ROAD AND DRIVER MONITORING SYSTEM

By

Sam Aleyadeh

A thesis submitted to the School of Computing

in conformity with the requirements for

the degree of Masters of Science

Queen's University

Kingston, Ontario, Canada

September, 2014

Abstract

In this work we present a driver and road monitoring system capable of scaling to the number of drivers in any region without incurring the high costs of implementation, thus allowing for safer driving conditions and shorter accident response times. We propose the Smartphone Road Monitoring (SRoM) system that is capable of sensing road artifacts such as potholes and slippery roads. It is also capable of detecting aberrant driver behavior such as speeding and drifting. This system uses both the driver's smartphone and vehicle as the main source of information. The information is collected through crowdsourcing and processed by a base station giving faster and more accurate responses compared to current systems, to address road safety related events in a timely manner. SRoM aims to accommodate various levels of reporting based on the nature of hardware available to each driver. It also allows pedestrians to share media pertaining to each event. The collected data is made available to the public through an interactive map updated with the authenticated events. Our system also incorporates the use of the vehicle's On-Board Diagnostic (OBD) device that helps with sensing road safety and vehicle health status. The OBD system acts as a streaming black box which in case of an accident gives Emergency Medical Services (EMS) personnel a better idea of the victims' condition and the location of the accident and allows police a more accurate rendition of the events for insurance and litigation purposes. We implemented a prototype of the system including OBD adaptation to perform the task of safety monitoring. System evaluation of the prototype shows that the system can be easily implemented in real-life using current technologies at little cost.

Co-Authorship

The original concept of SRoM is a collaborative effort with Sherin Abdelhamid, Mervat AbuElkheir, Hossam Hassanein and Sharief Oteafy. The incorporation of the OBD in the system is first proposed and implemented in this work though.

Acknowledgements

I would like to thank my research supervisor Dr. Hossam, his guidance was crucial in every step these past two years leading the completion of my thesis, more importantly I would like to show my gratitude for the mentoring he offered in the various aspects of my life, and helping me navigate through the events over the period of my master's studies.

I would like to thank my mom Dr. Najah Abu Ali, without her I would not be the man I am today, when everyone told me you have reached the limit of your abilities, she was the one that said no "I know you the best and you can do it", she was a constant beacon of support and encouragement, I only hope I can ever repay this debt to her.

I would like to thank my friend and brother Dr. Hamid Taha, he beyond a doubt understood me best among everyone and somehow knew exactly when to step-in with crucial help and advice that helped me reach my goals.

I would also like to show gratitude to my examining committee, Dr. Pat Martin, Dr. Nidal Nasser, and Dr. Aboelmagd Noureldin you have made the experience much easier than I anticipated and taught me the art of humor in high tense situations.

I would like to thank my family. My grandmother, who offered her encouragement through her constant phone calls – even when I got busy and didn't call she would remember me. With her own constant worrying over everything about my life, Jamelah Bader has been kind and supportive to me over my whole life. To my brother Wesam Aleyadeh it would be an understatement to say that, as a family, we have experienced some ups and downs in the past two years. However every time I needed help you were there to give it, and I am forever grateful.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

The problem of vehicle overcrowding on roadways is becoming more common around the world. This is mainly due to the steadily increasing population around the globe in developed and third world countries alike. Expansions were made to the network of roads to accommodate the increasing number of vehicles. However, these changes resulted in an increase in the rate of deterioration of road health, out-pacing current road health monitoring techniques and increasing the number of accidents.

The significant increase in the number of drivers can be put to use to help address these problems, through continuous monitoring and reporting of events by the drivers and their vehicles. However, to enlist the help of the public and their vehicles the system must be inexpensive to attract participants. To achieve such a goal we have to repurpose existing hardware available to all drivers that is capable of sensing, processing, and reporting events in real-time.

Current smartphones boast powerful processing capabilities supplemented by a number of sensors available to all smartphones with different degrees of quality and sensitivity. These phones are capable of acquiring information such as current location, direction, speed, and tilt and report this information through multiple forms of communication such as cellular, Wi-Fi, and Bluetooth.

In this thesis, we present a client server system that utilizes smartphone technology to monitor both road conditions and driver behavior collected from individual drivers, and to report any anomalies in real-time; thus providing better road conditions and a safer driving experience.

Maintaining traffic and road safety standards is becoming a burden on the current infrastructure due to the increasing number of vehicles on the road. This is manifested in the form of increasing incidents of traffic congestion, collisions, and hazardous road conditions. These problems pose a significant cost for both governments and drivers. The presence of more vehicles on the road in a poorly managed environment also results in more fatal accidents. In Canada alone, 166,725 people were injured during 2011 as a result of traffic collisions, 2,006 of those injured died within one month of the accident [1].

These numbers can be significantly reduced through better monitoring of road conditions and driving behavior. The information collected can help agencies concerned with road safety address issues such as prolonged traffic congestion, slippery roads, potholes, and collisions. This information can also be shared directly and instantaneously with other drivers to avoid problems, such as bad road conditions and collision locations. This allows for safer traffic and better emergency response when needed.

Information dissemination systems for traffic safety exist in a number of countries. The main purpose of such systems is to detect events such as collisions, congestion, and poor road conditions. In general such systems are referred to as Intelligent Transportation Systems (ITS), and are discussed in Chapter 2. Although they offer a solution to the problem of the increasing volume of vehicles on the road in a timely manner, they require an extensive network of infrastructure that is both costly to install and maintain.

Efficient capturing of information in ITS also requires a form of standardization to be implemented by car manufacturers. However, this process faces many problems due to the long life cycle of current light vehicles which on average is 9 years; moreover, at least 35% of new vehicles are expected to pass the 10 year mark [2], thus making the implementation of such systems an

impossibility for rapid deployment. The above reasons make such systems unfit for deployment in many regions of the world that lack the financial support or the time to implement such systems.

In order to address these shortcomings, we propose a system that is inexpensive, requires minimal hardware, and can be rapidly deployed using readily available technology. Our system does not seek to replace ITS technologies already in place, but offers solutions for the current environment until such systems are fully implementable.

The objective of this thesis is to present a system capable of detecting real-life events, on road conditions, and driver behavior, and report such events to a dedicated server. The system relies on smartphones and OBD to act as both the sensing platform and the reporting mechanism, thereby alleviating the need for an infrastructure-based approach. Data gathered from individual vehicles are aggregated to authenticate events based on the frequency of detection, and alert governmental agencies about the event, along with a measured level of urgency. The public is also allowed access to the collected information through an interactive map of recent unaddressed events to help avoid them. Smartphones are utilized in various ways to attain our goals. The system is designed to function without interrupting the driver's focus, and is seamlessly able to detect dangerous road conditions and driving behavior. In case the passenger would wish to contribute actively in the monitoring process, the system also allows for a detailed mode of operation that entails interaction and media collection (e.g., image of road condition).

We propose a real-time traffic, road conditions and driver behavior monitoring and archiving system designed and implemented on smartphones with and without OBD support. Following are the contributions of this work:

1. Integrating OBD and smartphone sensing and computational capabilities to enhance the performance of the monitoring and detecting system.
2. Designing a system that is interoperable and calibrated over the diversity of smartphone brands; i.e. sensors sensitivity.
3. Data fusion of OBD and smartphone sensors.
4. Automated calibration of smartphone built-in sensors customized to the car type and quality of suspension.
5. Designing and implementing a dynamic location prediction scheme to predict the car's current location relative to intermittent GPS readings; to prolong smartphone battery life.

This thesis is comprised of two main parts. The first consists of Chapter 1 and Chapter 2, which focuses on outlining the main problem along with current available solutions. Chapter 1 gives a short introduction followed by the motivation and the detailed description of the problem at hand. In Chapter 2, we present the current solutions in the market along with proof of concept solutions found in the literature addressing similar issues of the problem. The second describes the system design, prototype implementation and the performance evaluation and consists of Chapter 3, 4, and 6. Chapter 3 presents the design of the proposed system, its functionality and operation. Chapter 4 details the prototype implementation including hardware, software and networking. Our system performance evaluation is presented in Chapter 5. Finally, Chapter 6 concludes the thesis and presents our outlook on future work.

Chapter 2

Background

Intelligent Transportation Systems (ITS) are technologies and applications that mainly focus on providing services to improve traffic management and the timely dissemination of information among users, governmental agencies, municipalities and other concerned parties for safer, smarter and better utilization of the ITS networks [3]. ITS information flow is currently restricted between drivers and governmental transportation agencies (driver-ITS), while driver-driver information sharing is limited to social-network based smartphone applications.

Currently ITS information dissemination methods for on-the-road drivers regarding current and upcoming road-related events are often limited to road side information on electric billboards. Extending information dissemination from road side information billboards to a more accessible source of information such as the driver's smartphone, and in vehicle information console, will improve traffic safety and driver road experience. Additionally, collection of real-time road condition data is performed by specialized vehicles of the transportation sector of the government retrofitted with specialized sensors [4]. However, this sort of data collection cannot be relied on as the sole source of information due to their inability to monitor all the transportation networks for real-time problems.

Governmental agencies are aware of this limitation; hence arrays of stationary sensors are placed as a solution to counteract the lack of information collected by traditional means. These sensors are set up to monitor roads exhibiting high traffic volume or frequent road problems, thus requiring extensive infrastructure and maintenance to function properly [5].

2.1 Current techniques for traffic and road condition detection in ITS Systems

Currently, there are a number of systems to monitor road and traffic health, process data collected, and provide feedback both to the drivers and the governmental agencies. These systems can be classified into two categories; mobile-based sensing systems and stationary sensing systems. Mobile-based sensing systems comprise of sensors fitted on special purpose vehicles that perform routine road monitoring on the ITS system at different road locations. These systems implement detection methods such as 3D reconstruction, vibration, and smartphone based road monitoring. Stationary sensing systems, on the other hand, rely on stationary sensor nodes installed throughout the ITS system performing continuous monitoring of a single fixed location such as loop detectors, and closed circuit cameras. Both mobile and stationary systems share a number of shortcomings such as the need for extensive infrastructure, high-power requirements, and bulky hardware.

2.1.1 Mobile based sensing systems

To perform mobile based sensing both land and aerial vehicles are used, these vehicles are retrofitted with sensors, capable of sensing varying aspects of the road to detect anomalies using a number of methods.

2.1.1.1 3D Reconstruction based pothole detection

3D surface reconstruction method [59] is a visual type of detection that makes use of either range sensors using lasers, or stereovision algorithms using video cameras. While laser-scanning systems offer real-time detection, they require large devices mounted on the body of the vehicle, frequent maintenance and significant power consumption. This technology is impractical for lightweight passenger vehicles. Comparatively, stereovision-based pothole detection only requires two cameras, and is capable of detecting pothole clusters. However the system also requires complete 3D reconstruction to detect potholes exceeding the computational capabilities of most smartphones.

2.1.1.2 Vibration-based pothole detection using dedicated sensors

These require a small amount of storage, and are placed inside the vehicle thereby shielding it from adverse weather conditions. They mainly use the accelerometer, thus helping reduce power consumption. However, they suffer from a significant number of false positives resulting from road artifacts such as bridge expansion joints, cable protectors, and train rails.

2.1.1.3 Vibration-based pothole detection using smartphones

Smartphones offer better computational capabilities than dedicated systems, lower implementation cost, and no maintenance. However, this approach suffers from power limitations and low accuracy due to placement inside the vehicle. While research efforts showed success, prototypes have required repeated manual adjustments of the sensors' sensitivity to match car type.

Both manned and unmanned aircrafts are used to monitor large stretches of roads with mounted cameras and radars. The aircraft acts as a mobile speed trap.. However such methods incur a large cost to operate and maintain, and fail to provide results in bad weather conditions limiting its usability.

2.1.2 Stationary sensing systems

To perform stationary based sensing various sensors are used and are placed either inside the pavement or on the sides. These sensors are mainly focused on detecting dangerous driving behavior and traffic congestion of the road, and include

Loop Detectors: This approach relies on sensing the magnetic fluctuations to the iron present in car axels using a number of buried sensors. A hidden sensing apparatus that is triggered once a vehicle passes above it captures fluctuations in the magnetic readings indicating a vehicles passage while ignoring any other entities such as pedestrians and animals. While this method is fairly

accurate, and weather resistant to an extent it does not allow for proactive traffic solutions, and is incapable of distinguishing between the vehicles passing over them, loop detectors are embedded into the pavement this makes the sensor system quite inflexible and hard to fix in case of failure.

Visual detection: using manpower, and dedicated vehicles routine checks are performed to find road health problems and dangerous driving behavior. Either through public reports or through ministry personnel updates to problems are collected.

While the methods of detection differ in key aspects relating to their operations, the results remain the same. All aim to provide quick and accurate detection of dangerous road conditions and drivers. They also require elaborate setups, and fail to address all issues when used, such as loop detectors, while they offer high detection accuracy and weather resistance, they are very difficult to service, and in most cases require disruption of normal traffic while issues are getting addressed.

2.1.3 ITS systems

The array of sensors mentioned above forms a large network of data collecting nodes, the data collected is then aggregated at a central traffic control center which process the data alerts drivers of potential hazards, and governmental agencies of problem areas that require attention.

Recently a large number of developed nations such as Canada, United States, Japan, and Australia have automated the above system into what is known as the Intelligent Transportation Systems (ITS) to accommodate the more overcrowded areas specifically large metropolitan cities.

In Canada, the first major ITS project is named COMPASS developed by the Ontario Ministry of Transportation (MTO) on January 1991 [6] to manage traffic congestion on urban freeways. The COMPASS system consists of seven major components including vehicle detector stations, closed circuit television cameras, maintenance patrol, traffic operations center, changeable message signs, emergency response agencies, traffic, and road information systems.

The COMPASS system aims to reduce traffic congestion and increase road safety. To achieve this goal the system focuses on enhancing three aspects relating to road safety: (i) decreasing the time

latency for detecting and removal of freeway incidents and vehicle breakdowns, (ii) providing live updates to all drivers on the highways regarding delays and traffic conditions, and (iii) addressing peak rush hour traffic flow to avoid deadlocks and prolonged traffic jams through innovative traffic control devices.

Data collecting in COMPASS relies on loop detectors with a minimum of one loop detector per lane. Some lanes may contain two loop detectors to measure the average speed of vehicles in that lane, length of the vehicle, and occupancy information. Loop detectors are planted throughout the highway at inter-distance of 500-800m between the sensors. The information is then transmitted to the Traffic Operations Center (TOC) through fiber optic, and coaxial cables. The information is then processed by the Freeway Traffic Management System (FTMS) computer. The data is then sent to electronic signs to inform drivers of upcoming events. The information is also uploaded to a public interactive map which displays information regarding the whole highway for planning purposes. The COMPASS system also uses manual data collection as it relies on maintenance staff patrolling on the highway to call in any incidents they may observe. The system also relies on a network of Closed Circuit Television (CCTV) cameras stationed one kilometer apart along the freeway. The CCTV system is implemented to ensure the accuracy of the data collected from the loop detectors.

Once a road hazard is detected, the system alerts of the location and extent of the event and dispatches the appropriate emergency response agencies such as Ontario Provincial Police, fire departments, ambulance services, and towing companies to address the issue.

In the USA, the ITS system nicknamed Clarus, was introduced in 2004 the aim of this system is to monitor and provide near real-time updates regarding adverse weather conditions on surface transportation users by collecting data about both atmospheric and pavement conditions from the environmental sensor stations (ESS), and Automated Vehicle Location (AVL) equipped trucks.

The data is then uploaded to an interactive map as well as weather warning service to allow drivers to better plan upcoming trips and alert them of dangerous road conditions.

