Utilizing CAN-Bus and Smartphones to Enforce Safe and Responsible Driving

Abd-Elhamid M. Taha and Nidal Nasser
College of Engineering, Alfaisal University
PO Box 50927, Riyadh 11533, Saudi Arabia
{*ataha, nnasser*}*@alfaisal.edu*

***Abstract*—Road fatalities in Saudi Arabia are expected to reach an alarming rate of one death per hour in 2015. In this paper, we describe a framework for vehicular sensing that is directly aimed at instilling safe driving. The presented architecture utilizes access to vehicle's CAN-Bus through an OBD-II connector. The access is processed using both on-vehicle smartphone and in-the-cloud processing. Road conditions (e.g., potholes, speedbumps, slowdowns, etc.) are recognized and utilized to classify roads using a threshold-based engine. The proposed framework is designed with strong emphasis on extendability, and a service application providing drivers with Quailty of Road (QoR) status is demonstrated.**

***Keywords*-CAN-Bus; OBD-II Connectors; Crowdsourcing; Cloud-based services; Vehicular sensing.**

I. Introduction

A recent study by Hassan and Al-Faleh [1] indicate that fatalities in 2012 reached 7638. Considering the rate at which fatalities has increased since 2003, projections for 2015 surpass a rate of one death per hour. Such rate indicate the need for road safety solutions that overcome the limitations of traditionally employed mechanisms, such as police patrols and speed traps, [2].

An analysis of 11545 crash reports made during March 2004 to February 2011 highlights the leading causes of crashes were sudden lane change (44.4%), distraction (21.4%) and overspeeding (10.8%), with the main collision points being either head-on (21%) or at angle (23.1%). This finding, in addition to others in the report, illustrate a critical dependence on driving behaviour and awareness in most crashes, and suggest a great need for innovative mechanisms to instil safe driving not only at the licensing stage, but also post-licensing. The intent of this work is address this particular need.

In this paper, we present a preliminary overview of a system that exploits recent advances in sensing both in vehicles and smartphones. Specifically, we utilize the access to the vehicular Controller Area Network (CAN) facilitated through the second generation On Board Diagnostics (OBD), i.e., OBD-II. We also utilize advanced sensing and computing capabilities commonly available today's smartphone handsets. Through an application of crowd-sensing, a classification of roads traversed by drivers is generated and made available to various user services. This work demonstrates a Quality of Road (QoR) alert systems that raises the user's awareness of the states of the road currently being travelled.

The work presented herein goes beyond recognizing road features to facilitating a service platform that are aimed specifically at enhancing safe driving. The framework design approach also allows for extendability, as well enabling safety-based alerts and road navigation.

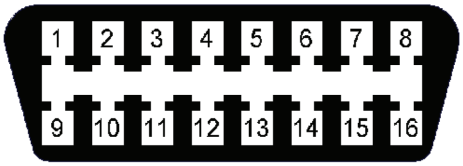
The remainder of this paper is organized as follows. Section II motivates the present work through a review of related work, and offers the necessary background for the proposed solution. Section III presents our architecture, named goodDrive, and details its main design components – including the architecture's road classifier. Section III-E showcases an implementation instance and output of the architecture. Finally, Section IV concludes.

II. Background and Related Work

Various smartphone applications (or apps) are already available in the main app outlets (e.g., Google play, iTunes store, etc.), including front-end apps for Google, Apple and Microsoft (Bing) map services. Other popular apps include Waze [3]. Most of these mapping apps provides rich routing and navigation based on various information sources, including mobile operator data as well as implicit crowdsourcing from user smartphones. Waze also utilizes explicit crowdsourcing, allowing users to report accidents and speed traps, in addition to identifying landmarks and relaying nearby gas prices.

In [4], Raj et al. consider pothole detection using a large dataset collected through *implicit crowdsourcing*, i.e., crowdsourcing without repeated user prompt/input. Thresholds are used on the phone's z-axis acceleration, with empirical data used for differentiating potholes from speed breakers. Bo et al. implement a naïve Bayesian classifier in [5] to identify whether the smartphone user *a)* has entered a vehicle; *b)* has boarded the vehicle from the left or the right; *c)* has sat in the front or back seat; and *d)* is texting. The designed system prevents vehicle passengers from texting while driving, and requires training in order to validate activity classification. Another system that differentiates drivers from passengers can be found in [6] by Chu. Castignani et al. employ fuzzy logic to profile driver behavior in [7]. The system utilizes the acceleration, gravity, magnetic and GPS sensors to estimate driving aspects such as jerk, orientation rate, speed variation and bearing variation. A fusion module then aggregates the input to feed a trained fuzzy inference system. In turn,

