Detection of Critical Events Using Limited Sensors

Examensarbete utfört i Fordonssystem
vid Tekniska högskolan vid Linköpings universitet
av

Henrik Hagelin

LiTH-ISY-EX--12/4620--SE
Linköping 2012
Detection of Critical Events Using Limited Sensors

Examensarbete utfört i Fordonssystem vid Tekniska högskolan vid Linköpings universitet av

Henrik Hagelin

LiTH-ISY-EX--12/4620--SE

Handledare: Kristoffer Lundahl
            ISY, Linköpings universitet
Anders Andersson
            FTS, VTI

Examinator: Jan Åslund
            ISY, Linköpings universitet

Linköping, 30 augusti 2012
Detektion av kritiska händelser med begränsade sensorer

Detection of Critical Events Using Limited Sensors

Unfortunately, people die and get injured due to accidents in the traffic. Furthermore, statistics of road accidents is limited and mostly composed of serious accidents, making it difficult to draw conclusions about how to improve the safety in the traffic. Thus, there is an interest in obtaining information about critical events in the traffic, i.e. potential accident situations, since they occur much more frequently.

One way of detecting critical events is to use sensors, such as accelerometers and gyroscopes. As the usage of cellphones with built-in sensors increases, it would be interesting to examine whether these sensors are good enough to detect critical events. This is where the focus of this thesis lies.

An application that collects data from the cellphone’s built-in accelerometer, gyroscope and GPS was developed and tested. The data was then analysed and compared to data from accurate sensors, represented by a VBOX coupled to an IMU.

The conclusions made in this thesis are that the sensors in the cellphone perform almost equivalent results compared to the VBOX. It is possible to use data from the sensors in order to detect critical events.
Abstract

Unfortunately, people die and get injured due to accidents in the traffic. Furthermore, statistics of road accidents is limited and mostly composed of serious accidents, making it difficult to draw conclusions about how to improve the safety in the traffic. Thus, there is an interest in obtaining information about critical events in the traffic, i.e. potential accident situations, since they occur much more frequently.

One way of detecting critical events is to use sensors, such as accelerometers and gyroscopes. As the usage of cellphones with built-in sensors increases, it would be interesting to examine whether these sensors are good enough to detect critical events. This is where the focus of this thesis lies.

An application that collects data from the cellphone’s built-in accelerometer, gyroscope and GPS was developed and tested. The data was then analysed and compared to data from accurate sensors, represented by a VBOX coupled to an IMU.

The conclusions made in this thesis are that the sensors in the cellphone perform almost equivalent results compared to the VBOX. It is possible to use data from the sensors in order to detect critical events.
Sammanfattning

Dessvärre dör och skadas folk på grund av olyckor i trafiken. Dessutom är statistik för trafikolyckor begränsad och bestående av mestadels allvarliga olyckor, vilket gör det svårt att dra slutsatser om hur man kan förbättra trafiksäkerheten. Det finns därför ett intresse i att få information om kritiska händelser i trafiken, det vill säga potentiella olyckssituationer, eftersom de förekommer mycket ofta.

Ett sätt att detektera kritiska händelser är använda sensorer, såsom accelerometrar och gyroskop. Eftersom användandet av mobiltelefoner med inbyggda sensorer ökar, skulle det vara intressant att undersöka om dessa sensorer är tillräckligt bra för att detektera kritiska händelser. Detta är fokus i denna avhandling.

En applikation som samlar in data från mobiltelefonens inbyggda accelerometer, gyroskop och GPS har utvecklats och testats. Data har analyserats och jämförts med data från noggranna sensorer i form av en VBOX med tillhörande IMU.

Slutsatserna i denna avhandling är att sensorerna i mobiltelefonen presterar likvärdiga resultat jämfört med VBOX:en. Det är möjligt att använda data från sensorerna för att detektera kritiska händelser.
Acknowledgments

I would like to thank VTI for giving me the opportunity to do this thesis. I want to thank everyone at the FTS department for a great time with lots of interesting discussions on all kinds of topics. A special thank to Anders Andersson and Jonas Andersson Hultgren for all the tips and advice during the course of this thesis and for their feedback on the report.

I would also like to send a special thank to Omar Bagdadi for interesting discussions related to critical events in traffic. I would also like to thank Harry Sörensen for the help with the VBOX. A thank also to my supervisor Kristoffer Lundahl for his feedback on the report.

Last but not least I would like to send a big thank to my family and my friends for all their support.

Linköping, August 2012
Henrik Hagelin
Contents

List of Figures xi

1 Introduction 1
  1.1 Background ................................. 1
  1.2 Objective ................................ 2
  1.3 Method .................................... 2
  1.4 Limitations ................................. 3
  1.5 Thesis Outline .............................. 3

2 Critical Events 5
  2.1 The Traffic Conflict Technique ................. 5
  2.2 Speed profiles .............................. 9

3 Test Equipment 13
  3.1 Cellphone ................................. 13
  3.2 VBOX ...................................... 14
  3.3 Vehicle ................................... 14
  3.4 Installation ................................. 15

4 The Application 17
  4.1 User Interface .............................. 17
  4.2 Data Collection ......................... 18
  4.3 Signal Processing ......................... 18
  4.4 Detection Alerting .......................... 19

5 Sensor Evaluation 21
  5.1 Accelerometer ............................. 21
  5.2 Gyroscope ................................ 25
  5.3 GPS ...................................... 26

6 Detection of Critical Events 29
  6.1 Braking in 30 km/h ......................... 30
  6.2 Braking in 40 km/h ......................... 32
List of Figures

2.1 Conflicts in relation to exposure ........................................ 6
2.2 Conflicts in terms of nearness to collision .......................... 7
2.3 Pyramidal representation of traffic events ......................... 7
2.4 The border between serious and non-serious conflict .......... 8
2.5 Speed profile for a hard but controlled brake ................... 11
2.6 Speed profile for a hard and sudden brake ....................... 11

3.1 The cellphone Google Galaxy Nexus .................................. 13
3.2 The VBOX and IMU ..................................................... 14
3.3 The test vehicle ....................................................... 14
3.4 Placing of the IMU .................................................... 15
3.5 Placing of the cellphone ............................................. 16
3.6 The coordinate system in the car .................................... 16

4.1 The application’s interface ............................................. 17

5.1 Longitudinal acceleration test ....................................... 22
5.2 Longitudinal acceleration test, filtered data ..................... 23
5.3 Longitudinal acceleration test, raw data and filtered data ..... 24
5.4 Swerving test, yaw rate .............................................. 25
5.5 GPS position test ...................................................... 26
5.6 GPS speed comparison ............................................... 27

6.1 Speed profile for a brake test in 30 km/h ......................... 30
6.2 Zoomed speed profile for a brake test in 30 km/h ............. 31
6.3 Speed profile for a brake test in 40 km/h ......................... 32
6.4 Zoomed speed profile for a brake test in 40 km/h ............. 33
6.5 Speed profile for a brake test in 50 km/h ......................... 34
6.6 Zoomed speed profile for a brake test in 50 km/h ............. 35
6.7 Speed profile for a brake test in 60 km/h ......................... 36
6.8 Zoomed speed profile for a brake test in 60 km/h ............. 37

A.1 The application screen .................................................. 43
A.2 Screenshot of the application ........................................ 44
A.3 The transfer command in cmd ........................................ 45
The following chapter provides general information about the thesis. It presents the background and the objective. It also shows the methods and limitations that are used.