In Australia, the ITS system SCATS (Sydney Coordinated Adaptive Traffic System) was introduced in 1982 [59], SCATS mainly manages the dynamic real-time timing of signal phases at traffic signals, ensuring a continuous flow of traffic thus avoiding traffic congestion. The system uses sensors situated at each traffic signal to detect vehicles and pedestrians waiting at the crossings. To detect vehicles inductive loops are used at each lane to calculate the total number of vehicles at each intersection. The pedestrian are detected via crossing buttons situated at the sidewalks. The system also relies on information collected from the vehicle sensors to adjust the timing of traffic signals in the network.

The current ITS systems do successfully enhance road safety, and lessen traffic congestion. However, these systems are rigid in nature, and incur a large cost when it comes to expansion or upgrading of the hardware used, and are susceptible to bad weather conditions. This makes these systems unfeasible to preexisting road networks, and developing nations.

2.2 Information Dissemination in Vehicular Area Networks

ITS implementations rely on stationary sensor stations, and dedicated vehicles to collect data pertaining to road health, and accident detection. The information must then pass through a centralized data collection server before it is sent to other drivers, and online resources such as interactive maps. However, with increasing processing capabilities of on board computers installed in vehicles this scheme becomes less effective.

With the increase in the complexity and sensing capabilities of built-in vehicle sensors, accidents and road conditions can be detected with high accuracy. It also becomes more viable to allow vehicles to share the information collected with each other directly by utilizing onboard communication devices avoiding the latency resulting from using a centralized approach for data distribution explained in Section 2.1.1. This sort of communication is not limited to vehicle-to-

vehicle (V2V) communication but also allows vehicles to communicate with Roadside Units (RSUs), e.g., vehicle-to-roadside (V-2-R) (V2R) communications [49]. These two types of vehicle communication are known as Vehicular Ad hoc Networks (VANETs)

The VANET communication infrastructure facilitates a wide range of promising applications including road safety applications such as accident reporting and collision warning to non-safety applications such as entertainment applications and applications to exchange road related information and improve the driving experience. For example, by utilizing the vehicle GPS and sensors devices, a vehicle can monitor road surface conditions and detect potholes on the road [50]. Then, the vehicle can disseminate the detected pothole warning through the V2R and/or V2V communication to the governmental agencies to fix the potholes and through V2V communication to neighboring vehicles. As a result, any approaching neighboring driver is able to drive with caution and avoid unnecessary risk of hitting a pothole.

The data disseminated through both communication infrastructures in VANETs fails to address information sharing in vehicle congested areas because the information sent through V2V communication received from the source vehicle is most likely resent by the neighboring vehicles resulting in an overflow of redundant information.

There are a number of research efforts to design VANETs protocols to maintain the effectiveness of communication while avoiding unnecessary flooding of messages in traffic congested areas. Routing protocols designed for VANETs to route the message along propagation path instead of simply multicasting the message over the whole network to achieve full coverage, thus avoiding the flooding problem of network overflow in congested areas. VANET routing protocols can be classified into two subcategories: topology-based routing and geographical routing. Topology-based routing requires the information about links within the network to be available to the sending node to perform packet forwarding. Geographic routing relies only on the last neighboring location

information received to perform packet forwarding. We briefly discuss a number of those protocols in the following sections.

2.2.1 Topology-based Routing

Topology-based routing requires the information of the links present before packet forwarding takes place, for this information there are two main approaches proactive (table driven) and reactive (on demand).

2.2.1.1 Proactive Routing (Table-Driven)

In table-driven routing information is always updated and maintained within all the network nodes. Using control packets, the routing table is constantly broadcasted among nodes to keep the routes or the link states between any pair of nodes up-to-date. Utilizing the routing table, the sending node decides the next hop node in the direction of the destination node. Proactive routing however suffers from a shortcoming as the maintenance of dead paths places a large overhead on the available bandwidth especially in cases of highly mobile VANETs such as highways.

Fisheye State Routing (FSR) [9] is a routing protocol that relies on efficient link state routing by maintaining a topology map of the whole network present inside each node, it differs from normal proactive routing protocols in two key aspects. First, network status updates are not broadcasted to the whole network but only to immediate neighbors. Second, the link state information is broadcasted in different frequencies based on the length of the hop distance. Entries that are further away are broadcasted with lower frequency than ones that are closer. While the differing frequency of updates lowers broadcast overhead, it negatively affects the accuracy of link status information.

Reactive routing takes place only after a node wishes to communicate with another. The routing protocol goes through a discovery phase where a packet is sent through the network to solicit members for location within the network. After the initializing phase is complete and a routing

map is created, messages are then sent based on the path of live links. Reactive routing requires less maintenance for the network routes than proactive routing. However, the introduction of the initialization phase does increase the time latency for a message to be sent between neighboring nodes.

Ad hoc on Demand Distance Vector (AODV) [7] routing is an example of reactive routing. Once a node decides to send a message, it first sends a broadcast query message. Upon receiving the broadcast query message, all nodes update their routing table with the sending nodes address. When the query message reaches its intended destination, a reply packet is then sent back in the same method the query message was sent and all nodes along the path update their routing table with the new information, however this time, the nodes record the address of the previous node. Upon arriving at the destination, a reply packet is then sent through the complete path obtained from the query message back to the source node. However, the message is appended with the last hop node address after each hop. The final message would finally have the total path between the source and distention. In case of failure, the source is informed, which triggers another cycle of query-response to remap the network to find a new path.

AODV – Preferred Group Broadcasting (PGB) [8] is an improvement over basic AODV as it has a different mechanism for mapping the network that reduces the overhead of route discovery making it more suitable for VANETs. Using the signal itself, the receiving nodes can determine if they are in a preferred cluster. A preferred cluster is a group of nodes with a leading node responsible for broadcasting and relaying data from the rest of the group. However AODV-PGB has some drawbacks as route discovery might take a longer period of time if the destination node is located outside of the sending node cluster.

2.2.2 Geographic Routing

Geographic routing depends on the physical positioning of the nodes with respect to each other. The source stores the position of the destination in its header. The source node uses the constant

updates of neighboring node's locations to determine the next hop, based on the last location update the nodes within radio range are labeled as neighboring node, from the pool of neighboring nodes the sending node chooses the node closest to the receiving node as the next hop.

Geographic routing protocols, unlike topology-based, do not rely on link state information to send packets among nodes. Hence, the geographical routing is better suited for use in VANETs. Geographical routing protocol reliance on the greedy approach to decide next hop can create some problems with dead end nodes, to address this drawback a number of solutions were introduced.

Non Delay Tolerant Network (Non-DTN) [11], is a geographical routing protocol that was introduced to take advantage of densely populated VANET networks. In geographical routing the sending node must pick a neighboring node to relay the message to the receiving node but in some cases no neighbor is closer to the destination than the node itself. In this case, the protocol declares the packet to have reached a closest distance before being sent and a recovery strategy to deal with the failure is performed.

Greedy Perimeter Stateless Routing (GPSR) [60] deals with problems arising in a VANET environment with scarce number of nodes. When a node forwards a packet to an immediate neighbor which is geographically closer to the destination node in some cases the packet reaches local maximum. The protocol then enters recovery mode. Consequently, the packet is forwarded to another node that is closer to the destination than the node where the packet encountered the local maximum. The packet resumes forwarding in greedy mode when it reaches a node whose distance to the destination is closer than the node at the local maximum to the destination.

Utilizing an efficient VANET routing protocol to facilitate V2V and V2R communication, ITS implementations started to adapt VANETs into its structure. VANETs increase amount of data collected and ease of communication. This allows for faster and more accurate detection and localization of accidents, traffic congestion, and road hazards, without incurring high additional costs to maintain and expand.

2.3 Smartphones for traffic and road condition detection in ITS Systems

In the last decade, smartphones have evolved beyond a simple device for communication. Current smartphones boast large processing capabilities with quad and octa-processors becoming the norm. They are also equipped with highly sensitive sensors with a very high sampling rate allowing the smartphone to become a sensory node capable of sensing any changes in its environment instantaneously [35].

To adapt the sensing of smartphones in a vehicular environment a number of sensors in particular form the main group for usage. These include: global positioning, acceleration in three dimensions, tilt sensing, visual and auditory sensing including ambient light, audio, video, and still pictures.

Global positioning capabilities rely on two forms of localization; normal GPS and Wi-Fi assisted GPS localization. When solicited, the outcome has a two variables corresponding to longitude and latitude coordinates capable of pinpointing the vehicles with varying degrees of accuracy based on location on the globe. Three dimensional acceleration sensors also known as accelerometers allow the smartphone to sense physical forces such as speeding, skidding and sudden stops. Smartphones also have multiple forms of communicating data such as Wi-Fi, Bluetooth and over 3G and LTE cellular networks.

Currently in the market today there are a number of private and governmentally supported applications of road monitoring using crowdsourcing techniques based on users volunteering their smartphone devices while driving, the following section gives details pertaining to them and any drawbacks they suffer from.

2.3.1 Boston reporting system (Street Bump)

The system was built in collaboration with Citizapps, a partnership between Fabio Carrera, a professor at Worcester Polytechnic Institute, and Joshua Thorp and Stephen Guerin of the Santa Fe Complex and the mayoral office of Boston [61]. The system works by utilizing the driver's smartphones to monitor the potholes in the roads during their in city trips. The data is then

collected and sent for analysis to authenticate events based on crowdsourcing, the authenticated locations are posted on the interactive map for the public, and the locations along with the severity of each pothole is sent to road services to address the issues as shown in Figure 2.1.1.

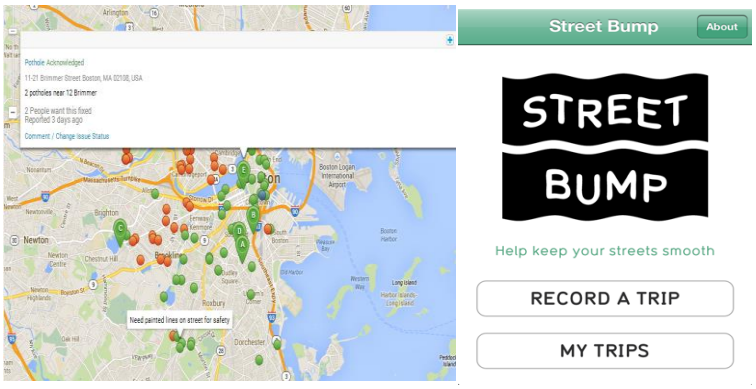

Figure 2.1: Street Bump main screen and Map

The system makes use of the accelerometer and GPS sensors within the smartphone, the system only requires that the smartphone be placed in a stable cradle on the dashboard and remain undisturbed during the trip. However, such status is not easy to maintain, which renders the system impractical as it is unable to adjust to the different car types, since the main target of such a system is to detect major potholes that can be detected regardless of the car's suspension rigidity. The system addresses the issue of the high volume of false positives by setting a relatively high threshold on the number of reports before allowing a pothole location to be authenticated, while the system might be able to detect and authenticate events in location with a high volume of traffic it fails to address areas with low traffic.

2.3.2 Waze

The waze mobile application is an application that uses social networks and built in sensors to share information regarding road condition between drivers within the same vicinity or over social posts made to the social networks [63]. The Waze application also offers augmented road maps with the social reports from multiple users to highlight the busy highway sections and the best routes between the current user location and their destination. The application makes use of the smartphone built in GPS sensor to localize the user and ascertain the user's current vehicular speed. Using that data the application can detect and locate traffic congestion and other similar events. Figure 2.1.3 shows a screen capture of the application map with the calculated route, the road events and the location of the other users that are currently using the application.

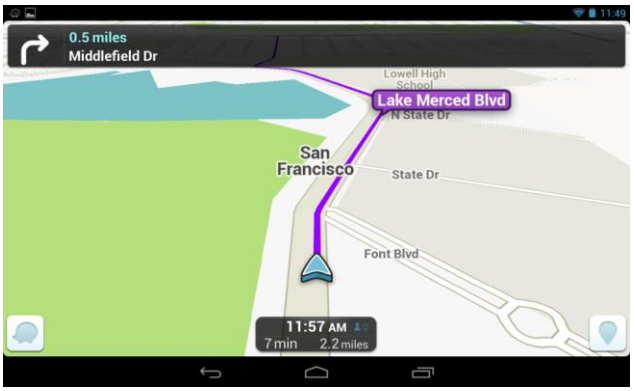

Figure 2.2: Waze routing map

Waze also incorporates active reporting where the users can pick the event type and allow the current GPS location to pinpoint the location of events such as an accident or bad road conditions. Figure 2.1.3 shows the list of events and how they appear on the map. This application proves the utility of using collaborative systems to provide road information. While this application's main focus is sensing traffic conditions it makes no attempts to detect the road conditions or driver

aberrant behavior using built-in sensors and relies on only active reporting to pinpoint the unique events along the road from hazards, accidents, and others.

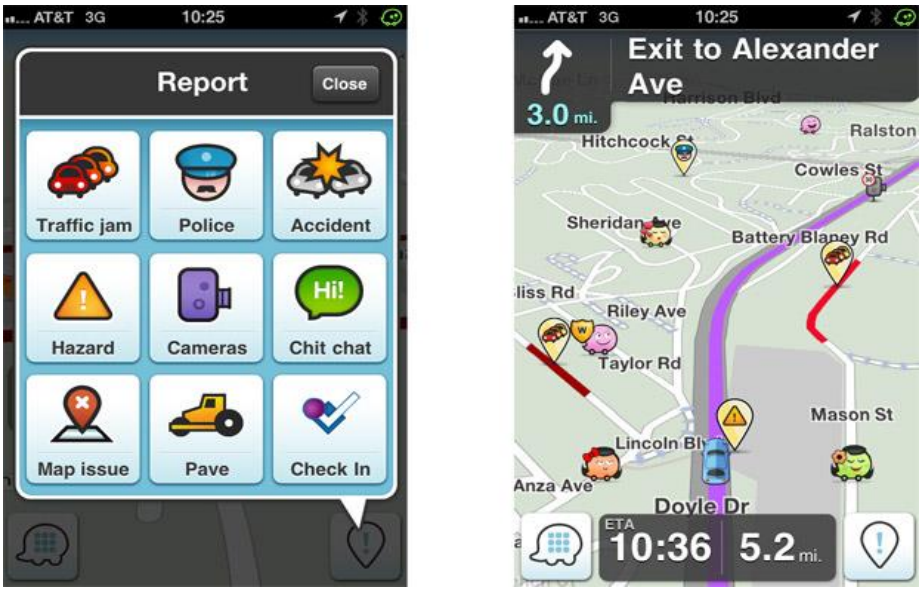

Figure 2.3: Waze Active Reporting

2.3.3 Inrix

Inrix is a mobile application that offers an interactive map with road events collected passively from participating smartphones to detect congestion events [64]. It also logs general data such as average speed and location along with time of day to ascertain general traffic behavior and predict traffic jam times to better plan current and future trips. Figure 2.1.5 shows the congestion map with the expected delay and the length of the jam.

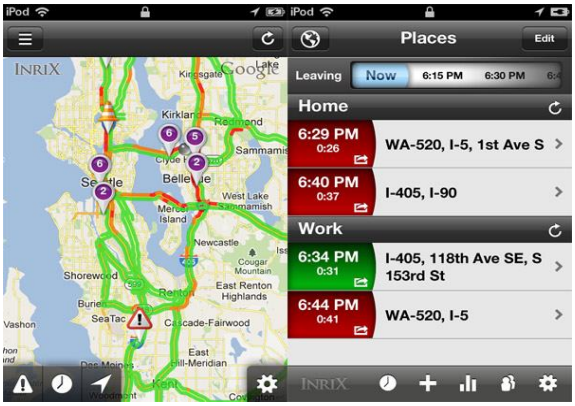

Figure 2.4: Congestion map in Inrix

2.4 On Board diagnostic system in vehicular sensing

The demand for more fuel efficient and reliable vehicles was answered with the introduction of the fuel injection based internal combustion engine. The new system relied on a pump to pressurize fuel and release it into the combustion compartments. Such a task requires much higher sensitivity and a shorter response time than what normal mechanical timing mechanism could offer.

