Effective Car Collision Detection with Mobile Phone Only

Mateusz Paciorek[1], Adrian Kłusek[1], Piotr Wawryka[1], Michał Kosowski[1], Andrzej Piechowicz[1], Julia Plewa[1], Marek Powroźnik[1], Wojciech Wach[2], Bartosz Rakoczy[1], Aleksander Byrski[1], Marcin Kurdziel[1], Wojciech Turek[1]

[1] AGH University of Science and Technology, Krakow, Poland
[2] Institute of Forensic Research, Kraków, Poland

Abstract Despite fast progress in the automotive industry, the number of deaths in car accidents is constantly growing. One of the most important challenges in this area, besides crash prevention, is immediate and precise notification of rescue services. Automatic crash detection systems go a long way towards improving these notifications, and new cars currently sold in developed countries often come with such systems factory installed. However, the majority of life threatening accidents occur in low-income countries, where these novel and expensive solutions will not become common anytime soon. This paper presents a method for detecting car collisions, which requires a mobile phone only, and therefore can be used in any type of car. The method was developed and evaluated using data from real crash tests. It integrates data series from various sensors using an optimized decision tree. The evaluation results show that it can successfully detect even minor collisions while keeping the number of false positives at an acceptable level.

Keywords: vehicle safety · collision detection · sensor data processing · decision tree

1 Motivation

According to the World Health Organization [9] 1.35 million people died in 2016 in road traffic. What is even more disturbing, this number has been constantly growing over the last 20 years. Despite the fast technological development observed in the automotive industry, providing efficient safety features is still a great challenge. Development of widespread, affordable means of improving road safety and post-crash care should receive more attention.

Severe road accidents, which threaten human lives, require immediate help of medical rescuers. The need for minimizing the time required to receive help after an accident is obvious, therefore a lot of effort has been put into organizing efficient rescue services. An immediate notification with detailed information about the location and expected consequences of the accident is a crucial element — one which can be improved by using automated collision-detection and classification systems. Such a solution would be especially important when victims are not able to call for help by themselves.

Solutions supporting emergency calls and automated crash detection are not a novel concept. In fact, the need for and the advantages of such systems are already recognized by many legislator worldwide — an interesting overview of acts, both planned and in

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

force, can be found in a report created by the ITS mobility cluster in Northern Germany [16]. The most advanced and strict regulations have been introduced in the Russian Federation where all cars registered after 2017 (new and imported) have to be equipped with the Accident Emergency Response System, also called ERA-GLONASS [19]. A similar system is being introduced in the European Union where new cars have to be fitted with an automatic emergency call system [11] called eCall. Simpler solutions, which do not interfere with a car's internal systems, are introduced by some insurance companies.

These are definitely valuable steps towards the reduction of deaths on roads. However, before these legal acts have any significant effects, many people will die. Older cars are still present on European roads, not mentioning less-developed countries, where the majority of fatal accidents occur.

The aim of the presented research is to develop a method for detecting car collisions which could be used in both new and older cars and could become a widespread, common solution. There are three main requirements that have to be met:

- independence of cars embedded systems which allows applicability in any car,
- reliability, which provides high accuracy in detecting collisions together with a low number of false alarms,
- availability and very low price which would motivate people to use the solution on a daily basis.

In the paper we are presenting the whole process of developing such a solution. After analyzing the existing approaches, the first step was to obtain data from real collisions. Data from 22 various crashes, recorded with different sensors, has been carefully analyzed in order to verify the solutions described in literature. The conclusions let us identify major issues and challenges, and create a novel method for detecting collisions. The method has been developed and carefully evaluated, providing very promising results.

2 Existing Approaches

Detection of car accidents is a problem that may be solved using either car-based or environmental sensors. Environmental sensors are usually based on cameras and the image is processed in order to signal an accident to some monitoring person (see, e.g. [6]). Of course environmental sensors are not mobile, unless mounted on a helicopter or a drone, so their usability is limited to the vicinity of well-known dangerous spots on the road. A real flexibility and usability can be delivered by telematics systems mounted inside the car participating in traffic.