1.1 Background

Unfortunately, people die and get injured due to accidents in the traffic. In Sweden there is a concept called Vision Zero ("Nollvisionen") which states that it is unacceptable that people are killed in traffic. The road safety work in the spirit of Vision Zero means that roads, streets and vehicles must be adapted to the human conditions to a greater extent. Measures shall be taken to prevent accidents. [20]

The Transport Administration carries out in-depth studies of all fatal road accidents. These studies are done to provide as clear picture as possible of what happened before, during and after the accident. An in-depth study may lead to immediate changes in the road environment. [21]

Statistics of road accidents is limited and mostly composed of serious accidents. Thus, there is an interest in obtaining information about critical events (also referred to as conflicts or near-crashes), since they occur much more frequently. There are between 3,000 and 40,000 conflicts for each police-reported personal injury accident, depending on the severity and type of conflict. [3]

In order to detect critical events, sensors can be used. As the usage of cellphones with built-in sensors increases [19], it would be interesting to examine whether these sensors are good enough to detect critical events.
1.2 Objective

The objective of this thesis was to investigate whether the sensors in today’s cell-phones can be used for detecting certain types of critical events in traffic where the driver has made an evasive maneuver.

Within the thesis, the following was made:

- Clarification on what distinguishes a critical event from an event that is not critical.
- An application that collects data from the cellphone’s built-in accelerometer, gyroscope and GPS was developed and tested.
- Offline analysis of sensor data, if data from the cellphone is consistent with data from the VBOX, or if consistency can be achieved by filtering the data.
- Study of options for detection, including acceleration and jerk (derivative of the acceleration) behavior.
- Further development of the application so that it filters and analyses the sensor data. It also warns when a critical event occurs, and store, with the help of GPS, the location of the event.

1.3 Method

To gain information about what a critical event in traffic is, a literature study was made by searching for articles on the subject. Parallel to the literature study, work began on developing the application for the cellphone that retrieves and stores sensor data. The data was retrieved from the cellphone’s built-in accelerometer, gyroscope and GPS. The accelerometer was used to detect critical events, while the GPS stored the locations where they occur.

In order to test the application, a number of test runs were designed, which described the maneuvers to be tested. These test runs were then performed while data was collected. Data was stored internally on the devices for both the VBOX and the cellphone, and was then transferred to a computer for offline analysis. The analysis consisted of comparing data from the cellphone with data from the VBOX to see if they were consistent. Low pass filtering was used to suppress the high frequency noise on the accelerometers.
1.4 Limitations

Tests were only carried out in a passenger car, since this is the type of vehicle used in most studies on critical events. Due to the time aspect of this thesis, only critical events consisting of a hard braking maneuver were examined, since this is the most common evasive maneuver.

The application was developed only for Android, since the cellphone used in this thesis runs on that operating system.

1.5 Thesis Outline

Chapter 1
Describes the background, objective, method and limitations of the thesis.

Chapter 2
Explains critical events, and how they can be used.

Chapter 3
Describes the components used in the tests, and how they were installed.

Chapter 4
Describes the application’s structure and functions.

Chapter 5
Shows the result of the sensor evaluation.

Chapter 6
Shows the result for the tests of detecting critical events.

Chapter 7
A summary of the thesis, including conclusions and future work.
This chapter describes why the traffic conflict technique is useful when it comes to traffic safety evaluations. The chapter also explains which critical events that are examined in this thesis, and why these are chosen.

2.1 The Traffic Conflict Technique

According to Chin et al. [5], most studies on road safety have traditionally been based on accident statistics for addressing security concerns. Since the occurrence of accidents can be caused by some unwanted problems in the traffic system, it could be useful to investigate these, and for instance look at the accident frequency. Data from accidents may prove to be useful in many cases, but there are several limitations in their usage. Among other things, the frequency of accidents is low, combined with the statistical nature of the problem, makes it difficult to get a statistical significance by simply examining the number of accidents. In addition, accident may not be reported in the same way from case to case, which can make it difficult to perform a comparative analysis. There is even the case that many accidents, especially those who do not lead to any damage, are not reported at all. Relying solely on accident reports may cause biased conclusions. But perhaps the most serious objection to the traditional approach, relying on the number of accidents for the safety assessment and improvement, is the ethical problem which requires that a sufficient number of serious accidents must take place before a dangerous place or situation can be identified and corrected.
To overcome these shortcomings, many ways has been suggested to use data from situations other than just accidents. One way is to use critical events, defined as critical incidents which do not necessarily include a collision, referred to as traffic conflicts. The concept of traffic conflicts was first proposed by Perkins and Harris in 1968 [14] as an alternative to accident data. Their goal was to define events in traffic that occur frequently, which can be easily observed and are related to accidents. This approach became known as “the traffic conflict technique”. They defined traffic conflicts as a potential accident situation that leads to an avoidance maneuver, such as braking or swerving. This simple definition has since been refined by introducing time and space between vehicles at the time of the conflict [5]. An internationally accepted definition of a traffic conflict made by Amundsen and Hydén in 1977 [1] is “an observed situation in which two or more road users approach each other in space and time for such an extent that there is a risk of collision if their movements remain unchanged”.

The most appealing aspect of the traffic conflict technique is that data from conflicts can be collected in a much shorter period of time in comparison with data from accidents. This also leads to, in addition to the analysis being less affected by time dependent aspects, that the ethical problem associated with the need of a long history of accidents is not a problem. The intuitive idea that conflicts are potential collisions, but to a lower level of risk, suggests that conflicts and accidents are related, as confirmed by Hydén in 1987 [11]. Several models have been used to describe this relationship. One of those models is shown in Figure 2.1 and describes accidents as a subset of serious conflicts, which in turn is a subset of less severe conflicts. This representation is derived with the assumption that accidents must be preceded by conflicts.

