SAARLAND UNIVERSITY

Faculty of Natural Sciences and Technology I
Department of Computer Science
Master's thesis

OpenIndoorMap

Smartphone-based capture of uninstrumented indoor environments

Frederic Kerber
Master's Program in Computer Science
March 2012

Advisor:
Florian Daiber, German Research Center for Artificial Intelligence, Saarbrücken, Germany

Supervisors:
Prof. Dr. Antonio Krüger, German Research Center for Artificial Intelligence, Saarbrücken, Germany
Dr. Tim Schwartz, Saarland University, Department of Computer Science, Saarbrücken, Germany

Reviewers
Prof. Dr. Antonio Krüger, German Research Center for Artificial Intelligence, Saarbrücken, Germany
Dr. Tim Schwartz, Saarland University, Department of Computer Science, Saarbrücken, Germany

Submitted
30th March, 2012

Saarland University
Faculty of Natural Sciences and Technology I
Department of Computer Science
Campus - Building E1.1
66123 Saarbrücken
Germany

Eidesstattliche Erklärung

Ich erkläre hiermit an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

Statement in Lieu of an Oath

I hereby confirm that I have written this thesis on my own and that I have not used any other media or materials than the ones referred to in this thesis.

Saarbrücken, 30^{th} March, 2012

Einverständniserklärung

Ich bin damit einverstanden, dass meine (bestandene) Arbeit in beiden Versionen in die Bibliothek der Informatik aufgenommen und damit veröffentlicht wird.

Declaration of Consent

I agree to make both versions of my thesis (with a passing grade) accessible to the public by having them added to the library of the Computer Science Department.

Saarbrücken, 30^{th} March, 2012

Acknowledgments

I thank a number of people for making this thesis possible. First of all, I thank Professor Antonio Krüger for providing the opportunity to work on such an interesting topic and for always being a helpful supervisor. I also thank my second supervisor, Dr. Tim Schwartz, who also provided advices during the writing of this thesis. Special thanks also to Dr. Johannes Schöning who advised me at the beginning of my work. I also thank my advisor Florian Daiber who spend many hours of discussion with me.

I thank Michael Mauderer who helped me on short notice by providing sensor data from his smartphone. Moreover, I say thank you to Pascal Lessel for his excellent co-operation not only during this thesis, but also during our whole duration of study. His remarks were always a source of inspiration and his help in proofreading indispensable. In this context, I also thank Fritz Rüdell for correcting my linguistic mistakes.

Finally, I would like to thank my parents and my wife for their appreciation and support whenever I needed it.

Abstract

During the last years, the relevance of digital maps has rapidly increased – services such as Google Maps are used by many people everyday. Another observable trend is the expansion of the OpenStreetMap project that aims at providing free geographic data of the world that is collected by a large number of volunteers. Currently, the existing digital map services focus on outdoor environments, but recently also the inside of buildings has come to the center of attention. With the help of an online questionnaire we gathered opinions of how people think about digital indoor maps and their usage in general and we have learned that people have at least a certain interest in this field and would not only use a digital map service but also as contribute to it. As traditional mapping methods mostly rely on GPS signals which are usually not available indoors or, if they are, only with low accuracy, we developed alternative techniques for indoor mapping. To enable the participation of many people we did not focus on systems that require particular hardware installations on location as it would be the case in instrumented environments. Instead, our system only requires a standard consumer smartphone, as these devices nowadays have many different types of sensors. We developed two indoor mapping approaches that employ some of the available sensors – in the first approach, they are used to keep track of a user's movements when walking along the walls of a room to capture its shape. This approach is better suited for larger rooms where no furniture is standing on the walls. In contrast, the second approach is better suited for smaller rooms where the placement of furniture is not of great importance. In this approach, the sensors are used to determine a room's dimensions with the help of trigonometric calculations. Some initial tests with both techniques showed promising results with respect to the accuracy needed for creating a floor plan of the captured environment. Furthermore, our system offers options for post-processing the captured data still on location as well as an export possibility for further usage of the captured data.

