This paper is a postprint of a paper submitted to and accepted for publication in Intelligent Transport Systems and is subject to Institution of Engineering and Technology Copyright. The copy of record is available at IET Digital Library - doi: 10.1049/iet-its.2014.0248

A Survey of Smartphone-based Sensing in Vehicles for ITS Applications

J. Engelbrecht, M.J. Booysen, G.-J. van Rooyen, F.J. Bruwer

Department of Electrical and Electronic Engineering
University of Stellenbosch
South Africa
E-mail: {mjbooysen,gvrooyen}@sun.ac.za

Abstract

Road crashes are a growing concern of governments and is rising to become one of the leading preventable causes of death, especially in developing countries. The ubiquitous presence of smartphones provides a new platform on which to implement sensor networks and driver assistance systems, as well as other ITS applications. In this paper, existing approaches of using smartphones for ITS applications are analyzed and compared. Particular focus is placed on vehicle-based monitoring systems, such as driving behavior and style recognition, accident detection and road condition monitoring systems. Further opportunities for use of smartphones in ITS systems are highlighted, and remaining challenges in this emerging field of research are identified.

Keywords: Vehicle safety, driving behavior, remote monitoring, wireless sensor networks, event detection, pattern recognition, reckless driving detection, vehicle telematics, insurance telematics.

1. Introduction

Most modern smartphones have a variety of embedded sensors — typically an accelerometer, gyroscope, light, proximity and magnetic sensor — as well as a Global Positioning System (GPS). These sensors can further be augmented by processing data from the camera and microphone. This richness in sensors enable support for a multitude of sensing applications. An example of such an application is gesture recognition, which is used to answer a call when bringing the phone to one's ear, or paging through a document by the wave of a hand [1, 2]. In a similar way, different activities such as walking, running, cycling and driving can be detected and classified using the inertial sensors of a phone that is carried in a user's pocket [3].

Worldwide more than a million deaths are caused by road accidents per year. The World Health Organization predicts that road fatalities will rise to become the fifth leading cause of death by 2030 [4]. Research done in the United States shows that, in more than 50% of fatal road accidents, unsafe driving behaviors were involved [5]. Although road accidents can be attributed to a variety of factors, reckless driving is one of the major preventable causes.

In the last decade, various vehicle manufacturers and other companies have been developing solutions to monitor a vehicle and its driver's behavior [6, 7, 8]. A problem with these solutions is that they are expensive and there is little incentive for individuals to buy them. These systems are mostly used for vehicle fleets. However, the increasingly ubiquitous presence of smartphones – with their variety of sensors – presents the opportunity to easily implement vehicle monitoring systems on a large scale.

Vehicle monitoring is an attractive sensing application for smartphones. For instance, drivers can be monitored to make them aware of their potentially dangerous driving behavior. Anonymous participatory sensing could also enable identifying areas where accidents are more likely to occur [9]. The authorities could also be notified to investigate extreme cases of aggressive driving.

The omnipresent connectivity of smartphones also allows the implementation of other vehicle monitoring features, such as traffic monitoring, traffic re-routing and accident reporting. Accident detection is possible using only the sensors in a modern smartphone, as shown by White et al. [10]. The swift automatic reporting of road accidents to authorities can prevent fatalities by minimizing the response time of emergency services. Additionally, using a machine-to-machine (M2M) communication platform would allow the redirection of other drivers away from an accident. Notifying drivers that they are approaching an accident scene could also increase their alertness and warn them to slow down, thereby preventing further accidents.

The remainder of this paper is organized as follows: Section 2 provides a brief overview of the current literature on smartphone sensing in vehicles; in Section 3, the limited number of available papers specifically describing vehicle monitoring systems that are entirely implemented on a smartphone are analyzed, reviewed and compared; Section 4 discusses the challenges facing the progress and adoption of vehicle monitoring systems; and Section 5 presents the concluding remarks.

2. Smartphone sensing in vehicles

Machine-to-machine (M2M) communications describes a system where multiple electronic devices communicate autonomously to enable the sharing of information [11]. Among the millions of M2M devices that will be deployed world-wide in the coming years, smartphones will be the most mobile, versatile and powerful devices that can be used as sensors and M2M gateways [12]. Therefore much research has been done on smartphone sensing applications in recent years [12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23].

In this section, a brief overview is given of the current literature on smartphone sensing in vehicles. The existing literature on vehicle sensing can be categorized according to the four types of information that is captured, which is then disseminated in various ways for different applications. The types of data are

- Traffic information, such as the location and movements of other vehicles or pedestrians
- Vehicle information, for example vehicle health telematics
- Environmental information, such as road conditions and weather conditions
- Driver behaviour information, including insurance telematics

The features and goals of the recent projects are listed in Table 1 to provide context of the systems discussed in the rest of this paper.

2.1. Traffic information

M2M platforms have been developed where smartphones are used as sensor gateways in vehicles to support traffic management applications such as detecting congestion and rerouting traffic [24, 25, 26]. The platform described by Ali et al. serves the same purpose, but information is manually entered by users on their smartphones [27].

SignalGuru [29, 13] is a service that uses smartphones to opportunistically detect traffic signals with their cameras and collaboratively share and learn traffic signal schedules. This enables Green Light Optimal Speed Advisory (GLOSA) applications which provide drivers with the schedule of forward traffic signals, allowing them to avoid coming to a complete halt and thereby lowering their fuel consumption. Jam Eyes [28] is an application that uses a smartphone's camera and wifi to detect vehicles around it in a traffic jam, in order to collaboratively calculate the length of a traffic queue.

2.2. Vehicle information

Zaldivar et al. [38] proposed using smartphones as an alternative on-board unit (OBU) in vehicles to access information in the vehicle's electronic control unit (ECU) wirelessly. An ECU is typically accessed through an industry-standard on-board diagnostic connector, known as OBD-II. By connecting an OBD-II-to-bluetooth adapter to the vehicle's controller area network (CAN) bus, a smartphone can gain access to the bus via bluetooth. Automatic accident detection is accomplished by using data obtained from the CAN bus together with the smartphone's GPS and accelerometer. Similarly, Yang et al. [37] developed a smartphone-based diagnostic system for hybrid electric vehicles that also accesses a vehicle's CAN bus through OBD-II.

Table 1: Literature relevant to smartphone sensing in vehicles.

Ref.	Year	Technology	Category	Goal
[12]	2007	mobile phone	traffic information	mobile phones as sensor gateways
[24]	2012	smartphone, external sensors	vehicle information	engine parameters collection from external sensors
[25]	2013	smartphone, external sensors, vehicle ECU	vehicle information	opportunistic transfer of external sensor- and CAN bus data
[26]	2011	smartphone	traffic information	intelligent driver guidance tool
[27]	2012	smartphone	traffic information	road incident and traffic crowd-sourcing
[28]	2012	smartphone	traffic information	traffic queue length detection
[13] [29]	2011	smartphone	traffic information	traffic signal detection and learning
[30]	2009	smartphone	driver behavior information	lane departure warning system
[9]	2011	smartphone	driver behavior information	aggressive driving detection
[31]	2012	smartphone	driver behavior information	driving style characterization
[32]	2010	smartphone	driver behavior information	drunk driving detection
[33]	2012	smartphone	driver behavior information, environmental information	advanced driver-assistance system
[34]	2008	smartphone	traffic information, driver behavior information, environmental information	road and traffic condition monitoring
[35]	2012	smartphone	traffic information, environmental information	pothole detection and notification
[36]	2011	smartphone	traffic information, environmental information	pothole detection and notification
[37]	2013	smartphone	vehicle information	hybrid electric vehicle diagnostics
[38]	2011	smartphone, vehicle ECU	vehicle information	accident detection and notification
[10] [39]	2010	smartphone	vehicle information	accident detection and notification
[40]	2011	smartphone	driver behavior information	eco-driving assistant
[41]	2012	smartphone	traffic information	eco-driving assistant
[22]	2014	smartphone	driver behavior information	dangerous cornering detection
[21]	2014	smartphone	driver behavior information	dangerous cornering detection
[42]	2014	smartphone	driver behavior information	driving risk level scoring and driver feedback
[23]	2015	smartphone	driver behavior information	fuzzy logic based driving risk level scoring

The WreckWatch [10] accident detection system developed by White et al. differs from the one developed by Zaldivar et al. [38] in that it detects accidents using only the accelerometer values from a smartphone, and not the values from a vehicle's electronic control unit (ECU). In addition to the advantage of not requiring additional hardware, WreckWatch also requires no permanent GPS-connection, which dramatically reduces power consumption.

Vehicle insurance telematics aids insurance companies in quantifying the actuarial risk associated with a vehicle. Händel et al. [42] provides a detailed investigation into the necessary characteristics of a smartphone-based insurance telematics system.

Wahlström et al.[21, 22], developed two systems that detect and quantify the risk level of cornering events using GPS measurements. These risk classifications are used for data driven insurance telematics.

2.3. Environmental information

Eriksson et al. [43] developed one of the first road condition monitoring systems that detects and maps road anomalies, such as potholes, using an accelerometer and GPS. Mednis et al. [36] and Ghose et al. [35] later both developed road condition monitoring applications for a smartphone which sends sensor data to a remote server and alerts drivers of potholes in the road.