This hurdle forced vehicle manufacturers to adapt built-in sensors to regulate the engine's functions such as the amount of oxygen available for combustion, the fuel flow rate and to perform diagnostics.

In the 1980's governmental mandates started relying on the data collected by these sensors to ensure the vehicles conformity to environmental protection laws. However, vehicle manufacturers developed independent systems with varying protocols to meet the government's demands making it difficult to perform the test using one unifying machine.

To address this issue, the On-Board Diagnostics (OBD) system was introduced. The term OBD refers to the standardized system which collects various metrics from the sensors built in the vehicle. The OBD serial input/output port was standardized in both the physical shape and the pin usage. The new interface serial port was called the J1962 connector [28, 29]. The port is shown in Figure 2.7. The permissible placement of the port is constrained to be at the area immediately below the steering column.

Initially the systems capabilities were focused on meeting the government's mandates only. With the mandate came a very important aspect for the use of the OBD port, while standardized it gave a number of pins for manufacturers to develop better public access to diagnostics tools and access to built-in sensors to allow vehicle repair shops and self-repair enthusiasts better accuracy and more information when to diagnose vehicle problems [26, 27].

The OBD-II standard addressed the issue of the wide variety of diagnostic connectors and standardized them into a common 16-pin connector named J1962 located under the steering wheel. Figure 2.3 shows the most common placement of the connecter.

Figure 2.5: The location of the OBD-II connector in a car [30]

The standardization of the OBDII also includes strict enforcement of pin usage where a number was set aside for the manufactures to use as they see fit while the rest was left for official EPA testing. Figure 2.4 shows the J1962 connector pin out and Figure 2.8 lists the pins primary task as set by the standardization. As aforementioned, the manufacturers are allotted a number of discretionary pins, these pins are employed to send data about the cars' various modules to monitor and troubleshoot. In the J1962 connector standardization also introduced a number of different communication standards. Currently there are five different signal protocols.

2.4.2 Signal protocols

The OBD-II as it is known in north America- also called the E-OBD in the euro region- makes use of five main protocols. Each protocol makes use of a different combination of pins while maintaining its readability by all EPA testing machines. The five protocols are the following [27]:

- SAE J1850 PWM: pulse-width modulation 41.6 kB/sec. Pins 2, 10.
- SAE J1850 VPW: variable pulse width 10.4/41.6 kB/sec. Pin 2.
- ISO 9141-2: asynchronous serial data in 10.4 baud. Pins 7, 15.
- ISO 14230: Keyword Protocol 2000, data rate between 1.2 to 10.4 kBaud. Pins 7, 15.
- ISO 15765 CAN: controller area network bus. It can provide 250 or 500 kBit/s. Pins 6, 14.

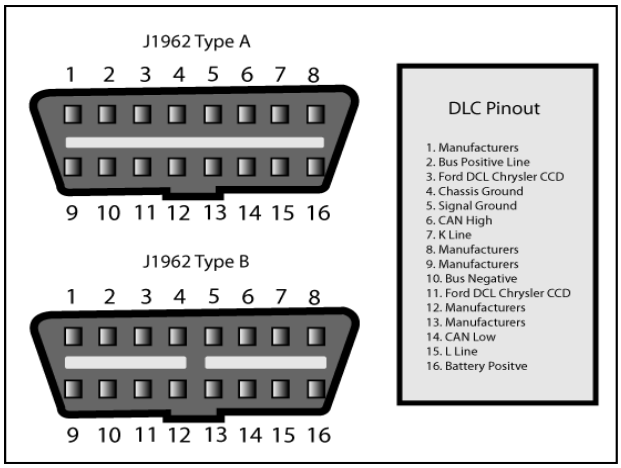

Figure 2.6: Pin description for the J1962 OBD-II connector

Currently, the controller area network (CAN) protocol is the most prevalent as it is the fastest of the five protocols allowing for the new drive by wire car types to function. The CAN protocol was initially introduced in 2003. Since then it has been continuously phasing out other protocols. Recently, all vehicles manufactured post 2008 carry the CAN protocol by default.

2.4.3 Diagnostics Information

The main and most complex module in the car is the Engine Control Unit (ECU). It is responsible for receiving the various sensory inputs from all the other car modules. It then synchronizes the engine mechanics to fit the new data at a very rapid pace. This makes the ECU the ideal location for the OBD connector as it allows it to access all the other modules without the need for dedicated wiring.

The method to request the data is defined by the SAE J1979 standard. The OBD system standardization introduced a list of parameters that must be collected from all the modules, and while manufacturers collect numerous amounts of parameters, they do fluctuate from one manufacturer to the other making them an unreliable source of information for crowdsourcing scenarios.

Each parameter is assigned a PID (parameter identification number). The EPA testing machines simply solicit the OBD for each PID required to ascertain the car eligibility for the road. While service stations solicit OBD for more module specific PIDs to troubleshoot errors or mechanical failures through real time data acquisition and any recorded diagnostic trouble codes (DTCs). A DTC is any flagged event deemed abnormal to normal working conditions. While there are a vast number of DTC codes that are standardized, manufacturers still use proprietary DTC codes.

PIDs are standardized by the OBD protocol allowing any generic OBD reader the ability to read the data supplied in the form of a tuple containing both the name and a brief description of the value. On the other hand, DTC data offers a wider spectrum of sensory data in a format that is shared for the most part among all manufacturers.

However, while the data format is common, the OBD reader is not be able to differentiate between the correct nomenclatures of a tuple from one manufacturer to the other. To address this void, new OBD readers capable of detecting car manufacturers through the decoding of the Vehicle identification number (VIN) of the vehicle attached to were developed.

In this work, the main focus is on making use of both generic PIDs and advanced DTCs. The use of the PIDs will focus mainly on updating the vehicles current speed, RPM, gear position among others. DTC codes will give the system the ability to detect events such as accidents through the detection of trouble codes such as the deployment of the airbags.

All generic trouble codes have a straightforward format that pinpoints both the type and location of failure or fault. The following are some research and market available implementations of OBD technology in road side detection, which are related to our proposed system.

2.4.4 Wreck Watch

Wreck Watch created for research purposes [53], makes use of the smartphone built-in sensors to detect collisions through the monitoring of the accelerometer for extreme decelerations events coupled with sudden changes in any direction followed by a static GPS readout confirming the vehicle is stationary after accident. Figure 2.8 illustrates the systems functionality, the application looks for large accelerometer events in the x and y axis to determine the presence of an accident once confirmed the application starts by sending a distress message including the GPS location of the vehicle to the EMS.

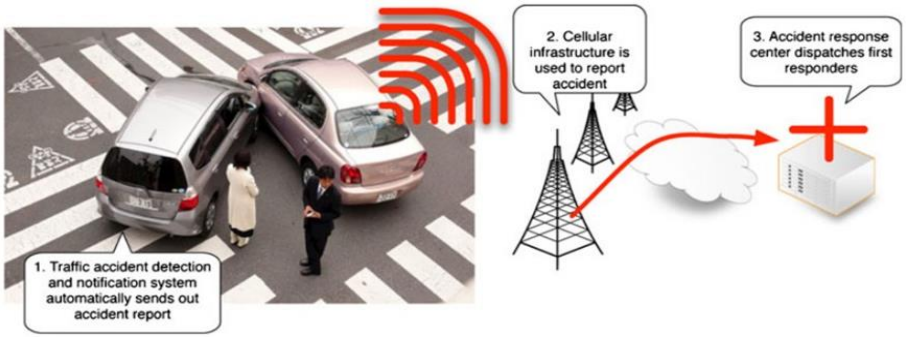

Figure 2.7: Wreck Watch accident detection

The system not only sends the location but also gives metrics about the extent of the damage detected to better inform the EMS of the situation allowing for the right facilities to be deployed. The accelerometer data logged during the accident also acts as a black box helping the police and

insurance companies to better reconstruct the accident as it happened and assign the liability more accurately, the system makes the assumption the smartphone will remain stationary during normal operation and the lack of an OBD device attached to the smartphone lowers the accident detection capabilities of the smartphone alone,.

Zaldivar et al. [56] created a system similar in function to wreck watch. The system incorporates the accident sensing capability of vehicles specially the airbag deployment sensors using an OBD dongle to tap into the vehicles own sensors the system looks mainly for airbag deployment DTC codes. Once triggered the system registers an accident and acts in a similar fashion to Wreck Watch. The system also detects other important metrics like the use of brakes, the speed before the accident and any changes in direction of steering during or before the accident. Such information allows the police to better understand the accident and whom to place the blame on.

2.4.5 Torque Pro

This application allows the user to turn their smartphone into an OBD reader [57], using a Bluetooth OBD dongle. The application gives the user access to more dedicated gauges such as exact engine RPM, O2 sensor, fuel to air ratio, and DTCs straight from the vehicle while running. Figure 2.8 shows two screen captures of real time data displays and the trouble codes list of Torque Pro. Torque Pro facilitates better identification of car trouble ahead of time through improved monitoring.

Figure 2.8: Torque Pro application. [62]

2.4.6 DriSMo

The work in [5] proposed a system called DriSMo. The system is more focused on driving habits, specifically sudden accelerations and decelerations [5]. The system relies solely on the accelerometer to determine dangerous behavior through monitoring sharp changes in speed or direction. DriSMo logs infractions with location to be shown in the interactive map, while also offering a report grade of the total behavior monitored. Figure 2.9 shows the statics of speed total driving behavior between, the start and end locations.

While the application does monitor speed and hard braking the other metrics used are limited, events such as skid sensors, steering wheel angle, and suspension position.

Figure 2.9 DriSMo report page [5]

2.5 Sensor fusion

In wireless sensor networks multiple sensors of the same or different types are placed in proximity to each other. Sensor fusion refers to when multiple sensors in a control system are used to provide more information, robustness and complementary information.

More information: refers to the use of secondary sensors of the same type to extend the range of the geographical environment covered, in order to improve the quality of data collected about the environment compared to that collected by a single sensor. Robustness: can be achieved by several techniques. For example, statistical techniques that refers to the use of multiple readouts of the same sensors type or multiple readings from a single sensor [12]. Robustness also involves fault tolerance with more sensors available it is easier to corrupt, faulty sensory input, it also allows the system as a whole to detect faulty or misconfigured equipment [54].

Complementary information: refers to sensory data that is not similar in type but can be used indirectly to increase the accuracy of the overall results by combining the data to give a more complete description of the immediate environment. They can also be used to perform plausibility check for each other to confirm acceptable working condition [54].

Sensor fusion can be implemented using both the accelerometer data from the OBD and the smartphone devices to make better and more accurate detection of events that rely on small changes such as weaving and small potholes.

Chapter 3

Road Monitoring System

We propose Smartphone Road Monitoring (SRoM) system that is capable of sensing road artifacts such as potholes and slippery roads. It is also capable of detecting aberrant driver behavior such as speeding and drifting. The system can function using a smartphone only and also can integrate the use of an OBD dongle to increase the types of events detectable and the overall accuracy. The SRoM system relies on crowdsourcing by making use of the crowds themselves to monitor road health and safe driving conditions.

3.1 Traffic and Road Conditions monitoring using Crowdsourcing

Crowdsourcing based systems are created in situations where a limited resource is used to address a situation with a large number of entities. It addresses the scarcity of resources by breaking down the problem into smaller tasks that can be done without incurring a larger cost.

There are two types for crowdsourcing. The first is *voluntary* where the cost of operation is kept to a minimum, while providing incentives for the participating users to encourage continuing participation [55]. The Second type is enforced participation where the users must offer their capabilities to be allowed to use the system. In the later type, the cost is higher than the voluntary mode, but it remains lower than that of a single entity based approach.

Existing traffic and road monitoring are based on heavily structured sensing platforms such as loop detectors, CCTV cameras, dedicated sensory vehicles, and constant police presence. As aforementioned in the Chapter 2, these technologies are difficult to scale with the rate of expansion in vehicle numbers and new road expansions, and incur large costs to maintain.

SRoM on the other hand utilizes crowdsourcing as a powerful tool to shift part of the burden onto the drivers to collect the data needed to properly monitor the road and traffic safety conditions. The result is an efficient, low cost system that provides better and more comprehensive coverage when

compared to its predecessor. SRoM combines information from the individual drivers, pedestrians, and governmental agencies, and combines this information in real time to locate deteriorating road conditions, or dangerous driving behavior and forward the data to the public and law enforcement agencies to help address these situations as they occur.

3.2 SRoM Architecture

The system is designed to meet the following requirements:

1. Integrated: Traffic and road conditions are assimilated from various sources including: smartphone, OBD sensors, and local transportation authorities.
2. Adaptable: The system is capable of scaling accuracy and functionality based on available hardware.
3. Human Reporting: Pedestrians/passengers are able to actively contribute to the system's event stream with more functions available through the system interface.
4. Context Aware: data shared with drivers such as alerts are based on their current location and mode of transportation.

The SRoM system fulfills the requirements above by allowing the same application to both sensory detection and active reporting making use of both drivers and pedestrians/passengers. The systems interactive map has the information collected from the users as well as MTO to better assist viewers in understanding the road condition on the routs they plant to travel. The system addresses the issue of varying car and smartphone types by applying a calibration phase for new users to adjust the newly registered smartphone vehicle combination to match the preexisting user pool.

The SRoM system is composed of four main components: the end user terminal composed of the smartphone and OBD dongle, traffic information supplied by preexisting ITS implementations, SRoM server, and Interactive map. The user terminal collects the majority of information while the ITS supplied information is mainly for scheduled road repair and congestion reports. The user terminal has 3 main modes of operation: (i) pedestrian/passenger mode which uses the smartphone

to make active reports of events, (ii) simple driver mode, which relies on the smartphone only to detect road conditions and abnormal driver behavior, and (iii) simple and advanced OBD mode, which is based on the vehicle and smartphone available to the driver. The SRoM server collects the reports from the end users and ITS alerts and combines the data to authenticate events, update the interactive map, and alert EMS and police patrols in case of infraction or accidents detected. Figure 3.1 describes the overall architecture of the SRoM system. The following subsections provide details regarding each component of the system.

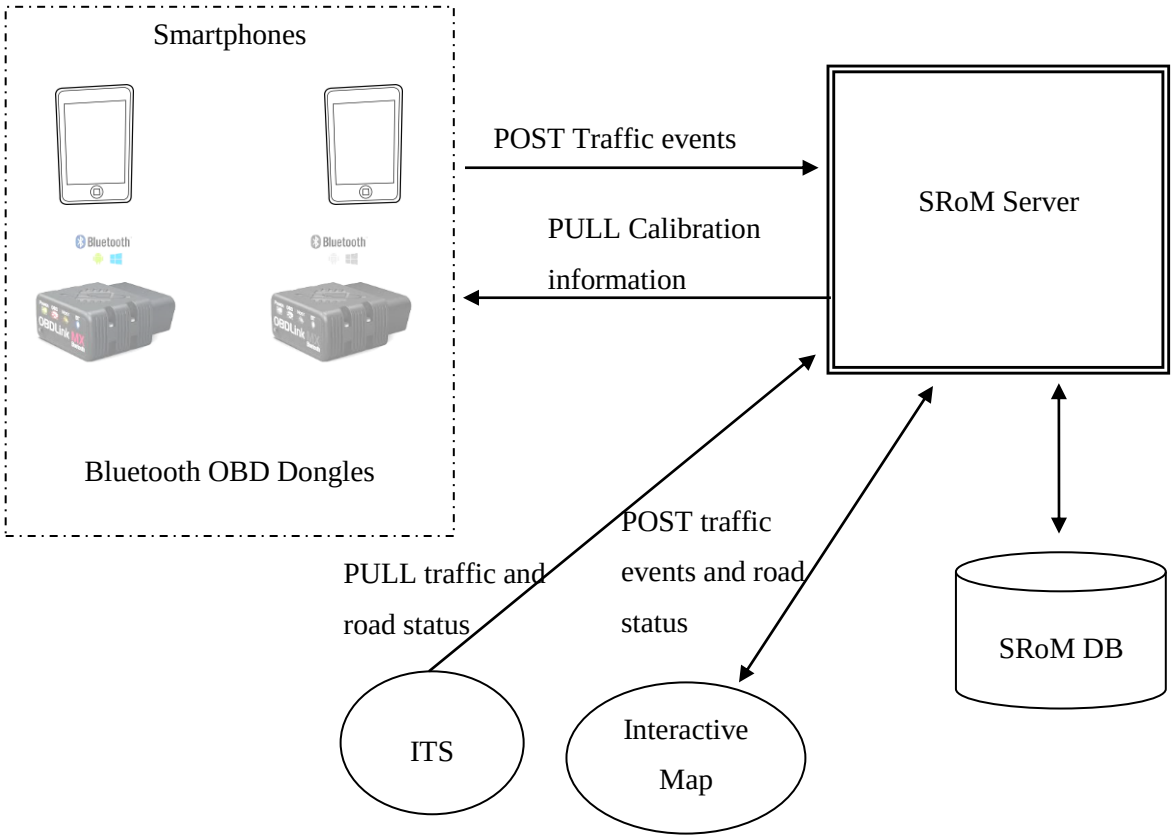

Figure 3.1 - Overall SRoM system architecture

3.2.1 End User

The end user component of the system consists of two major modes of operation, passive and interactive. Interactive mode is for use by pedestrians/passengers allowing them to report events using the system allowing for better accuracy. In the passive mode the drivers use the built-in smartphone sensors and the OBD to detect the presence of road deterioration, dangerous driving, or vehicular accidents. The accelerometer is the main sensor used in this mode as the slight variation in the x, y, and z axis translate the events experienced by the vehicle tilting during events such as dipping into pothole, slipping on sleet or driving over exposed gravel areas.

