Transportation Research Part C 79 (2017) 196–206

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Driving analytics using smartphones: Algorithms, comparisons and challenges

Eleni I. Vlahogianni, Emmanouil N. Barmpounakis *

National Technical University of Athens, 5 Iroon Polytechniou Str., Zografou Campus, 157 73 Athens, Greece

ARTICLE INFO

Article history:
Received 3 October 2016
Received in revised form 17 March 2017
Accepted 19 March 2017
Available online 31 March 2017

Keywords:
Driver's behavior
Critical driving patterns
Smartphones
Rough set theory
Machine learning

ABSTRACT

The present work investigates the use of smartphones as an alternative to gather data for driving behavior analysis. The proposed approach incorporates i. a device reorientation algorithm, which leverages gyroscope, accelerometer and GPS information, to correct the raw accelerometer data, and ii. a machine-learning framework based on rough set theory to identify rules and detect critical patterns solely based on the corrected accelerometer data. To evaluate the proposed framework, a series of driving experiments are conducted in both controlled and "free-driving" conditions. In all experiments, the smartphone can be freely positioned inside the subject vehicle. Findings indicate that the smartphone-based algorithms may accurately detect four distinct patterns (braking, acceleration, left cornering and right cornering) with an average accuracy comparable to other popular detection approaches based on data collected using a fixed position device.

1. Introduction

Driving analytics is a research field where smartphones are starting to have significant impact. Driving analytics mainly include the identification of extreme driving patterns, for example speeding, harsh braking/acceleration, harsh cornering (left or right turn with high speed), and harsh lane changing (Handel et al., 2014; Johnson and Trivedi, 2011). The above information can be leveraged to develop individual-based Intelligent Transportation Systems (ITS), connected fleet systems, as well as Usage Based Insurance (UBI) schemes (Eren et al., 2012; Handel et al., 2014; Husnjak et al., 2015; Johnson and Trivedi, 2011; Meseguer et al., 2013; Saiprasert and Pattara-Atikom, 2013; Tselentis et al., 2016; Wahlstrom et al., 2015; White et al., 2011). For a recent review of smartphone-based applications for driver's behavior monitoring see (Predic and Stojanovic, 2015).

The main advantage of using smartphones for extracting driving analytics is that they can form a non-intrusive environment for continuously collecting rich and more granular data on the actual driving task. As opposed to current naturalistic driving experiments, smartphone probes are a much more sustainable solution, when compared to instrumented vehicles. Instrumentation is costly and difficult to install and maintain, especially when experiments involve a large volume of vehicles (Vlahogianni et al., 2013, 2014). Smartphones are equipped with a variety of sensors (e.g. global navigation satellite system-GNSS, inertial measurement unit-IMU) and can alleviate the cost constraints.

The use of smartphones for monitoring vehicle's driving characteristics comes as a novel direction at a series of other telematics solutions, such as in-vehicle data recorders (Paefgen et al., 2014; Toledo et al., 2008), as well as the popular fixed

* Corresponding author.
E-mail address: manosbar@central.ntua.gr (E.N. Barmpounakis).

http://dx.doi.org/10.1016/j.trc.2017.03.014
0968-090X/

position devices plugged into the on-board diagnostics (OBD-II) port of vehicle (Amarasinghe et al., 2015; Tselentis et al., 2016). While OBD-II solutions have been pointed as the most accurate market solution for collecting vehicle's characteristics data, smartphones are gaining ground due to several advantages over OBD-II devices. First, they are a low-cost solution when compared to OBD systems that cost more than 100$ each, plus cloud or other services to transmit, analyze and maintain data. Second, smartphones are quite transparent data collection mechanisms; the users may resort to their data to check their behavior or even interact with them through the proper software, usually developed to work in a smartphone. Evidently, smartphones entail several disadvantages, such as the battery drain, the arbitrary position in vehicle, the noisy data encompassing various trips with various means of transportation and so on (Handel et al., 2014). These shortcomings may significantly affect the accuracy and reliability of smartphone based data collection systems.

Interestingly, there is little information on how accurate and reliable are smartphone-based collected data compared to other sources of information e.g. fixed GPS devices or Inertial Measurement Units (IMUs) with a combination of accelerometers, gyroscopes, and less often magnetometers. In (Paefgen et al., 2012) a study of driving event detection based on smartphones and OBDs is conducted in order to compare the performance of smartphones against an OBD in-car unit. Results revealed correlation between the events detected by smartphones and ODB devices. Smartphones were found to overestimate critical events when compared to OBDs, whereas roadway conditions and smartphone's position significantly affect the performance of smartphones. In (Saiprasert and Pattara-Atikom, 2013) it is shown that speed data from Smartphone are as accurate as the values from car's speedometer with a speed offset of approximately 4 km/h. In (Bergasa et al., 2014) experiments are conducted to detect driver's behavior and compared different approaches, but for fixed smartphone positions. Data fusion algorithms were developed in (Chowdhury et al., 2014; Ghose et al., 2016) based on GPS and inertial collected data to infer the speed of a vehicle and concluded that the estimated speed is comparable to the OBD based tachometer readings.

The above approaches converge to a single idea: to detect driving patterns one has to analyze the accelerometer and/or GPS data and trace - with some method - the outlying behavior. Literature has emphasized that GPS speeds are reliable and accurate, but need significant processing in order to be leveraged for extracting driver's analytics (Chowdhury et al., 2014; Ghose et al., 2016). On the other hand, the accelerometer data suffer from extreme levels of noise and are sensitive to device repositioning.

The present work proposes an algorithmic toolbox based on machine learning in order to leverage the data collection capabilities of smartphones and generate information on the driver's behavior. Two distinct research questions are treated:

- How to account for the noise induced in the sensor's signals due to the arbitrary positioning of a smartphone inside a moving vehicle or the differences between sensors and devices?
- What is considered an event and how to set the thresholds between regular and irregular (extreme) driving?

The proposed approach entails three processing steps: first, a simple and flexible device repositioning algorithm is implemented in order to continuously detect the true position of the smartphone inside a vehicle and correct the recorded data, so as to match its orientation to the one of the vehicle. Second, a flexible peak detection algorithm is developed to pick up irregularities in the rotated accelerometer data and tag them as possible harsh driving events. Third, a parameter free model based on rough set theory is developed to find the optimum accelerometer threshold values and extract critical driving events, specifically, harsh acceleration, harsh braking, and harsh left and right cornering. The accuracy of the proposed approach is compared to a fixed position device, which encompasses a threshold-based approach to detect critical events based on accelerometer data. The evaluation is based on two distinct experiments: a "free driving" experiment, where the subject driver conducts his daily driving habits with no intervention (naturalistic driving experiment) and a controlled experiment, where an observer monitors and annotates the behavior of drivers during the driving task.