Focusing on car-based crash detection methods, one can easily distinguish the following types according to the sensing system used:

- car-internal,
- dedicated device (beacon-based),
- universal device (smartphone-based).

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

A very good example of a **car-internal system** is eCall. This system must be mounted in all new cars sold in the EU starting from March 31st, 2018. The system can be activated either manually or automatically (monitoring e.g. the airbag status) and the infrastructure of mobile phone networks has undergone appropriate adaptation in order to support this system EU-wide (cf. the EC Regulation [11], making the vehicle fit for eCall, and EC Directive [10], making the public infrastructure fit for eCall).

In Russia, a fully interoperable system called ERA-GLONASS is being deployed with the aim to require an eCall terminal and a GPS/GLONASS receiver in new vehicles by 2015-2017. As ERA-GLONASS was the first system that started to operate, eCall is based on its technology [3]. At the same time, in North America, a similar service is provided by GM via their OnStar service [8].

Considering **beacon-based systems**, those are usually used by insurance companies and the installation of such a system in the car usually affects the price of the insurance. Selecting a third-party sensor makes it possible to utilize crash-detection facilities in older cars where no internal systems are present. A good example is the Octo system, which puts together a smartphone and a dedicated sensor, attached to the windshield. The system provides services like fleet monitoring, crash detection, driving-style modelling and even gamification in order to properly motivate the drivers [7]. A similar solution is the IoT DriveWell tag offered by Cambridge Mobile Telematics [13] or the PZU-GO offered by PZU [5]. Bosch offers a Telematic eCall Plug that can be configured for monitoring of driving style or for signalling a crash using the eCall connectivity [4].

The presented solutions, based on embedded car sensors or dedicated sensing devices, are gaining popularity in well-developed countries among experienced drivers. At the same time the majority of severe car accidents involve different drivers – often less experienced and using older cars. These people simply cannot afford the existing safety features. A reliable **Smartphone-based** accident detection system can become a solution to this situation.

Some research in this area also exists. Thompson et al. [14] describe an accident detection and reporting system that uses smartphone accelerometers to detect collisions. To decrease the number of false-positive detections, they trigger an alarm only when a GPS sensor indicates that the smartphone is moving faster than 15 miles per hour and when the recorded acceleration exceeds 4G. When a collision is detected, the smartphone app sends a notification to the central server. The notification includes location data and collision characteristics, such as the recorded acceleration and the vehicle's speed. In a follow-up work [17] this system was extended to include acoustic data in the collision detection procedure. The follow-up work also reports an evaluation against a publicly available dataset with acceleration readings from real accidents. Zaldivar et al. [18] describe an Android application that detects car collisions by monitoring smartphone accelerometer readings and the airbag status reported by a Bluetooth-connected OBD2 interface. An accident is suspected when the accelerometer readings exceed 5G or the airbags are triggered. The accident alarm can be canceled by the smartphone owner within 1 minute of the triggering event. Afterwards an accident notification is send via a text or an e-mail message. Finally, Amin et al. [1] describe and evaluate a collision detection procedure that employs accelerometer readings and GPS location

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

data. However, unlike the works mentioned previously, they use a dedicated accelerometer unit and a dedicated GPS receiver rather than smartphone sensors.

One of the real-world smartphone-only crash detection and notification applications is the SOSmart automatic crash detection app [12] designed by a startup based in Santiago, Chile. The authors claim that their crash detection algorithms are based on data gathered by the National Highway Traffic Safety Administration [15]. They need GPS data in order to detect a collision and have implemented detection of smartphone dropping, claiming that the acceleration readings in the case of an accident reaches hundreds of G while a fall leads to a reading of several G. There is no publicly available data (nor are there research articles) regarding the actual efficacy of the proposed application. Its last version was released to the Apple iTunes store on January 30th, 2017 (as the app is already unavailable through Google Play), and the user reviews are unfortunately very unfavourable, therefore while we fully acknowledge the effort of this startup company, we are unable to treat this application as a reliable reference point for presenting our research results.

The experiments presented in this paper show that it is not possible to create a credible crash-detection method by using only phone accelerometer readings threshold. The limitations and variety of sensors installed in contemporary smartphones require far more sophisticated algorithms, which will be presented in the following sections.