3.2.2 SRoM Server

The SRoM server is designed with three objectives:

1. Collecting events from End users and ITS road and weather alerts.
2. Track user confidence rating.
3. Push event data to interactive map engine.

End users, drivers specifically, create a continuous stream of events. However, due to the diverse nature of vehicles and smartphones used, the server must adjust the confidence rating based on each respective vehicle smartphone combination. The server is supplied with a number attributes associated with each event. Table 3.1 shows the details of each attribute and the task it performs in ensuring true positive reports.

Table 3.1 event data types

Attribute	Type	Task
Latitude	Float	Using GPS satellites the smartphone or OBD triangulates the signal using 3 or more satellites for better accuracy to locate the vehicles on earth in real in the form of two numbers longitude and latitude.
Longitude	Float	
Vehicle type	String	The users vehicle type is used to adjust the sensors sensitivity based on the vehicle type to maintain accurate readings
Smartphone type	String	The users first event is attached with the smartphone model number to adjust the systems operation to the sensors sampling rate and quality of data collected
OBD type	String	There is a wide variety of OBD devices with varying functions however this variable is only to indicate if the OBD is capable to read non standards car modules and if the advanced car modules are available, this variable is used to assign the confidence rating of a report.
Event types	String	The events authentication differs based on the severity, and the immediate impact of each event, the events available for reporting are predestinated such as pothole, traffic congestion, accident, and animal crossing etc.
Time stamp	Date	The time of detection, and time of reporting
Emergency message	Media file	In case of accidents the system allows the smartphone to act as a streaming black box allowing EMS services a better understanding of the situation while communication is available
Severity	Integer	A rating from 1 to 10 to assess the severity of the event reported where applicable to help with prioritizing event handling

3.2.1.1 Geographical Localization

End users have differing limitations when it comes to getting an accurate GPS location based on the SRoM mode:

1. Pedestrian/passenger mode: the users have access to smartphone only and the program can solicit the GPS localizing starts as soon as the active reporting mode is activated allowing the smartphone enough time to get a reading while the user fills the event report.
2. Driver mode: where the drivers are highly mobile requiring the geographical location acquisition time to a very short time.

a. Driver has access to a smartphone only: The GPS must be solicited periodically to ensure quick and more accurate GPS readout when compared to waking GPS sensor only after event detection. To save on battery power a predictive algorithm is used during the blind periods that take place between GPS reads. The algorithm takes the recently acquired GPS location as the starting point and using either the areas speed limit or the vehicle speedometer to detect the movement speed of the vehicle along with the accelerometer readings to detect the changes in direction a new predicted GPS locations is generated.

b. Drivers with access to OBD dongles: In the case of built-in GPS sensor inside OBD, the sensor is kept alive to ensure accurate readouts with no time latency. However, in case of a simple OBD device, the smartphone solicits the GPS sensors periodically. The speedometer and steering wheel angle are used to locate events based on last GPS reading corrected to the speed direction traveled using the OBD data to remap location from the last GPS readout.

3.2.1.2 Event communication

There are two forms of communication, Wi-Fi, and 4G. The smartphone is used for both pedestrian and driver mode. The events effect on road safety dictates the form of communication. However in some cases the drivers are required to use OBD with a built-in sim-card in cases such as repeat

offenders under probation to ensure continuous monitoring for any aberrant driving behavior that can pose a risk to other drivers.

3.2.1.3 Overall system functionality

Once users install the application and run for the first time they are registered into the system and assigned a unique ID, and a lifetime confidence score. The end user becomes part of the crowdsourcing pool allowed to the interactive map, report event in active and driver mode. The reports collected are then sent to the SRoM Server where events are weighted based on the current confidence rating of each user. The event is assigned a confidence rating based on the number of reports received and the individual confidence rating or the reporting users within a limited area (based on the GPS accuracy) of the initial report, once a predefined confidence threshold is achieved the event is classified true positive. The repair personnel are alerted to the location type and initial report time to be addressed. The repair personal upon arrival will confirm the presence of the event, and asses the degree of severity based on their report the reporting clusters confidence rating is either elevated or demoted. The system also has emergency mode where the driver mode experiences extreme sensor data indicating an accident, the user is sent a timed or untimed message to dismiss to indicated faulty report, if not dismissed EMS vehicles are sent to the scene, a streaming media distress call is activated and recorded. The processes of the SRoM system mentioned here are described in detail in the next section.

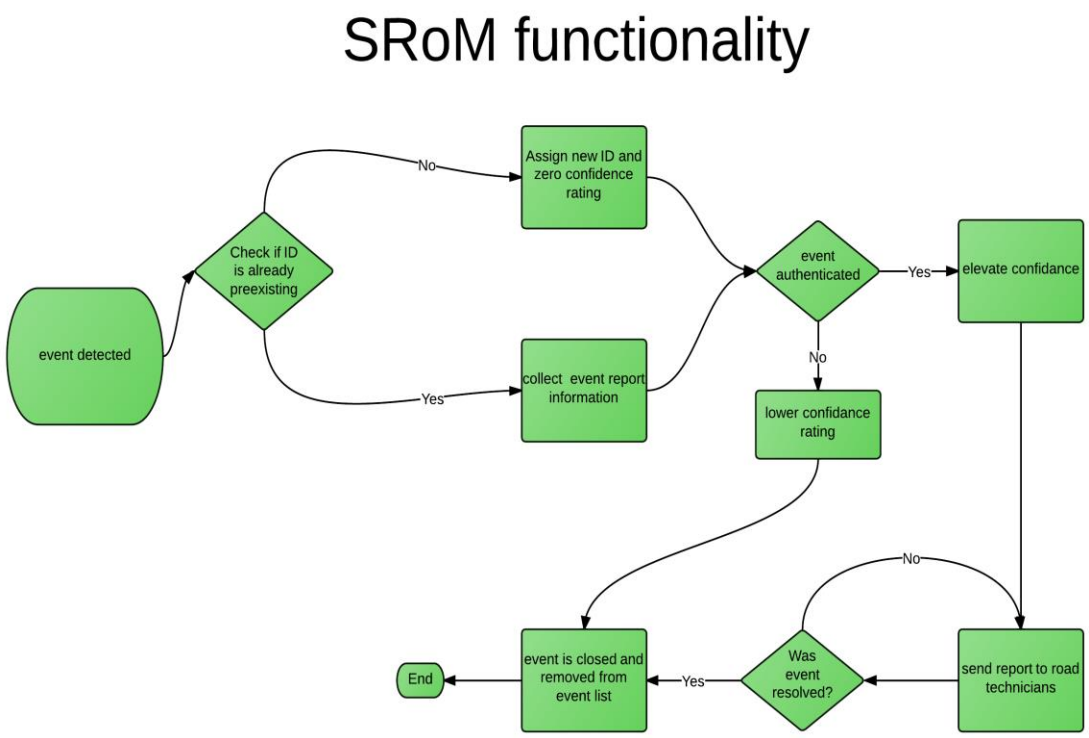

Figure 3.2 SRoM Functionality

3.3 SRoM System Processes

The SRoM processes consist of the following phases:

1. Registration phase: of newly joined smartphones.
2. Calibration phase: during this phase auto-calibration of the smartphone is carried out.
3. Operation: consists of two modes: active and passive.

In the following, we discuss each process in detail

3.3.1 Smartphone Registration

Once a user installs the application and runs it for the first time the system gets a registration request with the smartphone devices unique ID, the server responds with a system assigned ID that is used to report events both passively and actively.

3.3.2 Calibration Phase

This phase is used to normalize the reading of the various smartphones and vehicles combination to ensure more stable readings from the whole participatory user pool.

Once a new user engages the passive mode for the first time the application waits for a stable sensory reading indicating the placement within a cradle (minor noises are ignored). The application requests a training area from the server, the server responds with a list of locations of training dedicated baseline events along with the expected sensory reading. The driver is unaware of any changes in the application the system sets the sensory sensitivity to the lowest threshold preconfigured, while driving. The system keeps the GPS awake and waits to approach a known baseline event. Once the device is in the vicinity of an event it waits for any sensory input above the normal tries to match them to the supplied sensory data. Based on the comparison the system either passes the calibration incident or a failure is declared. In case of failure, the system adjusts the sensor sensitivity and repeats the process. Once a number of consecutive successes have been registered the last sensory adjustment is saved and the calibration phase's exits and the smartphone device is allowed to start reporting events as shown in Figure 3.3.

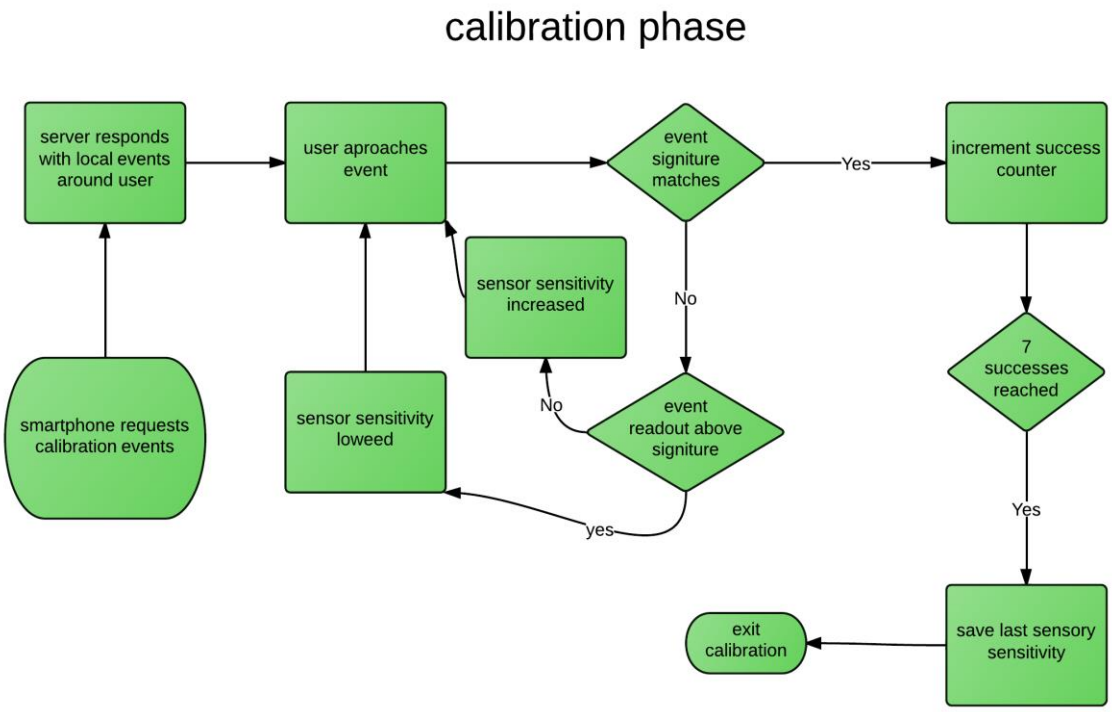

Figure 3.3 Calibration phase process

To ensure consistent results along with the smartphone ID the server assigns a confidence rating metric. This metric starts at zero. Once more events are physically checked and authenticated the confidence rating is either elevated or demoted. Compared to a predefined threshold, continuous bad reports by a certain user triggers a recalibration phase and mark the user's vehicle and smartphone as an unacceptable source of reports. After the user passes the recalibration phase, and if he/she continues to report false events, the user is then blacklisted from further use of driver mode for predefined period of time before being allowed to rejoin the system again.

3.3.3 Operation Phase

The application allows for active (pedestrian/passenger) and passive (driver) modes. In active mode the user is actively engage in the reporting of events through the use of the smartphone based

form. Driver mode, on the other hand, does not require active participation from the driver. Active mode can be either OBD assisted or smartphone only.

3.4 SRoM System Operation

3.4.1 Driver Mode

3.4.1.1 Standalone Smartphone Operation

In this mode the application upon booting up attempts to connect with an OBD active paired Bluetooth dongle. If device fails to connect, it operates in standalone smartphone mode.

3.4.1.2 Dangerous Driving Detection

In smartphone only mode, dangerous driving detection is mainly for warning purposes because the accuracy and potential for system abuse is high. Using the data from the accelerometer, gyroscope, and GPS, the smartphone is able to detect events such as

1. Weaving: using the gyroscope and GPS sensors a vehicle is recognized as weaving if within a 2 min period with constant movement based on the GPS if the vehicle registers a large value in alternating directions form the gyroscope that occurs when a driver performs rapid lanes changes at constant speeds.
2. Speeding: using the GPS sensor two readings are taken within a short period to help calculate the speed based on the distance traveled along with geofencing the system to detect users exceeding the speed limit of the specific road being used.
3. Drifting: drifting comes in many types however it forces the vehicle sideways causing the screeching of the tires, the system can detect such events by pulling the accelerometer data for and checking for high intermittent G-force readings in the x axis.
4. Stunt driving: in this mode several possible actions are observed such as: performing two wheeled stunts in which the vehicle is forced on two wheels and is driven around for a while

maintaining this state, using the gyroscope and GPS sensors, if the vehicles experiences a tilt of above 25 degrees while the GPS shows continuous motion the event is reported.

3.4.1.3 Pothole Detection

Once the driver has the smartphone inside the assigned cradle, the SRoM system is activated. State-of-art literature presents several methods of vibration detection to locate potholes and other similar road events [56]. While these schemes provide proof of concept of operation, they fail to provide an answer fitting for a crowdsourcing environment. Existing schemes require distinct manual tuning to achieve accurate results, and fail to adapt to the diversity of the smartphones types, vehicle models and makes and random smartphone placement inside the vehicle.

To address the aforementioned issues, the driver mode had to expand to allow for multiple operating modes to accommodate any smartphone type with differing sensory sampling rate, quality, and current battery charge to discern the proper sensing method to employ.

Currently there are three systems predominantly employed in current research and all rely on the accelerometer readings.