![Figure 2.1: Representation of conflicts in relation to exposure.](image)

Another type of representation is a severity scale, ranging from minor to very serious conflicts. There are two types of that scale of severity [5]. One type is the idea with a distribution function in terms of nearness to a collision, proposed by Glauz and Migletz in 1980 [7], as shown in Figure 2.2.
2.1 The Traffic Conflict Technique

![Graph showing traffic conflicts and nearness to collision]

Figure 2.2: Representation of conflicts in terms of nearness to collision.

The other scale is a safety pyramid proposed by Hydén in 1987 [11], as shown in Figure 2.3. Both figures illustrate that non-conflicts represents the majority of the observations and that the frequency/probability of an event decreases with increasing severity.

![Pyramidal representation of traffic events]

Figure 2.3: Pyramidal representation of traffic events. [12]
A variety of observation methods have been developed to measure traffic conflicts. A widely used measure is the time to collision (TTC), defined as “the time for two vehicles to collide if they continue at their present speed and on the same path” by Hayward in 1972 [10]. The value of TTC is infinite if the vehicles are not on a collision course. If the vehicles are in collision course, the TTC is finite and decreases with time. The minimum TTC that was reached when the vehicles were approaching each other in a collision course is used as a measure for estimating the severity of the conflict. [18]

The Swedish traffic conflict technique was developed at Lund University in the 1970’s and 1980’s before it finally reached its present level of development in 1987 [11]. The Swedish technique focuses on situations where two road users would have collided if none of them had performed some form of evasive maneuver. The point at which the maneuver is performed is recorded, through observation, as the “Time-to-Accident” (TA). The TA value together with the conflicting speed is used to determine whether or not a conflict is critical or not, see Figure 2.4. [2]

![Figure 2.4: The border between serious and non-serious conflict. [12]](image)

In a study by Dingus et al. [6], 100 cars were equipped with measuring instruments, such as accelerometer and gyroscope, to detect critical events in the traffic. The main objective of the study was to collect large scale naturalistic driving data. The measuring equipment offers the advantage that it provides much more detailed and accurate information about the events than what is available in police reports and crash investigations. This is because these reports and investigations
are based on eye-witness stories. Such data are found to have limited accuracy. For example, it is often the case that drivers do not remember specific rapidly-occurring events as a crash or near-crash scenario unfolds. This is exacerbated in cases where drivers or passengers are trying to hide the details because they are embarrassed of what happened, or because they are afraid of being prosecuted. Unlike experiments on a test track or in simulators, this approach has the advantage that the absence of an experimental leader means that the subjects did not change their driving behavior because he or she feels directly supervised. The idea to get around the issue with so few reported collisions is to use data from crashes together with data from near-crashes. The definitions of crash and near-crash used in the study is shown in Table 2.1. One of the conclusions of the 100 car study is that there is a strong frequency relationship between crashes and near-crashes.

### Table 2.1: The definition of crash and near-crash

<table>
<thead>
<tr>
<th>Severity level</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Crash</td>
<td>Any contact with an object, either moving or fixed, at any speed, in which kinetic energy is measurably transferred or dissipated. Includes other vehicles, roadside barriers, objects on or off the roadway, pedestrians, cyclists, or animals.</td>
</tr>
<tr>
<td>Near-crash</td>
<td>Any circumstance that requires a rapid, evasive maneuver by the subject vehicle (or any other vehicle, pedestrian, cyclist or animal) to avoid a crash. A rapid, evasive maneuver is defined as steering, braking, accelerating, or any combination of control input that approaches the limits of the vehicle's capabilities.</td>
</tr>
</tbody>
</table>

#### 2.2 Speed profiles

There are several ways to detect critical events, e.g. examining the vehicle’s lateral acceleration, longitudinal acceleration and/or yaw rate [6]. The most common evasive action is braking [13]. Thus, the focus in this thesis lies on detecting critical events that contains a critical braking maneuver.

One way of detecting a hard braking maneuver is to examine the acceleration of the vehicle and see if it is below a certain threshold, if so, the event could be critical. However, according to [4] it is insufficient to use acceleration as the only way of separating a normal braking maneuver from a braking maneuver in critical situations. This is due to that acceleration profiles in normal situations are quite similar to acceleration profiles from critical situations. Moreover, it is difficult to set a specific threshold for detection due to the large variance in person’s use of deceleration. This is seen to some extent in the study by Dingus et al. [6], where detections based on the longitudinal threshold only were valid in 33.6 % of the cases. The 66.4 % invalid detections were then filtered out manually.
by using the recorded video data. In order to avoid that many false detections it may be desirable to use a complementary indicator of traffic conflicts.

As stated, a hard braking maneuver does not necessarily mean that there has been a critical event. A hard braking maneuver can be both controlled and uncontrolled. One way of detecting a critical braking maneuver is to look at the derivative of the acceleration, called jerk, to see how sudden the braking maneuver was. A very sudden brake results in a very negative jerk. If this value is below a certain threshold, the brake is classified as critical.

The threshold values for detection of critical events using acceleration, and critical braking maneuvers using jerk, are set to

\[-0.6g = -0.6 \times 9.82 \text{ m/s}^2 = -5.892 \text{ m/s}^2\] for acceleration, and
\[-1g/s = -9.82 \text{ m/s}^3\] for jerk,

according to [6] and [13]. These thresholds were chosen to distinguish critical events from non-critical events, with the goal of detecting as many critical events as possible while keeping the false detections down.

To illustrate an example of how these definitions can be used, a comparison can be made between two different scenarios, each containing a hard braking maneuver, but only one of these is considered critical.

The first scenario symbolises when a traffic light turns red, then the driver is usually prepared for it, but there might still be a need of performing a hard braking maneuver to be able to stop in time. This scenario may result in something like what is shown in Figure 2.5.

The other scenario symbolises when a brake must be performed without the driver being prepared for it, for example if an animal crosses the road just in front of the vehicle. This may result in a sudden and abrupt braking maneuver, like in Figure 2.6.

By examining Figure 2.5 and Figure 2.6, a conclusion can be drawn that both braking maneuvers were hard, since their acceleration reached approximately -7 m/s², i.e. below the acceleration threshold. This value can be compared to a study of normal braking behavior by Wang et al. [22], where 92.5% of all measured trips had a weaker maximum braking force than one that results in an acceleration of -3.4 m/s².

But when comparing the jerk for these scenarios, it is clear that only the second scenario includes a critical braking maneuver, since it is only this jerk that reaches below the jerk threshold.
2.2 Speed profiles

Figure 2.5: Speed profile for a hard but controlled brake at 50 km/h.

Figure 2.6: Speed profile for a hard and sudden brake at 50 km/h.
To carry out a complete test, a cellphone, accurate sensors for validation and a vehicle were required. The following is a description of these, and how they were installed in the vehicle.