3 Crash Tests and Data Acquisition

To acquire the necessary data, we prepared dedicated mobile applications. The applications were intended to run as a background service and collect all possible information available on the mobile device. The following list enumerates and describes data gathered by our software:

- GPS - latitude, longitude, bearing and speed,
- accelerometer - acceleration for three coordinate axes,
- gyroscope - rate of rotation for three coordinate axes,
- magnetometer - magnetic field for three coordinate axes,
- gravity - force of gravity for three coordinate axes,
- rotation vector - orientation of the device.

It is important to note that many smartphone models are not equipped with all the sensors. To have reliable representation of a typical driver, we gathered phone usage statistics from the Internet. Based on that research, we selected 20 leading phone models from all popular manufacturers. The selected models varied from the cheapest ones up to flagship models. Thanks to this variety of phones, we have the access to sensor chips which vary up to the ranges and thresholds of measurements.

The applications were used in our day-to-day life for a few months. During that time many different types of drives were recorded: traffic jams while driving to work in a big city, casual rides through towns and country roads, and long journeys on highways. Selected parts of the collected data were then used as a counter example for recorded crashes.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

In order to collect reliable set of data from accidents we conducted series of crash tests using real cars. In order to verify the smartphone measurements, we installed a professional device, the PicDAQ5[2] (Data Aquisition Platform) designed by the Dr. Steffan Datentechnik for vehicle dynamics and crash test research. The basic technical data of the device:

– two 3-axial accelerometers ($\pm1.5g$ and $\pm200g$) for vehicle dynamics and impact testing respectively,
– one 3-axial angular velocity gyro-sensor ($\pm300deg/s$ for roll, pitch and yaw),
– 15 analog input channels (12 bit resolution),
– four digital inputs,
– user selectable sampling rate up to 1 kHz per channel (in our measurements: 500 Hz),
– 15 analog input channels are alternative usable for wheel revolutions, steering wheel angle and other measurements,
– 5 Hz GPS receiver,
– PC-based analysis software – PocketDAQ Analyzer.

The first set of experiments was conducted without drivers inside cars. Three vehicles were used to perform tests that were intended to imitate casual accidents in a parking lot, light clashes with one stationary car, and finally a head-on collision (see Figure 1).

Fig. 1: Head-on collision at 40.3 kph.

Because of the fact that there was no stuntman, we had limited control over the vehicles. Moreover, those tests could not be performed at very high speeds. During the crashes, the smartphones were placed in storage compartments near the gear shift and on the doors. There was always one device mounted using a mount holder on the windshield.

The second event was set up by our partner, PZU (the largest insurance company in Poland and at the same time one of the EU-wide players in the insurance market), at a race course in Poznań, the largest track in Poland. All experiments were performed by

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

professional stuntmen. During those tests we had an opportunity to collect data from different scenarios. The first one was driving onto a curb at a high speed. That scenario was followed by a rear collision and a head-on collision with a tree. (Fig. 2). The final test was a side collision at the highest speed recorded. The phones were mounted similarly as in the first crash event.

Fig. 2: Results of the collision with a tree (27.7 kph).

To sum up, in both experiments we have collected data from 22 unique crash events. We have prepared 53 crash examples collected by different phones, including 36 crash examples gathered by phones with gyroscope.

4 Proposed Solution

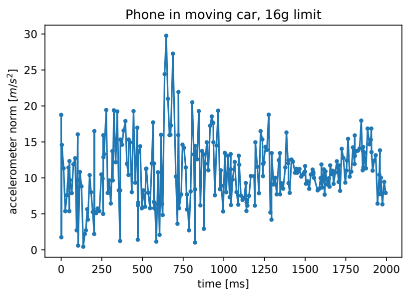
4.1 Data Analysis

The collected raw data covered a time frame that included the vehicles approaching one another before the collision and, in most cases, moving away from each other afterwards as well. To be able to analyze only those readings representing the behavior of the vehicles during a crash, we removed the unnecessary portions of the data using recorded footage of the events. In most cases the time span of useful data covers 1-3 s, of which 200-400 ms represent the time of physical contact between the vehicles.