978-1-4673-7194-0/15/$31.00 ©2015 IEEE



PIN	Description	PIN	Description
1	Vendor Option	9	Vendor Option
2	J1850 Bus +	10	j1850 Bus −
3	Vendor Option	11	Vendor Option
4	Chassis Ground	12	Vendor Option
5	Signal Ground	13	Vendor Option
6	CAN (J-2234) High	14	CAN (J-2234) Low
7	ISO 9141-2 K-Line	15	ISO 9141-2 Low
8	Vendor Option	16	Battery Power

Figure 1. The OBD-II connector and pinout.

the inference system distinguishes phenomena such as hard acceleration or over-speeding.

Despite substantial advances in the sensing capabilities of smartphones, certain characteristics limit the dependence on their use, especially for critical applications such as the one considered herein. Paefgen et al. explored this dependence aspect in detail in [8]. The work observes the interest of the insurance business to have "black-box"-like installations in vehicles through smartphones, and studies characteristics affecting reliability of smartphone date such as phone's orientation relative to that of the vehicle. External conditions such as weather, time of the day, and locale (e.g, urban vs. otherwise) are also considered. The authors find that despite correlation of smartphone measurements with fixed inertial measurements, smartphone measurements exhibited dependence on its position, orientation, event types and certain external factors.

A. OBD-II Based Systems

OBD-II was introduced in 1995 and standardized in 1996 in North America. Its main application at the time was to facilitate emission tests required for vehicle registration. A view of the standardized pin layout is provided in Figure 1. As of 2008 all vehicles sold in North America are required to use the CAN, which now facilitates data sampling speeds varying between 250kB/s and 500kB/s. This increase allowed car manufacturers redesign the vehicles sensors and control units in a compartmentalized fashion as the CAN allows them to communicate within the vehicles at such a sufficient speed to allowing them to work seamlessly yet remain isolated.

In [9], Araújo et al. consider a urban-sensing data collection setup through a Bluetooth-connected OBD-II and a data aggregation application called Torque-Pro [10]. The objective of the data collection is to identify fuel-efficient driving patterns through an application of neural networks on a 6-tuple data set that includes speed, acceleration, altitude, throttle, instantaneous fuel consumptions and engine rotations. Once the patterns are recognized, hints are provided in real time to drivers so that they would improve their driving patterns. An effort with similar objective was made by Campolo et al. in [11] with an Arduino/XBee interconnect used for connecting the OBD-II connector with the Android-based smartphone. Similarly, an expert system is implemented by Jhou and Chen in [12] that processes collected information to advise on the need for care (e.g., cooler fill up) or maintenance.

Meseguer et al. base their work in [13] on the need to establish the vehicle's driving context in order to deem a driver's behaviour aggressive. Similar to the works above, an OBD-II Bluetooth connector is utilized to collect speed, acceleration, engine revolution count and location. Data is aggregated in a data centre with a neural network applied to identify aggressive driving patterns.

B. Motivation

The main objective of this work is to offer a framework for designing vehicular sensing apps that monitor, represent and utilize road condition information based on a flexible in-vehicle setup. Similar to the work in [13], we believe that understanding the vehicle's context is necessary if safety on the road is to be enhanced. The scope of the current work is to alert the user to the state of the road currently being traversed in a timely manner, raising the driver's awareness of possibilities of road artifacts (potholes or speed bumps), weather effects (wetness, sand/dust), or aggregate behaviour of other drivers (e.g., sudden turns or stops).

III. The goodDrive System

A. System and Operation Overview

The proposed system, called goodDrive, employs OBD-II connectors with Bluetooth or WiFi connectivity, together a smartphone. Once the connector is paired to the smartphone, the two device transforming most on-road vehicles with CAN-Bus into a sensing vehicle that is capable of providing the required information to the goodDrive computation core in the cloud. In turn, the goodDrive aggregates data from vehicles as well as from other sources, e.g., weather, time-of-day, relevant news outlets, etc., and feeds the aggregated information to a road classifier module. Classifications of roads are maintained in an active database based upon which several services can be offered to drivers. While our current implementations is limited to a Quality of Road (QoR) alert system, other services are readily supported. The QoR system alerts the user the quality/state of the road traverse

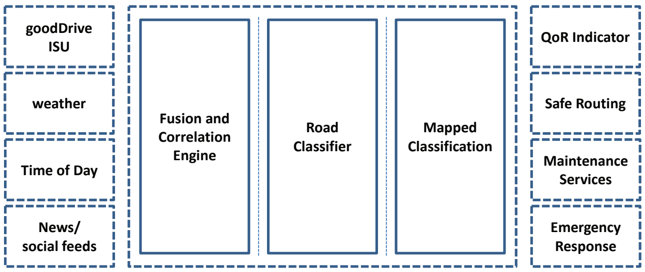


Figure 2. Overview of the goodDrive system.

