Machine Learning to Classify Driving Events Using Mobile Phone Sensors Data

Yazan A Alqudah (✉)
University of West Florida, Pensacola, USA
yalqudah@uwf.edu

Belal Sababha, Esam Qaralleh
Princess Sumaya University for Technology,
Amman, Jordan

Tarek Youssef
University of West Florida, Pensacola, USA

Abstract—With the introduction of autonomous and self-driving cars, innovative research is needed to ensure safety and reliability on the road. This work introduces a solution to understand vehicle behaviour based on sensors data. The behaviour is classified according to driving events. Understanding driving events can play a significant role in road safety and estimating the expense and risks of driving a vehicle. Rather than relying on the distance and time driven, driving events can provide a more accurate measure of vehicle driving consumption. This measure will become valuable as more ride-sharing applications are introduced to roads around the world. Estimating driving events can also help better design the road infrastructure to reduce congestion, energy consumption and pollution. By sharing data from official vehicles and volunteers, crowd sensing can be used to better understand congestion and road safety. This work studies driving events and proposes using machine learning to classify these events into different categories. The acquired data is collected using embedded mobile device motion sensors to train machine learning algorithms to classify the events.

1 Introduction

The rapid growth in population is leading to the unprecedented expansion of cities around the world. Such growth necessitates smarter traffic management systems to reduce congestion and its negative impact [1][2]. One key element for smart cities' management systems is intelligent transportation systems (ITS) [3]. Intelligent transportation systems provide solutions to save lives, energy, and environment through

optimized transportation paths and road conditions monitoring. Modern vehicles are equipped with active safety features, which interact with the road conditions trying to avoid accidents or at least minimize the severity of accidents caused by road conditions.

The ability to classify driving events is critical to a smart city infrastructure. It enables ride sharing and vehicle rental to have a better estimate the cost of a trip. Rather than relying on time and distance alone, driving events will factor into the calculation of a trip expense. Driving events enable authorities to better monitor their fleets of vehicles and trucks to determine when drivers need a break or should no longer be driving. Another application of driving events classification is to enable rental car companies to get a better estimate of vehicle consumption. Insurance companies can also benefit from the classification by including driving events to the estimate of vehicle insurance.

To better illustrate the need for driving events, consider the following scenario. Assume two vehicles are intended to travel the same distance from one location to another. The first vehicle maintains a travel path at a constant speed, while the second vehicle drives a path where it must speed up then slow down repeatedly. Based on the tools available today, both vehicles will show the same distance travelled and therefore would have the same expense for the trip. Introducing driving events as a metric allows for a better estimate of the wear and tear that a second vehicle is subjected to and thus provides a more accurate estimate of the trip expense.

Driving events can also be used to investigate accidents and collisions. By understanding vehicle behaviour prior to an accident, driving events act as a "Black Box" of information. Another application of logging driving event is to evaluate and improve vehicle driver safety and performance. The information from the vehicle's travel can be analysed to determine dangerous behaviour of the operator or an inefficient driving behaviour such as speeding and breaking.

The evolution of computational capabilities and sensors integration in communication devices (mobile phones) are enabling the development of smart applications and solutions in the areas of health, agriculture, transportation, and safety. In this work, mobile phones are used to collect sensors data and Machine Learning is utilized to classify driving events using collected data. Several classifications algorithms are applied and studied for accuracy. Machine learning is used to determine the minimum features required to successfully classify the events. This research builds on previously published work that relates different sensors' data to driving events [4].

Section 2 reviews literature and presents related work. Section 3 presents the proposed solution. Section 4 introduces Machine Learning. Section 5 presents the results. Finally, Section 6 concludes the paper and discusses future work.

2 Literature Review

Various aspects of driving events are researched and studied. In [5], vehicle sensor data is used to study longitudinal jerk to identify aggressive driving. The work finds the distribution of the jerk associated with the brake pedal operation has a wider range

on both negative and positive side compared to the jerk associated with the gas pedal operation. The jerk is found to positively correlate to the speed the driver was pressing the gas pedal. The work concludes that aggressive drivers were associated with significantly higher values of Jerk-based metrics. Large negative jerk seems to have better performance in identifying aggressive drivers.

Driver distraction and its role in fatal crashes is studied in [6]. The study found that inner cognitive distraction accounted for the greatest proportion of driver's distractions. Young drivers have a high probability of being distracted by in-vehicle technology-related devices/objects. Among six subcategories of distractions, older drivers are more likely to be affected by inner cognitive interference.