2. Smartphone-based driving analytics and modeling

2.1. Critical driving patterns from smartphone data

Smartphones are equipped with a variety of sensors, such as motion sensors (e.g. accelerometer and gyroscope), position sensors (e.g. magnetometer), global navigation satellite system (GNSS) receivers, environmental sensors (barometers, photometers, and thermometers), microphone, cameras, etc. Although, theoretically each sensor may add knowledge to the driver's behavior and conditions during driving, literature has systematically addressed the problem of harsh driving events detection using GNSS, as well as 3-axis accelerometer, gyroscope and magnetometer data (Wahlstrom et al., 2015).

Based on the above sensors various driving patterns may be detected. A concise review of the indicators and metrics extracted from smartphones and may characterize driving behavior is provided in Handel et al. (2014). The most popular of these, often met in usage based insurance schemes, include acceleration/braking, speeding, as an absolute value or in relation to a specific limit, right/left cornering and swerving (including lane changing). The frequency of these metrics in relation to the distance traveled, the time of day, etc. may be used to quantify risky driving, to rate driver's behavior, to explore aggressiveness during driving and so on (Chakravarty et al., 2013; Johnson and Trivedi, 2011; Musicant et al., 2014).

The task of converting data from multiple sensors that are often noisy and irregular to critical and meaningful information on the manner a driver behaves on the road is not easy. Literature has to demonstrate both simple rule-based approaches and more complex machine leaning paradigms. In Johnson and Trivedi (2011), a system is proposed that uses Dynamic Time Warping (DTW) and smartphone based sensor fusion approaches (accelerometer, gyroscope, magnetometer, GPS, video) to identify aggressive and driving. A non-intrusive method is proposed in Bhoraskar et al. (2012) that uses accelerometer, GPS and magnetometer sensor readings for traffic and road conditions detection. A mobile application is developed in Castignani et al. (2013), which using data from GPS, accelerometer, magnetometer, gravity sensors, and a Fuzzy Inference System, it produces a driver's rating based on driving behavior. A smartphone based solution is proposed in Saiprasert et al. (2013) to detect driving patterns using accelerometer and magnetometer data and pattern matching approaches. They compared their approach to a threshold-based approach (using in GPS data and showed that the pattern-based provides improved performance over the simple threshold based approach. Recently, (Predic and Stojanovic, 2015) developed advanced Machine Learning classifiers to detect harsh driving patterns and reported improved results when compared to classical methods of activity analysis from accelerometer data based on statistical metrics of standard deviation, entropy, energy, mean value, etc.

This characteristic makes traditional and well researched accelerometer data analysis techniques for physical activity detection mostly inadequate in traffic domain. In the second part of this paper we use this crowdsourced traffic events information in a demo service for drivers that offers dynamic routing and traffic events notifications and warnings.

Following current practice, the proposed system to quantify the behavior of drivers using smartphones aims to identify the following non-exhaustive list critical harsh events: i. Braking, ii. Acceleration, iii. Left/Right Cornering. In the present work, these driving events are reflected as peaks in the time series of accelerometer data in two out of the three axes. In a system, which needs "no user involvement" and operates in such way that it maximizes the reliability of the recorded data, two questions should be asked: i. which peak can be considered as an event? and ii. how to identify which event each peak corresponds to?

Intuitively, one could understand that not all peaks correspond to critical events, due to noise and the oscillatory nature of the time series of acceleration data. Consequently, one way to think of the specific problem is through finding specific thresholds in acceleration x and y or z axis to decide which peak may reflect critical driving behavior.

However, when it comes to annotating each peak to a specific event, the complexity of the problem increases in relation to the device positioning inside the subject vehicle. Evidently, the smartphone may be placed arbitrary in a vehicle (front seat, cup holder, back seat, bag, etc.), face towards any direction and even change position while driving. The various manners a driver may place the device inside the vehicle increases the complexity of the problem of detecting driver's behavior by increasing the noise in the observed time series and the uncertainty in detecting peaks in signals (Alanezi and Mishra, 2015). To illustrate the above, consider that the smartphone is placed in a landscape position seen in Fig. 1.

According to the direction of motion, when the driver accelerates, the device will record a negative acceleration in the y axis, and the opposite will occur in case of a harsh braking. In the case of abrupt left cornering, the device will be pushed to the right and a positive acceleration in the x axis will be recorded. If the device changes its position for some unknown reason, the specific time interval of position change should be detected and the new position of the phone should be recognized before proceeding to detecting the events. The above-described problems will be addressed in the following sections through two distinct modeling steps: i. The device positioning and reorientation algorithm and ii. The threshold rule induction classifier.

2.2. Device positioning and dynamic reorientation

This problem of identifying and correcting the positioning of a smartphone can be treated by applying strategies that match the orientation of the device with the one of the vehicle, e.g. by computing Euler angles. Reorientation strategies,

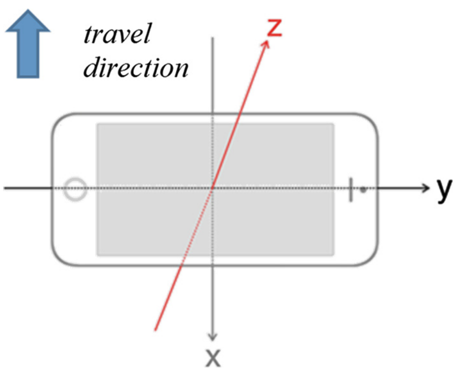

Fig. 1. Example of device positioning and direction of travel forces.

although easy to calculate, produce sensor data with significant errors, mainly because the angles evolve over time and exhibit significant noise. It should be noted that accelerometer values do not include the gravity component, so that all values are around 0 g. Therefore, to produce an efficient reorientation strategy, the main issue is to implement an efficient approach to assess when and how to correct the orientation of the device with respect to the orientation of the vehicle. In this paper a simple and flexible, yet, dynamically updated reorientation algorithm is implemented, which corrects the sensors' signals to address the uncertainties that stem from the arbitrary positioning of smartphones inside vehicles. The concept is to shift the smartphone's coordinate to match the vehicle coordinates as seen in Fig. 2. This is done using accelerometer, gyroscope and GPS data. Magnetometer data is not used as it may exhibit unpredictable errors due to environmental magnetic disturbances (Kang et al., 2012).