Mohan et al. [34] developed a road and traffic monitoring system, named Nericell, which also employs smartphone sensors to detect certain conditions. In addition to the driver behavior monitoring feature, Fazeen et al. [33] also added a road condition characterization and mapping feature to their system that uses a smartphone's GPS and accelerometer.

2.4. Driver behavior information

Johnson and Trivedi [9] developed one of the first complete driver behavior monitoring systems on a smartphone. Their system can detect and classify a number of aggressive and non-aggressive driving maneuvers when placed in a vehicle, by only using the internal accelerometer, gyroscope, magnetometer and GPS of a smartphone. Other driver behavior monitoring systems similar to the work in [9] has also been developed. Eren et al. [31] used a similar approach based on the same algorithms, but expanded their system by adding a driving style classification feature. Dai et al. [32] developed a system that specifically detects drunk driving. Fazeen et al. [33] developed a driver behavior monitoring system that advises a driver on dangerous vehicle maneuvers.

SmartLDWS [30] is a vision-based lane departure warning system developed for a smartphone. It employs a novel lane detection algorithm that provides satisfactory performance with the poor cameras typically found on older smartphones — while also being scalable to available computing power.

Eco-driving applications for smartphones aim to increase a driver's fuel efficiency by evaluating their driving and providing constructive feedback. Artemisa [40] is one such application which uses a smartphone's accelerometer to model a person's driving style and provides eco-driving tips to correct bad driving habits.

Castignani et al. [23] proposed the SenseFleet system for driver behavior monitoring. A novel, vehicle specific calibration procedure is used to determine fuzzy logic sets that enable the distinction between calm and aggressive driving to be made. Driver behavior is therefore identified independent of the vehicle's charecteristics.

3. Entirely smartphone-based vehicle monitoring systems

The advent of mobile technology has meant that many of the driver behavioral applications mentioned in Section 2 can use smartphone-based sensors. The use of smartphone-based sensing as opposed to sensing from a fixed installation in a vehicle, has certain advantages that include

- links behavior to an individual, rather than to a vehicle of which the driver might be unknown
- decouples the vehicle's age, technology, make, type, and interfaces from the sensing solution
- provides connectivity without any additional equipment

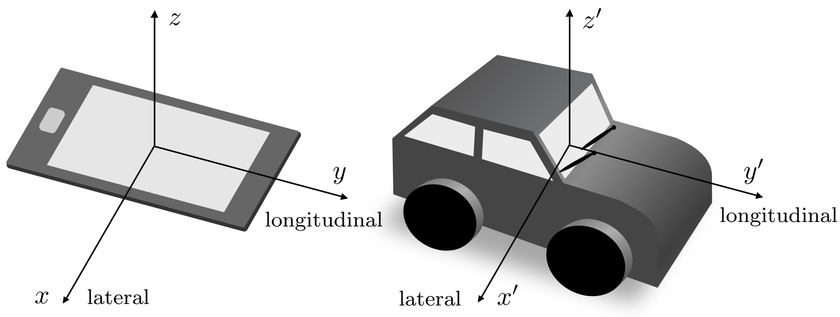

Figure 1: Smartphone and vehicle coordinate system used in analysis.

- negates installation costs
- enables the detection of phone usage while driving.

Despite these benefits, the use of only smartphone-based sensing introduces many challenges too, including

- sensor and algorithm complexity vs. limited battery power
- the changing orientation of the devices which makes vehicle acceleration sensing cumbersome
- broad range of smartphones
- inaccuracies in cost-effective sensors used in smartphones.

These challenges will be analysed in more detail through the paper, but demonstrates why the focus of this paper is purely smartphone-based solutions. The vehicle monitoring systems cataloged in Section 2 that solely rely on the embedded sensors of a smartphone are therefore further analyzed in this section. Only systems that use a smartphone's embedded sensors in an unobtrusive and energy-efficient manner are considered. For instance, systems using image processing techniques on a smartphone's camera are not considered, as it is processing and power intensive, as well as requiring the smartphone to be mounted on the dashboard. The most relevant published papers that are analyzed are listed in Table 2.

In the rest of the paper, readings from an accelerometer's three axes (x, y, z) are denoted as a_x, a_y and a_z. Readings from a gyroscope's three axes are denoted as ω_x, ω_y and ω_z. A vehicle's axes are denoted as x', y' and z'. Accelerometer readings are expressed in terms of the acceleration from gravity, g (9.8 m/s^2), and gyroscope readings in terms of rotation rate (rad/s). Acceleration vector components are denoted as $\mathbf{a}_x$, $\mathbf{a}_{x'}$, etc. As shown in Figure 1, the axes of a smartphone are defined as x pointing towards the right and y to the top from the phone's front, while z points out orthogonal to the screen. A vehicle's axes are defined as x' pointing towards the right and y' to the front of the vehicle, while z' points up towards the roof.

3.1. Nericell: Rich Monitoring of Road and Traffic Conditions using Mobile Smartphones — Mohan et al.

Nericell [34] is a smartphone-based system designed to detect certain conditions pertaining to vehicles such as braking, bumps in the road, honking and stop-and-go traffic. It uses a smartphone's accelerometer, microphone, GSM communications and GPS for this purpose. Nericell aggregates sensed data from multiple participating smartphones on a centralized server. Mohan et al. envisages the system being used to annotate existing traffic maps with information such as the condition of road surfaces and the level of chaotic traffic.

Nericell strives to use the sensors in a power-efficient manner. Only the accelerometer is sampled continuously with the GSM radio kept active, which is needed for communication anyway. The system relies on input from these devices to turn on the microphone or GPS only when they are needed in order to conserve energy. Data is filtered and processed locally on each smartphone before sending it to a server for aggregation.

Table 2: Summary of techniques, hardware, software, and sensors used by smartphone-based vehicle monitoring systems.

Reference	**Hardware**	**Software**	**Detection technique**	**Sensors used**
Mohan [34]	HP iPAQ hw6965 PDA, HTC Typhoon smartphone, Sparkfun WiTilt accelerometer	Windows Mobile 5.0 and 2003	pattern matching, orientation calibration	accelerometer, microphone, GPS
Dai [32]	HTC Dream (G1) smartphone	Android 1.6	pattern matching, orientation calibration	accelerometer, gyroscope
Johnson [9]	iPhone 4	iOS 4	endpoint detection, DTW	accelerometer, gyroscope, magnetometer, GPS
Eren [31]	iPhone 4	iOS 4	endpoint detection, DTW, Bayesian classifier	accelerometer, gyroscope, magnetometer
Fazeen [33]	HTC Google Nexus One smartphone	Android 2.1	pattern recognition	accelerometer, GPS
White [10]	HTC Magic (Google ION) smartphone	Android 1.5	pattern matching	accelerometer, microphone, GPS
Wahlström [21]	Samsung Galaxy S3, Galaxy Xcover 2, and Galaxy S4	unspecified	theoretical tire slip threshold detection	GPS
Wahlström [22]	iPhone 4, iPhone 5 and Samsung Galaxy S5	unspecified	theoretical tire slip threshold detection	GPS
Handel [42]	HTC Desire HD, iPhone 4, iPhone 5 and Samsung Galaxy S3	unspecified	multiple FoMs threshold detection	GPS
Castignani [23]	Samsung Galaxy Gio and a Samsung Galaxy S3	unspecified	fuzzy logic	GPS, accelerometer, magnetometer, weather, time of day

3.1.1. Virtual reorientation procedure

The accelerometer is the key sensor of the system, as it is used for braking, pothole and bump detection. However, the orientation of its axes in relation to the vehicle in which it is, must be known. Therefore, an algorithm was developed to virtually reorientate the smartphone's accelerometer to the vehicle's frame of reference.

An accelerometer measures the acceleration associated with the weight experienced by any mass — therefore, if the accelerometer is aligned correctly, it will measure $a_z = -1g$. The framework used for the reorientation of the accelerometer is based on Euler angles, which simplifies comprehension, but is computationally inefficient and presents problems with singularities and gimbal-lock. The orientation of the accelerometer can be described as a pre-rotation about z', a tilt about y' and a post-rotation about z', denoted as θ_{pre}, θ_{tilt} and θ_{post} respectively. Only the tilt operation changes the angle of z with respect to z', and since $|a_{z'}| = 1$ when assuming the vehicle is on flat ground,

$$\theta_{\text{tilt}} = \cos^{-1}\left(\frac{a_z}{a_{z'}}\right) = \cos^{-1}(a_z) \tag{1}$$

Also, since $|a_{z'}| = 1$, pre-rotation followed by tilt would result in non-zero a_x and a_y. Therefore

$$\theta_{\text{pre}} = \tan^{-1}\left(\frac{a_y}{a_x}\right) \tag{2}$$

To estimate θ_{tilt} and θ_{pre} using equations 1 and 2, periods when the vehicle is stationary or in steady motion have to be identified. However, a simpler method that proved to work well was to use the median values of a_x, a_y and a_z over a 10 second window. As long as no high-speed sharp turns are performed during the window, the values are notably stable, even on a bumpy road. Lastly, the post-rotation about z' has no influence on the forces experienced due to gravity, therefore another force is needed in order to estimate the angle of rotation.