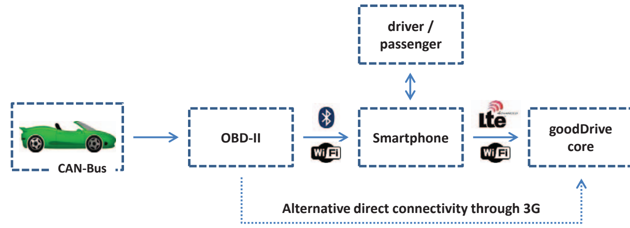


Figure 3. Illustration of the In-Vehicle Sensing Unit (ISU) with two possibilities for connecting to the goodDrive core.

and, where applicable, offers advice to enhance driving safety.

An illustration of the overall intended system architecture is shown in Figure 2. In what follows, we elaborate on the design of certain components of the architecture.

B. *The In-Vehicle Sensing Unit*

By In-Vehicle Sensing Unit (ISU) we refer to the potential mix of OBD-II connectors, dedicated sensors and smartphone set up on the vehicle. This view offers flexibility in ISU implementation. For example, certain OBD-II modules can be equipped with their SIM-cards[1]. Such a setup would forego the need for a smartphone in the sensing module. It is also possible to utilize standalone sensors (e.g., accelerometers, gyroscopes, etc.) that report to the goodDrive core through a dedicated communication module.

In our view, a typical implementation that would also reduce the cost and technical knowledge requirements on part of the users is based on pairing a Bluetooth or WiFi enabled OBD-II connector and a smartphone. Further advantages of this setup is that it would allow for *a)* smartphone preprocessing before transmission; *b)* logging of sensed-events in case of coverage outage; and *c)* involving users (drivers/passengers) in the validating sensed data. For safety consideration, user interaction can be limited to oral communication if the user is recognized to be a driver. In the meanwhile, passengers may chose either oral or direct input to the smartphone.

It is possible in certain instances that the smartphone preprocesses information to reduce over-the-air communication requirements. Communication between smartphone and the cloud can be performed either in cellular data or WiFi, depending on the availability of the access network type. If the smartphone is momentarily out of coverage, it maintains a log of events and reports to be transmitted as soon as the connectivity is recovered.

An illustration of the ISU is shown in Figure 3

[1] An example of a basic 3G-enabled OBD-II connector can be found at http://goo.gl/KhlCjw.

C. *The goodDrive Core*

The design of the goodDrive core involves the following main components:

- An interface for receiving sensed data by the ISU;
- A fusion and correlation engine;
- A road classifier;
- An active database for road quality; and
- A service interface.

We briefly elaborate on the role and operation of these components below. However, details of the road classifier are discussed in the next subsection.

In addition to data communicated by the ISU, the interface for receiving data is further designed to aggregate/crawl for information that would enrich the context of the ISU data. For example, there is high value in correlating sensed data with both the weather and time-of-day so as to enhance predictions for road classifications. Meanwhile, crawling engines can be aimed at local news outlets that are relevant to road conditions and quality, as well as public feeds from social networks that relate to road conditions.

The role of the fusion and correlation engine is twofold. First, it is possible that multiple feeds are made for the same data, e.g., vehicle location or speed. For example, in the absence or delay of access to the vehicle's CAN-Bus data, the goodDrive core can receive acceleration data from an on-board dedicated accelerometer and the smartphone's accelerometer. Properly combined (fused), the resulting understanding of the vehicle's acceleration can be made with reduced error. The correlation component of the engine is aimed at improving the quality of the gathered data, and attempts to synchronize and/or co-localize events indicated by the multiple information sources feeding the goodDrive core. For example, the bound on the sampling rate from the OBD-II connector limits the type of information sensed in a given epoch, as well as the rate at which it can be provided. A correlator is hence needed when preprocessing the sensed data from different ISUs.

An database is dedicated to maintaining and updating road classifications, which is to be discussed in the next subsection. The database storage and computation frequency are dependent on the intended granularity of the system. For example, the requirements for localizing a road feature

	potholes	pothole clusters	road stripped	accidents / congestion
Red 2	> 10 (large) + Illegal speedbumps	> 4	✔	✔
Red 1	> 10 (large)	> 4	✔	✔
Yellow 3	7 - 10 (large)	> 4	✔	×
Yellow 2	> 3 (medium) 3 – 6 (large)	> 4	×	×
Yellow 1	2-3 (medium) < 3 (large)	1-3	×	×
Green	> 5 (small)	0	×	×

Figure 4. An example road classification associated with color code.