Contents

Chapter 1
Introduction

This Master's thesis presents the OpenIndoorMap toolkit – a mobile application for the Android OS. The toolkit allows to capture indoor environments with the help of standard consumer devices such as smartphones. For this purpose, two different methods to gather information on location will be presented. The collected data can be exported to enable a usage in different scenarios.

1.1 Motivation

Nowadays, a lot of digital maps exist in everyday life. Several services such as Google Maps[1] or comparable map applications have rapidly increased the popularity of these maps as they offer information beneficial to a huge number of people. Compared to traditional i.e. paper-based maps, the digital versions offer numerous advantages. One of these advantages is the possibility to easily update existing maps. For paper-based maps this is only possible by obtaining a completely new map. In contrast, for digital versions it is sufficient to update only the changed parts – for example through a possibly automated synchronization process. Another benefit is the capability to easily combine different single layers to achieve new maps. Such a particular layer can contain information on a separate feature e.g. the population or the vegetation. With the help of digital maps, users can combine the individual levels to get a map perfectly fitting their needs. This is a clear advantage compared to paper-based maps where for each combination of different layers a separate version had to be prepared in advance. This is not necessary for digital maps where only the single layers have to be prepared beforehand and then arbitrary combinations can be generated. Furthermore, digital maps can profit from interaction possibilities offered by

[1] `http://maps.google.com`, last accessed March 27, 2012

computers or comparable devices. It is for example possible to show additional information on demand e.g. when the user clicks on a point of interest (POI). This enables the map designer to provide manifold information without the risk of overloading the map. The digitally available information to build a map can also be used for several other purposes e.g. Location-Based Services (LBS). A typical example for a LBS is turn-by-turn navigation. For such applications the availability of up-to-date map information is a large benefit.

Currently, most of the digital maps focus on outdoor environments. The most popular example for such maps are street maps. Especially in this context, several of the aforementioned advantages are of relevance: up-to-date information is needed to reflect structural changes in the road network, a separate layer containing information like service stations can be shown on request and also in-car navigation is enhanced through the availability of digital maps. The combination of these features allows for a reliable navigation system e.g. for cars. Furthermore, much additional information has been included in digital maps in recent years. A popular project showing this trend is the OpenStreetMap[2] project (OSM). Basically any information can be added to the database by users (see Section 1.2 for further details). This opportunity is widely used and a lot of objects e.g. post boxes or fire hydrants are currently present. In the last two years, another trend could be observed: an increasing amount of buildings are added to the database. Additionally, also indoor objects are present in growing number. Figure 1.1 shows a graphical representation of this evolution.

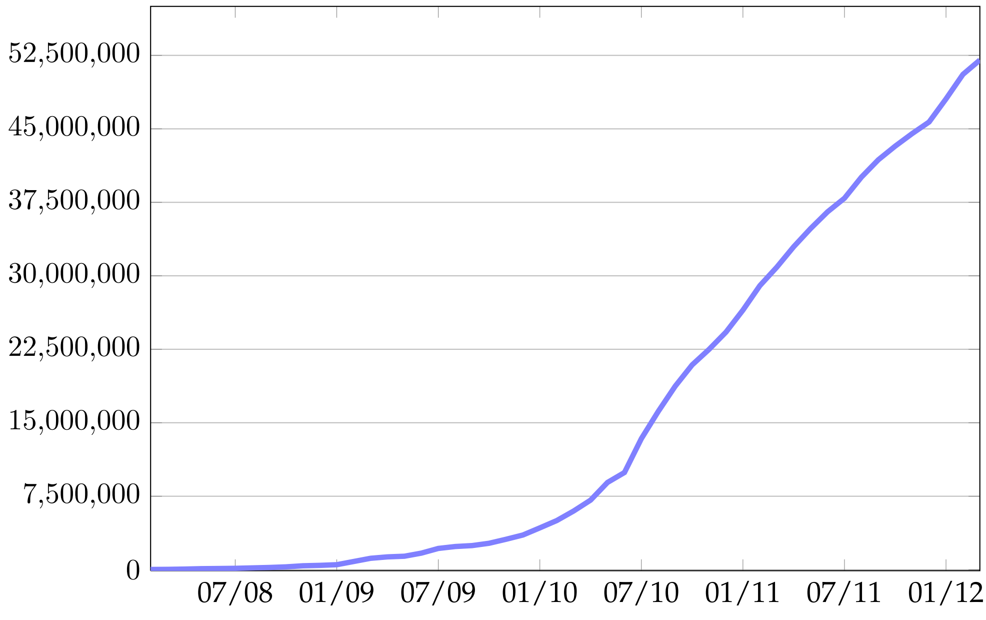


Figure 1.1: Number of usages of the building key in OSM between 2008 and 2012[3]