To reorient the smartphone-referenced coordinate system in relation to the vehicle coordinate system three-rotation angular parameters need to be estimated. According to Euler's rotation theorem, any rotation may be described using three angles. As seen in Fig. 2, α, β and φ represent the three Euler angles around the x′ axis (roll), y′ axis (pitch) and z′ axis (yaw) respectively. Then, the rotations are written in terms of rotation matrices R_x, R_y and R_z and a general rotation R can be estimated as follows:

$$R = R_z R_y R_x \tag{1}$$

The component rotations are given by the following matrices:

$$R_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\beta & -\sin\beta \\ 0 & \sin\beta & \cos\beta \end{pmatrix}, \; R_y = \begin{pmatrix} \cos a & 0 & -\sin a \\ 0 & 1 & 0 \\ \sin\alpha & 0 & \cos a \end{pmatrix}, \; R_z = \begin{pmatrix} \cos\varphi & \sin\varphi & 0 \\ -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{2}$$

Then the reoriented coordinate system can be found from:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = R \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{3}$$

To estimate the respective rotation angles, we use accelerometer and gyroscope to estimate β and α as follows:

$$\begin{aligned} a &= \sin^{-1}(a_x) \\ \beta &= \sin^{-1}(a_y) \end{aligned} \tag{4}$$

For estimating the rotation angle φ, with the combined use of GPS and gyroscope, the orientation of the device inside the vehicle and the vehicles' course are matched. Following, by performing the transformation described in Eq. (3), the corrected acceleration signals are produced.

To address the issue of smartphone position changes during driving, an updating mechanism based on calculating the median of each rotational angle in a time window is constructed. The use of median for every angle in 10-s time window suppress the noise and leads to a more stable reoriented acceleration signal. Fig. 3 depicts the time series of angle φ and the corrected time series of angle φ based on the median calculation in 10 s time windows.

Fig. 4 exhibits an example of the reorientation problem of a smartphone that was placed vertical to the car. Let us assume that the car is looking up north. In this case, the yaw angle was about 90° (perpendicular and horizontal to the ground level), while no significant variations appeared in roll and pitch angles. It is seen that the disoriented X axis is similar to the reoriented Y axis (with opposite sign) while the disoriented Y axis looks similar to the reoriented X axis. Obviously, no important changes appear in Z axis in both figures.

The second example illustrates a random orientation of the smartphone that changed during a specific trip. Fig. 5 shows that from 0 to 200 s no significant differences exist between the disoriented and the reoriented acceleration, which means the smartphone's axes were quite close to the car's axes (yaw ≈ 0°).

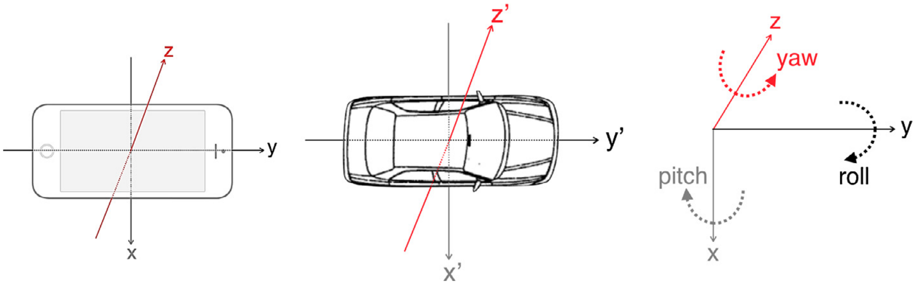

Fig. 2. Smartphone (a) and vehicle (b) coordinate system and Smartphone angular rotations (c).

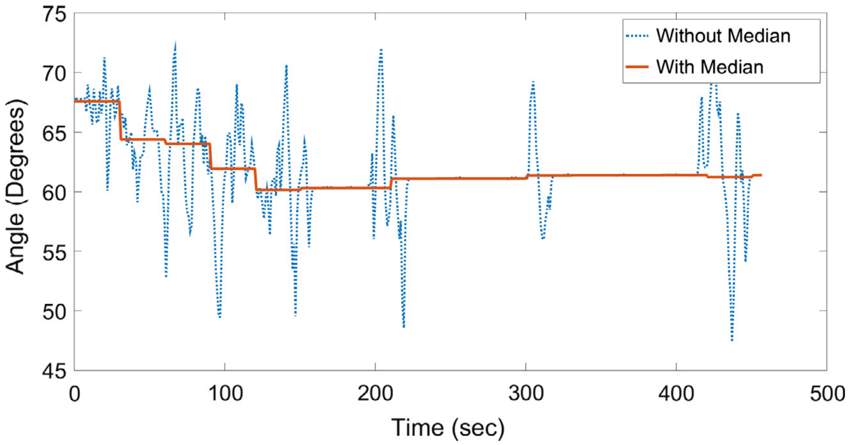

Fig. 3. Rotational angles with or without median estimation.