3.1 Cellphone

The cellphone used was a Google Galaxy Nexus (also known as Samsung Galaxy Nexus), see Figure 3.1. This cellphone was chosen because it was desired to use a recently released cellphone, and at the start of this thesis this was the only cellphone using the latest operating system from Google, called Android 4.0. The cellphone contains a GPS unit and the desired sensors, i.e. accelerometer and gyroscope.

*Figure 3.1: Cellphone Google Galaxy Nexus with the coordinate system used by the sensors.* [9; 8]
3.2 VBOX

In order to compare sensor data from the cellphone with data from accurate sensors a VBOX coupled to an IMU was used, see Figure 3.2. The VBOX is considered to provide accurate sensor data, and is used by VTI.

The VBOX collects data, consisting of for example accelerations, rotation rates and speed, in 100 Hz and saves it on a SD card. The GPS in the VBOX generates speed data with an accuracy of 0.1 km/h, a resolution of 0.01 km/h and a latency of 6.75 ms. The IMU generates acceleration data with an accuracy of $\pm 0.2g \approx 1.964 \text{ m/s}^2$ and a resolution of $0.001g \approx 0.00982 \text{ m/s}^2$. The rate sensors in the IMU has a resolution of $0.01 \, ^\circ /\text{s}$. [15; 16]

![VBOX and IMU with the coordinate system used by the sensors.](image)

**Figure 3.2:** VBOX and IMU with the coordinate system used by the sensors. [15; 16]

3.3 Vehicle

The vehicle used in the tests was one of VTI’s test cars, a Volvo 855, see Figure 3.3. The car is equipped with ABS, has a weight of 1530 kg, the dimensions 4.75 x 1.77 meters and a wheelbase of 2.66 meters.

![The vehicle used in the tests.](image)

**Figure 3.3:** The vehicle used in the tests.
3.4 Installation

Prior to a data collection test it was necessary to place the measuring devices in a certain way in the vehicle. In order to capture the main behaviour of the vehicle, the equipment was chosen to be placed as near the vehicle's center of gravity as possible, with the cellphone still being accessible to the user. The equipment was eventually placed between the front seats. The IMU was placed in the compartment under the armrest, see Figure 3.4. The cellphone shall be located as close to the IMU as possible so that their data differs as little as possible. Therefore, the cellphone was placed on the armrest, approximately 7 cm above the IMU. A rubber sheet was placed between the armrest and the cellphone to increase the friction so that the cellphone would not slip on the armrest. The rubber sheet and the cellphone were then tightened with two rubber bands, see Figure 3.5.

Data from the cellphone and the IMU may differ since the cellphone is located approximately 7 cm above the IMU, resulting in a different distances to the rotation axis when the car is pitching. For example, when performing a braking maneuver, the contribution from the pitch acceleration of the vehicle is greater the further away the devices are from the vehicle's rotation axis.

Figure 3.4: The IMU was placed in the compartment under the armrest.
Figure 3.5: The cellphone was placed on the armrest.

Based on the coordinate system in the IMU and the placing of the devices, the coordinate systems used in the vehicle is defined according to Figure 3.6. It is important to notice the different coordinate systems used by the cellphone compared to the one that is used in the vehicle. This results in that the y-axis used by the cellphone is regarded as the x-axis, and the x-axis in the cellphone is regarded as the y-axis, when placed in the vehicle.

Figure 3.6: The coordinate system in the car.
The application was developed in the Eclipse development environment and is compatible with cellphones using the Android operative system and are equipped with accelerometer, gyroscope and GPS.

### 4.1 User Interface

Figure 4.1 shows the user interface of the application. For more information about the application, see the user manual in Appendix A and the source code in Appendix B.

![The application's interface.](image)

**Figure 4.1:** The application's interface.
4.2 Data Collection

The following data, with corresponding sample rate, are collected from the cellphone’s accelerometer, gyroscope and GPS. The jerk is calculated by derivation of the longitudinal acceleration, see Section 4.3 and 4.4.

- Accelerometer, 100 Hz \([\text{m/s}^2]\)
  - Longitudinal acceleration
  - Lateral acceleration
  - Vertical acceleration
- Gyroscope, 100 Hz \([\text{rad/s}]\)
  - Pitch rate
  - Roll rate
  - Yaw rate
- GPS, 1 Hz
  - Longitude [Decimal degrees]
  - Latitude [Decimal degrees]
  - Speed [m/s]
- Time stamp [ns]
- Jerk [m/s\(^3\)]

4.3 Signal Processing

Low pass filtering was used on the longitudinal acceleration data in order to suppress the high frequency noise. The filtering consisted of averaging over 11 samples, according to Formula 4.1.

\[
accLP_i = \frac{\sum_{n=i}^{i+10} acceleration_{n-10}}{11}
\]  

(4.1)

The choice of using 11 samples was made since it reduced the noise so that the data became smooth while still not suppressing the peaks too much through averaging. Using an averaging filter also made it easy to implement in the application. The result of this filtering is most clearly shown in Figure 5.3.
4.4 Detection Alerting

Based on the discussions in Section 2.2 it was chosen that the best way to detect critical events with the use of a cellphone, and not having any video recordings for validation, is to use jerk as indicator for critical braking maneuvers, i.e. detect critical events that contains a critical braking maneuver. The jerk was calculated by the derivation of the filtered acceleration according to Formula 4.2, with a time difference of 0.2 seconds which is used in [13]. A sample frequency of 100 Hz yields an interval of 20 samples. The calculation of the jerk requires a smooth line to differentiate, thus the filtered acceleration is used. Using raw data would result in jerk data that is very hard to examine.

\[
\text{jerk}_j = \frac{accLP_j - accLP_{j-20}}{time_j - time_{j-20}}
\]  

(4.2)

The criteria used for detecting a critical braking maneuver is chosen to when the jerk value reaches below the threshold -1g/s, according to [13].
To test the performance of the cellphone’s sensors in comparison with the accurate sensors from the VBOX, various tests were carried out with data being saved both from the cellphone and the VBOX. The main focus was to compare the longitudinal acceleration from the accelerometers, and the rotational velocity around the z-axis, i.e. yaw rate, from the gyroscopes. This focus was chosen because such data is considered best suited to be investigated, among those available, in order to detect critical events. A test is also performed to examine the performance of the GPS in terms of speed and position.

5.1 Accelerometer

The purpose of the test was to investigate if the accelerometer has an undesirable time delay, and if it performs well during hard braking operations which results in large negative accelerations. The test also results in the possibility of comparing how noisy the accelerometers are in relation to each other.

In order to have the cellphone data and VBOX data in sync, the data collection from these were started at the same time. The test starts off by first reaching the desired speed, and then performing a sudden and short braking maneuver. Then the vehicle is accelerated up to the desired speed again. The test is then finished with a powerful brake to stop.