The work in [7] proposes using electroencephalogram (EEG) spectrum to monitor drivers' alertness. Using the relationship between changes in driving performance and EEG spectrum, it might be feasible to estimate driving errors based on multichannel EEG power spectrum estimation and principal component analysis algorithm. EEG data is also used in [8] to classify driving styles. EEG features extracted from power spectral density and classification results of the driving data were used to train a Support Vector Machine (SVM) model. To evaluate the performance, a leave-one-subject-out cross validation was utilized. Different driving styles were related with different driving strategies and mental states. The results suggest the feasibility of driving style recognition from EEG patterns.

To assist with lane change, an algorithm for the Advanced Driver Assistant System (ADAS) to help classify the driver's intention was proposed in [9]. Measurements from conventional on-board sensors are augmented using an artificial neural network (ANN) model. The information is fed to a Support Vector Machine (SVM) to detect the driver's intention with high accuracy.

Driving styles was studied in [10] using an onboard measurement and communication unit. Data was acquired through the diagnostics port along with accelerometer and GPS data. Experiments were conducted on drivers to show that it is feasible to differentiate driving styles in terms of safety, economy, and comfort. A score was assigned based on the eight measurable indicators such as bumping, cornering, and speeding.

A driver detection system was proposed in [11]. The system utilizes the gyroscope and magnetometer data and the interplay between electromagnetic field emissions and engine startup vibrations to detect the driver position and events. The system is evaluated experimentally with four participants and three different vehicles using varying vehicle-riding scenarios. The results show that the system can identify the driver with 89.1% average accuracy for different scenarios.

A driver monitoring system using driver mobile phone was proposed in [12]. The system is based on Multi-Task MobileNets (MT-MobileNets). It consists of the MobileNets' base and multi-task classifier. The classifier recognizes facial behaviors related to the driver status, such as distraction, fatigue, and drowsiness.

3   Proposed Solution and Experiments

This work proposes utilizing sensors data to classify driving events. The growing processing capabilities of mobile devices make it possible to utilize a mobile phone equipped with sensory information to collect and process data [13][14].

Drive tests are conducted to collect sensor data during travel. A mobile device with motion sensors is used to log time, sensors data, and GPS location. The drive is repeated several times while trying to maintain the same travel path. The path allows the vehicle to accelerate and decelerate and contain ramp up and ramp down. The travel path is illustrated in Figure 1.

The collected data includes Acceleration, Gyro Rotation, Yaw, Roll, Pitch, Rotation rate, Quaternion, Gravity, Magnetic field, and Orientation. Data is collected in x, y, and z direction. The software logs over 43 measurements and is sampled at a rate equal to 30 samples per second.

Using GPS locations and knowledge about the drive performed, it is possible to classify the driving event. A total of six events are identified for this experiment. They are: High Speed, Low speed, Stopping, take a Ramp UP to highway, Exit highway Ramp Down, and U-turn. Figure 2 shows the data from Gyro Rotation and Motion Pitch sensors and the corresponding event.
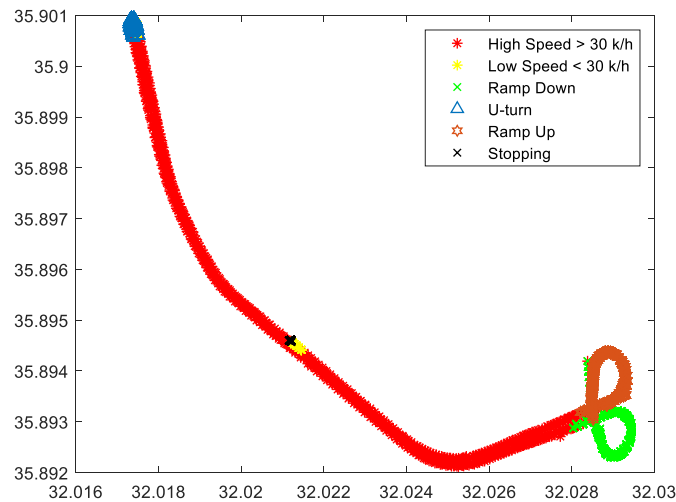