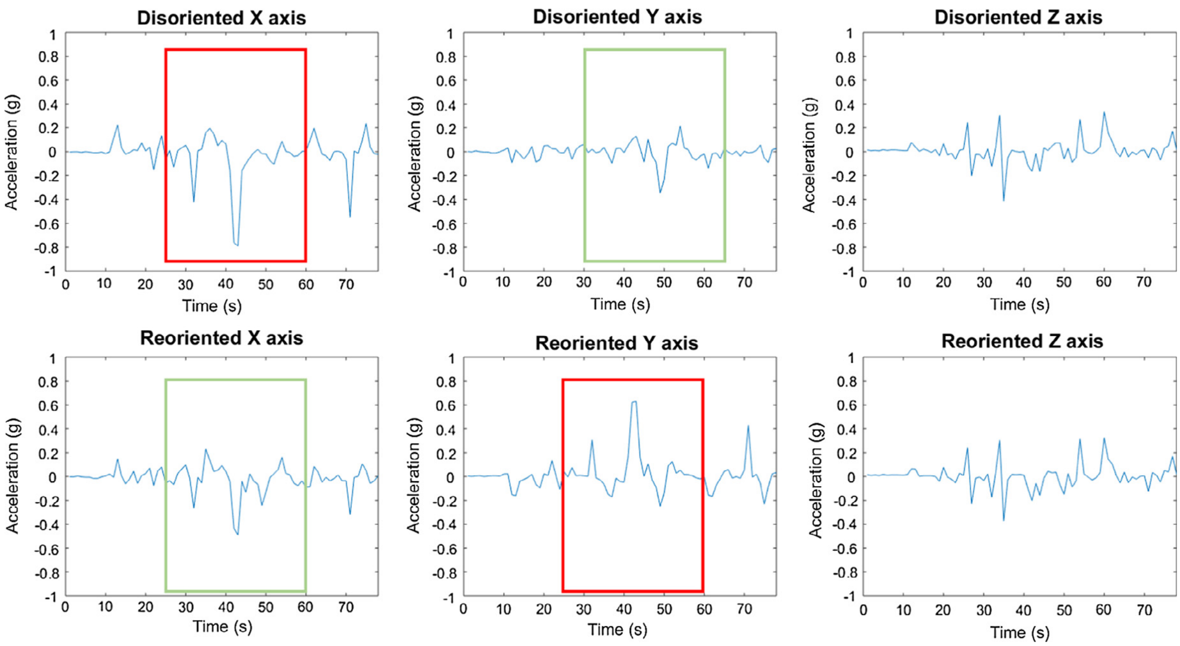

Fig. 4. Reorientation of a smartphone that was placed vertical to the car.

However, in the period between 200 s and 1000 s, the disoriented X axis looks similar to the Y axis which means that the smartphone was placed vertical to the car (yaw = 90°). From 1000 s until the end of the trip, it is seen that the disoriented X axis looks similar to the original X axis with opposite sign, which means that the smartphone is flipped vertically (yaw = 180°) to the original direction.

2.3. Calibration of the event detection algorithm based on rough set models

Following reorientation, the basic modeling decision we have to call is "which is the proper threshold in each of the three axis acceleration data above which a peak may be considered to be a critical (harsh) event (braking, acceleration, cornering)". Evidently, not one threshold may do the job in relation to different events, as well as different smartphone devices and positions inside a vehicle. To deal with these uncertainties and vagueness of the problem at hand, we introduce an approach based on the rough set theory. A rough set is a set characterized by a pair of precise concepts - called the lower and the upper approximations. Rough set theory is suitable for problems entailing a high degree of vagueness and uncertainty and aims to induce decision rules based on a decision variable (e.g. a decision class). Rough set models may be distinguished from clas-

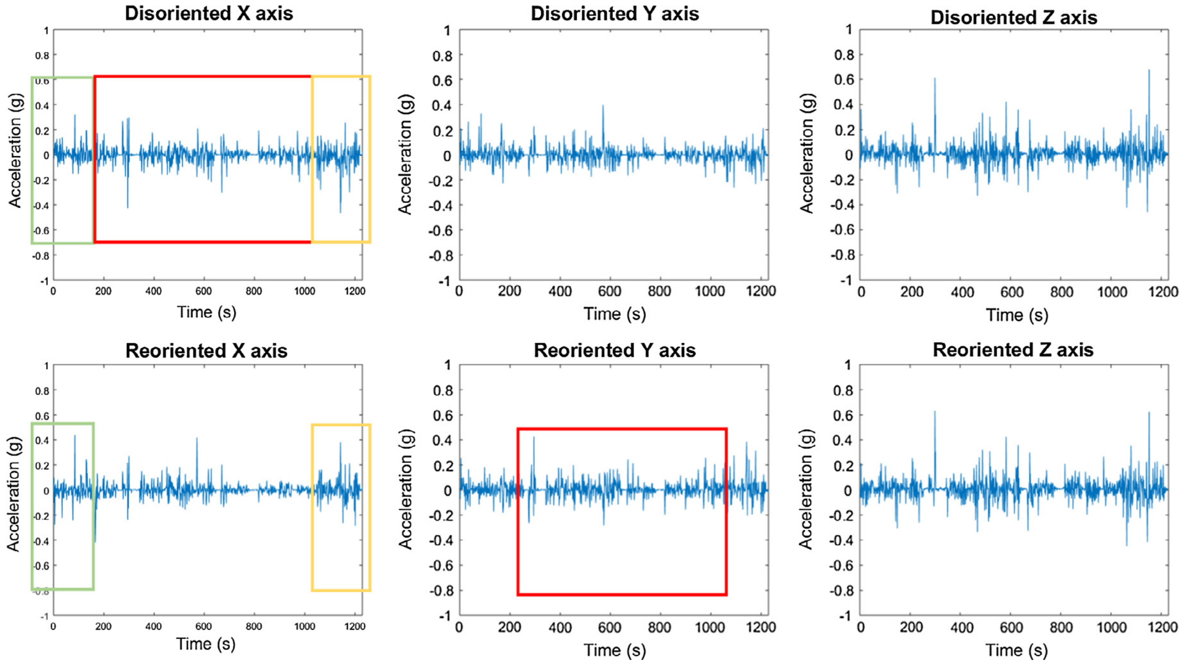

Fig. 5. Reorientation of a smartphone in various positions through a trip. Yaw angle for Green box is 0°, for Red 90° and for Yellow 180°. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

sical machine learning techniques in that they do not correct or aggregate data in order to treat for inconsistencies, but they induce lower (exact) and upper (possible) approximations of all decision concepts.

We implement the MODLEM algorithm (Stefanowski, 2002), which is based on the scheme of a sequential covering and it heuristically produces a minimal set of decision rules for every decision concept (e.g. decision class). The produced set of rules aim to address all (or the most significant) positive examples of a given concept. In the rule induction procedure, the first rule is constructed by choosing sequentially the "best" elementary conditions according to some chosen criteria. The first candidate condition is chosen forming one elementary condition and, if it does not fulfill the requirement to be accepted as a rule, then the second best elementary condition is added to the candidate condition part, and so on. This procedure repeats until a rule can be accepted. When the rule is formulated, all learning positive examples described by this rule are removed from the sample and the rule formation procedure is repeated for the remaining sample data (examples that remain uncovered by the set of rules). Then, the procedure is sequentially repeated for each set of examples from a succeeding decision concept. Elementary conditions are evaluated using class entropy (Stefanowski, 2002).

Let $(U, A \cup \{d\})$ be a decision table of a set of learning examples U and a set of condition attributes A describing examples with $\alpha: \ U \rightarrow V_a$ for every. Moreover, let $a(x)$ be the value of attribute $a \in A$ taken from $x \in U$ and $d \notin A$ a decision attribute which partitions examples into a set of decision classes K_j with $j = 1, \ldots, k$. Every decision rule r describing class K_j may be formulated in the form of *if* P *then* Q where $P = \omega_1 \wedge \omega_2 \wedge \ldots \wedge \omega_p$ is the conditions part and Q is the decision part of the rule (a specific example should be placed in a specific class). The elementary condition of a rule is presented as either $(\alpha < u_\alpha)$ or $(\alpha \geqslant u_\alpha)$, or even $(\alpha = [u_1, u_2])$ from the intersection of two different conditions $(\alpha < u_2)$ and $(\alpha \geqslant u_1)$ such that $(u_1 < u_2)$, where a denotes an attribute and u_α is its value.