- Threshold test: In this detection test a threshold is set based on the acceleration amplitude the sensory data is tested once a reading is larger than the predetermined threshold it is identified as a pothole, once the initial threshold is passed a number of tests are required to confirm a positive result to better classify the pothole event as a large pothole or a pothole cluster. In this mode the placement of the smartphone is very important as the orientation of the smartphone with respect to the ground is key. While this test is the simplest and requires rigid placement, it has the lowest computation and power cost when compared to the other tests [39].
- Accelerometer deference test: This test is an extension of the threshold test once the threshold test is passed two consecutive readings are taken, the two readings are then compared for the rate of change once the rate of change is large enough signifying an event took place. This test

like the previous one has the same drawback requiring the orientation of the smartphone to be upright compared to the ground [41]. However, this test is better than simple threshold test as is capable of detecting pothole clusters, but requires high sampling rate smartphones.

- Zero acceleration test: this test looks for a near zero readout in all axis of the accelerometer simultaneously, the orientation of the smartphone is a non-issue in this test allowing for better test results but requires a much higher sampling rate from the smartphone to have good accuracy to detect abrupt events such as potholes.
- High-end smartphones employ the three pronged test using all the above tests to ensure high accuracy and lower false positives. However, if the battery drops below certain threshold, the tests are restricted to both Z-DIFF and Z-THRESH. Z-THRESH alone is used once the battery reaches minimum threshold allowable. The system then shuts off pothole detection and only maintains emergency detection.

3.4.1.4 Accident Detection

a. Collision Detection

GPS location allows for accurate speed calculations. Using GPS reading, if a sudden speed change of more than 30 km/hr. is detected within a short period and the car remains stationary, an accident is suspected. To dismiss false emergency message, the driver is given a one minute timed alert message to deactivate an emergency message. If no response is detected, the server receives immediate location and contact information of the vehicle involved.

If the speed reduction is above 65km within a short period, which is beyond the capabilities of any braking system, this event triggers accident mode and an emergency message is sent immediately and a voice stream is activated and forwarded to a nearby emergency vehicle to promptly move to the accident location.

b. Rollover

Rollover detection requires the presence of an NFC (near field communication) capable cradle to ensure the readings are taken with the phone stationary relative to the vehicle To be able to detect rollover events without incurring false detection caused by the smartphone falling from the cradle. While the device remains inside the cradle, Rollover events can be detected by relying on both the accelerometer and gyroscope sensors, during the event of a rollover the vehicle will experience slight jumps during the rotation of the vehicle with the various sides of the vehicle hit the ground. This generates a distinct signature mainly the presence of large spikes in the accelerometer above 3g during the constant rotations of the vehicle by looking for an alternating gyroscope. Once the combination has been detected, the user is solicited for a response similar the collision detection.

c. Car falling

Car falling can be detected using built in elevation sensors either using GPS location or barometer. If a rapid change beyond the normal inclination of the road is detected, while the GPS readout detects very little horizontal changes, the event is registered and the driver is sent a timed message similar to that of the collision detection. Smartphone OBD Assisted Mode

The system makes use of OBD Bluetooth dongles to communicate data between the vehicle system and smartphone. The OBD can improve the performance of the SRoM system as follows:

1. Sensor Fusion: Dongles include built in GPS and accelerometer sensors which can be used to perform sensor fusion increasing the accuracy of event detection.
2. Accuracy: The use of the OBD dongle enhances the accuracy of event detection due to the rigid nature of the OBD dongle plug. It provides a better source of accelerometer data when compared to a smartphone placed in a cradle.
3. Continuity in sensor data readings: Unlike the smartphone GPS, the built in OBD GPS remains online while the vehicle is active because of the constant power supply availability provided by the vehicle power system, however, the smartphone GPS is still used in cases

where better accuracy can be achieved in location with multiple hotspot locations are available.

4. Possibility to categorize drivers into types: while using the smartphone only the ability to detect driving infraction was difficult and un-enforceable as the smartphone device is not adequate enough. However, utilizing the vehicle VIN number information and the driver information associated with the vehicle, the drivers can be categorized into three types; young drivers, repeat offenders, and normal drivers, and are treated based on their status.

The OBD assisted mode operates in two sub-modes based on the complexity of the vehicle used; normal mode and advanced mode.

3.4.1.5 Normal OBD Sub-mode

In the normal OBD sub-mode, we expect the system to operate with only the standard functions used by all vehicles. Hence, the use of the CAN network is not considered.

a. Pothole detection

In pothole detection the OBD supplies the engine Rotation per minute (RPM), and Speedometer reading of the vehicle along with its own set of accelerometer and GPS readings to be used with the smartphone to detect the events with better accuracy and sensitivity. The accelerometer readings are used along with the smartphones to confirm the presence of the event, while the speedometer and RPM readings are used to confirm that the vehicle is experiencing an increased power surge. This is achieved by detecting small changes in speed while the RPM remains fixed in areas with no change of elevation alerting the system to the presence of a pothole.

b. Dangerous driving

c. Speeding

In smartphone only mode the GPS is not kept alive all the time therefor it is incapable of detecting short events of speeding. However, with the presence of the OBD device the speedometer of the

vehicle is used to constantly monitor the driving speed for any infractions based on geofencing to change the allowed speed in each region.

d. Drifting

Drifting requires high engine (RPM) on the vehicle to be held while the vehicle is kept at low speed to perform the stunt. During the drifting stunt the driver rotates the back wheels of the vehicle at high speeds while the vehicle remains in the same location or moving at a very low speed using the RPM sensor, drifting can be detected in events of prolonged high RPM readings coupled with low speedometer readings, and localized GPS readings signifying drifting.

e. Collision

It is possible to detect if an accident occurred using the vehicles airbag deployment sensors, as well as front bumper sensors if available. However in some cases out of dealership repairs can leave the airbag sensor unusable. To address this issue, the system checks the airbags to ensure the system is responsive before starting to monitor it for a malfunction or deployment the application the airbag system as a metric to detect collisions however the system also relies on other data such as extreme deceleration without the activation of brakes to detect accidents where the airbags fail to deploy.

2.1.1.1 Advanced OBD Sub-mode

In this Sub-mode, the SRoM performs advanced functions besides standard functions. This sub-mode is made available for midgrade and high-end vehicles carrying a larger number of sensors than the one drive clean act enforces.

2.1.1.2 Pothole detection

With the use of the CAN network, the sensors available on the vehicles struts can be used to detect compression level. If a single wheel experience decompression in its struts before going back to its normal position, this points to the dipping that occurs when a wheel passes over a pothole. Hence, the system flags the event as a pothole event.

2.1.1.3 Dangerous driving

f. Rollover

The event of a rollover can be detected if the vehicle, for a short time has its wheels in the air allowing all the wheels struts to be fully decompressed signifying that the cars' wheels are not in contact with the road. This is coupled with all the wheels individual speed sensors registering increase in speed results from the lack of friction that allow them to spin faster.

g. Stunt driving

During the process of stunt driving, the driver balances the vehicle on two wheels, during that period the dongle detects two wheel struts fully decompressed, while the other two opposite to them are over compressed, while GPS and speedometer readings confirm continuing motion signifying stunt driving.

In the events involving willful driver unlawful behavior the system using differing OBD dongles can perform more tasks to limit and report such behavior.

h. Young driver

In each driving cycle the number of over-speeding instances is collected and the parent is alerted after a predetermined number of instances in case of any stunt driving a message is sent with the current location the event took place.

i. Repeat offender

During each driving cycle speed infractions are monitored for both frequency and magnitude with varying restrictions placed on a case-to-case basis. The server is updated with each location of the infraction along with vehicle type and number and police enforcement agencies are alerted based on preset thresholds.

j. Normal

The driver is alerted of the current speed limit and the degree of over-speeding using visual and audio alert. No report is forwarded to the server. While this may increase the chances of dangerous

behavior, privacy concerns can negatively impact the overall crowdsourcing pool rendering the system ineffective.

3.3.3 Pedestrian/passenger mode

This mode of operation the application requires human interaction, and offers ease of use in the process of reporting events by the user, thus allowing for a higher number of participants, to accomplish this goal the interface was simplified to shorten the time required to perform a successful report, the interface is comprised of a list of events to choose from after which the user is sent to another screen where they are promoted to either add a video, or picture to their report, or simply share their GPS location alone. The location, event type, and optional media files are then uploaded to the server.

Users are registered with the server before getting permission to post reports to ensure accountability the physical address of the phone is used when registering, this allows for black listing repeat offenders, who report false information or attach inappropriate media.

3.4.2 SRoM Server

In this section we explain the processes performed by the SRoM Server. Once an event is reported by a driver or a pedestrian/passenger the following processes are performed:

1. Update related event cluster: Once a number of events within a constrained physical area are present a cluster with a unique ID is created and further events of the same type after are amended to the current cluster.
2. Purging old events: Once an event has been flagged as resolved confidence points are distributed to the users involved and the event is then deleted
3. Storing Data: Updated list of users, respective rating, MTO alert areas, and unaddressed road issues are stored into the SRoM database.

These processes are event triggered allowing the SRoM server to function without the need for a continuous script constantly performing tasks. The SRoM server consists of a relational database

that is used to store all received event data. The database groups each cluster of event confined within a limited space as one event with the location centralized to the one corresponding to the majority of reports.

Reports are weighted by two factors: the quality of smartphone, and time delay until the first successful GPS solicitation after the event. Events pertaining to road condition are held for a period of time based on the severity of the event and the amount of expected traffic on the road this is used to increase the number of reports allowing for better accuracy before sending the location, event type, and severity to road technicians.

In the case of emergency events such as accidents and rollover the reports are sent immediately to EMS and the police, in the case of dangerous driving, the events are only reported under two circumstances the first if the driver is a repeat offender in this case any dangerous driving exceeding a preconfigured time threshold is reported, if the user is a minor with a parental watch in place only the parents are alerted to the infraction. Regions with repeated reports of stunt driving are marked and reported to police to better monitor these regions.

3.3.3 SRoM interactive Map

The interactive map allows users to view the collection of events in their region thus allowing them to better plan their tip and avoid bad road conditions. The interactive map has two modes of operation: heat map and report map. In the heat map mode, the system creates a list of events pertaining to road conditions only and uploads them to the map in clusters marked by circles radiating further for event condensed regions.

This allows users to better judge over all road health in the regions they wish to visit and avoid bad sections of road while traveling. The normal mode has more functions as it allows the users to click events and report events on the map directly by dropping a pin on the location they wish to report. After which the user is solicited for information regarding the event using a pop up page.

Users are allowed to view media relating to reported events by clicking preexisting pins generated by active reports and other interactive map users.

A special access interactive map is for use by governmental agencies; in this mode technicians can confirm reported locations by tapping them and deleting or confirming them. The interactive map also allows map operators to monitor the media reported for spam and unacceptable posts.

Chapter 4

Prototype Implementation

In this chapter we describe a prototype we developed that emulates the SRoM system. The goal of the SRoM application is to demonstrate the effectiveness of crowdsourcing for road and traffic monitoring purposes. The prototype implementation involves prototype architecture, prototype interface comprising from a smartphone with Bluetooth and 3G data connectivity, an OBD dice device with SD logger and Bluetooth capabilities, a Vida program capable of reading Volvo manufacturer data stacks, a computer, and a google map engine subscription to test the systems capabilities to detect road artifacts passively, detect hazardous driving, and active reporting.

4.1 Prototype Architecture

Our Prototype architecture emulates the original systems architecture. The components of the prototype are:

1. Samsung galaxy S3 running android OS v4.0.4.
2. Android Smartphone SRoM application as mentioned in Chapter 3.
3. Google map engine API to emulate interactive map.
4. OBD dongle to allow access to manufactures sensors stacks required to perform event detection. DICE (Diagnostic Communication Equipment) adapter from Volvo (Volvo S60 sedan, 2009) is used to provide the functionality of advanced OBD of a high-end light vehicle, and OBD from BMW (BMW li728, 2000) is used to resemble low-end light vehicle.
5. Software application is used to decrypt the data stacks of each sensor and display it in readable table format. VIDA (Vehicle Information and Diagnostics for Aftersales) application is used from Volvo to serve this purpose.

6. SRoM server: we used a standard i7 Lenovo desktop, 4GB ram, 2.7 GHz processing power to act as a data collector, and minor processing enough to perform the testing.

4.1.1 Hardware

Volvo vehicles' manufacturer has a dedicated OBD device called the DICE. DICE is installed in all vehicles manufactured by Volvo and is capable of adapting to each vehicles specific module setup. It collects data from each module such suspension position, steering wheel angle, and individual wheel speeds. However, the data collected is not instantly readable and requires some processing to allow for use in detection. The dice machine allows for two forms of communication USB and Bluetooth; we will rely on Bluetooth communication between the DICE OBD dongle and the smartphone to give freedom of movement while testing.

The DICE device functions in the same manner as a normal OBD dongle it is plugged into the OBD port under the steering wheel shaft and is paired to the smartphone of computer based on the usage required, While the project expects a plug and play approach for installation, it requires a decryption to translate the stacks of non-standard data from the OBD port, which would require an extensive amount of time to gather and compile.

Figure 4.1 DICE Device

The dice allows live streaming or logging for data during test cycle runs, we will use this functionality to mimic the behavior expected if a normal OBD device was available to read the vehicles data directly.

4.1.2 Software

The VIDA application is the main tool used by dealerships to troubleshoot Volvo vehicles, it accompanies the DICE device, and runs as a web applet. This application reads and recognizes vehicles via decrypting the vehicles VIN number and adjusts the stack decoding to the vehicle year and model.

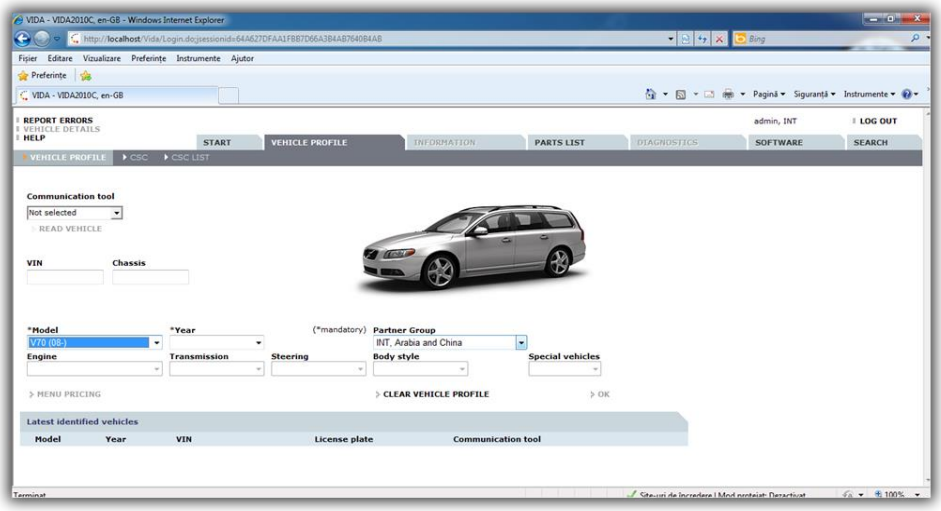

Figure 4.2: VIDA Application

After the initialization and connection is established, the user can access the various components of the vehicle and check their data in the Figure 4.2 the technician is checking for errors in the specific car module to troubleshoot.