(e.g., pothole, speed bump) to a directed lane varies from that made for undirected road. Meanwhile, updates for a certain road are ultimately dependent on the frequency that road is travelled, but also on the expected redundancy in the received data.

Finally, a service interface is implemented to facilitate access to the processed road information. In this work, we discuss the implementation of an alert application that indicates to the driver the quality of the road entered. Other services can be envisaged.

D. The Road Classifier

The road classifier yields an understanding of the Quality of Road (QoR) based on the input generated by the fusion and correlation engine. A framework-based approach is taken in designing a classifier that is open for both modification and expansion. In our implementation, we chose a grade-based system for the road classification in which each road grade is designated a color code. This coding facilitates ease of visualization both when viewing the road map for a city or for the QoR alert system to be discussed later in this work.

To elaborate, the classifier first processes confirmed data from the fusion and correlation engine in order to distinguish road artifacts such as potholes (small, medium, large), pothole clusters, speed-bumps, slow-downs, sharp-turns, etc. The classifier also utilizes correlations made with time-of-day, weather, and speed, among others. A threshold-based computation then aggregates these confirmed events into a multi-dimensional vector through which a profile-like understanding of the road is created.

Adding or removing a certain feature for a road or road segment must validate the presence or absence of a road feature across different driving points. This validation requires more sophisticated inference mechanisms and may entail more active sensing that employs a feedback path from the goodDrive core to the ISU.

An example color code is illustrated in Figure 4. The details of the threshold-based classification are as follows.

- **Green** Assigned to roads with no pothole or pothole clusters or less than ten small potholes. Roads, however, can experience temporary congestions.
- **Yellow 1** Assigned to roads with either one to three pothole clusters, two to three medium potholes or ten or more small potholes, with no large potholes present.
- **Yellow 2** Assigned to road with four or more pothole clusters; four or more medium potholes; and/or less than three large potholes.
- **Yellow 3** Assigned to stripped roads, i.e., roads with gravel and/or dirt present, with the number of large potholes less than 10.
- **Red 1** Assigned to roads with indications of accidents (through off-mark deceleration-to-stops or detours); congestion present; and detour availability to clear (or less-congested) routes.
- **Red 2** Assigned to roads to stripped roads (similar to Yellow 3), but with 10 or more large potholes presented and indicators of illegal/unmarked speed bumps.

The above classification may require quantization of road into segments.

E. Implementation Instance

An implementation instance of our system was made using a typical OBD-II connector (Munic.box with Bluetooth), an Android-based smartphone (Samsung 5S), and an SQL-based database in the cloud. Screenshots of implemented system are shown in Figure 5. An example map with roads classified is also provided in Figure 6.

IV. Conclusion

In this work we presented a vehicular sensing framework aimed at reducing accidents in Saudi Arabia. The framework is centered around recognizing the level of a safety for a traversed road, and utilizes sensing advances in smartphones as well as enhanced access to CAN-Bus on vehicles. A review of the extendible, cloud-based framework architecture was presented, and a brief discussion of an implementation instance was made. Currently undergoing work involves further validation and testing.

Acknowledgment

The authors would like to acknowledge the financial support provided by Alfaisal University through its Internal Research Grant NS3-201200101148.

References

[1] H. M. Hassan and H. Al-Faleh, “Exploring the risk factors associated with the size and severity of roadway crashes in riyadh,” *Journal of Safety Research*, vol. 47, pp. 67 – 74, 2013.

[2] “SAHER Traffic e-system.” available from http://goo.gl/fbynK; accessed March 30, 2015.

[3] “WAZE Navigation app.” available from https://goo.gl/3XzNqM; accessed March 30, 2015.