There exists a cover $[P]$ of the condition of a decision rule r which satisfies the elementary conditions in P. For a decision rule, $[P] \cap B \neq \varnothing$, where B is a set of examples belonging to class K_j. If $[P] \cap [\omega_i] \subseteq B$, the set of B depends on P. This rule is discriminant as it distinguishes positive examples of class K_j from its negative examples. P should be a minimal conjunction of elementary conditions satisfying this requirement. The set of decision rules R completely describes examples of class K_j, if each example is covered by at least one decision rules. Usually we require to construct such a description by a minimal set of rules. Let P be a set of conjunctions P of rules indicating class K_j, then P is the local covering of examples from B, if the following conditions are satisfied: i. each conjunction P is minimal; ii. $U_{P \in P}[P] = B$, iii. P is minimal, i.e. it has the smallest number of elements. Simply said, the minimum set of rules contains the smallest number of discriminant rules sufficient to cover the set of objects K_j.

3. Driving experiments

Two different types of experiments are conducted. The first type is similar to a naturalistic driving study, but with the exception that the participants were asked to enable a smartphone app, which automatically recoded the smartphone's sensor data, at the onset of a trip. The specific experiment targets to calibrate the smartphone detection process in such a way that it can replicate the detection process provided by an already calibrated OBD-II device. Data were recorded every 1 s. Three different drivers for one month each participated using interchangeably various smartphone and tablet devices. For this experiment, only high end devices were used, namely two iPhones 6, one LG Nexus 5, one iPad mini and one HTC Nexus 9. Although different devices may come with different acceleration detection ranges, it should be noted that the range in which driving events can occur is detectable by most modern hardware. The participants were allowed to place their phones or tablets in any position and they were allowed to talk, text, etc., with their phones during driving. Vehicles were equipped with OBD-II devices, which provided continuous monitoring of vehicles characteristics and driving behavior via calibrated threshold values for accelerometer data. The experiment was conducted on a BMW Series 1, a VW Golf and a Toyota Yaris. Overall, 2145 critical events were collected and analyzed; these correspond to approximately 180 h of travel and 8 critical driving events per trip on average.

The second type of experiments aims to compare the accuracy of the two calibrated devices (ODB-II and smartphone) on unseen harsh event data. To collect harsh event data during driving, a series of controlled experiments were conducted in different routes and with different driving directions. More specifically, first a route inside the NTUA campus road network was selected to evaluate the driving performance of the three drivers based on both smartphone and OBD-II recordings in the present of a trained observer (Fig. 6). Each driver was asked to drive naturally and perform specific driving tasks (braking, acceleration cornering) in predetermined road sections. During driving, the observer inside the car recorded the time a harsh event took place. Every driving event during which the observer felt uncomfortable with the driving or when the driver conducted a risky maneuver was considered as "harsh". Each trip was also recorded in video, so as the observer can have the opportunity to go back and observe the identified events. Moreover, a second series of controlled experiments were conducted in the presence of the observer, in which the three drivers were asked to drive as usual (DAU). In this controlled experiment, the observer had no communication with the driver. The DAU experiment was conducted in a predefined route in a densely-populated road network (Fig. 6). Both types of controlled experiments were conducted during the morning hours. Overall, during the controlled experiments, the observer detected 159 critical events that were further analyzed.

In both experiment types, the analysis is based on the smartphone thresholds that are presented in the previous sections. For the OBD-II device, the thresholds are set to the following values: harsh braking, 0.21 g, harsh acceleration, 0.18 g, harsh cornering (left/right), ±0.18 g.

4. Optimum acceleration thresholds identification

A MATLAB realization of the algorithms for device reorientation and peak detection is developed. Peak detection is accomplished using a simple heuristic algorithm, which traces "peaks" and "valleys" in a time series; given a specific threshold value, the algorithm annotates extreme values as local peaks, only if the series has returned to a low point. In the present work, the developed code is executed for all trips and for different threshold values seen in Table 1; for example, if the threshold is set to 0.16 g, each time the corrected acceleration in y axis (reoriented signal) exceeds the value 0.16 g the peak detection algorithm searches for the peaks (local maxima) and creates an annotated acceleration signal.

Following, a rule induction classifier (MODLEM) is developed with the aim to learn the optimum set of rules (threshold values) using which the detected events from smartphones are as similar as possible to those detected by OBD-II devices. The inputs of the classifier are all combinations of the threshold values of harsh braking, harsh acceleration, right and left harsh

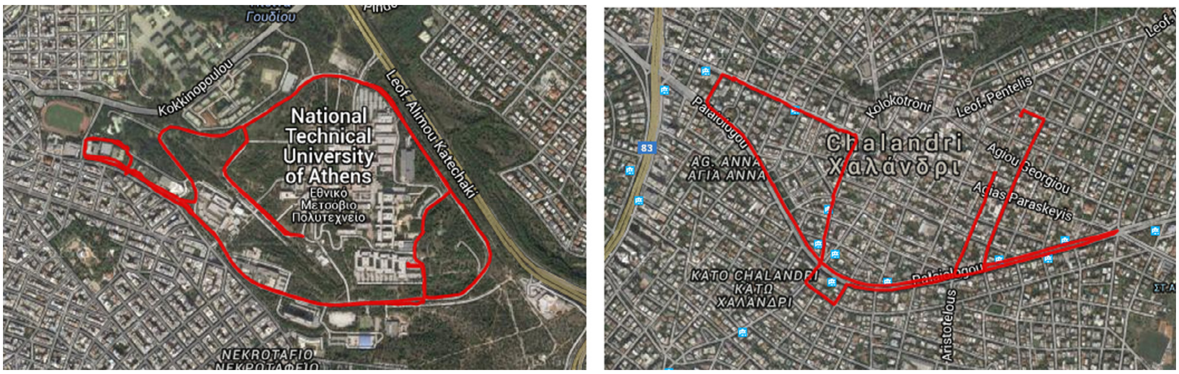

Fig. 6. The routes followed on the controlled (left) and DAU (right) experiments.

Table 1
Classifier input/output specifications.