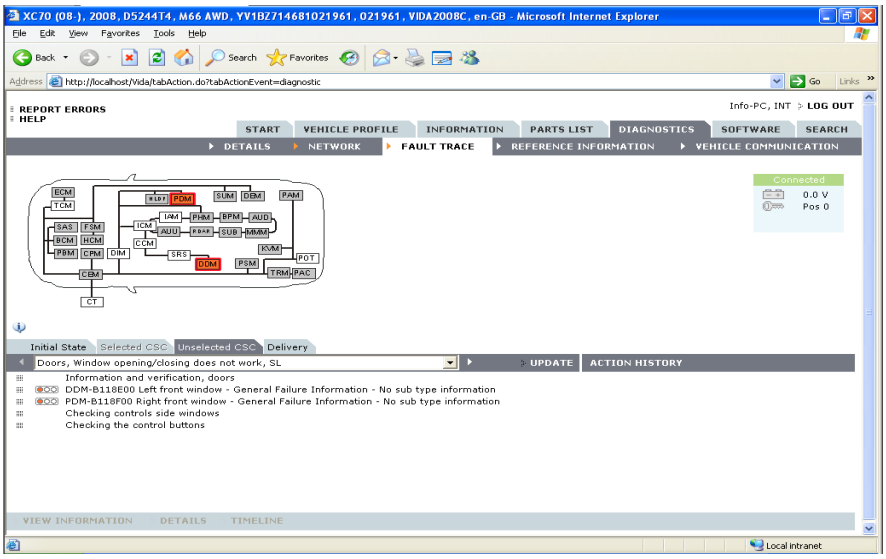

Figure 4.3: Modules for DTC data

The application can access the history of the vehicle, flagged events such as collisions, engine misfires, and failed car modules. The application can also provide access to useful live data, such as individual wheel speed, steering wheel angle, and suspension position; engine RPM, vehicle speed, fuel to air ratio and outside temperature. Android external libraries

As aforementioned, the open source nature of Android provides access to preexisting codes that help achieve our goals in the prototype. We use the Android OBD Reader library [36]. This library allows the program to connect to an ELM327 Bluetooth OBD dongle. It also allows the smartphone to read generic data from the OBD such as Engine RPM, running status, and airbag deployment flag status.

4.1.3 Programing environment

To program the android device, the android studio program was used to facilitate the process of creating and updating the programs interface with more ease. The program was made for jellybean or higher android operating systems to make use of the new libraries available to communicate with the of the shelf OBD dongles.

4.2 Prototype Real World Operation

A SRoM user downloads the application to his smartphone and places the smartphone inside the NFC cradle. The SRoM application starts by requiring the user to swipe left or right from the home screen to choose the form of operation; interactive mode for pedestrians/passengers, or passive mode for drivers. Once the user picks a mode of operation he is met by either of the screens shown in Figure 4.4.

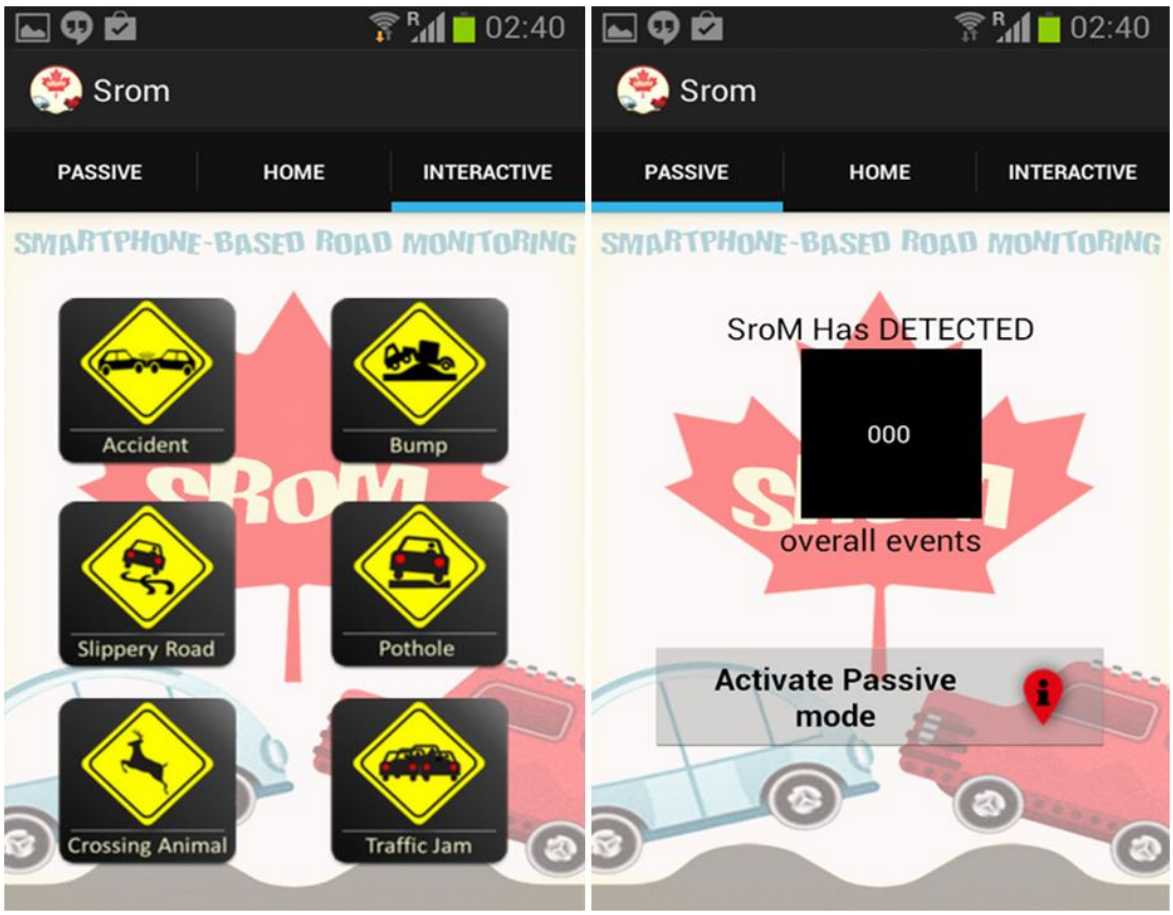

Figure 4.4 SRoM system active and passive home screens

Once a mode has been activated, the server verifies if the current user has an assigned ID or not. If not, new ID is assigned. If the user selects passive (driver) mode, the server solicits the

applications for its current location to initialize the training phase. Afterwards, the server responds by providing a list of event locations, which may include bridges, a number of potholes, and other relevant events. If the active mode is selected, the SRoM system verifies that the GPS sensor is on for the sake of correctness of event location reporting.

In case the Smartphone is paired with the OBD device, two modes of operations can be engaged: Direct simple OBD-Smartphone operation, or OBD logged-Smartphone operation via VIDA.

4.3 Interactive map

To simulate the functions of the SRoM system we make use of the Google engine API, as it allowed for the use of heat map and pin map modes. While such use usually requires licensing to use, Google offers free usage to universities and educational institutes. The map engine is limited in complexity, for the purposes of the project the map is supplied using PHP script an xml page using approved format. The heat map shows cluster location properly, however the radiation is not limited to streets

Chapter 5

Performance Evaluation and Analysis

In this chapter we evaluate the performance of the SRoM system. To evaluate the prototype we perform a number of scenarios involving the various modes of operation; *i.e.* OBD assisted, and Smartphone only. The tests measure a number of metrics to show the systems performance in performing its various tasks such as latency time of detection, location accuracy, and detection rate. While the system is setup to handle a multiplicity of events, we emphasize pothole detection. To perform the test scenarios we installed the program on a Samsung galaxy S2 android device, and made use of the DICA OBD reader and VIDA program to compensate for the lack of a traditional OBD device. The SRoM server used was setup on a Lenovo desktop device with 2.3 quad core processor, and Random access Memory (RAM) of 4 Giga bytes. The Operating system used is Windows 7 professional with an FTP server FileZilla to assist in data acquisition. By evaluating the prototypes various functions we aim to show the systems attributes: interactive reporting, passive reporting, OBD assisted, interactive map. In each scenario performed we outline the expected results the required input and the actual results using screenshots.

5.1 Performance metrics

Pothole detection rate: We use this metric to evaluate the number of true positives detected by the system versus the number of false reports made by the varying systems modes used.

$$PotholeDetectionRate = \left| \frac{DetectedEventes - TotalPotholeEvents}{TotalPotholeEvents} \right|$$

Detection time latency: We use this metric to evaluate the time lapse between the time the event occurred and the time the system detected and reported the event to the server.

$$DetectionTimeLatency = RealEventTime - ActualDetectionTime$$

Location accuracy: We use this metric to calculate the overall accuracy of our location when compared to actual coordinates. We have two modes of location detection. The first GPS reported and relies on live GPS reading from either the Smartphone or OBD sensors. The second is GPS predicted where the system performs a location prediction based on past GPS readings coupled with sensor data. We use this metric to affirm the rate of error in location when compared to a standing GPS at the same location.

Calibration accuracy: We use this metric to calculate the system's ability to correctly calibrate sensor sensitivity to match the systems global sensor sensitivity.

$$SensorAccuracy = \left| \frac{SensorAccuracy_{manual} - SensorAccuracy_{automated}}{SensorAccuracy_{manual}} \right|$$

5.2 Test environment

To properly test the systems various modes of operation we require a number of unique artifacts in the test environment including speed bumps, potholes of varying types, a long stretch of road, and a successive number of turns. Figure 5.1 shows the first test location. This location was chosen due to the presence of potholes speed bumps and a number of turns all within a small area.

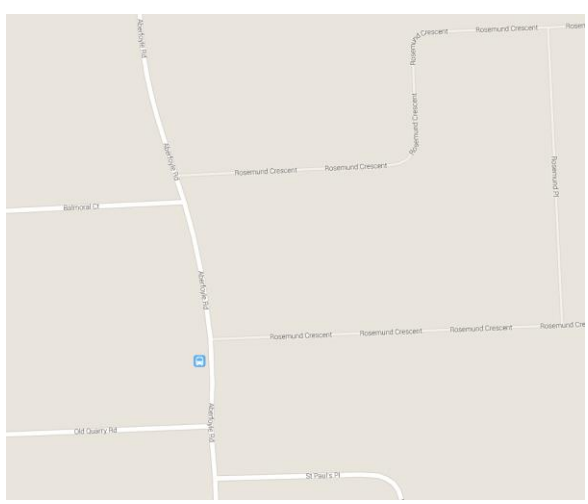

Figure 5.1: Test location #1 Aberfoyle Road and Portsmouth Road.

Figure 5.2 shows the second location containing a lengthy stretch of road with the possibility to lane switching allowing for the testing of over-speeding, weaving and hard stopping.

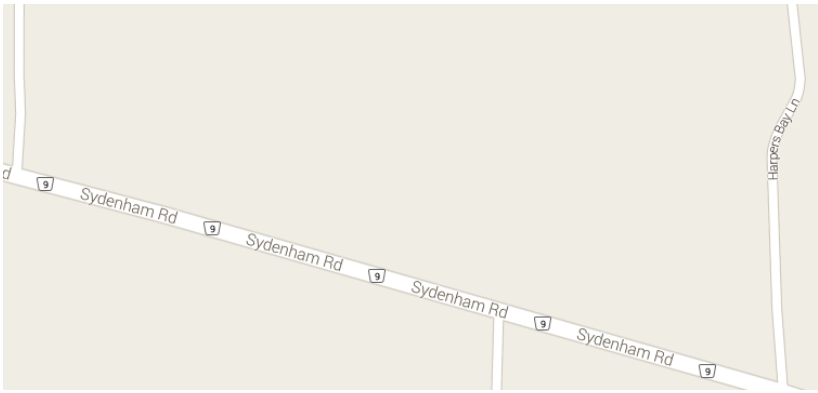

Figure 5.2: Test location 2 Sydenham Road between Sunnyside Road and Mildred Street

5.3 Scenarios and Experiments

Scenario 1: OBD pairing

In this test we check the programs ability to detect and operate in OBD mode to setup for this test we installed an OBD device into the vehicle turned the ignition on and kept it in that position, paired the OBD dongle as a Bluetooth device.

Results: The snapshot in Figure5.2 shows confirmation of the OBD application pairing the application will start using the sensory data supplied by the OBD to increase the accuracy of the system.

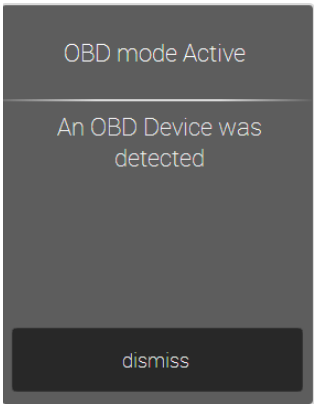

Figure 5.3: application detection of paired OBD device

Scenario 2: calibration accuracy

In this scenario we test the system's ability to accurately adapt a new device sensor sensitivity to match the current system sensitivity. Based on the systems specifications, the smartphone and OBD accelerometer sensors must generate the same sensory signature. To be able to generate the same signature while using varying types of vehicles the sensors must adjust, during the auto calibration phase mentioned in Chapter 3.

Execution Guidelines:

1. Manually set the Sensitivity of the accelerometer to 0.8g (predetermined to guarantee high detection rate of road artifacts).
2. Perform a signature collected specified run.
3. Clear the smartphone accelerometer settings and calibration flag.
4. Perform 1 lap with alternating entry direction at a constant speed of 40km/h (the speed limit in test location one)
5. Wait for a successful calibration record the setting and clear it.
6. Perform previous test with varying speeds.
7. Repeat the above steps for OBD calibration.

Expected Results:

1. The calibrated sensor setting must match the manual sensor sensitivity and remain within the error margin at same speed.
2. During the varying speed calibration the sensor sensitivity will fluctuate but remain within the threshold of acceptable readout.
3. OBD calibration will match more accurately the manually set sensitivity.

Results: For the first test using the smartphone the generated Sensitivity values were within range of the original setting. Small fluctuations attributed to the slight variations in the angle and speed of entry we observed. Figure 5.2 shows the sensitivity information using the same vehicle and same speed. All readings fall within range of the manual sensitivity setting of 0.8.

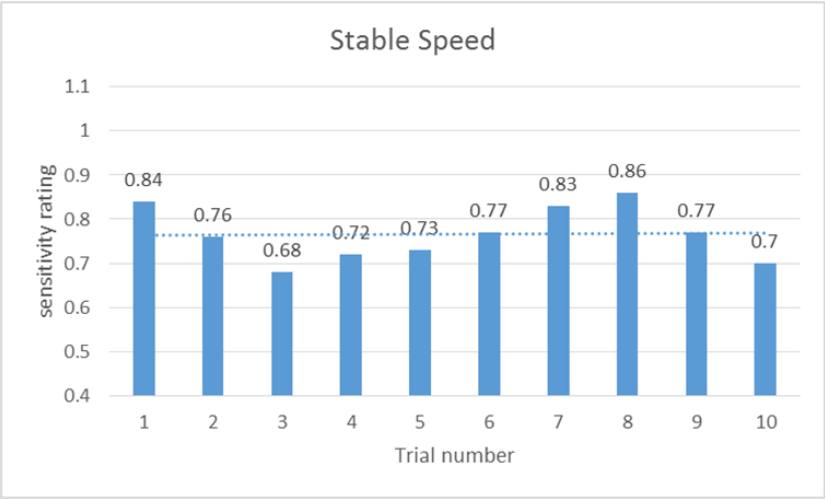

Figure 5.4 calibration results constant speed.

While the auto calibration data shown in Figure 5.4 fluctuates, these fluctuations are limited in range. Such small fluctuations are due to the fact that the vehicle's speed of entry when passing over the road artifacts used to calibrate is not fully controlled.

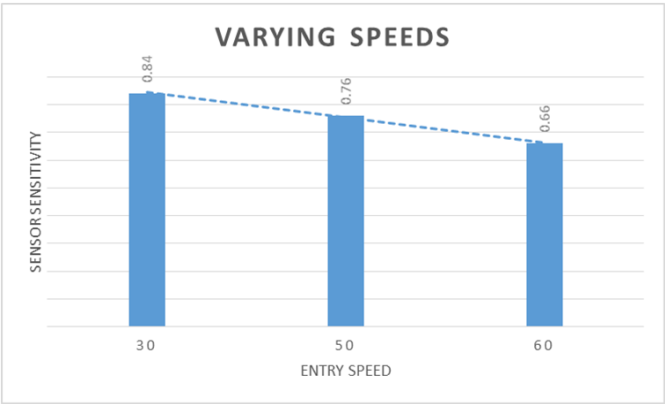

Figure 5.5 calibration results variable speeds

In Figure 5.5 shows the average sensitivity of 5 runs at 30, 50, and 60km/h. The results show the direct correlation between the data generated and the speed of entry however the fluctuations effect

is limited and does not hinder the systems capability. However during the testing phase the trials at 20km/h failed to produce a successful calibration as the speed of entry was insufficient to trigger an event detection frequently enough to complete the process.