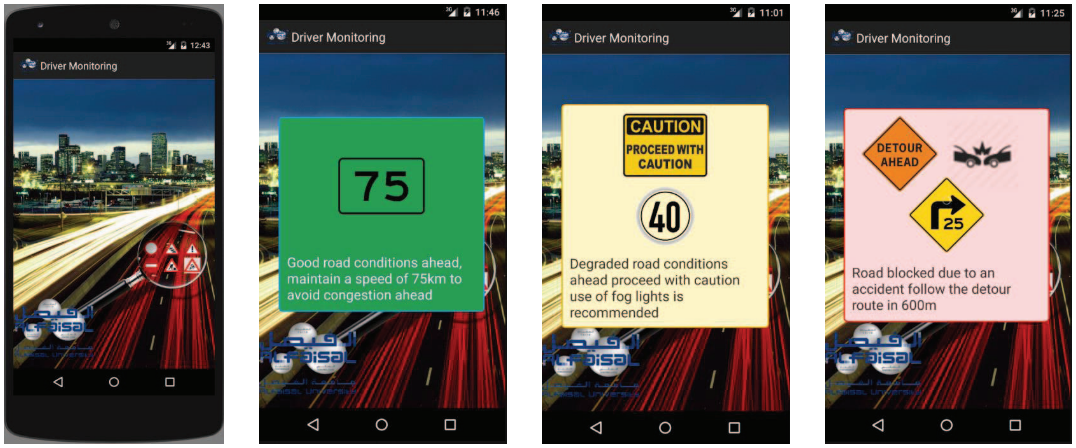


Figure 5. Example screenshots.

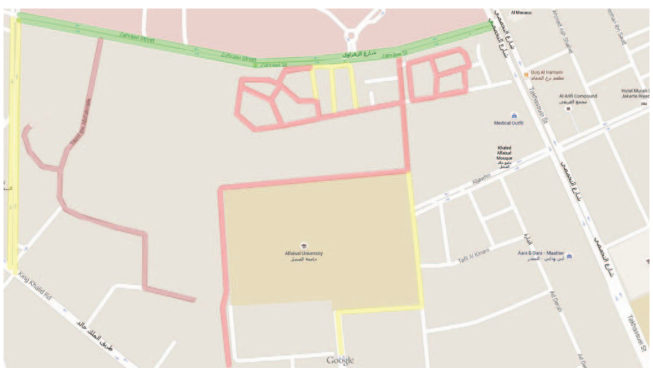


Figure 6. An example map with roads classified.

[4] S. Raj, A. Jain, and R. Misra, "Smartphone sensing for large data set collection of potholes," in *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, MobiSys '13, (New York, NY, USA), pp. 517–518, ACM, 2013.

[5] C. Bo, X. Jian, X.-Y. Li, X. Mao, Y. Wang, and F. Li, "You're driving and texting: Detecting drivers using personal smart phones by leveraging inertial sensors," in *Proceedings of the 19th Annual International Conference on Mobile Computing & Networking*, MobiCom '13, (New York, NY, USA), pp. 199–202, ACM, 2013.

[6] H. L. Chu, V. Raman, J. Shen, R. R. Choudhury, A. Kansal, and V. Bahl, "Poster: you driving? talk to you later," in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys 2011), Bethesda, MD, USA, June 28 - July 01, 2011* (A. K. Agrawala, M. D. Corner, and D. Wetherall, eds.), pp. 397–398, ACM, 2011.

[7] G. Castignani, T. Derrmann, R. Frank, and T. Engel, "Driver behavior profiling using smartphones: A low-cost platform for driver monitoring," *IEEE Intelligent Transportation Systems Magazine*, vol. 7, pp. 91–102, Spring 2015.

[8] J. Paefgen, F. Kehr, Y. Zhai, and F. Michahelles, "Driving behavior analysis with smartphones: Insights from a controlled field study," in *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*, MUM '12, (New York, NY, USA), pp. 36:1–36:8, ACM, 2012.

[9] R. Araujo, A. Igreja, R. de Castro, and R. Araujo, "Driving coach: A smartphone application to evaluate driving efficient patterns," in *IEEE Intelligent Vehicles Symposium (IV)*, pp. 1005–1010, June 2012.

[10] "Torque Pro Mobile app." available from http://goo.gl/CWD5VT; accessed March 30, 2015.

[11] C. Campolo, A. Iera, A. Molinaro, S. Paratore, and G. Ruggeri, "Smartcar: An integrated smartphone-based platform to support traffic management applications," in *First International Workshop on Vehicular Traffic Management for Smart Cities (VTM)*, pp. 1–6, Nov 2012.

[12] J.-S. Jhou and S.-H. Chen, "The implementation of obd-ii vehicle diagnosis system integrated with cloud computation technology," in *Intelligent Data analysis and its Applications, Volume I* (J.-S. Pan, V. Snasel, E. S. Corchado, A. Abraham, and S.-L. Wang, eds.), vol. 297 of *Advances in Intelligent Systems and Computing*, pp. 413–420, Springer International Publishing, 2014.

[13] J. Meseguer, C. Calafate, J. Cano, and P. Manzoni, "Drivingstyles: A smartphone application to assess driver behavior," in *IEEE Symposium on Computers and Communications (ISCC)*, pp. 000535–000540, July 2013.