[2] `http://www.openstreetmap.org`, last accessed March 27, 2012
[3] Underlying data provided by Marcus Götz (marcus.goetz@geog.uni-heidelberg.de), cf. also [22]

This is a remarkable progress that leads to further questions. One could be in which way people can profit from the availability of building information. A basic usage would be to enhance the above mentioned navigation system towards door-to-door navigation as it requires information on house numbers of individual buildings. This information can now be added to the building shapes that are available in the database. But this extension is directly followed by the idea to expand the navigation even further. Especially for larger buildings like shopping malls or airports it could be beneficial if the navigation did not end at the main entrance but continued to the destination of the user. Such a navigation approach could be of relevance for all types of buildings where a user is not only interested in reaching the building itself but a specific shop or office inside the building. Without digital maps, only little possibility is offered for people to inform themselves a priori how to reach a specific location inside a building as printed maps are virtually unavailable. A lot of different use cases are imaginable in which people could profit from the ability to be directly navigated towards a specific location inside a building – for example, a family that tries to catch a flight at the airport and therefore looks for their gate. Without a digital map the only possibility would be to look for signs when already on location with the potential disadvantage of a certain time pressure. In contrast, if such a map is available beforehand, the best way could be determined unhurriedly and it could also be estimated how much time has to be calculated for reaching the gate without any time pressure. Another example would be to find a specific office in a large administrative building or a certain sickroom inside a hospital. Normally, only the room number is known and in the best of cases this allows for inference on the level. When arriving at the correct floor, the only possibility is to inspect the name plates at every door until the desired room is found. However, if indoor information is available, the desired destination can be found easily without such a brute-force approach. The quick localization of rooms inside a building is also of interest for emergency units. Even trained units could have problems with a complicated building's architecture [2]. For example in case of a medical emergency minutes can be important and therefore many people could benefit from the availability of indoor maps.

1.1.1 Availability of digital indoor maps

Several companies have already started projects that deal with indoor maps in different ways. In December 2010, Bing Maps introduced mall directory maps[4]. With the help of these maps people were to be able to figure out where the best place for parking is located to reach a specific store without making a detour. NAVTEQ, the company behind Nokia's Ovi maps, announced their new product "Destination Maps"[5] in March 2011 and thereby claimed to be "the first global map

[4] `http://www.bing.com/community/site_blogs/b/maps/archive/2010/12/07/bing-s-new-mall-maps-get-in-get-out-and-the-avoid-the-crowds.aspx`, last accessed March 27, 2012

[5] `http://press.navteq.com/?s=4260&item=30551`, last accessed March 27, 2012

provider to offer this type of indoor mapping product". The system includes a three-dimensional data model to provide an advanced exploration and guidance experience. Furthermore, access restrictions are supported to offer a precise navigation for pedestrians to POIs like public convenience. In October 2011 Ericsson Labs announced a new "Indoor Maps and Positioning" platform[6]. In contrast to other indoor mapping systems, they stressed that they could offer a complete system including a map studio to create maps, an Android map API as well as an Android Wi-Fi positioning API. The system is currently in the public beta phase. In November 2011, Google extended their maps application to also offer indoor maps[7]. Initially, indoor maps of some of the largest retailers, airports and transit stations in the U.S. and Japan were available. Also for the OSM project some buildings were added with detailed indoor coverage. According to their Wiki[8] there exist 15 buildings worldwide in the database with such coverage – a comparably small amount. This is strange, especially if we recall the increasing number of mapped building shapes. In the following section we will investigate the idea behind projects such as OSM to illustrate how such information is made available to the public.