The first braking maneuver is used primarily to examine the response of the devices, i.e. to see how quickly the cellphone’s accelerometer reacts to a sudden brake compared to the VBOX’s accelerometer. The second braking maneuver is used to examine if the cellphone’s accelerometer data reaches the same amplitudes as the VBOX’s at powerful braking maneuvers.
Figure 5.1 shows the result of an accelerometer test. The green line represents the vehicle speed, based on data from the VBOX. The blue line represents the longitudinal acceleration from the cellphone, and the red line represents the longitudinal acceleration from the VBOX.

The sequence in Figure 5.1, from 207 to 223 seconds, consists of an acceleration from a standstill up to 55 km/h around the time of 214 seconds. At the times 208.5 and 210.5 seconds gear changes were performed, resulting in a decrease in acceleration. The short and sudden braking maneuver is performed at 214.5 seconds, which results in a decrease in speed and that the acceleration becomes negative. The same thing happens for the second braking maneuver, starting at 219.5 seconds, but with a longer time lapse.

The conclusion from this test is that the accelerometers perform similar results. Data from the cellphone follows data from the VBOX in a satisfactorily way. The same levels are reached during both accelerations and decelerations, and they react equally fast on sudden maneuvers.
However, the cellphone’s accelerometer is a bit noisier than the VBOX’s, but the noise can be reduced by processing the data. The processing consisted of reducing the high frequency noise by using a low pass filter. This was done by averaging over 11 samples, according to Formula 4.1. Such filtering on the acceleration data in Figure 5.1, from both the cellphone and the VBOX, results in Figure 5.2.

![Accelerometer test, longitudinal acceleration, filtered data](image1)

**Figure 5.2: Longitudinal acceleration test, filtered data.**

As seen in Figure 5.2, the data looks very alike after the low pass filtering. The noise is negligible, and the spikes have disappeared. Now it is easy to determine the levels that the acceleration reaches during the braking maneuvers.

To get a better view of a braking maneuver, i.e. the maneuver that is the focus in this thesis, Figure 5.3 shows the second braking maneuver zoomed in to the time interval 219-223 seconds. The figure presents both the raw data and the filtered data from the cellphone and the VBOX.

![Acceleration zoomed in](image2)
Figure 5.3 also illustrates the results of the low pass filtering even better. There will be a very small difference in relying on the cellphone’s accelerometer compared to relying on the VBOX’s accelerometer since they, after the low pass filtering, result in almost equivalent data. The small differences in the filtered data may be the result of a number of things, such as the devices not being placed at the exact same place in the vehicle, or that the cellphone and/or the IMU were not sufficiently tightened and were allowed to move from its position in powerful braking maneuvers or accelerations.

![Acceleration test, raw data and filtered data](image)

**Figure 5.3:** Longitudinal acceleration test, raw data and filtered data.
5.2 Gyroscope

The purpose of the test was to investigate if the gyroscope has an undesirable time delay, and if it performs well during fast swerving maneuvers, which can be an indicator of a critical event. The test also results in the possibility of comparing how noisy the gyroscopes are in relation to each other.

The test starts off by reaching the desired speed, and then slow swerving maneuvers are performed followed by fast swerving maneuvers. The test is finished with a deceleration to stop.

The swerving maneuvers are used to see how quickly the cellphone’s gyroscope reacts compared to the VBOX’s gyroscope. The maneuvers are also used to examine if the cellphone’s gyroscope reaches the same levels as the VBOX’s gyroscope in both slower and faster swerving maneuvers.

Figure 5.4 presents the result of a swerving test. The green line represents the vehicle position in meters, starting from (0,0). The blue line represents the yaw rate from the cellphone’s gyroscope, and the red line represents the yaw rate from the VBOX’s gyroscope. The sequence in Figure 5.4 lasts 23 seconds, from 122 to 145 seconds and starts with an acceleration from a standstill up to 40 km/h. At 129 seconds, slow swerving maneuvers begin which the gyroscope reacts to. Fast swerving maneuvers begin at 137 seconds.

This test also results in that data from the cellphone and the VBOX are almost equivalent. There is no noticeable time delay, and there is no need to low pass filter the data since the noise is negligible. It is easy to determine which levels the yaw rate reaches, and there would be no problem in detecting a swerving maneuver using data from the cellphone in comparison of using data from the VBOX.

![Figure 5.4: Swerving test in 40 km/h, yaw rate.](image-url)
5.3 GPS

The purpose of the GPS test was to investigate whether the cellphone's GPS is capable of determining the location where a critical event occurs. The investigation also examined how well the GPS estimated the vehicle speed, because the speed can be an interesting aspect of a critical event.

In order to test the position data from the cellphone’s GPS, data was collected during a test run. The position data from the GPS were then plotted on a map, taken from Google Maps, over the area that included the entire test run. For the test to be successful it was required that, based on the obtained location data, it would be possible to determine where the vehicle was driving. For example, if a critical event occurred in an intersection, it would be possible to determine at which intersection the event occurred. Such a test run is seen in Figure 5.5, the route was driven in clockwise direction, according to the arrows.

The conclusion from this test is that the positions from the cellphone’s GPS correspond very well to the map, since the figure presents the correct route of the test run. There will be no problem to determine where a critical event occurs, based on the cellphone’s position data.

Figure 5.5: GPS position test.
The second part of the GPS test was to examine the speed data from the cellphone’s GPS. To evaluate the cellphone’s speed data a comparison was made with the speed data from the VBOX. Figure 5.6 shows a 90-second sequence consisting of both acceleration and braking maneuvers.

The conclusions from this test is that the cellphone’s speed data follows the VBOX’s speed data, but with a time delay. Another difference is that the cellphone’s GPS only updates at 1 Hz, while the VBOX generates speed data at 100 Hz.

This results in that it would be possible to use the cellphone’s speed data in order to determine the speed in the starting moment of a critical event, in this case at a hard braking maneuver. But it would be difficult to make accurate speed analysis of the sequence of a critical event based on the cellphone’s speed data, due to the time delay and the slow sampling frequency.

It would be interesting to examine if it is possible to improve the speed data in a similar way done by the VBOX, i.e. measuring the Doppler effect in the GPS signals from the satellites, and using the IMU and a Kalman filter to generate accurate and smooth GPS data. [17]
Detection of Critical Events

This chapter shows how well the cellphone’s accelerometer performs in the levels around the detection thresholds. If the sensors in the cellphone performs well, the data should not result in false nor missed detections. The chapter is divided into sections, where each section presents the result of a brake test for a certain speed. The result is presented as speed profiles, according to Section 2.2. Two speed profiles are presented in each section.

The tests were performed by a test driver with the objective of carrying out braking maneuvers in a certain speed. The driver started with the brake test in 30 km/h and then gradually increased the speed with 10 km/h until reaching the final speed of 60 km/h.