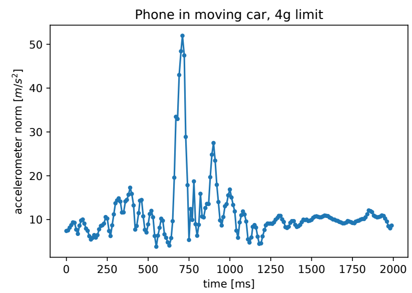
Accelerometer is the most commonly used sensor in this application, as it is designed to measure the forces acting on the object to which it is attached. A sample of accelerometer data collected from one of the collisions can be seen in Figure 3. In this specific event, one vehicle was stationary with the handbrake engaged and its front oriented towards the second, moving vehicle. The collision occurred at a speed of 20 kph, the cars hit each other with the left corners of their hoods. Subfigures of Figure 3 represent the magnitudes of acceleration vectors recorded during the same event by different devices:

(a) phone in the moving vehicle, accelerometer limited to 16g on each axis,
(b) phone in the moving vehicle, accelerometer limited to 4g on each axis,
(c) phone in the stationary vehicle, accelerometer limited to 4g on each axis,

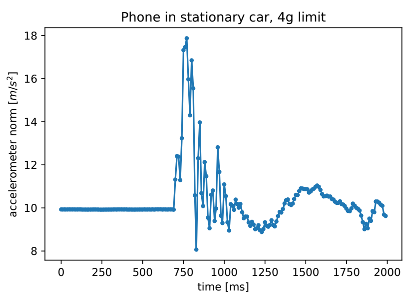
ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24



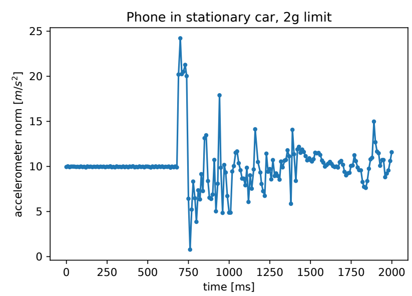
(a)



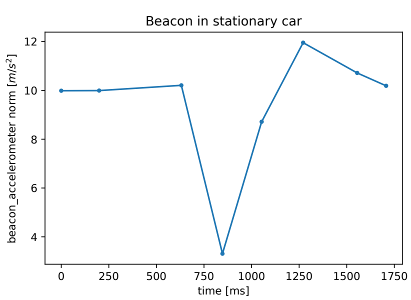
(b)



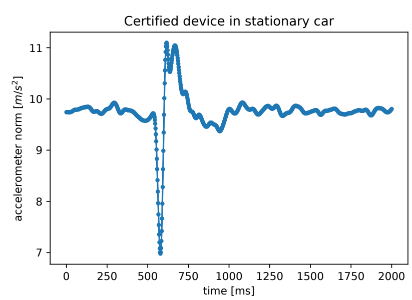
(c)



(d)



(e)



(f)

Fig. 3: Comparison of accelerometer data collected with different devices

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

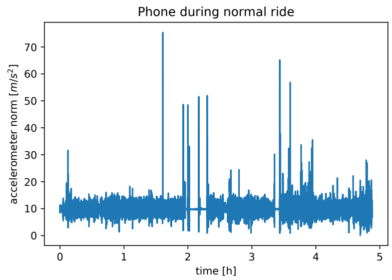


Fig. 4: Accelerometer readings collected during normal ride

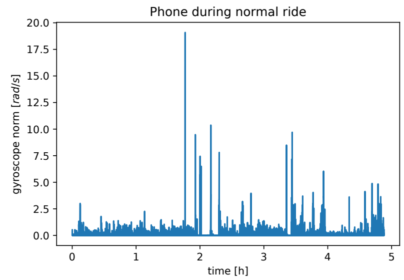


Fig. 5: Gyroscope readings collected during normal ride

(d) phone in the stationary vehicle, accelerometer limited to 2g on each axis,
(e) simple bluetooth beacon with accelerometer in a stationary vehicle,
(f) certified device in the stationary vehicle.