1.2 Volunteered Geographic Information

The general concept behind collaborative mapping projects like OSM is the idea of Volunteered Geographic Information (VGI). The term was coined by Michael F. Goodchild as "a special case of the more general Web phenomenon of user-generated content" [21]. Several different technologies were mentioned as prerequisite for the concept of VGI: a key factor is the evolution of the Internet towards Web 2.0. The new possibility of adding user-generated content via the Internet has created entirely new types of Web pages as for example Wikis. For these Web pages only little or even no moderation by an administrator is necessary as other users of the system take over this task. Another necessity is the availability of georeferencing tools, i.e. addressing a location on the Earth's surface using a well-defined interoperable system. A lot of the consumer devices available today provide such functionality by using the Global Positioning System (GPS). Examples of such devices are cameras that are equipped with GPS and allow to automatically tag taken photos with geo-coordinates. Moreover, some receivers offer the possibility to log complete GPS tracks that can later be used in digital form. A great change in the availability of GPS receivers could be achieved with the help of the mobile phone market. Many new smartphone devices are equipped with GPS technology and provide therefore functionality similar to a discrete GPS receiver. Another remarkable element of the evolution was the

[6] `https://labs.ericsson.com/developer-community/blog/complete-indoor-mapping-start-finish`, last accessed March 27, 2012

[7] `http://googleblog.blogspot.com/2011/11/new-frontier-for-google-maps-mapping.html`, last accessed March 27, 2012

[8] `http://wiki.openstreetmap.org/wiki/Indoor#Places`, last accessed March 4, 2012

change in the computational and graphical power of household computers. This enables for example the visualization of three-dimensional objects as it is used in applications like Google Earth[9]. Finally, the impact of a widely available broadband Internet access on a VGI project should not be underestimated. Examples of popular VGI projects are given in Figure 1.2.

(a) Marked places in Wikimapia

(b) Map section from OpenStreetMap

Figure 1.2: Examples of VGI projects

Wikimapia[10], founded in 2006, is a combination of Google Maps and a wiki system. Users can mark areas by rectangles and polygons as well as add notes to any location of the world. At the time of this writing more than 18 million places have been created. Since December 2010, all contents have been available under Creative Commons Attribution-NonCommercial-ShareAlike license 3.0[11]

[9] `http://earth.google.com`, last accessed March 27, 2012
[10] `http://www.wikimapia.org`, last accessed March 27, 2012
[11] `http://creativecommons.org/licenses/by-nc-sa/3.0`, last accessed March 27, 2012

through the API but as the collected data is built on aerial photos of Google Maps, it could be subject to copyright issues as derived work.

In contrast to this, the OSM project tries to build a free map of the world that is not affected by such copyright issues. All content is currently licensed under Creative Commons Attribution-ShareAlike 2.0[12] but should soon be relicensed under the Open Database License[13]. Currently, more than 550.000 users are registered and have the possibility to add new data or change existing entries. Over the past eight years nearly 1.4 billion nodes, more than 127 million ways (ordered lists of nodes) and 1.3 million relations (aggregations of different nodes and ways) have been added to the database. The collected data is based on data from user-generated GPS tracks, appropriate aerial images, other free sources or local knowledge of the mappers.

For the their new Floor Plan[14] project Google Maps also tries to use the power of volunteers to collect new information on indoor environments. People can upload images of floor plans and thereby increase the database of buildings equipped with indoor information.

1.3 Research aims

The main aim of this thesis is to investigate data acquisition methods that can be used in the course of a VGI project for indoor environments. This especially addresses methods that can be used by a large number of people without any particular prerequisites – a key factor for the success of a VGI project. The main focus is therefore on methods that do not require any special hardware, but are applicable with standard consumer devices e.g. smartphones.

In the course of this thesis the following four main research aims will be covered:

Investigation of the public interest in digital indoor maps

It will be examined to which extent people are interested in the field of indoor maps in general. This covers on the one hand questions about situations in which people would be contented to have indoor maps or about the expected frequency of indoor map usage. But on the other hand it will also be examined if and under which circumstances people are willing to contribute to a VGI project dealing with indoor environments. Similar questioning will be conducted with regards to the OSM project to enable a classification of the gathered results as the OSM project is an example of a very successful VGI project in a similar domain. The results of this investigation will be presented in Chapter 3.