The acceleration and deceleration of a vehicle in a straight line provides a force in the positive and negative y' direction respectively. Deceleration (braking) tends to induce a stronger force than acceleration and is therefore used for the estimation procedure. The GPS trace is used to monitor the vehicle for a sharp deceleration event in a more or less straight line. As explained in the paper [34], a maximization procedure yields an equation for θ_{post} dependent on θ_{tilt} and θ_{pre}. Therefore, to estimate θ_{post}, θ_{tilt} and θ_{pre} must first be estimated using equations 1 and 2. The GPS trace is then evaluated for a braking event and during the transient surge the mean of a_x, a_y and a_z are recorded. This is done to account for the time delay in the speed estimate received from the GPS.

The effectiveness of the virtual reorientation procedure was validated through testing. Values from a separate, well-oriented accelerometer, was recorded simultaneously with accelerometer values from a smartphone running Nericell, during numerous drives. The cross-correlation between the data was calculated to quantify the results, which are shown in Table 3. The cross-correlation is far from perfect, but this is mostly due to sensor noise and the fact that no two accelerometers will give the exact same values. In general, the cross-correlation between a well-oriented accelerometer and a disoriented smartphone accelerometer, improves substantially when the latter has been virtually reoriented.

The process can be streamlined and the computational efficiency significantly improved by implementing rotation matrices or quaternions instead of Euler angles.

3.1.2. Driving event detection

After a smartphone's accelerometer has been successfully reoriented, it can be used to detect certain events from which road and traffic conditions can be deduced. The first problem pertains to detecting braking events. A high occurrence of braking on a stretch of road could indicate poor road conditions or heavy traffic. Braking causes a surge in the negative y' direction acceleration. Detecting a braking event is fairly easy, since the surge typically spans more than a second. The mean of $a_{y'}$ is calculated over a sliding window of N seconds, and if it exceeds the threshold λ, a braking event is assumed. The GPS could also

Table 3: Cross-correlation between two well-oriented (f, g), a disoriented (h) and a virtually reoriented (h') accelerometer. [34]

$f \star g$	$\theta_{\text{pre}}/\theta_{\text{tilt}}/\theta_{\text{post}}$	$h \star f$	$h' \star f$	$h \star g$	$h' \star g$
0.90	7°/38°/106°	0.30	0.88	0.20	0.91
0.75	174°/34°/-107°	0.43	0.72	0.54	0.87
0.94	174°/34°/-107°	0.59	0.84	0.67	0.90
0.74	4°/42°/12°	0.65	0.72	0.63	0.68
0.76	3°/44°/-1°	0.62	0.71	0.69	0.79
0.78	-80°/42°/121°	0.65	0.73	0.64	0.73

be used to detect braking, as with the orientation procedure, but it uses more power and it is susceptible to the GPS localization error.

Detecting a bump or pothole in a road is the second problem for which the accelerometer is used. This is more challenging to accomplish, since the accelerometer's signal can vary considerably when driving over a bump, depending on the size of the bump and the vehicle's speed. The duration of such events are typically also very short (in the order of milliseconds). Nericell uses two methods for bump detection — one for low speeds and one for high speeds. When a vehicle's wheel hits the bottom of a pothole, a sharp force is induced which causes a distinct spike in the curve of $a_{z'}$. Therefore, a surge in $a_{z'}$ greater than a threshold is recognized as a bump. However, at low speeds (<25 km/h) the spike is not distinct enough from noise. The sustained crossing of a threshold for at least 20 ms is indicative of a bump at low speeds, while at high speeds slight unevenness also causes sustained peaks and dips. Monitoring for a peak in $a_{z'}$ is therefore appropriate for high speeds (*z-peak*); while evaluating for a short sustained rise or dip in $a_{z'}$ is appropriate for low speeds (*z-sus*). Lastly, the coarse estimate of speed obtained from GSM-localization is sufficient to govern which method to apply.

Nericell also uses a smartphone's microphone for horn detection. Although audio recording and processing consumes a considerable amount of power, it is only triggered when frequent braking is detected. To protect user privacy, only anonymous information obtained after processing is sent to the server. The goal of horn detection is to conjecture chaotic traffic conditions in some places, such as at unregulated intersections. The discrete Fourier transform is performed on short audio samples to be able to detect significant energy spikes in the frequency domain. The detection algorithm is based on empirical observations: if at least two spikes (harmonics) are detected, with one in the 2.5 to 4 kHz range, the audio sample is classified as containing the sound of a horn.

Braking detection was tested during a 35 km drive with varied traffic conditions. The detection performance is measured in terms of false positives (FP) and false negatives (FN). For the ground truth, 45 braking events were identified from the GPS trace with a threshold of $\lambda = 0.1g$ that must be exceeded for at least 4 seconds. A 4 second window and threshold values of $\lambda = 0.11g$ and $\lambda = 0.12g$ were used for the accelerometer braking detection algorithm. For $\lambda = 0.11g$, a FN and FP rate of 4.4% and 31.1% was obtained respectively. While for $\lambda = 0.12g$, a FN and FP rate of 11.1% and 17.7% was obtained respectively.

Bump detection was tested by manually annotating bumps on a route to use as the ground truth. To lessen subjectivity, a route was repeated a few times and a consensus was reached between two or three people. On a 30 km route of mixed road conditions, a total of 101 bumps or potholes were noted. At low speed, FN and FP rates of 37% and 14%, respectively, were obtained by *z-sus*. At high speed, FN and FP rates of 41% and 8%, respectively, were obtained by *z-peak*.

The horn detector was tested simultaneously on four phones at a chaotic intersection. The ground truth was establish by listening to an audio recording and manually noting the time when a horn is heard. With a high enough threshold to avoid false positives, FN rates of 0% to 50% was obtained between the different phones while placed inside and outside of an enclosed vehicle.

The virtual reorientation algorithm implemented does not account for the vehicle being on an incline. If the vehicle is on a steep incline when the reorientation procedure is started, the accuracy of the system could be considerably reduced.

3.2. Mobile Phone-Based Drunk Driving Detection — Dai et al.

Dai et al. [32] developed a system which can detect drunk driving by solely using a smartphone's accelerometer. As far as is known, they were the first to develop a system that uses smartphone sensors for driver behavior recognition. Their motivation for designing such a system is the fact that most of the time drunk driving goes unnoticed by the authorities, which puts many people's safety at risk.

They summarized drunk driving related behaviors from a study done by the Unites States National Highway Traffic Safety Administration (NHTSA). There are two categories of behavioral cues which correspond to a high probability of drunk driving. The first category is related to lane positioning problems such as drifting and swerving. The second category is related to speed control problems such as sudden acceleration or erratic braking. Both these categories of cues can be detected by using an accelerometer to map these cues into the lateral and longitudinal acceleration of a vehicle.

The system is designed with four software components: a monitoring daemon module, calibration module, pattern matching module and an alert module. The calibration module determines the orientation of the smartphone within a moving vehicle. This enables the system to function irrespective of where and how the smartphone is placed in a vehicle. The monitoring daemon continuously examines accelerometer samples in order to start the calibration module when vehicle movement is detected. The initial acceleration of a vehicle induces a continuous longitudinal force on the accelerometer in either the forward or backward direction. This acceleration is denoted as vector $\mathbf{a}_l$. It was determined empirically that $\mathbf{a}_l$ must exceed $0.265g$ for several seconds before the calibration module is started.

3.2.1. Calibration Procedure

When the calibration module is started, the virtual orientation sensor of the smartphone is first used to obtain its yaw, pitch and roll — denoted as θ_z, θ_x and θ_y, respectively.

The orientation sensor [44] is a virtual sensor programmed into certain smarphones' operating systems. It uses proprietary algorithms to estimate a device's orientation with respect to gravity and magnetic north, usually from accelerometer- and magnetometer—in some cases gyroscopic– readings.

The horizontal acceleration components of the smartphone's x and y-axis, denoted as $\mathbf{a}_{xh}$ and $\mathbf{a}_{yh}$, are then obtained from

$$\begin{aligned} \mathbf{a}_{xh} &= \mathbf{a}_x \cos(\theta_y) \\ \mathbf{a}_{yh} &= \mathbf{a}_y \cos(\theta_x) \end{aligned} \tag{3}$$

Next, the magnitude of $\mathbf{a}_l$ is obtained by

$$|\mathbf{a}_l| = \sqrt{|\mathbf{a}_{xh}|^2 + |\mathbf{a}_{yh}|^2} \tag{4}$$

The angle between vector $\mathbf{a}_{xh}$ and $\mathbf{a}_l$ is denoted as α, while the angle between vector $\mathbf{a}_{yh}$ and $\mathbf{a}_l$ is denoted as β. These angles are obtained from

$$\begin{aligned} \alpha &= \arccos(\mathbf{a}_{xh}/|\mathbf{a}_l|) \\ \beta &= \arccos(\mathbf{a}_{yh}/|\mathbf{a}_l|) \end{aligned} \tag{5}$$

Lastly, the lateral acceleration vector $\mathbf{a}_{x'}$ and longitudinal acceleration vector $\mathbf{a}_{y'}$ of the vehicle is obtained from the equations in 6.

$$\begin{aligned} \mathbf{a}_{x'} &= \mathbf{a}_{xh} \sin\alpha + \mathbf{a}_{yh} \sin\beta \\ \mathbf{a}_{y'} &= \mathbf{a}_{xh} \cos\alpha + \mathbf{a}_{yh} \cos\beta \end{aligned} \tag{6}$$

3.2.2. Pattern Matching

The pattern matching module is only activated once the calibration procedure is done. It evaluates the difference between the maximum and minimum value of lateral acceleration ($\mathbf{a}_{x'}$) within a pattern checking time window. If the difference exceeds a set threshold, the module reports that an abnormal curvilinear movement has occurred. The module also checks whether the longitudinal acceleration ($\mathbf{a}_{y'}$) exceeds fixed positive or negative thresholds at any given time, indicating a speed control problem. Multiple rounds of pattern matching are performed on both the detection algorithms, which is necessary in order to detect drunk driving with a high degree of certainty.