It is easy to notice that the frequency of the data collected by the beacon disqualifies it from being used for reliable crash detection. Only one reading represented on the chart (e) captured a significant acceleration value change. All the other devices registered clearly visible spikes of acceleration. Comparing the data collected by phones to the certified device (e), phone sensors register a lot of noise in addition to a spike caused by the collision. Nonetheless, the presence of unusually high or unusually low values suggests that it is possible to set a threshold – any values exceeding this threshold indicate an occurrence of a crash.

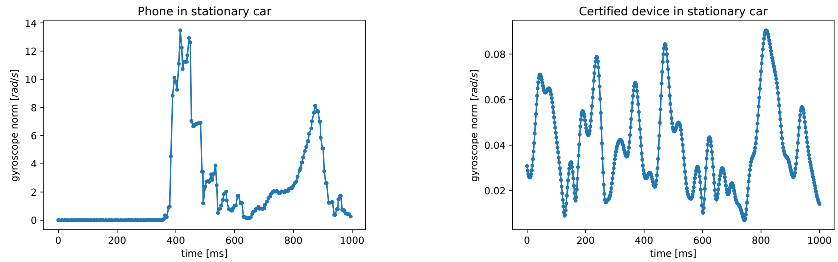
This method has proven to be incorrect due to the capability of phone sensors to register other vehicle activities. Figure 4 shows data collected during an ordinary ride, during which no collision occurred. The accelerometer recorded several significantly high values, in some cases higher than the values captured during the crash presented in Figure 3. The source of those values is not certain, however further synthetic tests indicated that it could be one of following:

– phone being used during the ride,
– phone falling on the floor of the vehicle,
– phone being located in the car door storage compartment while the door is being rapidly closed.

All of these possibilities are unavoidable in case of any device that is not fixed to the body of the car.

We applied a similar analysis process to gyroscope data. The charts presented in Figure 6 show that the certified device attached to the vehicle did not register any significant angular velocity (b), but the phone gyroscope observed a large spike (a). These result were promising, showing that an unattached device is capable of registering much more significant angular motion compared to a more massive vehicle. Unfortunately, an analysis of the readings from an ordinary ride shown in Figure 5 suggests that the threshold method would yield numerous false positives in this case as well.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24



(a) (b)

Fig. 6: Comparison of gyroscope data collected on different devices

4.2 Data Filtering and Aggregation

Before inputting collected data to any detection algorithm, we had to change the coordinate system of the collected data. With the device positioned flat on a surface with the screen facing upwards, the axes of the initial system are oriented as follows:

- x axis points to the right of the screen,
- y axis points to the top of the screen,
- z axis points upwards.

Since all the sensor axes are fixed in reference to the phone, a movement of the device causes a movement of the axes in relation to the vehicle. To counteract these changes, we used the virtual "rotation vector" sensor built into most modern devices. This sensor represents a rotation quaternion that, if applied to any other 3-axis sensor, reorients its axes to the following:

- x axis points to the geographic east,
- y axis points to the geographic north,
- z axis points upwards (away from the center of Earth).

To further clarify, those two systems are identical when the phone is positioned perfectly flat with its screen facing upwards, its display's top edge facing north and its display's right side facing east.

However, the x and y axes of the resulting system are still independent from the axes of the vehicle. As the final step, we rotated the coordinate system around the z axis to obtain the following axes in relation to the vehicle:

- x axis points to the right side of vehicle (perpendicular to the driving direction),
- y axis points to the front of the vehicle (along the driving direction),
- z axis points upwards.

Taking the results from the initial analysis into consideration, instead of relying on raw rotated sensor readings, we introduced two additional aggregation methods. The first aggregate comes from the observation that a single short peak of acceleration or

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

angular velocity is not an indication of a collision. Since a collision implies a significant change in the resultant speed of the vehicle, the accelerating force must persist over time. Similarly, to change the orientation of the vehicle, the angular velocity must persist over time. To reflect this observation, we aggregated accelerometer and gyroscope readings into a sequence of integrals calculated from each axis in a sliding time window.