[12] `http://creativecommons.org/licenses/by-sa/2.0`, last accessed March 27, 2012
[13] `http://opendatacommons.org/licenses/odbl`, last accessed March 27, 2012
[14] `http://maps.google.com/help/maps/floorplans`, last accessed March 27, 2012

Technical possibilities for indoor mapping on location

It will be investigated which technical methods could be appropriate for indoor mapping. In indoor environments special attention has to be paid to structural circumstances like metal packages in walls that can disturb measurements. Furthermore, it has to be considered which limitations exist when using standard consumer devices, e.g. which types of sensors are available at all. It should also be taken into account that the available sensors are mostly small and comparably low priced. Hence, the provided data could be more inaccurate or provided with lower frequency compared to other more complex sensor units. Details on the properties of the different sensor types are discussed in Chapter 4.

Algorithmic approaches for indoor mapping

In the course of this thesis we will not only provide one single solution but different ones in order to find their advantages and shortcomings. In Chapter 2 several related systems are presented and it will be discussed to which extent their approaches could be of use for our OpenIndoorMap toolkit. The implementation of the algorithms is described in Chapter 5. To assess the quality of the proposed algorithms we will test how accurate the provided results are and under which circumstances the best measurements can be obtained. The results of an initial evaluation are provided subsequent to the description of the implementation.

Making the captured data available

The data that can be collected on-location should be made available for as many different use cases as possible. Such a widespread distribution can enormously increase the number of people that are interested in collecting the data themselves. Therefore, one aim of this thesis is to provide an output format that could later be used to transfer the collected data to already existing applications. In Chapter 7 we will outline some systems for which the collected data could be useful.

Chapter 2
Related Work

In the following chapter, related work in the field of indoor mapping is investigated. In the first section, several systems are presented that make use of a so-called Dead-Reckoning approach. Although they are not intended for room measurement, the underlying technical principles can be used for this purpose. Afterwards, some mobile apps for smartphones are introduced which provide functionality for automated measurements of object dimensions.

2.1 Personal Dead-Reckoning

In the following section several systems in the field of pedestrian tracking will be presented. As GPS signals are seldom available indoors with reasonable precision, only techniques whose main functionality is not based on GPS will be considered. The remaining systems can be classified into those that need external references and also others that do not require such installations. The major drawback of the former is the requirement to equip the environment with either observable signal emitters or otherwise observable passive waypoints. Examples for such techniques can amongst others be based on infrared light as presented by Butz et al. [4] or ultrasound as done by Cho and Park [11] as well as Priyantha et al. [49]. Another approach is based on so-called opportunistic positioning systems that exploit already available instrumentations originally installed for a different purpose [52]. For example, locally installed Wi-Fi can be utilized for pedestrian tracking. Although no extra installation is required beforehand, there is still the need to initially create a database with all the Wi-Fi senders' positions as it was the case in the development of Yamasaki et al. [62] and in the one by Golden and Bateman [19]. Another approach which first requires a setup phase to build a database for comparing reasons is based on computer vision: Liu et al. [39] present a system that allows image matching against a previously created database. A similar approach has also been taken by Kourugi and Kurata [32].

As we target a system that is usable without any expensive setup phase, this chapter will focus on the second group of systems, namely those that do not require any external instrumentation. To achieve the goal of tracking a pedestrian's movements a so-called Inertial Measurement Unit (IMU) can be used. Such an IMU consists of several gyroscopes and accelerometers, but also other sensors can support the tracking process (see Section 4.1 for a discussion of the different sensor types). With the help of the collected sensor data, the person's movements as well as changes in his heading can be determined. The process of determining the current position by considering heading and speed in combination with a known past position is called Dead-Reckoning (DR) [50]. In this special context also the term Pedestrian or Personal Dead-Reckoning (PDR) is used.

In the following sections, several implementations are presented. First, systems with particular hardware requirements are introduced and compared. The ones realizable with widely available consumer hardware like smartphones are presented afterwards.