Tests were conducted to obtain a total of 72 sets of data for drunk driving related behaviors and 22 sets of data for regular driving. The time window for lateral acceleration was set to 5 seconds, as most acceleration patterns happen within this period. The threshold values were set to achieve a low number of FN and a sensible FP probability. The approach achieved a FN rate of 0% for both lateral and longitudinal driving events, and a FP rate of 0.5% and 2.4% for lateral and longitudinal driving respectively. Testing also showed that the system has tolerable power consumption. Initially with a fully charged battery, the smartphone's battery level was at 78% after 7 hours of operation — compared to 92% when the smartphone and system was idle for 7 hours.

The system assumes the accuracy of the virtual orientation sensor, this sensor, however cannot correctly determine device orientation relative to earth when the device experiences a significant acceleration [45] and a significant acceleration is required to determine the front of the vehicle. Secondly, only the horizontal components of acceleration are used by the system, therefore any incline encountered during operation would result in inaccurate accelerometer readings. The orientation of the device relative to the vehicle must also remain constant throughout operation, as calibration is only performed once at startup.

Another limitation of the system is that it cannot determine the speed of the vehicle, which would improve the ability of the system to identify dangerous driving patterns. The smartphone's GPS can provide the speed of the vehicle, but at the expense of increased power consumption. The system could also be improved by using the GPS to match the movement of the vehicle to road maps.

3.3. Driving Style Recognition Using a Smartphone as a Sensor Platform — Johnson and Trivedi

The driver behavior monitoring system developed by Johnson and Trivedi [9], named MIROAD, solely relies on the internal accelerometer, gyroscope, magnetometer and GPS of a smartphone. They were the first to develop a more complex pattern recognition approach. They also argue that anonymous participatory sensing would allow the number of aggressive drivers in an area to be established. This gives foresight into where accidents may possibly occur.

A smartphone running MIROAD can detect and classify a number of aggressive and non-aggressive driving maneuvers. However, unlike in [34] and [32], the smartphone has to be mounted on the dashboard of the vehicle. The system assumes the smartphone's axes are orientated with x, y and z in the direction of the top (z'), left side ($-x'$) and back ($-y'$) of the vehicle respectively. Output from the accelerometer, gyroscope and magnetometer of the smartphone is used for maneuver recognition. With the magnetometer, corrections are made with respect to magnetic north. The Euler rotation, also used in [34], can therefore be determined more accurately from a reference attitude.

The maneuvers associated with aggressive driving are hard left and right turns, swerving, and sudden braking and acceleration patterns. For the detection of longitudinal movements, the rotation rate ω'_x and the a'_y acceleration are used. For lateral movements, the a'_x acceleration and rotation rate ω'_z, as well as the Euler angle about y' is used. The accelerometer and gyroscope are continuously sampled at a rate of 25 Hz. In order to detect maneuvers, the start and end of driving events are determined by using the endpoint detection algorithm. For lateral maneuvers, a simple moving average (SMA) of ω'_z is continuously calculated for a short window of k samples. From the current sample i, we have

$$\mathrm{SMA}(i) = \frac{\omega'_z(i)^2 + \omega'_z(i-1)^2 + ... + \omega'_z(i-k-1)^2}{k} \tag{7}$$

The beginning of an event is detected if the SMA goes above a set upper threshold. The succeeding gyroscope values are concatenated until the SMA falls below a bottom threshold, signifying the end of the event. The event is dismissed if it exceeds 375 samples, or 15 seconds.

When a valid driving event has been detected, the signals recorded during the event are compared to a set of template events using the Dynamic Time Warping (DTW) algorithm. DTW finds an optimal alignment between two signal vectors with different lengths. Consider a matrix of the Euclidean distance between each point of two signal vectors. Both vectors start at the bottom left corner. An optimal warping path constitutes the minimum sum of distances, or cost, while adhering to monotonicity, boundary and step size conditions. The template event with the lowest warping path cost is the closest match to the detected event.

Video footage is continuously recorded by the rear camera of the smartphone facing towards the front of the vehicle, but no image processing is performed. MIROAD is able to playback video and sensor data synchronously to provide a recreation of an incident. To limit memory usage, all data is recorded in five minute intervals and if it is not flagged for any detected events, it is deleted. The system audibly alerts drivers of aggressive driving events through a software speech synthesizer. Alerts are also sent with the vehicle's location to external systems via the GSM internet connection.

MIROAD was tested in three different vehicles, with three different drivers. They collectively accumulated 201 driving events on highways, urban and rural roads, of which about 50 were considered possibly aggressive. While the detection rates vary for different events, in total, 97% of all the aggressive driving events were correctly identified by die DTW algorithm.

The system does not, however allow the labeling of driving styles, only the detection of maneuvers. Another limitation of the system is that the smartphone has to be kept in a fixed position within the vehicle, as no reorientation algorithm is implemented. The system also consumes more power than the other systems, due to the computationally intensive DTW algorithm and the GPS. The tests show only a slightly higher detection accuracy than the more efficient systems.

3.4. Estimating Driver Behavior by a Smartphone — Eren et al.

The system developed by Eren et al. [31] characterizes a person's driving as either safe or risky. Sudden maneuvers, turns, lane departures, braking and acceleration are seen as risky events. These events are detected by only using the accelerometer, gyroscope and magnetometer of a smartphone.

The system detects driving events similar to [9]. Moving average filters are applied to the raw sensor data to eliminate noise. Likewise, the endpoint detection algorithm is used to identify events, and the dynamic time warping algorithm is also used to compare input data to template events. However, another layer is added to the system that Johnson and Trivedi's system lacks — labeling of the driver behavior, as can be seen in Figure 2.

A Bayesian classifier is used to label a driver's behavior as either safe or risky according to a calculated probability. The existence of two classes, r_1 and r_2, related to safe and risky driving is assumed. The calculation is based on the number of occurrences of different driving events (s) over time. The mathematical expression for determining the probabilities is given by

$$P(r_1|s) = \frac{P(r_1)P(s|r_1)}{P(s)} = \frac{P(r_1)P(s|r_1)}{P(r_1)P(s|r_1) + ... + P(r_n)P(s|r_n)} \tag{8}$$

The classification is made by comparing the calculated probabilities

$$\begin{aligned} P(r_1|s) > P(r_2|s) &\Rightarrow \text{Safe} \\ P(r_1|s) \leq P(r_2|s) &\Rightarrow \text{Risky} \end{aligned} \tag{9}$$

The driving patterns of a selected group of drivers were analyzed. The group consisted of five experienced, five novice and five randomly chosen drivers. Two tests were done with each driver in different weather conditions in order to evaluate the reliability of the system. Two other experienced drivers were also selected to sit in the passenger sides of the vehicles while the tests were performed. They were required to fill in a short survey after each test. The Bayesian classifier correctly identified the driving style as safe or risky, as well as driving event types, for 14 out of the 15 drivers. However, the accuracy of the classifier

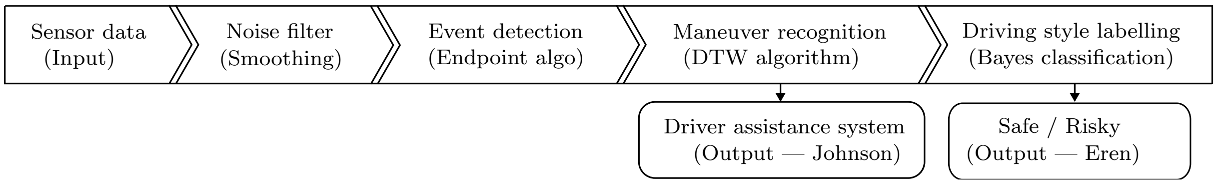

Figure 2: Block diagram illustrating the extra layer of Eren et al.'s system over Johnson's and Trivedi's system.

is based on completed surveys by test participants — therefore the results could have been influenced by subjectivity. Having to place the smartphone in a fixed position within the vehicle is also a limitation of the system. No data is provided regarding the power usage of the system, but the lack of GPS sampling and the simple Bayesian classifier used is indicative of a slightly lower power consumption than the MIROAD [9] system .

3.5. Safe Driving Using Mobile Phones — Fazeen et al.

Fazeen et al. [33] envisages implementing an advanced driver-assistance system (ADAS) on a smartphone. Such a system advises a driver on dangerous situations that emerge from vehicle maneuvers and environmental factors. The system uses the accelerometer and GPS of a smartphone to enable typical features found in ADAS-equipped vehicles. The aim of the system is to recognize and classify driving behavior and to map road surface conditions.