Fig. 1. Travel path. Around trip consists of U-turn, speeding taking a ramp to highway, exiting a highway, and slowing down.
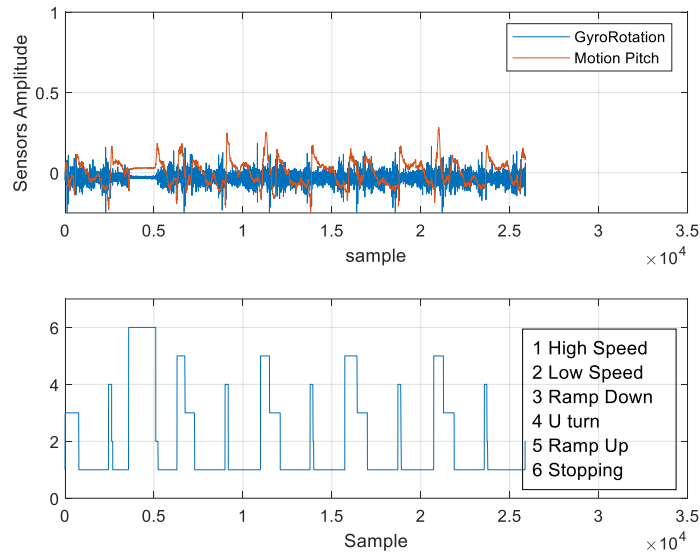
Fig. 2. A sample of sensors data and corresponding driving event

A new column is added to data with the classification of the event. A numeric value between 1 and 6 is used to represent the events according to Table 1 below:

Table 1. Events and their labels

Event	Label
High Speed	1
Low Speed	2
Ramp Down	3
U turn	4
Ramp Up	5
Stopping	6

4   Machine Learning

Machine learning refers to the ability to train machines (computers) to make decisions or predict future data. Training is required to learn how to map input to output. It enables a system to update and modify its structure. Machine learning is utilized to discover patterns in data. The algorithms used can broadly be classified as supervised learning and unsupervised learning.

Supervised learning utilizes training data. Data teaches the algorithm what conclusion it should have. The training data is pre-generated and should be large enough so that the system is able to produce correct results. They should cover all expected in-

puts and outputs. Unsupervised learning does not require training data and is used to discover patterns in data. Thus, providing insight into the data is unknown in advance.

Different classifiers exist in machine learning. These classifiers differ in their performance i.e., speed, memory usage, and interpretability. Different classifiers of supervised learning are studied and their ability to classify driving events are compared. Below, we briefly describe the classifiers used, namely:

- Decision Trees
- Discriminant Analysis
- Naïve Bayes
- Fit k-nearest neighbour
- Ensembles
- SVM

Support Vector Machines (SVM) are supervised learning models with associated learning algorithms that classify data. The algorithm builds a model that assigns known data into categories with the goal of using training data to predict target values. In SVM, input data is mapped nonlinearly into a higher dimension feature space. A separation hyperplane is generated in feature space that solves the classification [15].

A classification decision tree is based on features contained in the data set[2]. Decision Trees make decision by generating a tree-like graph. A binary tree splits branching node based on the values of a column of data. The tree is composed of decision, chance, and end nodes. Classification trees produce a categorical output while regression trees produce number outputs.

Naïve Bayes uses Bayes' theorem and assumes features at independent. Given features vector x, Naïve Bayes calculates the probability that these features belong to each classification i.e., $P(C_k|x_1, x_2, \dots x_n)$. Using Bayes' theorem and the fact that $P(C_k)$ is known

$$P(C_k|x) = \frac{P(x|C_k) * P(C_k)}{P(x)} \tag{1}$$

By chain rule, the joint probability $P(x|C_k)$ is equivalent to its joint probability $P(C_k, x_1, x_2 \dots x_n)$ i.e.

$$P(C_k, x_1, x_2 \dots x_n) = P(x_1|x_2 \dots xn, Ck) * P(x2|x3 \dots xn, Ck) * P(x_{n-1}|x_n, C_k) * P(x_n|C_k) * P(C_k) \tag{2}$$

Assuming that conditional probabilities are independent, i.e., $P(x_i|x_{i+1}, \dots x_n, C_k) = P(x_i|C_k)$,

$$P(C_k, x_1, x_2 \dots x_n) = P(x_1|C_k) * P(x_2|C_k) * P(x_{n-1}|x_n, C_k) * P(x_n|C_k) * P(C_k) \tag{3}$$

Using likelihood function to model the probability distribution, a prediction is made that results in the highest $\sum_k p(C_k) * p(x|C_k)$.