2.1.1 Personal Dead-Reckoning with particular hardware

As already outlined, the tracking of a pedestrian's movements can be detected with the help of IMUs and additional sensors. The use of supplementary sensors is necessary as the temporal integration of IMU sensor data will propagate large errors even when using high accuracy sensors [13]. The different implementations vary depending on the part of the body on which the sensors are worn (see Figure 2.1 for examples). A common place for the sensors are the user's feet or shoes – these systems are presented in the next section. Other possible spots like head, legs or waist are discussed afterwards.

(a) Shoe-mounted[15] (b) Helmet-mounted[16] (c) Belt-mounted[17]

Figure 2.1: Different mount points for sensor units

[15] Picture taken from [25]
[16] Picture taken from [3]
[17] Picture taken from [54]

MEMS Based Pedestrian Navigation System

The Pedestrian Navigation System [11] presented by Cho and Park is a Dead-Reckoning approach based on the idea of a pedometer. With the help of a biaxial accelerometer mounted on a shoe, they detect individual steps and use this information by multiplying it with the step length to calculate the covered distance. The determination of the step length is a critical part as it depends on the walking speed, the ground inclination and other factors. It could be shown that the step length is proportional to the walking frequency and the variance of the accelerometer readings (cf. [9, 12, 28, 33, 36, 38]), but Cho and Park also illustrated that it has a complex tendency according to the ground inclination. Therefore, they solved the non-linear problem about the ground inclination through a neural network using the walking frequency, the variance of the accelerometers and the ground inclination as inputs. Finally, they calculated the user's azimuth using a biaxial magnetometer also mounted on the shoe. The error of the magnetometer is bounded unlike the one of a gyroscope, which can grow arbitrarily large due to integration. But the error may include components resulting from tilt generated by the inclination. In general, such errors can be avoided by using a triaxial magnetometer or an inclinometer but as the authors found them too bulky, they applied a tilt-compensation algorithm for biaxial magnetometers developed by Cho and Park [10]. Although this error can be compensated, errors will remain and increase over time due to the Dead-Reckoning construction. To overcome this problem, Cho and Park combined their system with readings from GPS and incorporated these values with the help of a Modified Receding Horizon Kalman Filter (MRHKF) based on a Receding Horizon Extended Kalman Filter (RHEKF) presented in [8]. The advantages of the modified version are a decreased computational load and an enhanced convergence characteristic. A field test with a closed-loop of 3,730 meters was conducted and showed an error of 58 meters (1.55%) which is, according to the authors, better than in previous work.

The work provides interesting concepts especially regarding the different error correction and filtering approaches. Nevertheless, the question is whether the different algorithms, particularly the neural network, are too complex and power consuming to be feasible on standard mobile devices like smartphones.

Personal Odometry Sytem / Personal Dead-Reckoning system

Ojeda and Borenstein presented a Personal Odometry system (POS) [44, 45] based on a six degrees of freedom IMU (3-axis gyroscope and 3-axis accelerometer) made by BAE[18] attached to one of the user's boots. Based on the sensor data the POS computes the complete trajectory of the boot on each step. For legged motion like walking, running or climbing, the POS can compute linear displacement (i.e. odometry) as well as a position estimation (i.e. Dead-Reckoning). To reduce drift from the sensor readings, a method called Zero Velocity Update (ZUPT) is

[18] `http://www.baesystems.com`, last accessed March 27, 2012

used: during a normal stride, there is a period of time in which the bottom of the sole is in contact with the ground. During this time, there is no movement of the IMU relative to the ground and the accelerometer readings should be zero. The difference between zero and the actual values is the result of drift. With this information, subsequent readings can be corrected. As this procedure can be repeated with every stride, there is always at least one point in every stride where all drift is removed. With a similar reasoning, the ground speed is also reset with every stride to the known value zero. With this approach, any error produced during one stride only affects this one and is not carried over subsequently. To detect the period of time when the drift correction can be applied, the accelerometer readings alone are not applicable because they will not show zero due to drift. Therefore, the readings from the gyroscope are taken into account and compared to a threshold value. All sensor values below this threshold are considered and the smallest identifies the time instance at which the drift correction should be applied. Several experiments presented in the paper showed an error rate of up to 0.8% for walking a straight line of 40 meters length where only the linear displacement was considered. An experiment with a closed loop in one plane (2D) results in error rates of up to 0.9% for a distance of 65 meters. When also the third-dimension was considered, the error rate grew up to 1.4% for a closed loop path of 104 meters length including two stairways. Finally, two longer-duration experiments with distances of 1,010 meters and 896 meters were made and resulted in error rates of up to 2.33%. The general problem for longer walks is the increasing drift of the gyroscopes and hence the resulting errors that grow unbound as a function of time. Another problem the authors mentioned was the used IMU as it is bulky, heavy and expensive. Therefore, they continued their work with a much smaller IMU from MemSense[19] as presented in [46, 47]. The conducted experiments lead to comparable results.