The first speed profile in each section shows the entire brake test, including five or more braking maneuvers with a braking behavior ranging from normal to critical. The second speed profile shows the speed profiles of five of those braking maneuvers in a shorter time interval to make it easier to examine and compare the different signals. These braking behaviors are represented in a color scale, consisting of the colors green, cyan, blue, magenta and red. The color green represents the weakest braking and the color red represents a critical braking.

The chapter is concluded with a discussion about the result of these tests.
6.1 Braking in 30 km/h

The entire test run for the brake test in 30 km/h is presented in Figure 6.1. Five of these braking maneuvers, with a braking behavior ranging from normal (green) to critical (red), is presented in Figure 6.2.

Figure 6.1: Speed profile for a brake test in 30 km/h.
Figure 6.2: Zoomed speed profile for a brake test in 30 km/h.
6.2 Braking in 40 km/h

The entire test run for the brake test in 40 km/h is presented in Figure 6.3. Five of these braking maneuvers, with a braking behavior ranging from normal (green) to critical (red), is presented in Figure 6.4.

![Figure 6.3: Speed profile for a brake test in 40 km/h.](image-url)
6.2 Braking in 40 km/h

Figure 6.4: Zoomed speed profile for a brake test in 40 km/h.
6.3 Braking in 50 km/h

The entire test run for the brake test in 50 km/h is presented in Figure 6.5. Five of these braking maneuvers, with a braking behavior ranging from normal (green) to critical (red), is presented in Figure 6.6.

Figure 6.5: Speed profile for a brake test in 50 km/h.
Figure 6.6: Zoomed speed profile for a brake test in 50 km/h.
6.4 Braking in 60 km/h

The entire test run for the brake test in 60 km/h is presented in Figure 6.7. Five of these braking maneuvers, with a braking behavior ranging from normal (green) to critical (red), is presented in Figure 6.8.

Figure 6.7: Speed profile for a brake test in 60 km/h.
Figure 6.8: Zoomed speed profile for a brake test in 60 km/h.
6.5 Discussion

In order to detect critical events using the sensors in a cellphone, it is important that these sensors perform satisfactorily in the working range that is used when driving a car. Especially in the levels around the detection thresholds.

When examining the figures, an interesting discovery can be seen in Figure 6.2 where a braking maneuver ends up next to the jerk threshold, but it is only the cellphone data that reaches below it. The same thing can be seen in Figure 6.8. This might be the result of that the acceleration data from the cellphone reaches slightly lower levels compared to the VBOX data, which can be seen by examining the zoomed speed profiles. The lower levels obtained by the cellphone might be the result of the cellphone and IMU not being placed exactly the same.

Another point worth mentioning is the time delay between the minimum values of acceleration and jerk. The jerk reaches is minimum value at the beginning of the braking maneuver, while the acceleration reaches its minimum value later in the braking sequence. This must be taken into consideration if a detection criteria is used based on both the acceleration and jerk data.

In the 60 km/h test, the test driver performed four braking maneuvers that trigged or almost trigged based on the jerk threshold. This might be the result of the test driver was becoming more used to the critical braking maneuvers and therefore underestimating how sudden the braking maneuvers were.

When examining the figures in this chapter, the overall picture is that the sensor data from the cellphone corresponds well to the sensor data from the VBOX. This is the case for all braking maneuvers tested, regardless of speed.

According to [3] there is no significant difference in maximum deceleration between different approach speeds. Thus, there is no need to perform tests with greater approach speed, since such tests would result in approximately the same maximum deceleration as in the tests in this chapter, where the hardest braking maneuvers resulted in that the wheel locks. The conclusion that can be made is that using this cellphone could be a substitute for using a VBOX, in regard to detecting critical events using acceleration data.
7 Summary

The thesis has consisted of performing a literature study on the subject critical events, and developing an application that collects data from the cellphone's sensors and GPS. This application has then been used in test runs. Data from the cellphone and the accurate sensors have then been compared and analysed.

The thesis focused on braking maneuvers, which is the most common evasive action. Since the data collection was done without the ability to validate events through video recordings, jerk was preferable for detection of critical events. Therefore, the detection consisted in critical events which involves a critical braking, i.e. braking maneuvers resulting in a large negative jerks. By just using acceleration as an indicator of a critical event results in many false detections.

7.1 Conclusions

The cellphone’s accelerometer performs almost equivalent results to the VBOX’s accelerometer after low pass filtering. The cellphone’s accelerometer also performs sufficiently well in order to calculate jerk in a desired way, so that it can be used for detection of critical braking maneuvers. The cellphone’s gyroscope does also perform results almost equivalent to the VBOX’s gyroscope.

The GPS in the cellphone can be used to determine where a critical event occurs. It can also be used to determine the speed in the starting moment of a critical event. The downside is that the GPS in the cellphone is not that fast, making it difficult to obtain useful speed data during the actual braking process.
7.2 Future Work

There are a few further developments that would be interesting to perform. Some examples of future work are:

• Complement the data collection with video recording by using the camera in the cellphone. The camera could for example be facing the road in front of the vehicle, and the recording could then be used to validate whether a detection really was critical or not.

• Improving the application so that it is aware of the direction in which the cellphone is pointing. It could then be placed in a completely optional way in the vehicle.

• Performing a naturalistic driving study, to further test the application and to collect driving data.

• Implement more ways to detect critical events with the application, such as using the yaw rate as a trigger for swerving maneuvers.

• Improve measurement data by fusing data from the accelerometer, gyroscope and GPS, using sensor fusion.


In order to start the application, press the *Sensordata* icon, located among the other applications, that is highlighted with a dotted red box in Figure A.1.

*Figure A.1: The application screen.*
When the application is started the screen will display its interface, see Figure A.2. The following list describes the functions highlighted in this figure.

1. This is the location where the name of the save file is entered. The name have to be specified before the data collection starts.

2. This button starts the data collection. Data is written, in real time, to the save file on the SD card.

3. Here is data from the GPS presented, longitude, latitude and speed.

4. Here is data from the accelerometer presented, acceleration in the directions x, y and z.

5. Here is data from the gyroscope presented, pitch rate, roll rate and yaw rate.

6. This is a text string that indicates whether a critical event has been detected or not. The threshold used is based on jerk and has a value of -1g/s.

7. This button stops the data collection.

*Figure A.2: Screenshot of the application.*
In order to transfer the save files to a computer, use the cmd and type the command "adb pull sdcard/nameOfFile", where nameOfFile is the name of the file. In Figure A.3 the name of the file is Test1.