Ensembles combines several models into one predictive model to improve machine learning. Ensembles decreases variance (bagging), bias (boosting) and improve predictions (stacking). Ensemble methods are either sequential or parallel. Sequential methods exploit the dependence between base learners. Performance is improved by assigning higher weight to mislabelled examples. Parallel methods exploit the independence between base learners. Most ensemble methods use a single base learning algorithm to produce base learners.

Discriminate Analysis is based upon the concept of searching for linear combination of predictors that best separates two classes. To assess the effectiveness of discrimination, Mahalanobi distance between groups is calculated. A distance greater than 3 means that two averages differ by more than 3 standard deviations. Thus, the probability of misclassification is small.

The K-Nearest Neighbour is a nonparametric and instance-based learning algorithm. The algorithm makes no explicit assumptions about the functional form. The algorithm does not explicitly learn a model. Instead, it chooses to memorize the training instances which are subsequently used as knowledge for prediction. The classification is performed by a plurality vote between the K most similar observations. Similarity is defined as a distance metric between data points. The algorithm runs through the data set and computes distances to each of the training observations. It then estimates conditional probability for each class i.e., fraction of training points with class label. Finally, the input is assigned to the class with largest probability.

5 Results

The five round trips conducted results in very large sensors data. The data collected contains over 25000 rows and 30 columns. To reduce storage and processing, irrelevant data is removed from the data set. Only motion and gyro sensors data are kept. A table that contains the sensors data and event classification label is created to train the machine learning classifiers.

5.1 All sensors classification

In order to train the classifier, 70% of the data is selected randomly as the training data set. The table below lists the number of samples taken from each event to train the classifier.

Table 2. Events and Training Data

Event	Training Data
1	12537
2	228
3	2157
4	586
5	1549
6	1064

A cross partition of data is created for 5-fold cross validation on the samples. The partition divides N observation into 5 disjoint subsamples chosen randomly but with equal size. The algorithms above offer optimization options that improve their accuracy. Table 3 compares the loss for different algorithms with and without optimization. The loss is calculated as the ratio of misclassified labels to total samples.

Table 3. Algorithms Accuracy with and without optimization

Algorithm	With Optimization	No Optimization
SVM	0.1255	0.1374
Decision Tree	0.0191	0.0193
Discriminate	0.1931	0.1994
Naïve	0.054	0.1394
KNN	0.0218	0.0417
Ensembles	0.0076	0.0374

Figure 3 shows that the accuracy of the classification depends on the algorithm used. The results show that all algorithms perform well at classification of driving events. With no optimization, the Decision Tree performs best. With optimization, Ensembles has the best performance, while Discriminate appears to perform the worst.

Although the above classification performs well, it uses large number of features making it difficult to be deployed on an embedded system.