Variable		Threshold Range (g)
Input		
Harsh braking	y-axis Corr. Acceleration	0.16 to 0.36
Harsh acceleration	y-axis Corr. Acceleration	-0.26 to -0.16
Right harsh cornering	x-axis Corr. Acceleration	0.16 to 0.26
Left harsh cornering	x-axis Corr. Acceleration	-0.26 to -0.16
Output		
CLASS	1: accurate detection, 0: otherwise	

cornering seen in Table 1. The output (CLASS) is a binary 0/1 variable, where 1 is for the cases that the thresholds' set may accurately predict OBD-II recorded driving events with an error of less than 3 events per 100 km (CLASS).

The procedure is as follows: first, the 3D accelerometer data is corrected by applying the reorientation algorithm explained in Section 2.2. Second, for different acceleration threshold values sets in x-axis and y-axis, the peak detection algorithm is executed to detect the critical driving events. Third, the detected events are compared to those identified by the ODB-II devices and the metric of binary class variable depicted the accuracy of the peak detection is estimated. Fourth, the classifier is trained to relate the threshold values sets to the class variable and the low approximation rules are extracted. These rules reflect the boundary values of 3D acceleration data, which are optimized, based on the available driving trips dataset.

The classifier uses the *strength and support* strategy to classify each object to a specific class based on the estimated induced set of rules, where strength is the total number of learning examples correctly classified by the rule during training and support is the sum of scores of all matching rules from the concept (Stefanowski, 2004). The class K_j for which the support is the highest, is the winner class. In the cases where complete matching is not possible, an additional parameter is introduced which quantifies the ratio of the matching conditions to all conditions in the rule (Matching factor). Again, the winning class is the one, which has the maximum support, which is defined as the sum of the product of strength and matching factor.

Classification results based on a 10-fold cross validation are seen in Table 2 with respect to: i. the TPR: rate of true positives (instances correctly classified as a given class), ii. FPR: rate of false positives (instances falsely classified as a given class), iii. Precision: proportion of instances that are actually of a class divided by the total instances classified as that class, iv. Recall: proportion of instances classified as a given class divided by the actual total in that class (equivalent to TP rate), and v. F-Measure: A combined measure for precision and recall calculated as: $2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$. True Positive (TP) refers to how many harsh events were correctly identified, False Negative (FN) to harsh events that were not identified, False Positive (FP) to regular driving that was incorrectly identified as harsh events and True Negative (TN) to regular driving that was not identified as harsh events. The performance of MODLEM algorithm is compared to other previously used models for classification of driving behavior, e.g. C4.5 trees (Predic and Stojanovic, 2015), the Multilayer Perceptron classifier and the ZeroR classifier for benchmark predictions. All algorithms use the re-oriented accelerometer data in order to identify the four different event types.

All classifiers tested are of high accuracy, especially when compared to ZeroR the simplest classification method, which predicts the majority category (class). MODLEM achieves the optimum accuracy. Results further analyzed with respect to each category show that the developed classifier performs equally well (Table 3).

The resulting set of low approximation rules (certain) for those cases where the smartphone could detect the same events as the OBD-II device (CLASS 1) is seen below:

$$R1: \ (HarshLeftTurn \geqslant 0.24\text{ g})\&(HarshRightTurn < -0.24\text{ g})\&(HarshBrake \geqslant 0.21\text{ g})$$
$$R2: \ (HarshRightTurn < 0.23\text{ g})\&(HarshLeftTurn \geqslant 0.24\text{ g})\&(HarshAccelerationin[-0.19\text{ g}, -0.17\text{ g}])$$
$$R3: \ (HarshRightTurn < -0.23\text{ g})\&(HarshBrake[0.24\text{ g}, 0.27\text{ g}])$$
$$R4: \ (HarshBrake < 0.2)\&(HarshLeftTurn < [0.19\text{ g}, 0.21\text{ g}])\&(HarshRightTurn < -0.19\text{ g})$$
$$R5: \ (HarshBrake < 0.17)\&(HarshRightTurn < [-0.25\text{ g}, -0.23\text{ g}])\&(HarshLeftTurn > 0.24\text{ g})$$

Table 2
Comparative evaluation of MODLEM classifier.

	TPR	FPR	PRECISION	RECALL
MODLEM	0.966	0.031	0.967	0.966
MLP	0.932	0.127	0.932	0.932
C4.5 trees	0.94	0.058	0.943	0.94
ZeroR	0.701	0.701	0.491	0.701

Table 3
MODLEM classification results.

	CLASS 0[a]	CLASS 1[a]
TPR	0.963	0.971
FPR	0.029	0.037
PRECISION	0.988	0.919
RECALL	0.963	0.971
F-Measure	0.975	0.944

[a] CLASS 1: correct detection of harsh events having as reference the OBD-II device, 0: otherwise.

The first rule can explain the 54% of the sample, whereas the second and third may describe the 30% of the sample. The last two rules account for the 7% of the sample. Based on the induced rules, the thresholds of the acceleration signals in x and y axis can be set. The threshold for critical braking events is 0.21 g, whereas the threshold for acceleration events is −0.17 g. Regarding left and right harsh cornering, the thresholds equal to 0.24 g and −0.24 g respectively. These thresholds depict the average general picture of driver's behavior as reflected in the available data. These values should be constantly estimated as the sample of drivers and driving hours are increased. This means that the optimized thresholds may not be transferable. Evidently, as in all data mining approaches, it is the methodological framework that is transferable and should be calibrated every time different data are available or different experiments are conducted (Karlaftis and Vlahogianni, 2011).

5. Driver's analytics from fixed and free position devices

This section presents the results of the smartphone-based detection algorithm on the controlled driving experiments, which are, then, compared to the detected events from the OBD-II device. In this case, the modeler knows the actual events (which are annotated by the observer) and wants to evaluate the detection power of the developed smartphone procedure, in comparison to the OBD-II procedure. The results are presented in Table 4 per category of event and in total.

These results from the two experiments suggest that the total accuracy for Smartphone and OBD-II device is 99.4% and 99.3% respectively. TPR (sensitivity) is 88.1% and 86.6% while the FPR is 0.3% and 0.4% for Smartphone and OBD-II device respectively. The results indicate that both devices have impressive results, when it comes to identifying harsh events.

The weak spot of the smartphone procedure is related to the detectability of the harsh acceleration events. This can be explained by the fact that, in contrast to the braking events that are usually more intense, acceleration events are usually smooth and, thus, reflected to less explicitly observed peaks. For the OBD-II device, no sensitivity in identifying different longitudinal event categories has emerged, since a similar rate in failing to detect both acceleration and braking events can be observed.