![Image of cmd output](image)

**Figure A.3:** The transfer command in cmd.

The save file will have a content as follows.

```
<table>
<thead>
<tr>
<th>AccX</th>
<th>AccY</th>
<th>AccZ</th>
<th>GyroX</th>
<th>GyroY</th>
<th>GyroZ</th>
<th>Longitude</th>
<th>Latitude</th>
<th>Speed</th>
<th>Time</th>
<th>Jerk</th>
</tr>
</thead>
<tbody>
<tr>
<td>m/s²</td>
<td>m/s²</td>
<td>m/s²</td>
<td>rad/s</td>
<td>rad/s</td>
<td>rad/s</td>
<td>DDD.dddd</td>
<td>DDD.dddd</td>
<td>m/s</td>
<td>ns</td>
<td>m/s³</td>
</tr>
<tr>
<td>1.0825489</td>
<td>0.69246</td>
<td>9.666598</td>
<td>-0.02238307</td>
<td>0.052226186</td>
<td>-0.051160924</td>
<td>15.572456941008568</td>
<td>58.39616728015244</td>
<td>0.0</td>
<td>5889892</td>
<td>0.0</td>
</tr>
<tr>
<td>1.0395882</td>
<td>0.8301737</td>
<td>9.837093</td>
<td>-0.005329518</td>
<td>0.1065845</td>
<td>-0.06075363</td>
<td>15.572456941008568</td>
<td>58.39616728015244</td>
<td>0.0</td>
<td>32409668</td>
<td>0.0</td>
</tr>
<tr>
<td>0.8457413</td>
<td>0.8469388</td>
<td>9.677226</td>
<td>-0.024514664</td>
<td>0.17373317</td>
<td>-0.041568484</td>
<td>15.572456941008568</td>
<td>58.39616728015244</td>
<td>0.0</td>
<td>36346435</td>
<td>0.0</td>
</tr>
<tr>
<td>0.69096315</td>
<td>1.0371932</td>
<td>10.15144</td>
<td>-0.062885754</td>
<td>0.20357683</td>
<td>-0.026646258</td>
<td>15.572456941008568</td>
<td>58.39616728015244</td>
<td>0.0</td>
<td>47515869</td>
<td>0.0</td>
</tr>
<tr>
<td>0.5510042</td>
<td>0.6373746</td>
<td>10.050999</td>
<td>-0.05649044</td>
<td>0.190787</td>
<td>0.004263188</td>
<td>15.572456941008568</td>
<td>58.39616728015244</td>
<td>0.0</td>
<td>59875488</td>
<td>0.0</td>
</tr>
</tbody>
</table>
```

...
The main task of the application is to collect data from the sensors and GPS in the cellphone. Data collection is performed when the accelerometer updates. The update frequency of the sensors is chosen to SENSOR_DELAY_FASTEST, i.e. as fast as possible, resulting in a sample rate around 100 Hz. Data from the accelerometer, gyroscope and GPS is saved, together with the time and jerk, each time the accelerometer is updated. Data is written directly to the SD card.

The application also detects critical braking maneuvers. This is accomplished by filtering the longitudinal acceleration, and then calculating the jerk. In the case of a detection, a text string is changed which states that a critical event has been detected.

```java
public class Hello4Activity extends Activity implements SensorEventListener {

    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
        getWindow().addFlags(WindowManager.LayoutParams.FLAG_KEEP_SCREEN_ON);
        mCriticalTextView = (TextView) findViewById(R.id.critical);
        dataSetEdit = (EditText) findViewById(R.id.dataSetName);
        mSensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
        mSensorAcc = mSensorManager.getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
        mAccXTextView = (TextView) findViewById(R.id.acc_x);
        mAccYTextView = (TextView) findViewById(R.id.acc_y);
        mAccZTextView = (TextView) findViewById(R.id.acc_z);
        acc33 = new float[33];
        time33 = new long[33];
        mSensorGyro = mSensorManager.getDefaultSensor(Sensor.TYPE_GYROSCOPE);
        mGyroXTextView = (TextView) findViewById(R.id.gyro_x);
        mGyroYTextView = (TextView) findViewById(R.id.gyro_y);
        mGyroZTextView = (TextView) findViewById(R.id.gyro_z);
        mStartButton = (TextView) findViewById(R.id.startButton);
        mStopButton = (TextView) findViewById(R.id.stopButton);
```
nf2 = NumberFormat.getInstance();
    nf2.setMinimumFractionDigits(2);
    nf2.setMaximumFractionDigits(2);

nf6 = NumberFormat.getInstance();
    nf6.setMinimumFractionDigits(6);
    nf6.setMaximumFractionDigits(6);

locationManager = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
mGpsLongTextView = (TextView) findViewById(R.id.gps_long);
mGpsLatTextView = (TextView) findViewById(R.id.gps_lat);
mGpsSpeedTextView = (TextView) findViewById(R.id.gps_speed);

locationListener = new LocationListener() {
    public void onLocationChanged(Location location) {
        longitude = location.getLongitude(); // Default format: degrees
        latitude = location.getLatitude(); // Default format: degrees
        speed = location.getSpeed(); // m/s;
        
        gpsRound = String.format("%" + nf6.format(longitude));
        mGpsLongTextView.setText("%" + gpsRound);
        
        gpsRound = String.format("%" + nf6.format(latitude));
        mGpsLatTextView.setText("%" + gpsRound);
        
        gpsRound = String.format("%" + nf2.format(speed * 3.6));
        mGpsSpeedTextView.setText("%" + gpsRound);
    }
    
    public void onStatusChanged(String provider, int status, Bundle extras) {}
    public void onProviderEnabled(String provider) {};
    public void onProviderDisabled(String provider) {};
};

@Override
protected void onResume() {
    super.onResume();
    mSensorManager.registerListener(this, mSensorAcc, SensorManager.SENSOR_DELAY_FASTEST);
    mSensorManager.registerListener(this, mSensorGyro, SensorManager.SENSOR_DELAY_FASTEST);
    locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, locationListener);
}

@Override
protected void onPause() {
    super.onPause();
    mSensorManager.unregisterListener(this);
}

@Override
public void onAccuracyChanged(Sensor sensor, int accuracy) {
    // Auto-generated method stub
}

@Override
public void onSensorChanged(SensorEvent event) {
    if (!STARTFLAG)
        return;
    else if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {
        mSensorAccX = event.values[0]; // m/s^2
        mSensorAccY = event.values[1]; // m/s^2
        mSensorAccZ = event.values[2]; // m/s^2
        currentTime = System.nanoTime() - startTime;
        
        accRound = String.format("%" + nf2.format(mSensorAccX));
        mAccXTextView.setText("%" + accRound);
        
        accRound = String.format("%" + nf2.format(mSensorAccY));
        mAccYTextView.setText("%" + accRound);
        
        accRound = String.format("%" + nf2.format(mSensorAccZ));
        mAccZTextView.setText("%" + accRound);
        
        if (jerk < -9.82)
            mCriticalTextView.setText("CRITICAL EVENT DETECTED. ...
            \n            " + String.format("%" + nf2.format(jerk / 9.82) + " g/s"));
        
        if (STARTFLAG)
            jerkCalc();
        buildDataMatrix();
    }
    else if (event.sensor.getType() == Sensor.TYPE_GYROSCOPE) {
        double piDouble = Math.PI;
        float piFloat = (float) piDouble;
        
        mSensorGyroX = event.values[0]; // rad/s
        mSensorGyroY = event.values[1]; // rad/s
        mSensorGyroZ = event.values[2]; // rad/s
        