5.2 Reduced features

Using Component Analysis (NCA), it is possible to select a small subset of features that carry the most relevant information to classifications and reduce redundancy in features. Neighbourhood Component Analysis (NCA) is a non-parametric method for selecting features with the objective of maximizing prediction accuracy of classification algorithms. Figure 4 shows the relative weights of features and the regularized objective function for NCA.
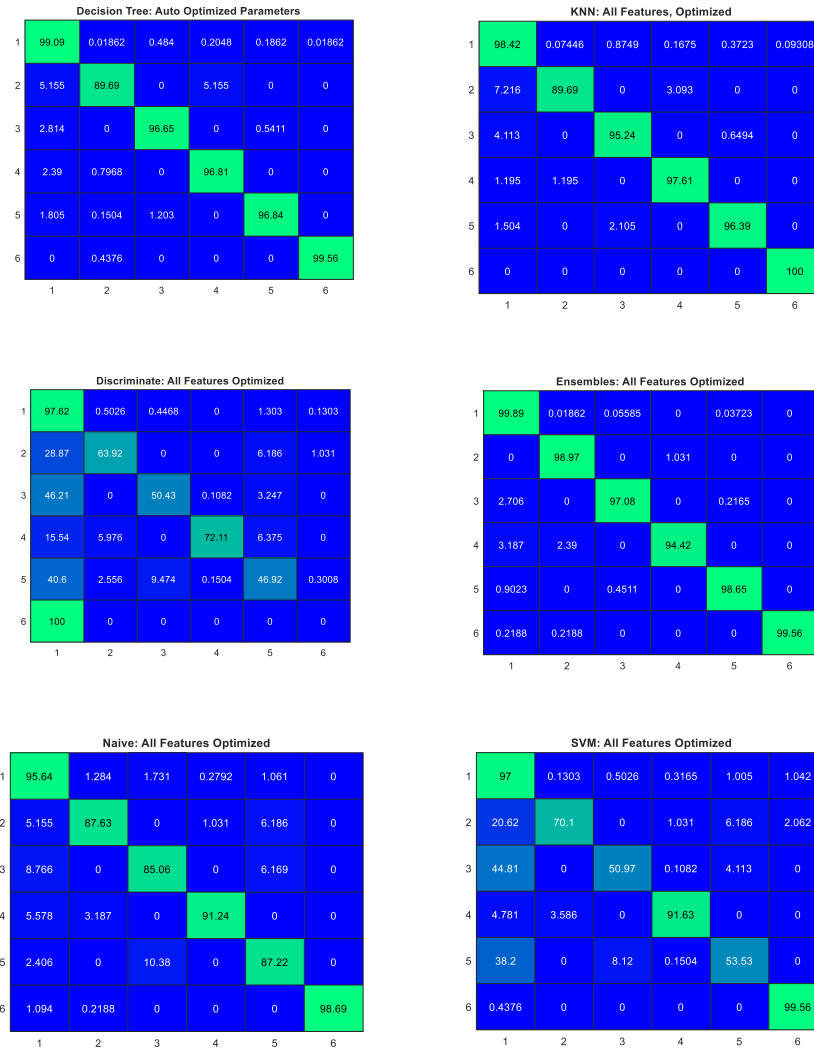
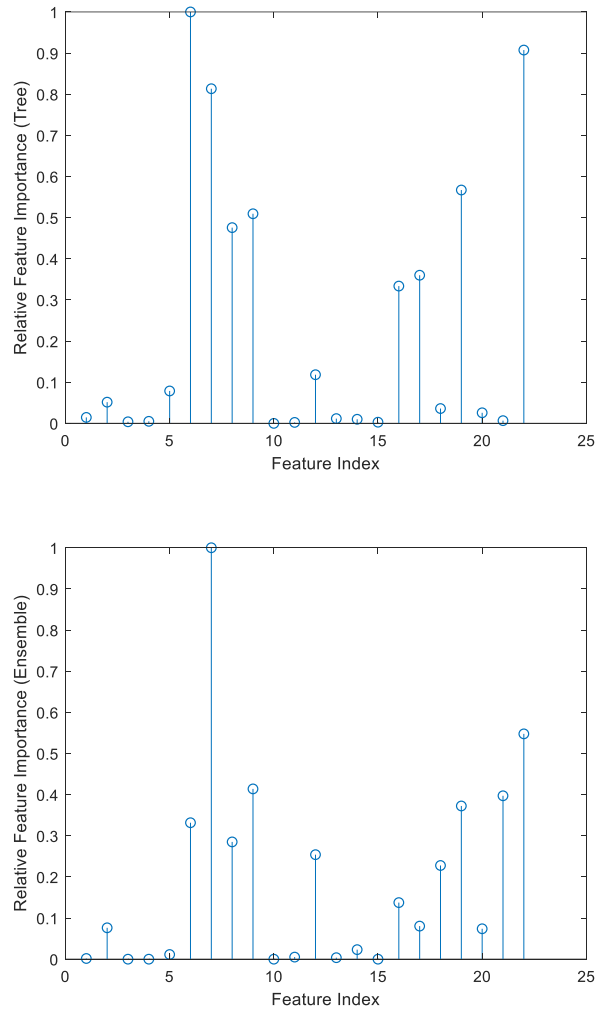**Fig. 3.** Heat map of the classification accuracy of different algorithms.