Data identified as part of a driving event or road anomaly is stored on the phone and the user has full control over it. Any data that is sent to a server for mapping and machine learning techniques is kept anonymous. Drivers are audibly alerted by the system of dangerous situations.

Road anomalies can be detected because of the vibrations experienced by a vehicle when driving on a rough road. When a vehicle drives over a bump, it ascends onto it, resulting in a sharp rise in the z-axis value of the accelerometer. An increase in the y-axis value is also observed, depending on the shape of the bump, because of the longitudinal force exerted on the vehicle's wheels. The difference between successive accelerometer readings is continuously evaluated. A bump in the road is presumed if the difference exceeds a dynamic threshold, which is dependent on the speed of the vehicle. The height of a bump can be roughly calculated by using dynamics equations. The accuracy of the approximation could be improved if the dynamics of the vehicle's suspension is known.

When a road anomaly is detected, the current GPS coordinates are saved with a corresponding value indicating the condition of the road. The system classifies a section of a road as either smooth, rough, uneven, or as containing a bump or pothole. The data can then be used to map the condition of entire stretches of road.

During testing, an overall accuracy of 85.6% was achieved by the road condition classification system. It was determined empirically that safe acceleration and deceleration never exceeds $\pm 0.3g$, while sudden maneuvers approaches, but does not exceed $\pm 0.5g$. A gradual lane change, in comparison, exerts an average lateral acceleration of only $\pm 0.1g$. It is therefore possible to differentiate between safe and unsafe driving maneuvers. It was also found that recognizing gear shifts is possible, which would enable the system to advise a driver when to shift gears in order to achieve efficient fuel usage.

The system relies on the GPS for calculation of the vehicle's velocity, consuming power additional to the sampling of the accelerometer. To overcome this limitation, the vehicle's velocity could rather be calculated by integrating the sampled acceleration curve between each gear shift, thereby easing battery usage.

The accuracy of a Nexus One smartphone's accelerometer was tested by comparing it to calculated data from dynamics equations. The system requires the smartphone to be orientated with the top facing to the front of the vehicle and the screen facing upwards. Tests were conducted with the smartphone placed in different locations in the vehicle. The results showed that placement in the center of the vehicle is the best location for monitoring driving behavior. The required fixed orientation of the smartphone in the vehicle is also a limitation of the system.

3.6. *WreckWatch: Automatic Traffic Accident Detection and Notification with Smartphones — White et al.*

The WreckWatch [10] system detects accidents using only the sensors from a smartphone, as opposed to similar original equipment manufacturer systems which use values from a vehicle's electronic control unit (ECU). White et al. argue that it is unrealistic to expect drivers to connect their smartphones to the ECU for every journey. Another concern is that older vehicles do not have ECUs, therefore, an accident detection system that is not dependent on an ECU is advantageous.

WreckWatch uses a soft real-time (close to real-time) approach sampling the accelerometer, microphone and GPS of a smartphone. An accident is detected by threshold filtering the sensor readings. Data recorded preceding and during an accident is sent via GSM to a centralized server. Important information about an accident can then be relayed to the relevant authorities from a stored database on the server.

False positives are more likely to occur with a system using only smartphone sensor data. Dropping the phone on the floor or making a sudden stop may be detected as an accident. Therefore, context information obtained from filters must be used to prevent false positives. Firstly, the determined acceleration is filtered by ignoring any values below $4g$. Secondly, a user is assumed to be in a vehicle if they are moving faster than 25 km/h. The smartphone's GPS is used to determine the speed of the user. Accelerometer information is only evaluated when the user is traveling faster than 25 km/h. This reduces power consumption and prevents false alerts from occurring if the phone is accidentally dropped while being outside a vehicle.

The WreckWatch system has been developed further to improve low-speed collision detection by adding acoustic data analysis. The microphone of the smartphone is used to listen for high-decibel sounds such as impact noise, car horns or airbag deployment (170 dB peak).

If an accident is detected, emergency responders are automatically notified by the system. Situational awareness is provided to the first responders to enable them to assess the severity of the accident. The GPS coordinates of the accident is immediately sent to the server with other accident characteristics. Thereafter, bystanders and uninjured victims can provide critical information through the WreckWatch application. For instance, pictures of the accident can be taken with the smartphone's camera and shared with the first responders.

A few different tests were performed to evaluate the possibility of false positives occurring. The empirical results demonstrate that dropping the smartphone in a moving vehicle is not likely to cause a false positive. The filter threshold of $4g$ on accelerometer readings are sufficient to prevent it. Furthermore, it was found that threshold filtering can not be used for acoustic detection of airbag deployment. Playing music at full volume or people shouting in the vehicle causes sound signal clipping at 145 dB on the smartphone. Therefore, the system relies on the acoustic signature of a detected event as a secondary indicator of an accident. As with the systems of Sections 3.3, 3.4 and 3.5, the required fixed orientation of the smartphone in the vehicle is also a limitation of the system.

3.7. *Detection of Dangerous Cornering in GNSS Data Driven Insurance Telematics – Wahlström et al.*

3.7.1. *Quantifying Dangerous Cornering*

The two main dangers presented during harsh cornering are tire slippage(slipping) and vehicle rollover. With this in mind, Wahlström et al.[21] defined a dangerous cornering event relative to the thresholds where slipping or vehicle rollover are theoretically likely to occur. The theoretical slipping threshold is defined with respect to the tangential velocity(v), rotational velocity (w) and tangential acceleration(a) of the vehicle as well as the coefficient of friction (μ) between the vehicle's tires and the road. To avoid slipping, inequality 11 must hold. A function $T(v, a, w)$ is defined as the ratio between the horizontal F_h and normal F_n forces exerted by the road on the vehicle.

$$T(v, a, w) = \frac{F_h}{F_n} = \frac{1}{g}\sqrt{v^2w^2 + a^2} \tag{10}$$

$$T(v, a, w) \leq c1\mu \tag{11}$$

$$c1\mu = \gamma_{ns} \tag{12}$$

Where $c1$ is a constant safety factor chosen empirically and γ_{ns} is the adjusted no slip threshold.

For the theoretical rollover threshold, the torque around the vehicle's center of gravity(CG) must be considered. The vehicle's track width (l) as well as the height of the CG (h) need also be taken into account. It can then be proven that, to avoid rollover, inequality 13 must hold.

$$T(v, 0, w) \leq c2\frac{l}{2h} \tag{13}$$

$$c2\frac{l}{2h} = \gamma_{nr} \tag{14}$$

Where $c2$ is a constant safety factor chosen empirically and γ_{nr} is the adjusted no roll threshold.

Evaluating these equations for various vehicles it can be shown that, for a typical case with no additional roof loads, zero road pitch- and roll angles and no tripped rollovers, the no-slip threshold will be reached before the no-rollover threshold. Wahlström et al defines a single cornering event from the moment where $T(v, a, w) \geq \gamma_{ns}$ to the moment where $T(v, a, w) \leq \gamma_{ns}$. However, when $T(v, a, w) \geq 0.35$ (0.35 being empirically chosen) between two events, they are combined as a single cornering event. The risk level of the event is defined as the maximum value of T(v,a,w) during the event.

3.7.2. Filtering of GNSS Measurements

A number of input variables are needed to enable the use of the methods in section 3.7.1. These variables need to be obtained from Global Navigation Satellite System (GNSS) measurements, according to Wahlström et al. however, the GNSS receivers found in smartphones are not of very high quality and the data they provide is prone to errors. The variables are therefore estimated using a Kalman Filter.

Most mobile phone based GNSS systems supply at least the two dimensional coordinate position as well as the Doppler-based heading and velocity of the device. To improve the accuracy of the estimated variables, the position measurement is modeled as a combination of discrete, zero-mean white noise as well as a slowly varying bias caused by clock errors and atmospheric effects. The slow moving bias is effectively removed by using

$$\Delta\mathbf{p_n} = \mathbf{p_n} - \mathbf{p_{n-1}} \tag{15}$$

instead of the position measurement for calculations. Due to the non-linear nature of the equations governing the system's dynamics and its ability to apply constraints to individual sigma points, an Unscented Kalman Filter is used to estimate the required variables. The filter's parameters such as system and measurement noise covariances are chosen to correspond to typical vehicle dynamics as well as expected GNSS error magnitudes.

A framework for parameter tuning and performance evaluation is described, where events detected by the system are mapped to events provided by a reference system. Various options for loss functions based on the number of missed and false detections are given and certain parameters can be modified to minimize the loss function.

3.7.3. Testing and Results

The system was field tested using 31 minutes of aggressive driving data. A Microstrain 3DM-GX3-35 was used to collect reference data using it's IMU and GNSS sensors. The IMU and GNSS sensor data was combined to provide a reverence value for $T(v, a, w)$. Data was simultaneously collected from three different android smartphones. Thresholds for γ_{SL} were chosen between 0.5 and 0.75, according to Wahlström et al. these values correspond to what should intuitively be labeled as aggressive cornering.

For thresholds in the range of $0.5 \leq \gamma_{SL} \leq 0.6$ The system showed a average 40% missed detection and false alarm rate over the three phones used. This percentage deteriorates for higher thresholds. The estimator is also less effective when dealing with high g-force cornering, due to these events typically having a shorter duration and the GNSS update rate being too slow.