In Figure 5.6 we show the data collected by the OBD. In this Figure the presence of the OBD gives access to the speedometer readings allowing fluctuations to be addressed.

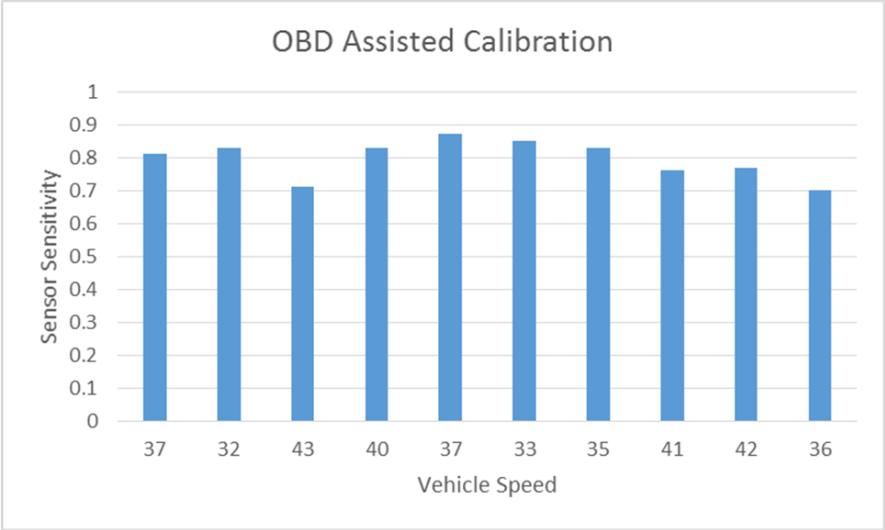

Figure 5.6 Calibration using OBD Speedometer and accelerometer data

In instances where the vehicular speed was below the set speed generated a weaker sensory response forcing the calibration phase to over compensate allowing the sensitivity to go above the manually set selectivity and reaching values such as 0.88g. While sensitivity remains within the allowed threshold, it shows the potential of failure based on driver behavior inconsistences, which

Scenario 3: Pothole Detection

In this scenario we deal with road events detection. The vehicle involved performs a number of laps around a designated area shown in Figure 5.1. Initially the pothole locations are collected based on pedestrian/passenger smartphone GPS readings and the types of the potholes were classified into three categories: cluster, large, and small potholes. This scenario is divided into two sub scenarios based on the mode of operation used either smartphone only or OBD assisted.

Execution Guidelines:

1. Perform two laps through the test track activating each detection method listed below at 50km/h after each lap take the reverse route.
   a. Threshold.
   b. Freefall, differential.
   c. Majority vote based test using the three above tests running simultaneously.
2. Repeat step 1 with the following speeds 25 and 65km/h.
3. Repeat step 1 and 2 using the OBD mode.

Expected Results:

1. The accuracy of the freefall test and differential test to be significantly more than the threshold test specially at low speeds.
2. The OBD mode to have much higher accuracy at all speeds when compared to the smartphone only mode.

Execution Results:

1. In this section we compare the results of the multiple pothole detection mechanisms used in smartphone only mode.

   a. Constant speed.

   The performance results of the three detection mechanisms under a constant speed are shown in Table 5.1. As expected, the threshold test being the most simple is the most reliant on accurate sensor sensitivity, while Zero-G shows the best detection rate. As the table shows the three pronged test offers the best accuracy with comparable number of false positives.

Table 5.1: Pothole Detection Comparison

Type	Total	Threshold	Zero-G	Threshold difference	Three pronged test
Large pothole	6	6	6	6	6
Small pothole	4	2	2	2	4
Pothole cluster	4	1	3	1	3
False positive	-	4	2	1	1
Total	14	9	11	9	13

b. Variable speeds

The performance results of the three detection mechanisms under the three speeds are shown in Table 5.7 as expected, at low speeds the testing mechanism starts to fail, the threshold based test suffering the most loss of accuracy due to its simple approach to detection, while the Zero-G test remains impervious to the low speeds the vehicles are going during event detection. As shown in Figure 5.5

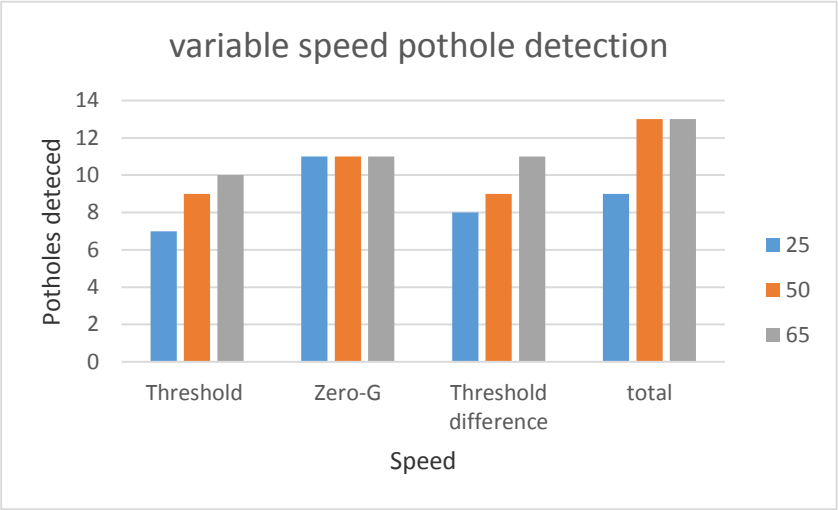

Figure 5.7 Pothole Detection Variable Speed

2. In this section we compare the results of the variations of pothole detection using both the smartphone and OBD dongle

   a. Constant speed

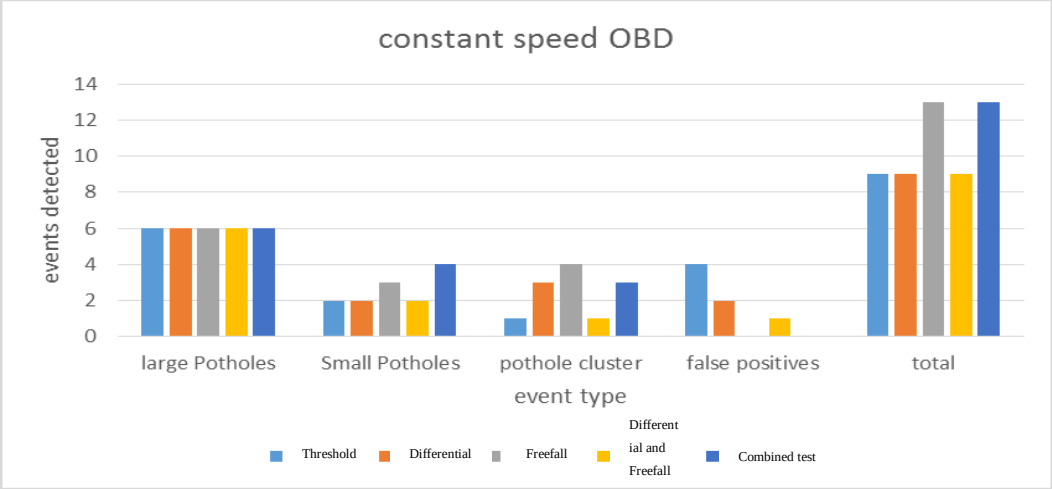

Figure 5.8 pothole detection constant speed variable tests

As shown in the Figure 5.8 above, the OBD's impact on the systems accuracy is great, it lowered the number of false positives dramatically, when using the advanced OBD we were able to detect all events with only one false positive.

   b. Variable speed

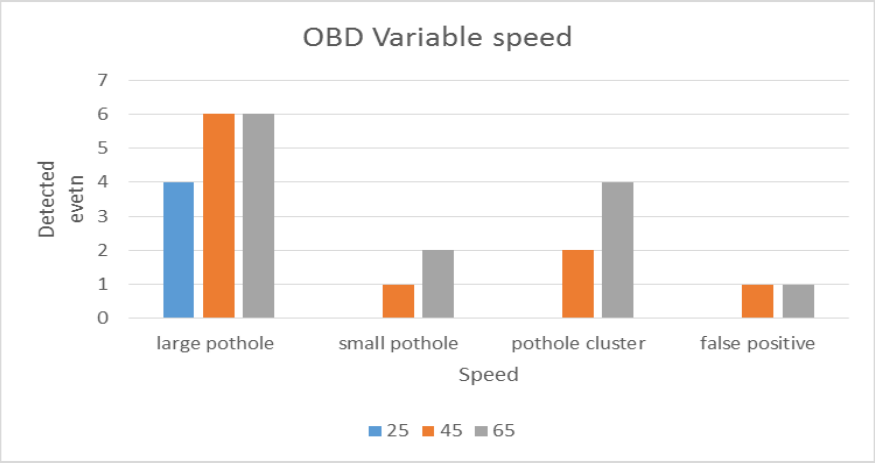

Figure 5.9 Pure OBD Pothole Detection

Scenario 4: Location prediction

In this scenario we test the system's ability to predict location in the blind periods between GPS reads, the duration of the test is 2 minutes, the vehicle will perform turns randomly between the start location and end location.

Execution Guidelines:

1. Install the application after adjusting the movement speed of GPS prediction algorithm.
2. Await a GPS reading, turn off the GPS sensor.
3. Perform 5 laps inside the test track at each speed 55, 45, 25 km/h.
4. Turn on GPS location to get read to match with prediction algorithm.

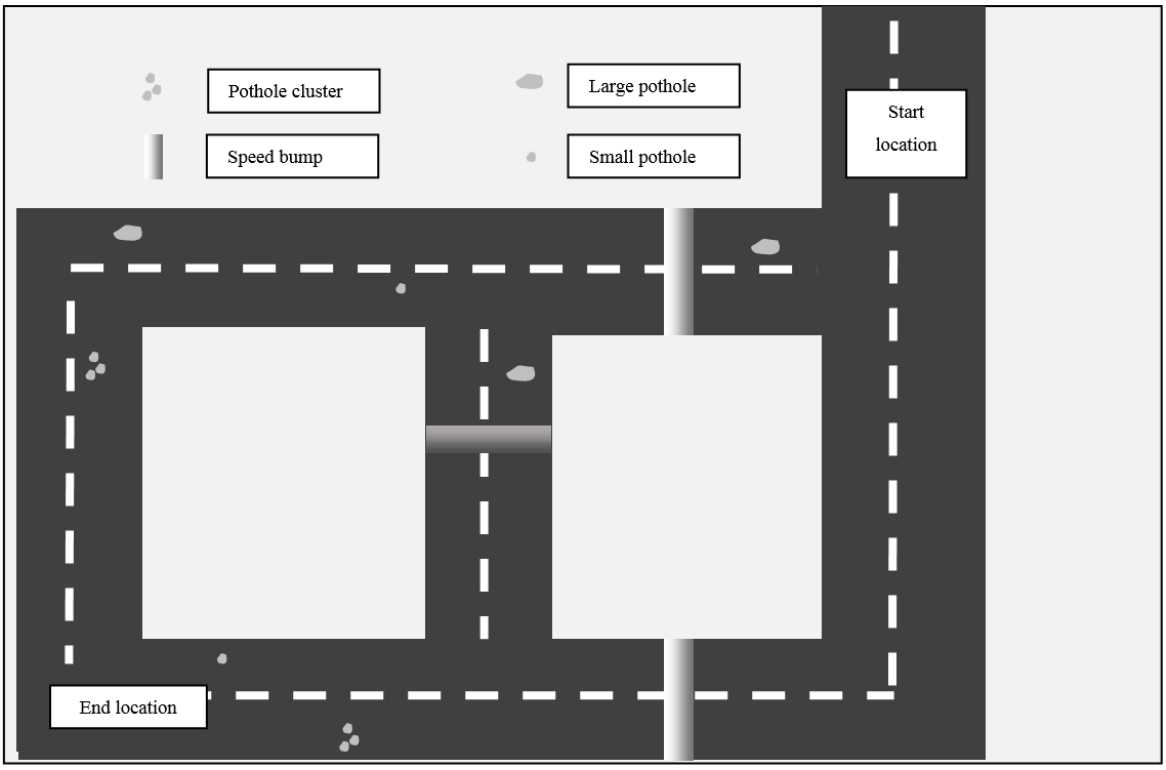

Figure 5.10 location prediction start and end locations

Expected Results:

While the speed of predictions is adjusted to match the vehicles speed however with lower speeds the ability to detect turns is lowered causing higher error.

Execution Results:

3. The table below shows the average of distance between the GPS reading and the GPS prediction algorithm

Table 5.2: Prediction Error Margin

Speed	Lap 1	Lap 2	Lap 3	Lap 4	Lap 5
55	3m	4m	5m	4m	3m
45	4m	5m	5m	6m	5m
25	7m	9m	20m	4m	9m

The performance results of the algorithms under a constant speed are shown in Table 5.1 as expected, the threshold test being the most simple is the most reliant on accurate sensor sensitivity, while Zero-G shows the best detection rate. As the table shows the three pronged test offers the best accuracy with comparable number of false positives.

Scenario 5: Speeding

In this test case the application is placed in young driver mode, repeat offender, and normal mode to ensure the correct response based on detected speeding.

Execution Guidelines:

1. Application with OBD in passive mode activated.

2. Speed over the global threshold of 50km per hour.

Expected Results:

1. In case of young driver: an email and a warning message of the same material must be issued to the driver and parents or supervising guardian.
2. In case of repeat offender: generate timed message to stop illegal behavior to alert the authorities.
3. In case of normal mode: message regarding the areas speed limit.

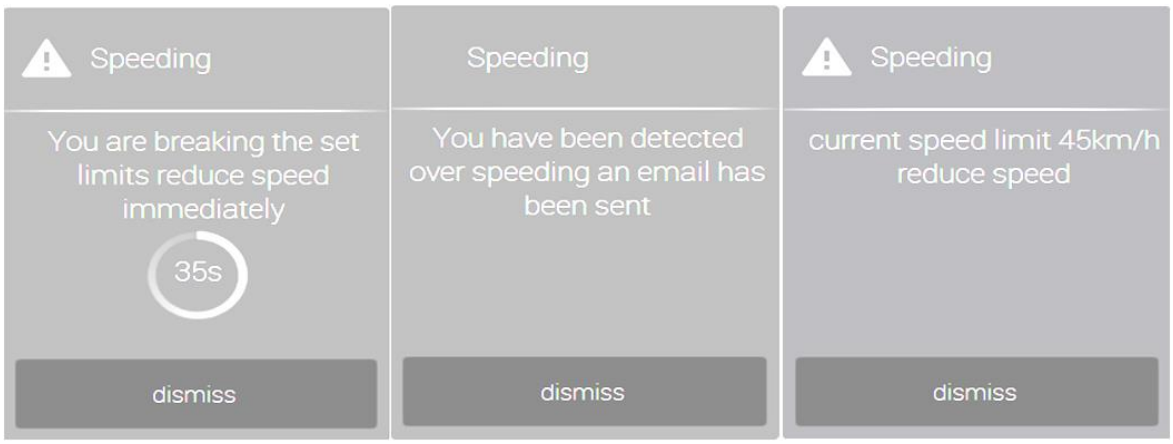

Figure 5.11 Speeding repeat offender mode, young driver, and normal

Results: The test is detected by removing the application and installing with the varying modes and repeating the above steps as Figure 5.3 shows the applications had successfully detected and appropriately respond to speeding events. In the case of the repeat offender a timed warning is given after which an event is sent to the server, in the case of a young driver an email is sent to the parents in case of speeding, normal drivers are only warned as their use of the program is voluntary.