    }
mSensorGyroXdeg = mSensorGyroX * (180/piFloat);
mSensorGyroYdeg = mSensorGyroY * (180/piFloat);
mSensorGyroZdeg = mSensorGyroZ * (180/piFloat);
gyroRound = String.format("% s\n", nf2.format(mSensorGyroXdeg));
mGyroXTextView.setText("% s\n",gyroRound);
gyroRound = String.format("% s\n", nf2.format(mSensorGyroYdeg));
mGyroYTextView.setText("% s\n",gyroRound);
gyroRound = String.format("% s\n", nf2.format(mSensorGyroZdeg));
mGyroZTextView.setText("% s\n",gyroRound);
}
else
return;
}

public void jerkCalc() {
for(int i=1 ; i<33 ; i++) {
acc33[33-i] = acc33[32-i];
}
acc33[0] = mSensorAccY;
time33[0] = currentTime;
acc11sum1 = 0;
acc11sum2 = 0;
for(int i=1 ; i<12 ; i++) {
acc11sum1 += acc33[i];
acc11sum2 += acc33[i+20];
}
acc11LP1 = acc11sum1/11;
acc11LP2 = acc11sum2/11;
if (currentTime > 1000000000) // One second elapsed
jerk = 1000000000*((acc11LP1 - acc11LP2)/(time33[6] - time33[26])); // m/s^3
else
jerk = 0;
}

public void buildDataMatrix() {
try {
oss.write(mSensorAccX + "\n" + mSensorAccY + "\n" + mSensorAccZ + "\n" + mSensorGyroX + "\n" + mSensorGyroY + "\n" + mSensorGyroZ + "\n" + longitude + "\n" + latitude + "\n" + speed + "\n" + currentTime + "\n" + jerk + "\n") ;
} catch (IOException e) {
e.printStackTrace();
}
}

public void onStartButtonClick(View view) {
if (mStartButton.getText() == "Collecting data!")
return;
else if (mStartButton.getText() != "Data collection stopped!") {
datasetName = dataSetsEdit.getText().toString();
if (datasetName.isEmpty()) {
Toast toast = Toast.makeText(getApplicationContext(), "Enter file name first!", Toast.LENGTH_SHORT);
toast.setGravity(Gravity.CENTER_HORIZONTAL|Gravity.CENTER_VERTICAL, 0, 75);
toast.show();
return;
}
else {
try {
sdCard = Environment.getExternalStorageDirectory();
dir = new File(sdcard.getAbsolutePath());
dir.mkdirs();
file = new File(dir, dataSetName);
Out = new FileOutputStream(file);
oss = new OutputStreamWriter(Out);
oss.write("AccX,AccY,AccZ,GyroX,GyroY,GyroZ,Longitude,Latitude,Speed,Time,Jerk\n");
oss.write("[m/s^2],[m/s^2],[m/s^2],[rad/s],[-rad/s],[-rad/s],[DDD,ddd],[DD.DDDD],[m/s][-ms][-ms],[m/s^3]\n");
} catch (IOException e) {
Log.d("WriteSdCard", e.getMessage());
}
startTime = System.nanoTime();
STARTFLAG = true;
mStartButton.setText("Collecting data!");
}
public void onStopButtonClick(View view) {
    if (mStopButton.getText() == "Data collection stopped!")
        return;
    else if (!STARTFLAG) {
        Toast toast = Toast.makeText(getApplicationContext(),
            "No data collection running!", Toast.LENGTH_SHORT);
        toast.setGravity(Gravity.CENTER_HORIZONTAL|Gravity.CENTER_VERTICAL, 0, 75);
        toast.show();
    } else {
        Toast toast = Toast.makeText(getApplicationContext(),
            "Data saved to file " + dataSetName, Toast.LENGTH_LONG);
        toast.setGravity(Gravity.CENTER_HORIZONTAL|Gravity.CENTER_VERTICAL, 0, 75);
        toast.show();
        mStopButton.setText("Data collection stopped!");
        mStartButton.setText("Data collection stopped!");
        STARTFLAG = false;
        try {
            osw.flush();
            osw.close();
            fOut.close();
        } catch (Exception e) {
            Log.d("WriteSdCard", e.getMessage());
        }
    }
}
Upphovsrätt

Detta dokument hålls tillgängligt på Internet — eller dess framtida ersättare — under 25 år från publiceringsdatum under förutsättning att inga extraordinära omständigheter uppstår.

Tillgång till dokumentet innebär tillstånd för var och en att läsa, ladda ner, skriva ut enstaka kopior för enskilt bruk och att använda det oförändrat för icke-kommersiell forskning och för undervisning. Överföring av upphovsrätten vid en senare tidpunkt kan inte upphäva detta tillstånd. All annan användning av dokumentet kräver upphovsmannens medgivande. För att garantera äktheten, säkerheten och tillgängligheten finns det lösningar av teknisk och administrativ art.

Upphovsmannens ideella rätt innefattar rätt att bli nämd som upphovsman i den omfattning som god sed kräver vid användning av dokumentet på ovan beskrivna sätt samt skydd mot att dokumentet ändras eller presenteras i sådan form eller i sådant sammanhang som är kränkande för upphovsmannens litterära eller konstnärliga anseende eller egenart.

För ytterligare information om Linköping University Electronic Press se förlagets hemsida http://www.ep.liu.se/

Copyright

The publishers will keep this document online on the Internet — or its possible replacement — for a period of 25 years from the date of publication barring exceptional circumstances.

The online availability of the document implies a permanent permission for anyone to read, to download, to print out single copies for his/her own use and to use it unchanged for any non-commercial research and educational purpose. Subsequent transfers of copyright cannot revoke this permission. All other uses of the document are conditional on the consent of the copyright owner. The publisher has taken technical and administrative measures to assure authenticity, security and accessibility.

According to intellectual property law the author has the right to be mentioned when his/her work is accessed as described above and to be protected against infringement.

For additional information about the Linköping University Electronic Press and its procedures for publication and for assurance of document integrity, please refer to its www home page: http://www.ep.liu.se/

© Henrik Hagelin