Given $[v_1, v_2, ..., v_n]$ - values along a single axis, $[t_1, t_2, ..., t_n]$ - times of collecting those values, we calculate trapezoidal integrals as in (1) and sum integrals within a time window of size k as in (2).

$$tr_i = \frac{(v_{i+1} + v_i) \cdot (t_{i+1} - t_i)}{2} \tag{1}$$

$$trw_i^k = \sum_{j=i}^{i+k-1} tr_j \tag{2}$$

The second aggregate is inspired by the observation that in multiple cases both the peak and the integral are reduced by an unrestricted motion of the phone. The same resultant loss of velocity is distributed over a longer time. As a result, sensors generated an oscillating pattern that did not reach values as large as the single peak in other cases, but the change of amplitude between any two consecutive readings was noticeable. As a result, we the introduced second method of aggregation which captured abrupt changes in data.

Using the same notation: $[v_1, v_2, ..., v_n]$ — values along a single axis, $[t_1, t_2, ..., t_n]$ — times of collecting those values, we calculate average absolute derivatives over time as in (3) and sum these values within a time window of size k as in (4).

$$ds_i = \frac{|v_{i+1} - v_i|}{t_{i+1} - t_i} \tag{3}$$

$$dsw_i^k = \sum_{j=i}^{i+k-1} ds_j \tag{4}$$

4.3 Collision Detection Method

Following the introduction of aggregation methods we attempted the threshold approach once again, but to no avail. Neither one of the aggregates carried enough information to precisely determine whether a crash has occurred or not.

As a next attempt, we took another observation into consideration. Phone sensors are limited in the maximal value that can be registered along any single axis. As a result, devices with lower limits may experience "clipping" of readings if the actual value would exceed the limit. To address this issue we grouped devices by the limits of their sensors and calculated thresholds for each group separately. Again, we were able to identify all the collisions, but at the cost of a significant number of false positive indications.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

Since the method of calculating thresholds for aggregates did not yield satisfying results, we decided to use a decision tree. This method accepts a series of labeled examples and, based on the observed relations, creates a ruleset for assigning labels to new, unlabeled examples. The single example consists of a number of values, each representing one feature of the observed phenomenon, while a label represents the class of the observation.

Although this method itself also operates on thresholds on the low level, it is able to produce complex rules involving several input values. Moreover, the decision tree is able to identify the most important features - something that we utilized later.

Due to the fact that a significant fraction of modern phones is not equipped with a gyroscope, we created two sets of features, which implied two different decision trees. The set of features for an accelerometer-only tree included the following 17 features:

- raw accelerometer — x axis, y axis, z axis, magnitude,
- accelerometer integrals — x axis, y axis, z axis, magnitude,
- accelerometer average derivatives — x axis, y axis, z axis, magnitude,
- absolute accelerometer values — x axis, y axis, z axis,
- accelerometer threshold and frequency.

The magnitude of the absolute values is identical to the magnitude of raw data, therefore it is not included to avoid duplication. The set of features for a gyroscope and accelerometer tree included additional 17 features calculated using gyroscope values in the same manner. Our set of labels included two classes: crash and non-crash.

As a first iteration, we trained a decision tree with a depth limit of 8 levels. This tree was created in order to identify the most important features. In this process we learned which features are crucial in determining the class of the observation. Then, we repeated the learning process, but in this iteration we limited the depth to 3 levels and reduced the feature set to the 8 most significant ones obtained in the previous iteration. This process was repeated for the both tree types: including and excluding gyroscope. As a final step, we tested the 3-level trees, and we present our results in the next section.

5 Results

In order to verify the effectiveness of the decision tree and compare it to the threshold-based algorithm, we collected necessary measurements from the crash tests described in section 3. For negative samples (data without crashes) we used measurements obtained in two long rides between cities in Poland. Half of this data was used as negative examples for training and the rest was used for evaluation. In these road experiments we utilized several mobile devices with different accelerator and gyroscope limits.