Fig. 4. Weight of different features using Tree and Ensemble

For Tree classification, the features that have weights greater than threshold value (10%) are 9 features namely: gyroRotationZ, motionYaw, motionRoll, motionPitch, motionRotationRateZ, motionQuaternionX, motionQuaternioY, motionQuaternionW, and motionGravityZ. The Ensemble shares the above features in addition to motionGravityY. Using this reduced set of features to predict the travel event

The corresponding loss for using Tree and Ensemble is 0.0155 and 0.0488, respectively as shown in Figure 5. The results show even with reduced feature, an accurate classification of the event can be achieved.
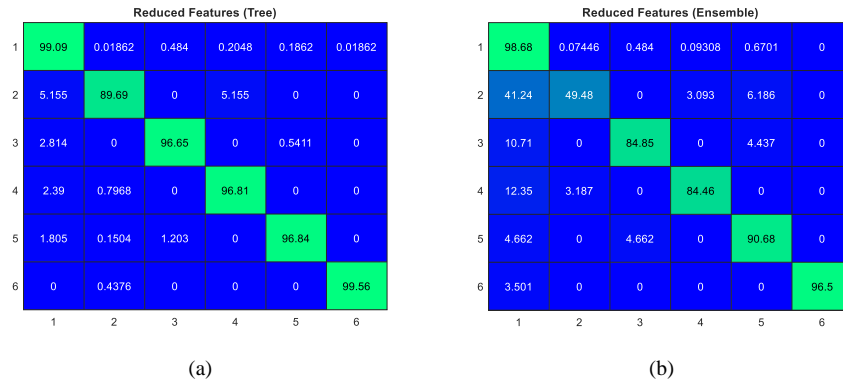
Fig. 5. Heat map with reduced features for (a)Tree and (b) Ensemble

6 Conclusion

With the growing road traffic around the world, the introduction of autonomous cars, and advances in automobile technology, new research is needed to take advantage of sensory data to improve efficiency and improve safety. This work introduces a technique that uses Machine Learning to classify driving events. It utilizes sensors data available from a mobile device with motion sensors to classify different events. Understanding driving events is essential for estimating cost of travel for vehicles and improving their efficiency by planning new routes that might reduce certain events. Our results show that by using machine learning with reduced features, it is possible to accurately classify events with very high accuracy. Using Decision Trees, the accuracy of classification is over 98%. Future work will consider using sensors data to classify driver behaviour in terms of efficiency and safety.

7 References

[1] A. M. de Souza, C. A. Brennand, R. S. Yokoyama, E. A. Donato, E. R. Madeira, and L. A. Villas, "Traffic management systems: A classification, review, challenges, and future perspectives," Int. J. Distrib. Sens. Networks, vol. 13, no. 4, p. 155014771668361, Apr. 2017, https://doi.org/10.1177/1550147716683612.

[2] T. Van Nguyen, V. D. Nguyen, T. T. Nguyen, P. C. P. Khanh, T. A. Nguyen, and D. T. Tran, "Motorsafe: An android application for motorcyclists using decision tree algorithm," Int. J. Interact. Mob. Technol., vol. 14, no. 2, pp. 119–129, 2020, https://doi.org/10.3991/ijim.v14i02.10742.

[3] "Intelligent Transportation Systems - Joint Program Office Home Page." [Online]. Available: https://www.its.dot.gov/ . [Accessed: 06-Jan-2020].

[4] Y. A. Alqudah and B. H. Sababha, "A Statistical Approach to Estimating Driving Events by a Smartphone," in Computational Science and Computational Intelligence (CSCI), 2016 International Conference on, 2016, pp. 1021–1025. https://doi.org/10.1109/csci.2016.0195

[5] F. Feng, S. Bao, J. R. Sayer, C. Flannagan, M. Manser, and R. Wunderlich, "Can vehicle longitudinal jerk be used to identify aggressive drivers? An examination using naturalistic driving data," Accid. Anal. Prev., vol. 104, no. February 2016, pp. 125–136, 2017, https://doi.org/10.1016/j.aap.2017.04.012.

[6] L. Qin, Z. (Richard) Li, Z. Chen, M. S. Andi Bill, and D. A. Noyce, "Understanding driver distractions in fatal crashes: An exploratory empirical analysis," J. Safety Res., vol. 69, pp. 23–31, 2019, https://doi.org/10.1016/j.jsr.2019.01.004.

[7] S. F. Liang, C. T. Lin, R. C. Wu, Y. C. Chen, T. Y. Huang, and T. P. Jung, "Monitoring driver's alertness based on the driving performance estimation and the EEG power spectrum analysis," Annu. Int. Conf. IEEE Eng. Med. Biol. - Proc., vol. 7 VOLS, pp. 5738–5741, 2005, https://doi.org/10.1109/iembs.2005.1615791.

[8] F. Yan, M. Liu, C. Ding, Y. Wang, and L. Yan, "Driving style recognition based on electroencephalography data from a simulated driving experiment," Front. Psychol., vol. 10, no. MAY, pp. 1–13, 2019, https://doi.org/10.3389/fpsyg.2019.01254.

