "Road traffic injuries." [Online]. Available: https://www.who.int/en/news-room/fact-sheets/detail/road-traffic-injuries

[2] "A press conference of ministry of interior about the traffic situation during 2017 in qatar - ministry of interior postal." [Online]. Available: t.ly/jz0bV

[3] "What is NDS - UDRIVE, European Naturalistic Driving Study." [Online]. Available: http://www.udrive.eu/index.php/about-udrive/what-is-naturalistic-driving

[4] I. van Schagen and F. Sagberg, "The Potential Benefits of Naturalistic Driving for Road Safety Research: Theoretical and Empirical Considerations and Challenges for the Future," *Procedia - Social and Behavioral Sciences*, vol. 48, pp. 692–701, 2012.

[5] T. Dingus, J. Hankey, J. Antin, S. Lee, L. Eichelberger, K. E. Stulce, D. McGraw, M. Perez, and L. Stowe, "Naturalistic Driving Study: Technical Coordination and Quality Control (No. SHRP 2 Report S2-S06-RW-1)," Virginia Tech Transportation Institute, Tech. Rep., 2015.

[6] S. Ingrid, E. Rob, and N. Nicole, "Promoting real life observations for gaining understanding of road user behavior in Europe," *Proceedings of the Road Safety on Four Continents Conference*, vol. 15, pp. 200–207, 2010.

[7] K. L. Campbell, "The shrp 2 naturalistic driving study: Addressing driver performance and behavior in traffic safety," *Tr News*, no. 282, 2012.

[8] K. Nahrstedt, H. Li, P. Nguyen, S. Chang, and L. Vu, "Internet of mobile things: Mobility-driven challenges, designs and implementations," in *Proceedings - 2016 IEEE 1st International Conference on Internet-of-Things Design and Implementation, IoTDI 2016*. Institute of Electrical and Electronics Engineers Inc., may 2016, pp. 25–36.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," Tech. Rep., 2016. [Online]. Available: http://image-net.org/challenges/LSVRC/2015/

[10] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," Tech. Rep. 4, 2017.

[11] V. Vijayan and E. Sherly, "Real time detection system of driver drowsiness based on representation learning using deep neural networks," in *Journal of Intelligent and Fuzzy Systems*, vol. 36, no. 3. IOS Press, 2019, pp. 1977–1985.

[12] K. Dwivedi, K. Biswaranjan, and A. Sethi, "Drowsy driver detection using representation learning," in *Souvenir of the 2014 IEEE International Advance Computing Conference, IACC 2014*. IEEE Computer Society, 2014, pp. 995–999.

[13] W. Liu, J. Qian, Z. Yao, X. Jiao, and J. Pan, "Convolutional two-stream network using multi-facial feature fusion for driver fatigue detection," *Future Internet*, vol. 11, no. 5, 2019.

[14] M. García-García, A. Caplier, M. Rombaut, and M. Rombaut, "Sleep Deprivation Detection for Real-Time Driver Monitoring using Deep Learning," Tech. Rep., 2018. [Online]. Available: https://hal.archives-ouvertes.fr/hal-01837080

[15] 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," in *Procedia Computer Science*, vol. 130. Elsevier B.V., 2018, pp. 400–407.

[16] C. H. Weng, Y. H. Lai, and S. H. Lai, "Driver drowsiness detection via a hierarchical temporal deep belief network," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10118 LNCS. Springer Verlag, 2017, pp. 117–133.

[17] S. Park, F. Pan, S. Kang, and C. D. Yoo, "Driver drowsiness detection system based on feature representation learning using various deep networks," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10118 LNCS. Springer Verlag, 2017, pp. 154–164.

[18] J. M. Guo and H. Markoni, "Driver drowsiness detection using hybrid convolutional neural network and long short-term memory," *Multimedia Tools and Applications*, vol. 78, no. 20, pp. 29 059–29 087, oct 2019.