First, we present the results for the threshold-based detection. In Figure 7 we plot the maximum value of acceleration integral registered in each crash (per each involved device). In Figure 8 we present the histogram of acceleration integrals during normal rides. Finally, in Figure 9 we plot an average number of false positives per hour of normal ride versus crash detection sensitivity. Note that in this and subsequent plots we consider any sequence of detections shorter than 5 seconds as one positive detection.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

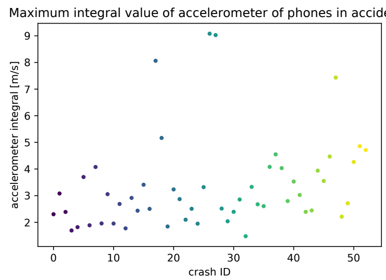


Fig. 7: Maximum value of accelerometer integral registered by devices in crashes

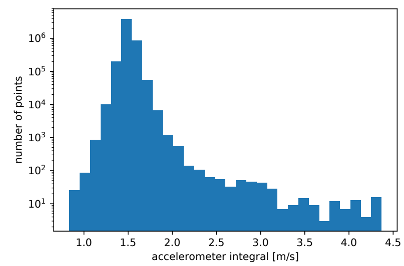


Fig. 8: Histogram of accelerometer integral values during normal rides

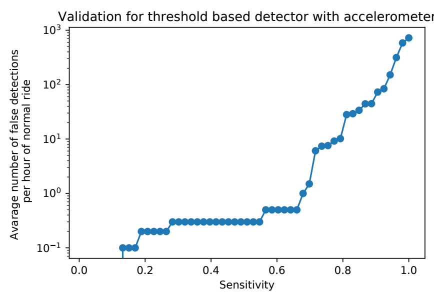


Fig. 9: Sensitivity vs. number of false positives for threshold based crash detector (using accelerometer data only)

This reflects the fact that during accidents, or false positives due to e.g. poor road conditions, violent phone movements may span an interval lasting several seconds. Results in Figure 9 demonstrate that threshold based detection is unreliable due to excessive number of false alarms.

Because the number of collected crash data instances is limited, we opted to evaluate the performance of decision tree detector with leave-one-out cross validation. Therefore, each detector was trained on all but one crash data instance and then used to classify that held-out positive example. In each case we used the same set of negative examples. Also, all trained decision trees were run on test data from normal rides, where we counted the number of false positive detections. Figure 10 report an average number of false positive detections per hour of normal ride versus crash detection sensitivity for detectors that use only the accelerometer data. Results for accelerometer and gyroscope sensors are reported in Figure 11.

As we can see, decision tree detector gives vastly better results than simple threshold based detection. For a sensitivity of 80%, the threshold-based detector gives around 10 false positive detections per hour of driving. For a decision tree detector this number is around 0.6 when using accelerometer data and around 0.5 for devices with accelerometer and gyroscope.

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24

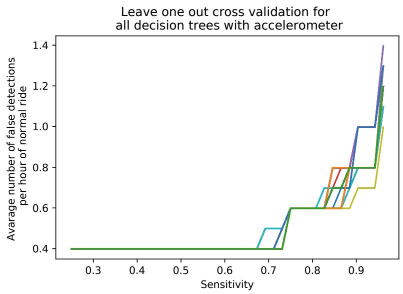


Fig. 10: Sensitivity vs. number of false positives for decision tree detectors that uses only the accelerometer sensor.

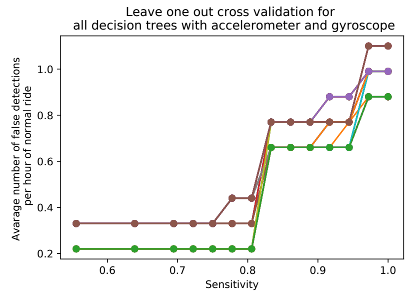


Fig. 11: Sensitivity vs. number of false positives for decision tree detectors that uses both accelerometer and gyroscope.

It is important to mention here that the positive data collected in our tests came from recorded accidents in which speeds varied from 12 kph to 43.5 kph, while the negative data came from drives where speeds reached up to 140 kph. The increased number of false positives at high sensitivity, shown in Figures 9, 10 and 11 may result directly from the very small margin between the interface of negative and positive examples in which readings from the accelerometer and gyroscope at low speed accidents may resemble some readings from fast car driving.

6 Conclusions

Although we have dealt with certain problems of data quality and spotted significant differences between the data readings in different smartphones, we are sure that these problems can be overcome by processing methods for the reduction of the inevitable false-positives detection.