[9] I. H. Kim, J. H. Bong, J. Park, and S. Park, "Prediction of driver's intention of lane change by augmenting sensor information using machine learning techniques," Sensors (Switzerland), vol. 17, no. 6, pp. 1–18, 2017, https://doi.org/10.3390/s17061350.

[10] B. Jachimczyk, D. Dziak, J. Czapla, P. Damps, and W. J. Kulesza, "IoT on-board system for driving style assessment," Sensors (Switzerland), vol. 18, no. 4, pp. 1–22, 2018, https://doi.org/10.3390/s18041233.

[11] D. H. Ahn, H. Park, K. Shin, and T. Park, "Accurate Driver Detection Exploiting Invariant Characteristics of Smartphone Sensors," Sensors (Basel)., vol. 19, no. 11, pp. 1–11, 2019, https://doi.org/10.3390/s19112643.

[12] W. Kim, W.-S. Jung, and H. K. Choi, "Lightweight Driver Monitoring System Based on Multi-Task Mobilenets," Sensors, vol. 19, no. 14, p. 3200, 2019, https://doi.org/10.3390/s19143200.

[13] K. T. Nguyen, T. Van Pham, V. D. Nguyen, L. T. Do, A. Van Tran, and D. T. Tran, "Development of a smartphone application for safe car driving using Ggoogle API and built-in sensor," Int. J. Interact. Mob. Technol., vol. 14, no. 2, pp. 178–195, 2020, https://doi.org/10.3991/ijim.v14i02.11118.

[14] C. Huda, H. Tolle, and F. Utaminingrum, "Mobile-based driver sleepiness detection using facial landmarks and analysis of EAR Values," Int. J. Interact. Mob. Technol., vol. 14, no. 14, pp. 16–30, 2020, https://doi.org/10.3991/ijim.v14i14.14105.

[15] M. Paluszek and S. Thomas, MATLAB Machine Learning. 2017.

8 Authors

Yazan A Alqudah received his PhD degree from The Pennsylvania State University, USA in 2003 in electrical engineering. He joined Intel corporation, Oregon in 2003 as a Senior Researcher where he worked with Logic Technology Development (LTD) and Mobile Wireless Groups (MWG). In Fall 2018, Dr. Alqudah joined the Department of Electrical and Computer Engineering at the University of West Florida as an Associate Professor.Dr. Alqudah is a Senior Member of IEEE. In 2014, he won best The Best Researcher Award at PSUT. His current research interests include Broadband Optical Wireless Communication, Next Generation Mobile Networks Deployment and Performance, and Mobile Development.

Belal Sababha, is the Dean of the King Abdullah II School of Engineering, Princess Sumaya University for Technology, Amman, Jordan. He is also an Associate Professor of Electrical and Computer Engineering with the Computer Engineering Department. He is a Senior Member of IEEE and a member of several national and international professional organizations.

Esam AlQaralleh received the B.Sc. degree from the University of Jordan, Amman, in 1995. He went on to obtain his M.Sc. and PhD degree from National Chiao Tung University, Taiwan, in 2006, all in Electrical Engineering. He served as the Head of the Department of Computer Engineering from 2009 to 2011. He is a member of the IEEE computer, and RAs societies. He has participated in organizing and conducting many local and international conferences and events. His current research interests include video coding/motion estimation, algorithm and implementation using FPGA, Embedded Systems/RTOS, Engineering Education and Arabic Handwritten Recognition.

Tarek Youssef is an Assistant Professor of Electrical and Computer Engineering Department, has worked as a researcher in several projects funded by the department of energy (DoE) and the office of naval research (ONR). He authored and co-authored more than 15 journal and 25 conferences articles, two books chapters and three patents in energy storage, energy management, and smart grid security. Youssef joined UWF in 2018 after working as a postdoctoral at Florida International University for one year. His research during his Ph.D. and the postdoctoral period were funded by DOE's SEEDS cybersecurity centre for securing smart grid. Prior to that, he worked for 12 years in communication and security field in Egypt. Youssef holds an M.S. degree in power electronics and B.S. degree in Electrical Engineering from Helwan University, Egypt and a Ph.D. degree in electrical engineering from Florida International University, USA. His research interests include cyber-physical security, energy management and intelligent control of the smart grid.