As far as lateral events are concerned, the smartphone seems to be sensitive to lateral forces leading to overestimate harsh turning movements, even when the conditions of the experiments are controlled by an observer. For the OBD-II device, this shortcoming did not appear mainly due to its fixed position. Further research is needed in order to understand the relationship between smartphone positioning and detected lateral events.

The results from the second controlled experiment (DAU) are poorer compared to those of the first controlled experiment for both smartphone and OBD-II device (Table 4). In this experiment, the FPR of smartphone algorithm is higher than OBD–II by approximately 20% of 87 harsh events. A closer look at the smartphone sensors data taken from the free driving exper-

Table 4
Classification for Smartphone vs Fixed Device events in controlled and the DAU experiment (in parenthesis the number of events per each category).

	Smartphone				*OBD-II Device*			
	Controlled Experiment							
	Total(72)	HA(27)	HB(30)	HC(15)	Total(72)	HA(27)	HB(30)	HC(15)
TPR	0.903	0.889	0.967	0.800	0.958	0.958	0.963	0.933
FPR	0.001	0.000	0.000	0.001	0.001	0.001	0.000	0.000
ACCURACY	0.998	0.999	1.000	0.999	0.999	0.999	1.000	1.000
PRECISION	0.903	0.960	0.967	0.706	0.920	0.920	0.929	0.875
F-Measure	0.903	0.923	0.967	0.750	0.939	0.939	0.945	0.903
	DAU Experiment							
	Total(87)	HA(33)	HB(34)	HC(20)	Total(87)	HA(33)	HB(34)	HC(20)
TPR	0.770	0.758	0.882	0.600	0.816	0.727	0.971	0.700
FPR	0.006	0.002	0.001	0.002	0.005	0.03	0.01	0.001
ACCURACY	0.991	0.996	0.998	0.997	0.992	0.996	0.999	0.998
PRECISION	0.650	0.625	0.789	0.480	0.703	0.585	0.846	0.667
F-Measure	0.705	0.685	0.833	0.533	0.755	0.649	0.833	0.683

iment shows that the time series of acceleration in all axes are significantly more irregular and exhibit sudden shifts to extreme values. This is probably because the drivers were allowed to perform a variety of behaviors, e.g. talking on the phone, texting, etc. Moreover, there is a variety of micro-movements that might have been recorded as events, as the smartphone could have been placed to any position. The above driver's behavior patterns are not introduced to the modeling, consequently, in the free experiment, their occurrence induce discrepancies.

6. Limitations and challenges

The documented performance is very encouraging, but linked to a number of limitations. In a conceptual level, for the driver's profile to be complete, the list of indicators that the proposed artificial intelligent framework has learnt to detect should be expanded to include at least speeding, lane changing and more advanced indicators such as talking on the phone while driving, texting, or roadway conditions identification.

This expansion will certainly be accompanied by expanding the list of sensors that are currently used to detect critical patterns. The use of corrected GPS speed will be beneficial. Yet, attention should be placed on its reliability and accuracy (Ghose et al., 2016). An example largely documented in literature is the use of GPS to detect speeding that becomes inaccurate around large buildings and in tunnels; therefore, special algorithmic treatment is suggested to acquire a signal that is as accurate as possible (Quddus et al., 2007).

Moreover, according to literature, data should be first filtered to exclude noise coming from trips with other transportation means (bus, metro, etc.), walking, as well as driver from passenger trips. In this paper, all experiments conducted were exclusively trips, where the participant was driving his/her own car. Although, detecting car trips is a relatively straightforward task and is mainly based on speed information from GNSS and accelerometer data or only accelerometer data (Ilarri et al., 2015), passenger/driver trip detection is much more complicated and of outmost importance for accurately identifying driver's profile. Passenger/trip detection requires accurate high-resolution data (probably more than 1 Hz) in order for the micro-activities of users to be analyzed and identify those patterns that may help detect driving (Wahlstrom et al., 2015).

Finally, the identification of the mobile usage can give a better insight not only for identifying the driver's behavior and distraction, but also for clearing out noise coming from rough phone movements. Specifically, smartphone sensors may get triggered by mobile phone usage while driving, which affects the event detection process and the critical events' threshold values. The identification of these movements may reduce the false positive events and improve the detection power of the approach.

7. Conclusions

The present paper attempts to evaluate the use of smartphones as probes to identify critical driving patterns, such as harsh cornering, harsh acceleration and braking. First, a device reorientation algorithm is applied to the raw data, which leverages gyroscope, accelerometer and GPS information to estimate the positioning of the smartphone inside a vehicle and match its orientation to the course of the vehicle. Then, a modeling framework based on rough set theory is developed and evaluated in order to identify rules to detect critical patterns solely based on the corrected accelerometer data in x-axis and y-axis.

Findings indicate that the smartphone may accurately detect four distinct patterns (braking, acceleration, left cornering and right cornering). When compared to OBD-II device detected events, which is found to be highly accurate, the data collection approach based on smartphone sensors may be considered as reliable and accurate, as it identifies specific driving patterns with high accuracy. In a DAU experiment, accuracy drops probably due to macro- (e.g. texting or talking on the phone) and micro- (small displacements from interactions with other objects e.g. keys, bags, etc.) user patterns that are not accounted for in the detection framework.

The reduced performance to the DAU experiments was discussed in the framework of a series of limitations, for example the univariate nature of the detection algorithm (detection based solely on accelerometer data), the signals' noise coming from the mobile use and the device positioning, as well as the detection problems stemming from trips with other transportation means. Although these limitations may significantly affect the driving analytics, yet, their treatment are major challenges, if modelers are to turn to the use of smartphones for driving analytics.

This work provided evidence that, at least algorithmically, such a shift may be accomplished. Nevertheless, the smartphone use should be extensively tested under specific roadway (urban networks, motorways, etc.), weather (rain, fog, etc.), and driver's conditions (fatigue, driving during risky hours, etc.). Future work should also focus on issues such as the diversification of sensors technologies and recordings in relation to smartphone devices of different age, brand, etc.

Acknowledgements

The authors would like to thank Oseven Telematics (https://www.oseven.io/) for their valuable technical support during the data collection period.

References

Alanezi, K., Mishra, S., 2015. Design, implementation and evaluation of a smartphone position discovery service for accurate context sensing. Comput. Electr. Eng. 44 (May), 307–323.