The novel and useful framework for parameter tuning can be better utilized by applying large amounts of driving data, as the 31 minutes of driving data used cannot be seen as a fully representative sample. The advantage of using only GNSS data is that it is independent of device orientation and movement relative to the vehicle. GNSS on a smartphone, however, has a high power consumption, slow update rate (typically around 1Hz, but 4Hz in this case) and is often unreliable in terms of satellite signal and accuracy.

Fusing readily available smartphone IMU data with GNSS data could increase system reliability and the higher update rate of the IMU sensors could aid in detecting shorter high g-force events.

3.8. Risk Assessment of Vehicle Cornering Events in GNSS Data Driven Insurance Telematics – Wahlström et al.

Similar to the other paper presented by Wahlström et al.[21], theoretical thresholds for rollover and slip are calculated and used as a reference for assessing the risk level of cornering events, but instead of $T(v, a, w)$ in equation 10, $T(v, a, r)$ is used, where r is the radius of the specified turn. This leads to equation 16.

$$T(v, a, r) = \frac{F_h}{F_n} = \frac{1}{g}\sqrt{\frac{v^4}{r^2} + a^2} \tag{16}$$

The radius r is estimated using a circle fitting technique to fit circles to position measurements. The velocity (v) and acceleration (a) parameters are estimated from the GNSS measurements using a Rauch-Tung-Striebel (RTS) Kalman smoother.

The system functionality was first demonstrated by simulation with generated data. A field study was then conducted with three smartphones (iPhone 4,iPhone 5 and Galaxy S5) collecting GNSS data simultaneously. The differences in smartphone position measurements was expected to have a limited effect on the radius estimates, but the one smartphone(iPhone 5) produced a outlier datapoint resulting in a false radius estimation and increased risk. No attempt was made to compare the smartphone data with data from a reference system, therefore, despite the theoretical merit of the system no comparison in terms of accuracy can be made.

3.9. Insurance Telematics: Opportunities and Challenges with the Smartphone Solution – Händel et al.

The article presented by Händel et al. [42] does not aim at the development and testing of a complete smartphone-based reckless driving detection system, but rather discusses the technological aspects of vehicle insurance telematics and the use of smartphones as a platform for vehicle insurance telematics.

Eight of the most common Figure of Merits (FoMs) that can be used for insurance telematics are identified and characterized in terms of a number of quality measures, namely: observability, event stationarity, actuarial relevance and driver influence. Observability indicates how accurately the figure of merit can be derived from the sensor measurements. Stationarity is defined as the time length in which the measurements that relate to a FoM can be measured. Actuarial relevance indicates how indicative the event is of the risk-level of insuring the driver. Driver influence indicates to what degree the driver has an effect on the FoM. The characterization of the 8 FoMs, if obtained from GNSS-measurements, are shown in table 4.

An alternative method to direct differentiation(which amplifies high frequency noise) of GNSS speed data is proposed, to obtain "clean" acceleration data as well as a quality index indicating the quality of the calculated data. A test of the percentage GNSS coverage and harsh breaking detection ability of 7 popular smartphones was done by using a vehicle's OBD data as a reference. The test confirmed the unreliability of the directly differentiated smartphone data and showed that the low-level data cleansing routine decreased false detections.

A scoring system for individual FoMs is developed. A score between 0 (bad) and 1 (good) is allocated using equation 17.

$$\mathbf{S} = \frac{1}{1 + \alpha f} \tag{17}$$

With f denoting the number of detected events and α being a chosen constant. Given a statistical distribution of f, modeled using the information in table 4 one can calculate an appropriate value of α for each FoM.

To combine scores for all FoMs, the scores can be scaled according to their actuarial relevance. A trade off has to be made between a reliable system which places less weight on FoMs with low observability and

Table 4: Characterization of foms in insurance telematics when calculated using gnss-data. [42]

FoM	Description	Observability	Stationarity	Driver-Influence	Actuarial Relevance
Acceleration	Number of rapid acceleration events and their harshness	Medium	Low	High	Medium
Braking	Number of harsh braking events and their harshness	Medium	Low	Medium	High
Speeding (Absolute)	Amount of absolute speeding	High	High	High	Medium
Speeding (Relative)	Amount of speeding relative a location dependent limit	Medium	High	High	High
Smoothness	Long-term speed variations around a nominal speed	High	High	Medium	Low
Swerving	Number of abrupt steering maneuvers and their harshness	Low	Low	Medium	Low
Cornering	Number of events when turning at too high speed and their harshness	Medium	Medium	High	Medium
Eco-ness	Instantaneous or trip-based energy consumption or carbon footprint	Low	Medium	High	Low
Elapsed time	Time duration of the trip	High	High	Low	Low
Elapsed distance	Distance of the trip	High	High	Low	High
Time of day	Actual time of day when making the trip	High	High	Low	High
Location	Geographical location of the trip	High	High	Low	Medium

stationarity versus placing emphasis on actuarially relevant FoMs. The relevance of the scoring feedback to the driver must also be carefully evaluated, because actuarially relevant data such as day or nighttime driving is out of the drivers control and low scores due to such FoMs could cause drivers to lose confidence in the system.

3.10. Driver Behavior Profiling Using Smartphones – Castignani et al.

The SenseFleet [23] system proposed by Castignani et al. uses smartphone based accelerometer, magnetometer and GPS sensors to identify and rate the riskiness of driving events. This is done by defining Fuzzy logic sets for each parameter after a calibration phase in a specific vehicle. The relevant parameters chosen are the standard deviation of the Jerk(derivative of accelerometer measurement), mean yaw rate(using orientation sensors) as well as the speed and bearing variations(from GPS measurements). Calibration is vehicle-specific, as different vehicles have unique driving characteristics. Low, medium high and very-high values are calibrated for all parameters except speed variation, for which HIGH-DEC, LOW-DEC, STABLE, LOW-ACC and HIGH-ACC values are defined. Detection can be done using a basic fuzzy inference system. Weather and time of day data are also gathered for each event so driver scoring can be adjusted accordingly. A simple scoring system is implemented where a driver starts a trip with 100 points and a predefined amount of points are subtracted for each type of event and the conditions during which the event takes place(i.e. more points will be subtracted for an event happening at night in snow than in the daylight in sunny weather.). Points are added when no events are detected for a set interval.

The calibration method proposed, requires the user to drive the vehicle while a set amount of samples are collected within each of three speed intervals. The necessity of this cumbersome vehicle specific calibration is a notable disadvantage of the system.

To experimentally test the system, ten drivers were asked to, after the calibration phase, drive a predefined route twice. The first lap was driven calmly and the second lap aggressively. The results showed an average driver score of 74.4 for the calm lap and 20.3 for the aggressive lap. A Principle Component Analysis (PCA) was done on the data and the resulting data could easily be clustered according to the aggressive and calm laps. The system therefore allows clear distinctions to be made between aggressive and calm driving behavior. A test was also executed using a high end and entry level android smartphone mounted in a similar way within the vehicle. The two phones showed similar results, demonstrating the compatibility of the system with different smartphone models.

3.11. Review and comparison of the summarized papers

In this section, the detection techniques, hardware, software and objectives of the different systems are compared. Although the sensors and detection techniques used by the different systems are similar, as can be seen in Table 2, their objectives differ slightly.

3.11.1. Driving maneuver recognition versus driving behavior classification

The system of Dai et al. [32] explicitly attempts to determine whether a driver is drunk. This is achieved by detecting and positively identifying a combination of dangerous driving maneuvers associated with drunk driving. Johnson and Trivedi's [9] system can detect and identify a number of different driving maneuvers, but does not draw any conclusions from them. Their intent is to use the system to support a holistic driver assistance system (DAS) by providing it with additional information. The system of Eren et al. [31] also detects driving maneuvers, but incorporates another layer where a person's driving style is labeled as either safe or unsafe with a given probability. Fazeen et al. [33] aims to implement an ADAS entirely on a smartphone. Their system records and analyses various driver behaviors and external road conditions.

It is important to note the difference between driving maneuver recognition and driving behavior classification. A system could detect various maneuvers, but not necessarily infer anything from them, whereas another system may be able to deduce and classify a driver's behavior from detected driving maneuvers. These different systems demonstrate the variety of driving behavior classifications that can be made. A person's normal driving style can be classified as safe or risky, fuel-efficient or inefficient, skilled or unskilled — and recommendations can be given accordingly to improve their driving. On the other hand, a person's driving behavior can sometimes differ from normal due to certain circumstances. A person could be driving under the influence of alcohol, drugs or other sensory impairments. In such situations a driver could be warned of their dangerous behavior or the relevant authorities could even be notified of the driver's behavior and location.

3.11.2. Accuracy versus simplicity

It is difficult to quantitatively compare the performance and power consumption of the different systems. All of the systems were implemented on different smartphones that have varied sensors and computing power. The test studies were performed in various countries with different road and traffic conditions. Their methods of establishing the ground truth for tests were not necessarily the same and could vary due to subjectivity.

Figure 3 shows a qualitative comparison of the accuracy versus simplicity of the different systems. A system that achieves high detection accuracy with a simple algorithm is considered superior. The assumption is made that a simpler system uses less resources and therefore consumes less power, which is a critical aspect of a successful system. The experimental and empirical test results of the systems as given in each paper was used to compare detection accuracy, although the testing procedures differed as mentioned. The perceived simplicity of each system is based on what each system is trying to detect, what sensors its using and how its algorithms function.