It is important to note that in the real world application of the presented solution, false positives can be accepted and mitigated by introducing a dedicated loopback approach: the emergency notification service should first try to contact the driver. It is quite obvious that false-negatives might do much more damage and therefore a reasonable attitude must be maintained towards the classification efficacy of the proposed system.

In the future we are planning to further extend the application, perform more real-world tests based on individual drivers and fleets and extend the features of the application, focusing on driving-style modelling and displaying suggestions relevant to the driver when necessary based on their current driving style and the features of the route.

Acknowledgment

This research was partially supported by the Polish National Center for Research and Development under the project no. TANGO2/340869/NCBR/2017 and by the funds of

Polish Ministry of Science and Higher Education assigned to AGH University of Science and Technology. We would also like to thank our partners from PZU for supporting the research and the experiments.

References

1. Amin, M., Ibne Reaz, M., Bhuiyan, M., Nasir, S.: Kalman filtered gps accelerometerbased accident detection and location system: A low-cost approach. Current Science **106**(11), 1548–1554 (2014)
2. Datentechnik, D.S.: Picdaq5 (2019), http://www.dsd.at/index.php?option=com_content&view=article&id=327:pic-daq-deutsch&catid=53&Itemid=175&lang=de
3. commercial Partnership for Development, N., of Navigation Technologies, U.: Era-glonass (2015), http://en.glonassunion.ru/era-glonass/
4. Devices, B.C., GmbH, S.: Telematics ecall plug (2019), https://www.bosch-connectivity.com/products/telematics-ecall-plug/
5. Group, P.: Pzu go (2019), https://www.pzu.pl/pzugo
6. Jiansheng, F.: Vision-based real-time traffic accident detection. In: Proceeding of the 11th World Congress on Intelligent Control and Automation. pp. 1035–1038 (June 2014). https://doi.org/10.1109/WCICA.2014.7052859
7. Octo: Insurance telematics solutions for personal auto (2019), https://www.octousa.com/personal-insurance-telematics-solutions/
8. OnStar: Welcome to onstar (2019), https://www.onstar.com/us/en/home/
9. Organization, W.H.: Global status report on road safety 2018: Summary (2018), https://www.who.int/violence_injury_prevention/road_safety_status/2018/en/
10. Parliament, E.: European commission decision 585/2014/eu on the deployment of the interoperable eu-wide ecall services. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32014D0585 (2015)
11. Parliament, E.: European commission regulation 2015/758 on ecall type-approval ecall amending directive 2007/46/ec. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32015R0758 (2015)
12. SpA, S.: Automatic car crash detection app (2015), http://www.sosmartapp.com/
13. Telematics, C.M.: The drivewell tag (2019), https://www.cmtelematics.com/drivewell-tag/
14. Thompson, C., White, J., Dougherty, B., Albright, A., Schmidt, D.C.: Using smartphones to detect car accidents and provide situational awareness to emergency responders. In: Mobile Wireless Middleware, Operating Systems, and Applications. pp. 29–42 (2010)
15. of Transportation, U.S.D.: National highway traffic safety administration (2019), https://www.nhtsa.gov/
16. Visser, M.: Overview on emergency call worldwide - status 2016 (2016), https://www.its-mobility.de/download/eCallDays2016/presentations/Visser_Gemalto_eCallDays_2016.pdf
17. White, J., Thompson, C., Turner, H., Dougherty, B., Schmidt, D.C.: Wreckwatch: Automatic traffic accident detection and notification with smartphones. Mobile Networks and Applications **16**(3), 285 (2011)
18. Zaldivar, J., Calafate, C.T., Cano, J.C., Manzoni, P.: Providing accident detection in vehicular networks through obd-ii devices and android-based smartphones. In: 2011 IEEE 36th Conference on Local Computer Networks. pp. 813–819 (Oct 2011). https://doi.org/10.1109/LCN.2011.6115556
19. Öörni, R., Meilikhov, E., Korhonen, T.O.: Interoperability of ecall and era-glonass in-vehicle emergency call systems. IET Intelligent Transport Systems **9**(6), 582–590 (2015)

ICCS Camera Ready Version 2021
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-77980-1_24