The approach shows that very accurate computations can be done with the help of a six degrees of freedom IMU. The problem of the increasing errors when used for longer walks for our approach is not of such great importance as typical room contours seldom have more than 100 meters. Meanwhile, a different solution for the drift correction must be found as the ZUPT approach [44, 45] is not feasible for sensors not mounted to the shoe.

A comparison of PDR algorithms using a low-cost MEMS IMU

In their publication [25], Jiménez et al. presented a comparison of different Dead-Reckoning algorithms with low-performance sensors. They used an IMU built by Xsens[20] consisting of triaxial accelerometers, gyroscopes and magnetometers. The IMU was placed at the user's shoe to get a more robust step detection. The first step detection method is based on the accelerometer reading's magnitudes and local variance. Steps are detected if there is a transition from high ($> 2m/s^2$)

[19] http://www.memsense.com, last accessed March 27, 2012
[20] http://www.xsens.com/, last accessed March 27, 2012

to low ($< 1m/s^2$) acceleration. A second step detection method was implemented using gyroscopes and magnetometers. Initially, the total angular rate magnitude is computed, followed by a thresholding process ($1rad/s$). Next, a median filter is applied to remove outliers and finally, transitions to a motionless state are detected. As a variation, a third algorithm using magnetometer readings was developed: first, a high pass filter to remove undesired measurements is applied and then the same steps of the aforementioned algorithm are used. All three algorithms were tested with a test set of 955 steps to detect. The best results were achieved with the acceleration algorithm (error rate 0.1%), the one using the gyroscopes performed only slightly worse (error rate 0.2%) and even the third method achieved not too bad results (error rate 0.94%).

To estimate the stride length, also two algorithms were considered. The Weinberg stride length algorithm assumes that the stride length is proportional to the vertical movement of the hip (see [59] for details). The second algorithm computes the stride length by integrating accelerometer readings using ZUPT [44, 45] for drift correction. With both algorithms tests at different walking speeds were conducted and yielded error rates between 0.3% and 0.78% of travelled distance (360 meters) for the first approach and between 0.62% and 1.15% for the second.

For position estimation three different algorithms were examined. Two of them were based on the addition of foot displacements (using the estimated stride lengths) whereas the third accumulates the positioning increments gained during the ZUPT calculations. All algorithms were tested indoors and outdoors with closed-loop trajectories between 100 and 320 meters. All algorithms produced a final position error between 5 and 15 meters (error rate < 5% of travelled distance), but the third algorithm performed a little worse than the other two.

The comparison discussed above provides interesting insights into different algorithms. The step detection tests showed that it is especially hard to identify steps at the beginning or end of a motion as well as when the user is walking at a very low speed. The similar results for the position estimation show that also sensor units without gyroscopes can be used for a Dead-Reckoning approach.

A helmet-mounted PDR system

Beauregard presented a system [3] based on a combination of a GPS device for outdoor usage and calibration as well as an IMU (Xsens MT9 IMU) consisting of accelerometer, gyroscope and magnetometer for indoor tracking. The step length was estimated based on a linear combination of step frequency, variance in accelerometer magnitudes and vertical velocity as it was presented in [28, 37] in combination with a neural network considering patterns from GPS position fixes outdoors. Due to the neural network the cumulated step distances showed a low error rate of about 5%. The heading estimation was based on the IMU's yaw output after some calibration steps, but no further details on this were published. An outdoor experiment including GPS fixes showed an error of about 40 meters after a walk of 30 minutes. Another experiment including indoor and outdoor transitions was also conducted but no clear statement on its performance was given, but the results showed some distortions of greater magnitude indoors.