WreckWatch of White et al. [10] is empirically proven to be 100% accurate and is the simplest system, because it only detects accidents and nothing else. The road condition monitoring feature of Mohan et al. [34] is more accurate than that of Fazeen et al. [33], and its implementation is simpler. Regarding driver behavior monitoring — the drunk driving detection system of Dai et al. [32] is the most accurate, achieving a FN rate of 0%.

Dai et al. [32] implemented a simple yet effective pattern matching approach that requires very little computation. Essentially, only the difference in subsequent values on the relative longitudinal and latitudinal axes are calculated. If the difference exceeds a certain threshold, an aggressive driving maneuver is assumed. The algorithm used by Nericell of Mohan et al. [34] works in a similar manner. Both systems have satisfactory performance and consume minimal power. In contrast, Johnson and Trivedi [9], as well as Eren et al. [31],

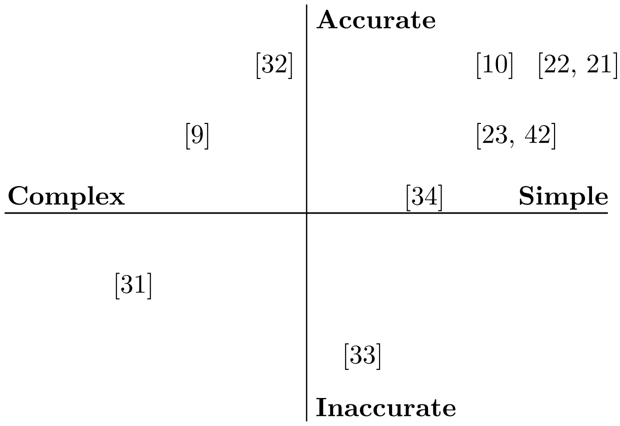

Figure 3: Qualitative comparison of accuracy versus simplicity of the different systems.

implemented a more complex pattern recognition approach derived from speech recognition techniques. Their systems also perform well and do not consume an overly large amount of power. Although it can not be explicitly proven here, the simpler approaches probably consumes less power while achieving similar performance to the more complex approaches. Arguably Dai et al. [32] accomplished the same functionality as Eren et al. [31] with a simpler algorithm, as both systems can infer a certain aspect of a driver's behavior from detected driving maneuvers.

In Table 2 the hardware and software on which each system was developed can be seen. The systems of Mohan et al., Dai et al. and White et al. [34, 32, 10] were all developed on hardware and software that is now considered obsolete, yet their systems were simple and accurate. This suggests the performance of embedded sensors used in smartphones has not improved significantly in the last decade. The computing power and efficiency of modern smartphones, however, has increased dramatically, which provides headroom for more complex solutions. Therefore there is still merit in implementing a more complex approach as used by Johnson and Trivedi [9] — if the accuracy could be improved to such an extent as to have no false negatives or positives whatsoever.

3.11.3. Contributions and best practices

In terms of contributions made, Dai et al. and Mohan et al. [32, 34] were the only authors to implement a procedure to calibrate the system to any arbitrary orientation of the smartphone. All of the other systems assume that the smartphone is placed in a fixed position within a vehicle. Automatic virtual reorientation of a smartphone's axes to a vehicle's axes is considered a best practice for any smartphone-based vehicle monitoring system. Ideally, the reorientation should take place each time the device is moved within the vehicle.

4. Challenges facing vehicle monitoring systems

In this section, a number of remaining challenges facing the future progress of vehicle monitoring systems are discussed.

4.1. Algorithm performance

There is currently no one accepted algorithm that stands out in terms of performance. It is difficult to compare the accuracy and performance of the different implemented algorithms. The algorithms must each be tested using the same hardware and ground truth before the distinction between them can be fully appreciated. A study that compares various algorithms' performance would help to establish best practices on which to base further development.

4.2. Data aggregation

The collection and aggregation of data is a key component of intelligent transportation systems (ITS). The utility of vehicle monitoring systems could be substantially improved with effective participatory sensing. Sharing data with aggregation servers would allow providing additional services, such as warning users of accidents and heavy traffic. Aggregated data could also be classified and used for machine learning based algorithm training and parameter tuning in other systems. An unified framework for participatory vehicle sensing and data aggregation must be developed.

4.3. Wide-scale deployment

A key challenge facing vehicle monitoring systems is accomplishing wide-scale deployment and use. Few individuals are willing to buy expensive dedicated hardware systems. Additional hardware not only increases costs, but also makes acquisition and installation cumbersome and inefficient, while a mobile phone application can be downloaded within seconds and with no cost to the user. It is for that reason that implementing such systems on smartphones is an attractive solution. There are no associated hardware costs, and smartphones as well as their communications infrastructure are prevalent, even in developing countries. In fact, Booysen et al. [46] expects ITS to be mobile-phone based in the developing world. To facilitate adoption by road users, insurance companies [47] have begun to provide discounts on premiums for users who allow themselves to be monitored. The monitoring allows the companies to adjust premiums according to the actuarial risk associated with a driver or vehicle.

4.4. Automation

The operation of the smartphone application must be fully automatic and unobtrusive. Cumbersome calibration routines should therefore be avoided. It is necessary that during driving, user interaction with the smartphone is minimized — in order to keep the driver's focus on the road.

4.5. Safe Feedback

Alerting drivers of notifications and warnings received from aggregation servers or tips on driving technique must not take their focus off the road. It must also not require any distracting interaction with the smartphone. Careful consideration must be given to the interface between a smartphone and driver. In most cases simple synthesized speech should be considered.

4.6. Power consumption

The power consumption of such applications running on a smartphone must be minimized so that it will not affect a user's normal smartphone usage habits. The power consumption could be decreased by developing computationally efficient algorithms and minimizing the use of GPS or the processing of microphone or image data. Thus far, not much thought has been given to the effective power consumption of monitoring systems on a smartphone.

4.7. Multi-platform

For a smartphone-based vehicle monitoring system to become widely used, it must work on a wide variety of smartphones. Another challenge is therefore to develop a multi- and cross-platform solution, and obtaining consistent performance between different smartphone hardware and software. The amount of noise in sensor readings can vary between different smartphones, for example. Typically, access to sensors are limited by the firmware and operating system of a smartphone. Therefore, techniques to standardize sensor readings on different smartphones through software must be investigated.

5. Conclusion

This paper provides an analysis of the use of smartphones to support novel ITS applications. Existing approaches are categorised as collaborative driving, vehicle telematics as well as driver behavior -, and road condition monitoring. Existing systems that solely depend on smartphones are analysed and compared. Weaknesses of each of the approaches are identified and where possible, improvements are suggested. The hardware and software platforms of each of the existing approaches are compared, and the complexity vs. accuracy qualitatively evaluated. From the review it is clearly possible to implement a complete vehicle monitoring system, and even a driver assistance system (DAS), on a smartphone, with acceptable performance. However, there are still a number of challenges facing the future progress of such systems. A key issue is accomplishing wide-scale deployment and use. In many cases these systems require many users before they become useful, and since there is in some cases little incentive for individuals to use them, adoption will be slow. If the majority of drivers on the road can be alerted of dangerous behavior, - situations and - road conditions, there would consequently be fewer road accidents and fatalities. Such systems could also help lower traffic congestion by informing drivers of optimal routes to their destinations and alerting relevant authorities to problems causing congestion. We envisage that smartphone-based vehicle monitoring and driver assistance systems will be a crucial part of ITS in the future, especially in developing countries.

6. Acknowledgement

This research was supported by MTN (through the MTN Mobile Intelligence Lab), the South African Transport Conference, and MiX Telematics.

References

[1] R. Xu, S. Zhou, W. J. Li, MEMS accelerometer based nonspecific-user hand gesture recognition, IEEE Sensors Journal 12 (5) (2012) 1166–1173.

[2] A. Akl, C. Feng, S. Valaee, A novel accelerometer-based gesture recognition system, IEEE Transactions on Signal Processing 59 (12) (2011) 6197–6205.

[3] S. Reddy, M. Mun, J. Burke, D. Estrin, M. Hansen, M. Srivastava, Using mobile phones to determine transportation modes, ACM Transactions on Sensor Networks (TOSN) 6 (2) (2010) 13.

[4] Global status report on road safety: time for action, Geneva, World Health Organization, [ONLINE] Available: `http://whqlibdoc.who.int/publications/2009/9789241563840_eng.pdf`. [Accessed 5 August 2013] (2009).

[5] Aggressive driving: Research update, AAA Foundation for Traffic Safety, [ONLINE] Available: `http://www.aaafoundation.org/pdf/AggressiveDrivingResearchUpdate2009.pdf`. [Accessed 5 August 2013] (April 2009).

[6] How DriveCam works, DriveCam, [ONLINE] Available at: http://www.drivecam.com/our-solutions/how-drivecam-works. [Accessed 5 August 13] (2010).

[7] How it works, AutoHabits, [ONLINE] Available at: http://autohabits.com/how-it-works. [Accessed 5 August 13] (2012).

[8] Fleet safety, FleetMind, [ONLINE] Available at: http://www.fleetmind.com/fleet-management-products/fleet-safety. [Accessed 5 August 13] (2013).