Scenario 6: Accident Detection

The detection of the accidents is done in the same way as the detection of other events using the accelerometer and OBD. Once an accident is detected a timed message is displayed, the user must

interact with the message to interrupt incident response procedure. In the case of static GPS readout once the countdown expired, the application immediately sends an email to the preprogramed address with the GPS coordinates and the message that an accident took place. To mimic the behavior of an accident a class was used to feed pseudo sensory data with both static and continuous GPS readings to ensure correct event response.

Execution Guidelines:

1. Run program with tester class.

Expected Results:

1. In case of continuous GPS readings the message will be removed after a timer.
2. In case of static GPS readings the system will issue an accident alert.

Results: an alert was issued to the driver in case of an accident to be dismissed, if not dismissed the program sends a distress message to the authorities.

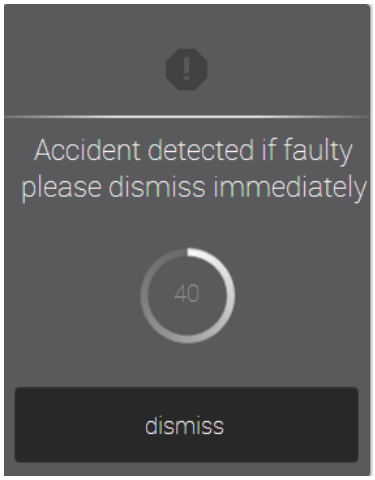

Figure 5.12 Accident alert

In case of a higher event of deceleration the application will forgo the wait time and immediately send a report, this is due to the improbability of a faulty accelerometer reading resulting from human interference.

Chapter 6

Conclusion and Future Work

In this thesis we describe how we created a system and a prototype capable of using smartphones and OBD devices to turn a normal vehicle into a smart vehicle capable of detecting road events and hazardous driving. The use of the OBD-Smartphone assisted systems can help users address car problems and road problems before they become detrimental to the driver's and occupants' health and the vehicle's usability.

OBD is normally used for regular maintenance to address specific problems, but OBD can be instrumental to immediately assisting in addressing other issues related to road, traffic, and driving pattern monitoring. Best accuracy can be achieved with the presence of the OBD dongles.

OBD dongles are getting cheaper to buy and easier to install for the average user.

SRoM introduces a good accident reporting mechanism, but it can be improved by adjusting the reporting mechanism to allow for streaming media. This may allow incoming EMS services to locate the event and the extent of the damage via the media stream. It may be beneficial to employ a dedicated line or hotspots along the highway and other problem areas to allow for such a stream to take place since android permissions do not allow for application-generated calls to emergency numbers.

The proposed system provides a proof of concept for using smartphone as a sensory node within vehicles assisted with OBD. However, full development requires a number of expansions before reaching the state of public usage.

The most important aspect is the ability to decode OBD manufacturer's stacks. It was difficult during the development and testing of SRoM to make use of the data collected from the DICE OBD device, mainly due to the need to pass the data through the VIDA program first. All vehicles

can be identified by a VIN number allowing us to identify each vehicle make and model allowing us to use the specific approach to decode the stacks based on each manufacturer's requirements, which accomplishes a task similar to the VIDA operation. This will allow us to solicit the server for the car-specific dictionary whenever driving. However this will require the car manufacturer's assistance in making the data public. The increase in accuracy collected for crowd sourcing is expected to increase significantly once this data becomes available providing a more accurate detection of events.

The Bluetooth dongle poses a security risk for the driver, since car alarms in the past were disabled using dongles that were left active. Changes must be made to restrict the OBD dongles' access to the vehicle by changing their status to only reading data. Participation is expected to increase significantly once these changes are made.

While using the application in high-end vehicles, it might be more beneficial to run the program using the car's built-in computer as it offers more privacy for the user allowing the data to remain within the vehicle. This may increase the price, albeit not significantly.

It may be beneficial to make some events available publicly through services such as Waze; an event detection followed by a number of reports of traffic congestion can alert incoming drivers to take alternate routes. This alert may remain until an EMS service has cleared the event. Providing the information on such publically accessible services allows for better traffic management, and if deployed properly can also facilitate shorter arrival times for EMS services.

The prototype's interface was made in a predefined list of events, but it can be further developed by allowing the user or traffic authorities to update and alter the list content to become better suited for the region. For example, we might introduce sleet and black ice alerts during the winter and construction alerts during the summer.

Bibliography

[1] Canadian Motor Vehicle Traffic Collision Statistics (2011). Retrieved May 2014, from http://www.tc.gc.ca/media/documents/roadsafety/TrafficCollisionStatisitcs_2011.pdf

[2] European Automobile Manufacturers Association Average Vehicle Age Statics (2010). Retrieved May 2014, from http://www.acea.be/statistics/tag/category/average-vehicle-age.pdf

[3] A. Benslimane, "Optimized dissemination of alarm messages in vehicular ad-hoc networks (VANETs)," High Speed Networks and Multimedia Communications, pp. 655-666, 2004.

[4] European Environment Agency. Average age of road vehicles. URL http://www.eea.europa.eu/data-and-maps/figures/average-age-of-road-vehicles. Last visited 2013-08-19.

[5] ELM Electronics. Elm Electronics Products - OBD ICs. URL http://www.elmelectronics.com/obdic.html#ELM327. Last visited 2013-09-02.

[6] Gary "Chunky Ks" Briggs, OBD GPS Logger, URL http://icculus.org/ obdgpslogger/. Last visited 2013-09-03.

[7] Jonas Gehring. GraphView Library. URL http://www.jjoe64.com/p/graphview-library.html. Last visited 2013-09-13.

[8] Instituto Nacional de Estad´ıstica. Relaci´on de municipios, provincias, co- munidades y ciudades aut´onomas y sus c´odigos, URL http://www.ine.es/ jaxi/menu.do?type=pcaxis&path=/t20/e245/codmun&file=inebase. Last visited 2013-10-03.

[9] Relacionarel C´odigo INE con su C´odigo Postal,
http:// switchoffandletsgo.blogspot.com.es/2013/04/relacionar-el-codigo-ine- con-su-codigo.html.
Last visited 2013-09-04.

[10] C.-W. You, M. Montes-de-Oca, T.J. Bao, N.D. Lane, H. Lu, G. Cardone, L. Torresani, and A.T. Campbell, "CarSafe: a driver safety app that detects dangerous driving behavior using dual-cameras on smartphones," ACM Conference on Ubiquitous Computing, pp. 671-672, 2012.

[11] K.I. Aoyama, “Universal traffic management system (UTMS) in Japan,” IEEE Vehicle Navigation and Information Systems Conference, pp. 619-622, 1994.

[12] M. Arikawa, S. Konomi, and K. Ohnishi, “Navitime: Supporting pedestrian navigation in the real world,” IEEE Pervasive Computing, vol. 6, issue 3, pp. 21-29, 2007.

[13] G.S. Tewolde, "Sensor and network technology for intelligent transportation systems," IEEE International Conference on Electro/Information Technology, pp. 1-7, 2012.

[14] M. Liu, "A Study of Mobile Sensing Using Smartphones," International Journal of Distributed Sensor Networks, 2013.

[15] R. Bishop, “Intelligent vehicle technology and trends,” 2005.

[16] S. Biswas, R. Tatchikou, and F. Dion, “Vehicle-to-vehicle wireless communication protocols for enhancing highway traffic safety,” IEEE Communications Magazine, vol. 44, pp. 74–82, 2006.

[17] S.S. Chawathe, “Using dead drops to improve data dissemination in very sparse equipped traffic,” IEEE Intelligent Vehicles Symposium, pp. 962-967, 2008.

[18] S. Eichler, C. Schroth, T. Kosch, and M. Strassberger, “Strategies for context - adaptive message dissemination in vehicular ad hoc networks,” IEEE International Conference on Mobile and Ubiquitous Systems: Networking & Services, pp. 1-9, 2006.

[19] P. Ekler and T. Lukovszki, “Similarity distribution in phonebook-centric social networks,” IEEE International Conference on Wireless and Mobile Communications, pp. 359-364, 2009.

[20] D. Work and A. Bayen, "Impacts of the mobile internet on transportation cyberphysical systems: traffic monitoring using smartphones," National Workshop for Research on High-Confidence Transportation Cyber-Physical Systems: Automotive, Aviation, & Rail, pp. 18-20, 2008.

[21] C. Campolo, A. Iera, A. Molinaro, S.Y. Paratore, and G. Ruggeri, “SMaRTCaR: An integrated smartphone-based platform to support traffic management applications,” Vehicular Traffic Management for Smart Cities, 2012.

[22] P.S. Henry and H. Luo, “Wi-Fi: what’s next?” IEEE Communications Magazine, vol. 40, no. 12, pp. 66-72, 2002.

[23] J. Howe, “Crowdsourcing: how the power of the crowd is driving the future of business. Century,” 2008.

[24] M. Picone, M. Amoretti, and F. Zanichelli, "A decentralized smartphone based traffic information system," IEEE Intelligent Vehicles Symposium, pp. 523-528, 2012.

[25] U. Lee, J. Lee, J.S. Park, and M. Gerla, “Fleanet: a virtual market place on vehicular networks,” IEEE Transactions Vehicular Technology, 2010.

[26] R. Araujo, A. Igreja, R. de Castro, R.E. Araujo, "Driving coach: a smartphone application to evaluate driving efficient patterns," IEEE Intelligent Vehicles Symposium, 2012.

[27] J. Paefgen, F. Kehr, Y. Zhai, and F. Michahelles, "Driving behavior analysis with smartphones: insights from a controlled field study," ACM International Conference on Mobile and Ubiquitous Multimedia, pp. 36-43, 2012.

[28] A.K. Shaout and A.E. Bodenmiller, "A mobile application for monitoring inefficient and unsafe driving behaviour," 1996.

[29] S.M. Tornell, C.T. Calafate, J.-C. Cano, P. Manzoni, M. Fogue, and F.J. Martinez, "Implementing and testing a driving safety application for smartphones based on the eMDR protocol," IFIP Wireless Days, pp. 1-3, 2012.

[30] P. Mohan, V.N. Padmanabhan, and R. Ramjee, “Nericell: rich monitoring of road and traffic conditions using mobile smartphones.” ACM Conference on Embedded Network Sensor Systems, pp. 323–336, 2008.

[31] C. Shi, H.J. Lee, J. Kurczak, and A. Lee, "Routine driving infotainment app: gamification of performance driving," International Conference on Automotive User Interfaces and Interactive Vehicular Applications, pp. 181-183, 2012.

[32] B. Kirk, "Connected vehicles: an executive overview of the status and trends," Globis Consulting, November 2011.

[33] M.T. Sun, W.C. Feng, T.H. Lai, K. Yamada, H. Okada, and K. Fujimura, “GPS-based message broadcasting for inter-vehicle communication,” IEEE International Conference on Parallel Processing, pp. 279–286, 2000.

[34] S. von Watzdorf and F. Michahelles, “Accuracy of positioning data on smartphones,” ACM International Workshop on Location and the Web, pp. 2, 2010.

[35] C. Thompson, J. White, B. Dougherty, A. Albright, and D.C. Schmidt, "Using smartphones to detect car accidents and provide situational awareness to emergency responders," Mobile Wireless Middleware, Operating Systems, and Applications, pp. 29-42, Springer Berlin Heidelberg, 2010.

[36] A. Mednis, G. Strazdins, R. Zviedris, G. Kanonirs, and L. Selavo, "Real time pothole detection using android smartphones with accelerometers," IEEE International Conference on Distributed Computing in Sensor Systems and Workshops, pp. 1-6, 2011.

[37] G. Strazdins, A. Mednis, G. Kanonirs, R. Zviedris, and L. Selavo, "Towards vehicular sensor networks with android smartphones for road surface monitoring," International Workshop on Networks of Cooperating Objects, vol. 11, 2011.

[38] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan, “The Pothole Patrol: using a mobile sensor network for road surface monitoring,” ACM International Conference on Mobile Systems, Applications, and Services, pp. 29-39, 2008.

[39] A. Mednis, G. Strazdins, M. Liepins, A. Gordjusins, and L. Selavo, “RoadMic: Road surface monitoring using vehicular sensor networks with microphones,” Networked Digital Technologies Communications in Computer and Information Science, vol. 88, pp. 417-429, 2010.

[40] P. Mohan, V.N. Padmanabhan, and R. Ramjee, “Nericell: rich monitoring of road and traffic conditions using mobile smartphones,” ACM Conference on Embedded Networked Sensor Systems, pp. 357-358, 2008.

[41] URL http://www.cdn.mto.gov.on.ca/english/traveller/trip/compass.shtml#vds.
Last visited 2013-08-19.

[42] C.E. Perkins and E.M. Royer, "Ad-hoc on-demand distance vector routing," IEEE Workshop on Mobile Computing Systems and Applications, pp. 90-100, 1999.

[43] V. Naumov, R. Baumann, and T. Gross, "An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces," ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp. 108-119, 2006.

[44] G. Pei, M. Gerla, and T.-W. Chen, "Fisheye state routing: a routing scheme for ad hoc wireless networks," IEEE International Conference on Communications, vol. 1, pp. 70-74, 2000.

[45] R. Flury and R. Wattenhofer, "MLS: an efficient location service for mobile ad hoc networks," ACM International Symposium on Mobile Ad Hoc Networking and Computing, pp. 226-237, 2006.

[46] B. Karp and H.-T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," ACM International Conference on Mobile Computing and Networking, pp. 243-254, 2000.

[47] URL http://eurlex.europa.eu/LexUriServ/LexUriServ.do?uri=CONSLEG:1998L0069::PDF Last visited 2013-08-19.

[48] X. Lin, R. Lu, C. Zhang, H. Zhu, P.-H. Ho, and X. Shen, "Security in vehicular ad hoc networks," IEEE Communications Magazine, vol. 46, no. 4, pp. 88–95, 2008.

[49] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan, "The pothole patrol: using a mobile sensor network for road surface monitoring," ACM International Conference on Mobile Systems, Applications, and Services, pp. 29–39, 2008.

[50] H. Kathiriya, N. Kathiriya, and A. Bavarva, "Review on V2R Communication in VANET," International Conference on Innovations in Automation and Mechatronics Engineering, pp. 167-172, 2013.

[52] S.S. Kanhere, "Participatory sensing: Crowdsourcing data from mobile smartphones in urban spaces," IEEE International Conference on Mobile Data Management, vol. 2, pp. 3-6, 2011.

[53] C, Thompson, J. White, B. Dougherty, and D.C. Schmidt, "Optimizing mobile application performance with model–driven engineering," Software Technologies for Embedded and Ubiquitous Systems, pp. 36-46, Springer, Berlin, Heidelberg, 2009.

[54] R.R. Brooks and S.S. Iyengar, "Multi-sensor fusion: fundamentals and applications with software," Prentice-Hall, Inc., 1998.

[56] B. Plana Puig, "Smartphones for smart driving: a proof of concept," unpublished master's thesis for master's degree, Universitat Politecnica de Catalunya, Barcelona, 2013.

[57] URL http://torque-bhp.com/wiki/Talk:Main_Page. Last visited 2013-08-19.

[58] C.C. Robusto, “The cosine-haversine formula,” The American Mathematical Monthly, vol. 64, no. 1, pp. 38-40,1957.

[59] Koch, Christian, and Ioannis Brilakis. "Pothole detection in asphalt pavement images." Advanced Engineering Informatics 25, no. 3 (2011): 507-515.

[60] Rao, Sunder Aditya, et al. "GPSR-L: Greedy perimeter stateless routing with lifetime for VANETS." ITS Telecommunications, 2008. ITST 2008. 8th International Conference on. IEEE, 2008.

[61] URL http://www.cityofboston.gov/doit/apps/streetbump.asp Last visited 2013-09-02.

[62] URL http://torque-bhp.com/ Last visited 2013-09-02.

[63] URL https://www.waze.com/about/dev Last visited 2013-09-02.

[64] URL http://www.inrixtraffic.com/ Last visited 2013-09-02.