Amarasinghe, M., Kottegoda, S., Arachchi, A.L., Muramudalige, S., Dilum Bandara, H. M.N., Azeez, A., 2015, May. Cloud-based driver monitoring and vehicle diagnostic with OBD2 telematics. In: IEEE International Conference on Electro/Information Technology (EIT), 2015. IEEE, pp. 505–510.

Bergasa, L.M., Almería, D., Almazan, J., Yebes, J.J., Arroyo, R., 2014, June. DriveSafe: an app for alerting inattentive drivers and scoring driving behaviors. In: IEEE Intelligent Vehicles Symposium Proceedings, 2014. IEEE, pp. 240–245.

Bhoraskar, R., Vankadhara, N., Raman, B., Kulkarni, P., 2012. Wolverine: traffic and road condition estimation using smartphone sensors. In: Fourth International Conference on Communication Systems and Networks (COMSNETS), 2012. IEEE, pp. 1–6.

Castignani, G., Frank, R., Engel, T., 2013, October. Driver behavior profiling using smartphones. In: 16th International IEEE Conference on Intelligent Transportation Systems-(ITSC), 2013. IEEE, pp. 552–557.

Chakravarty, T., Ghose, A., Bhaumik, C., Chowdhury, A., 2013, December. MobiDriveScore-A system for mobile sensor based driving analysis: a risk assessment model for improving one's driving. In: Seventh International Conference on Sensing Technology (ICST), 2013. IEEE, pp. 338–344.

Chowdhury, A., Chakravarty, T., Balamuralidhar, P., 2014. A novel approach to improve vehicle speed estimation using smartphone's INS/GPS sensors. In: International Conference on Sensing Technology (ICST).

Eren, H., Makinist, S., Akin, E., Yilmaz, A., 2012, June. Estimating driving behavior by a smartphone. In: IEEE Intelligent Vehicles Symposium (IV), 2012. IEEE, pp. 234–239.

Ghose, A., Chowdhury, A., Chandel, V., Banerjee, T., Chakravarty, T., 2016, January. An enhanced automated system for evaluating harsh driving using smartphone sensors. In: Proceedings of the 17th International Conference on Distributed Computing and Networking. ACM, p. 38.

Handel, P., Skog, I., Wahlstrom, J., Bonawiede, F., Welch, R., Ohlsson, J., Ohlsson, M., 2014. Insurance telematics: opportunities and challenges with the smartphone solution. IEEE Intelligent Transp. Syst. Mag. 6 (4), 57–70.

Husnjak, S., Peraković, D., Forenbacher, I., Mumdziev, M., 2015. Telematics system in usage based motor insurance. Proc. Eng. 100, 816–825.

Ilarri, S., Stojanovic, D., Ray, C., 2015. Semantic management of moving objects: a vision towards smart mobility. Expert Syst. Appl. 42 (3), 1418–1435.

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

Kang, W., Nam, S., Han, Y., Lee, S., 2012, September. Improved heading estimation for smartphone-based indoor positioning systems. In: IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC), 2012. IEEE, pp. 2449–2453.

Karlaftis, M.G., Vlahogianni, E.I., 2011. Statistical methods versus neural networks in transportation research: differences, similarities and some insights. Transp. Res. Part C: Emerging Technol. 19 (3), 387–399.

Meseguer, J. E., Calafate, C.T., Cano, J.C., Manzoni, P., 2013, July. Driving styles: a smartphone application to assess driver behavior. In: IEEE Symposium on Computers and Communications (ISCC), 2013. IEEE, pp. 000535–000540.

Musicant, O., Bar-Gera, H., Schechtman, E., 2014. Temporal perspective on individual driver behavior using electronic records of undesirable events. Accid. Anal. Prev. 70, 55–64.

Paefgen, J., Kehr, F., Zhai, Y., Michahelles, F., 2012. Driving behavior analysis with smartphones: insights from a controlled field study. In: Proceedings of the 11th International Conference on mobile and ubiquitous multimedia. ACM, p. 36.

Paefgen, J., Staake, T., Fleisch, E., 2014. Multivariate exposure modeling of accident risk: insights from Pay-as-you-drive insurance data. Transp. Res. Part A: Policy Practice 61, 27–40.

Predic, B., Stojanovic, D., 2015. Enhancing driver situational awareness through crowd intelligence. Expert Syst. Appl. 42 (11), 4892–4909.

Quddus, M.A., Ochieng, W.Y., Noland, R.B., 2007. Current map-matching algorithms for transport applications: state-of-the art and future research directions. Transp. Res. Part C: Emerging Technol. 15 (5), 312–328.

Saiprasert, C., Pattara-Atikom, W., 2013. Smartphone enabled dangerous driving report system. In: 46th Hawaii International Conference on System Sciences (HICSS), 2013. IEEE, pp. 1231–1237.

Saiprasert, C., Pholprasit, T., Pattara-Atikom, W., 2013. Detecting driving events using smartphone. In: 20th ITS World Congress.

Stefanowski, J., 2002. Bagging and induction of decision rules. In: Intelligent Information Systems. Physica-Verlag HD, pp. 121–130.

Stefanowski, J., 2004. The bagging and n2-classifiers based on rules induced by MODLEM, w: Tsumoto, S., et al. (red.), Proceedings of the 4th Int. Conference Rough Sets and Current Trends in Computing, RSCTC'2004, Uppsala, Sweden, June 2004, Lecture Notes in Artificial Intelligence, vol. 3066, Springer-Verlag, 2004, str. 488–497.

Toledo, T., Musicant, O., Lotan, T., 2008. In-vehicle data recorders for monitoring and feedback on drivers' behavior. Transp. Res. Part C: Emerging Technol. 16 (3), 320–331.

Tselentis, D., Yannis, G., Vlahogianni, E.I., 2016. Innovative motor insurance schemes: A review of current practices and emerging challenges. Acc. Anal. and Prevent. 98, 139–148.

Vlahogianni, E.I., Yannis, G., Golias, J.C., 2013. Critical power two wheeler riding patterns at the emergence of an incident. Accid. Anal. Prev. 58, 340–345.

Vlahogianni, E.I., Yannis, G., Golias, J.C., 2014. Detecting powered-two-wheeler incidents from high resolution naturalistic data. Transp. Res. Part F: Traffic Psychol. Behav. 22, 86–95.

Wahlstrom, J., Skog, I., Handel, P., 2015. Detection of dangerous cornering in GNSS-data-driven insurance telematics. In: IEEE Transactions on Intelligent Transportation Systems, available online.

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