[9] D. A. Johnson, M. M. Trivedi, Driving style recognition using a smartphone as a sensor platform, in: 14th International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2011, pp. 1609–1615.

[10] J. White, C. Thompson, H. Turner, B. Dougherty, D. C. Schmidt, WreckWatch: automatic traffic accident detection and notification with smartphones, Mobile Networks and Applications 16 (3) (2011) 285–303.

[11] M. J. Booysen, J. Gilmore, S. Zeadally, G. J. Van Rooyen, Machine-to-machine (M2M) communications in vehicular networks, KSII Transactions on Internet and Information Systems 6 (2) (2012) 529–546.

[12] D. Trossen, D. Pavel, NORS: An open source platform to facilitate participatory sensing with mobile phones, in: Fourth Annual International Conference on Mobile and Ubiquitous Systems: Networking & Services (MobiQuitous), IEEE, 2007, pp. 1–8.

[13] E. Koukoumidis, M. Martonosi, L. S. Peh, Leveraging smartphone cameras for collaborative road advisories, IEEE Transactions on Mobile Computing 11 (5) (2012) 707–723.

[14] M. V. Ramesh, A. Jacob, R. Aryadevi, Participatory sensing platform to revive communication network in post-disaster scenario, in: 21st Annual Wireless and Optical Communications Conference (WOCC), IEEE, 2012, pp. 118–122.

[15] S. S. Kanhere, Participatory sensing: Crowdsourcing data from mobile smartphones in urban spaces, in: Distributed Computing and Internet Technology, Springer, 2013, pp. 19–26.

[16] C. Costa, C. Laoudias, D. Zeinalipour Yazti, D. Gunopulos, SmartTrace: Finding similar trajectories in smartphone networks without disclosing the traces, in: 27th International Conference on Data Engineering (ICDE), IEEE, 2011, pp. 1288–1291.

[17] B. Predic, Z. Yan, J. Eberle, D. Stojanovic, K. Aberer, ExposureSense: Integrating daily activities with air quality using mobile participatory sensing, in: 11th International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), IEEE, 2013, pp. 303–305.
[18] T. Ichimura, S. Kamada, A generation method of filtering rules of Twitter via smartphone based participatory sensing system for tourist by interactive GHSOM and C4.5, in: 2nd International Conference on Systems, Man, and Cybernetics (SMC), IEEE, 2012, pp. 110–115.
[19] M. Liu, A study of mobile sensing using smartphones, International Journal of Distributed Sensor Networks 2013.
[20] N. De Caro, W. Colitti, K. Steenhaut, G. Mangino, G. Reali, Comparison of two lightweight protocols for smartphone-based sensing, in: Communications and Vehicular Technology in the Benelux (SCVT), 2013 IEEE 20th Symposium on, 2013, pp. 1–6.
[21] Detection of Dangerous Cornering in GNSS Data Driven Insurance Telematics, Accepted for publication in IEEE Transactions on Intelligent Transportation Systems.
[22] J. Wahlstrom, I. Skog, P. Handel, Risk assessment of vehicle cornering events in gnss data driven insurance telematics, in: Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on, 2014, pp. 3132–3137. doi:10.1109/ITSC.2014.6958194.
[23] G. Castignani, T. Derrmann, R. Frank, T. Engel, Driver behavior profiling using smartphones: A low-cost platform for driver monitoring, Intelligent Transportation Systems Magazine, IEEE 7 (1) (2015) 91–102. doi:10.1109/MITS.2014.2328673.
[24] C. Campolo, A. Iera, A. Molinaro, S. Y. Paratore, G. Ruggeri, SMaRTCaR: An integrated smartphone-based platform to support traffic management applications, in: First International Workshop on Vehicular Traffic Management for Smart Cities (VTM), IEEE, 2012, pp. 1–6.
[25] O. Briante, C. Campolo, A. Iera, A. Molinaro, S. Y. Paratore, G. Ruggeri, M. J. Booysen, ITSPhone: An integrated platform for participatory ITS data collection and opportunistic transfer, IEEE Infocom 2013 (2013) 1420–1421.
[26] K. Perera, D. Dias, An intelligent driver guidance tool using location based services, in: 1st International Conference on Spatial Data Mining and Geographical Knowledge Services (ICSDM), IEEE, 2011, pp. 246–251.
[27] K. Ali, D. Al Yaseen, A. Ejaz, T. Javed, H. S. Hassanein, CrowdITS: Crowdsourcing in Intelligent Transportation Systems, in: Wireless Communications and Networking Conference (WCNC), IEEE, 2012, pp. 3307–3311.
[28] X. Zhang, H. Gong, Z. Xu, J. Tang, B. Liu, Jam eyes: A traffic jam awareness and observation system using mobile phones, International Journal of Distributed Sensor Networks 2012.
[29] E. Koukoumidis, L. S. Peh, M. R. Martonosi, Signalguru: leveraging mobile phones for collaborative traffic signal schedule advisory, in: Proceedings of the 9th international conference on Mobile systems, applications, and services, ACM, 2011, pp. 127–140.
[30] M. Lan, M. Rofouei, S. Soatto, M. Sarrafzadeh, SmartLDWS: A robust and scalable lane departure warning system for the smartphones, in: 12th International Conference on Intelligent Transportation Systems (ITSC'09), IEEE, 2009, pp. 1–6.
[31] H. Eren, S. Makinist, E. Akin, A. Yilmaz, Estimating driving behavior by a smartphone, in: Intelligent Vehicles Symposium (IV), IEEE, 2012, pp. 234–239.
[32] J. Dai, J. Teng, X. Bai, Z. Shen, D. Xuan, Mobile phone based drunk driving detection, in: 4th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth), IEEE, 2010, pp. 1–8.
[33] M. Fazeen, B. Gozick, R. Dantu, M. Bhukhiya, M. C. González, Safe driving using mobile phones, IEEE Transactions on Intelligent Transportation Systems 13 (3) (2012) 1462–1468.
[34] P. Mohan, V. N. Padmanabhan, R. Ramjee, Nericell: rich monitoring of road and traffic conditions using mobile smartphones, in: Proceedings of the 6th ACM conference on Embedded Network Sensor Systems, ACM, 2008, pp. 323–336.
[35] A. Ghose, P. Biswas, C. Bhaumik, M. Sharma, A. Pal, A. Jha, Road condition monitoring and alert application: Using in-vehicle smartphone as internet-connected sensor, in: 10th International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), IEEE, 2012, pp. 489–491.
[36] A. Mednis, G. Strazdins, R. Zviedris, G. Kanonirs, L. Selavo, Real time pothole detection using android smartphones with accelerometers, in: Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on, IEEE, 2011, pp. 1–6.
[37] Y. Yang, B. Chen, L. Su, D. Qin, Research and development of hybrid electric vehicles can-bus data monitor and diagnostic system through obd-ii and android-based smartphones, Advances in Mechanical Engineering 2013.
[38] J. Zaldivar, C. T. Calafate, J. C. Cano, P. Manzoni, Providing accident detection in vehicular networks through OBD-II devices and Android-based smartphones, in: 36th Conference on Local Computer Networks (LCN), IEEE, 2011, pp. 813–819.
[39] C. Thompson, J. White, B. Dougherty, A. Albright, D. C. Schmidt, Using smartphones and wireless mobile sensor networks to detect car accidents and provide situational awareness to emergency responders, in: ICST Conference, June, 2010.
[40] V. C. Magaña, M. M. Organero, Artemisa: Using an Android device as an eco-driving assistant, Cyber Journals: Multidisciplinary Journals in Science and Technology: Journal of Selected Areas in Mechatronics (JMTC).
[41] R. Araujo, A. Igreja, R. de Castro, R. Araujo, Driving coach: A smartphone application to evaluate driving efficient patterns, in: Intelligent Vehicles Symposium (IV), IEEE, 2012, pp. 1005–1010.
[42] P. Handel, I. Skog, J. Wahlstrom, F. Bonawiede, R. Welch, J. Ohlsson, M. Ohlsson, Insurance telematics: Opportunities and challenges with the smartphone solution, Intelligent Transportation Systems Magazine, IEEE 6 (4) (2014) 57–70. doi:10.1109/MITS.2014.2343262.
[43] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, H. Balakrishnan, The pothole patrol: using a mobile sensor network for road surface monitoring, in: Proceedings of the 6th international conference on Mobile systems, applications,

and services, ACM, 2008, pp. 29–39.
[44] Sensormanager — android developers, http://developer.android.com/reference/android/hardware/SensorManager.html#getOrientation(float[],float[]), (Visited on 05/07/2015).
[45] G. Milette, A. Stroud, Professional Android Sensor Programming, Wiley.
URL http://sun.eblib.com.ez.sun.ac.za/patron/FullRecord.aspx?p=821861
[46] M. J. Booysen, S. J. Andersen, A. S. Zeeman, Informal public transport in Sub-Saharan Africa as a vessel for novel intelligent transport systems, in: 16th International Conference on Intelligent Transportation Systems (ITSC), IEEE, 2013.
[47] Discovery insure - android apps on google play, https://play.google.com/store/apps/details?id=za.co.discovery.insure.drivingapp&hl=en, (Visited on 05/07/2015).