The placement of the IMU and the presumably acceptable position calculation indoors shows that shoes are not the only sensor mounting points that can provide useful measurements. This finding is of importance as we will also have a system without shoe-mounted sensors. However, a big difference lies in the fact that the sensors in this approach are at fixed positions which will not be the case when the user is holding a mobile device.

Pedestrian Navigation Module

Ladetto and Merminod presented a Pedestrian Navigation Module (PNM) [34] consisting of a high performance GPS receiver, an IMU with accelerometers and magnetometers with an embedded Dead-Reckoning algorithm, a barometer as well as a gyroscope. The system is designed for several use cases like navigation for the blind or locating rescuers in case of an emergency. The optimal position is considered to be belt-attached. The PNM is able to detect step occurrences with the help of accelerometer readings as well as giving the direction of displacement (forward, backward, left or right). It is also possible to determine the type of movement (e.g. going up or down stairs) and the type of ground over which the person walks (e.g. hard or soft surface). The underlying model for stride length estimation can be calibrated using either speed measurements or a known distance. The PNM calculates the person's speed and with this information, different physiological models can be applied to determine step length. The heading of the person is estimated via magnetometer and gyroscope readings. If magnetic disturbances are present, the magnetometer will react while the gyroscopic data remains unaffected. Hence, in such a situation only the gyroscopic data is considered. Otherwise the readings from magnetometer and gyroscope are merged through a Kalman filter. If GPS signals are available, the bias of magnetometer and gyroscope is corrected. With the help of barometric data, the system is also able to provide 3D positioning data. Several tests with the PNM were reported and provided promising results both outdoors and indoors.

The PNM shows again that Dead-Reckoning is also feasible with sensors that are not shoe-mounted. The approach provides an inspiring idea by combining gyroscopic data with magnetometer readings to cope with magnetic disturbances.

PDR System with Phone Location Awareness Algorithm

The PDR system [54] presented by Shin et al. uses a phone location awareness algorithm which should improve the overall performance of Dead-Reckoning approaches. As step detection method a so-called zero crossing method was used on the accelerometer readings: a step is detected if the acceleration changes from a positive to a negative value. It was considered most appropriate for this use case as a peak detection method would be too greatly affected by the walking velocity whereas a flat zone detection method cannot be used when the PDR system is mounted to the user's waist belt as it was the case here. The observations reported in the paper show that the stride length is influenced linearly by the walking frequency and by a variance of the accelerometer readings. The exact linear combination can be learned during a calibration phase. One major problem of the PDR approach is the mis-detection of steps as the covered distance is computed as the sum of all estimated stride lengths. Typical situations for a false detection are sit down or stand up movements as in these cases a zero crossing is detected. Therefore a location awareness algorithm was considered helpful.

Furthermore, three different sensor mount positions were investigated: hand (swing), hand (fix) and pants pocket. The algorithm was designed using variance of the triaxial gyroscope norm, difference of pitch angle and norm of horizontal acceleration. For hand (fix) the variance of the gyroscope norm and the difference of pitch angle are smaller compared to other locations. For these, the norm of horizontal acceleration and difference of pitch angle are taken into account: they are smaller for pants pocket compared to hand (swing). Some experiments were made and showed error rates of less than 2% for the phone location awareness.

The location awareness algorithm discussed above showed very promising results. It can be expected that the overall performance of PDR systems is improved when such an algorithm comes into play as it reduces mis-detections of steps. Especially for situations where it is unclear how the user is holding the device, such an approach could be of interest. For systems with mobile devices like smartphones the use of improvements like this can increase the usability as the user has more possibilities to hold the device. Unfortunately, no information was given on the heading estimation of the used algorithm.

2.1.2 Personal Dead-Reckoning with consumer devices

In the following section, systems are presented that do not require special hardware components, but can be used with consumer devices like smartphones. As sensor units in such devices have to be relatively small and cheap, often only sensors with low update performance and low accuracy are used. It might be more challenging to design algorithms that can cope with the therefore inaccurate data, but which can still provide correct orientation estimates and user positions.

DRec: exploring indoor navigation with an un-augmented smart phone

The DRec system by Dekel and Schiller [15] implements a Dead-Reckoning navigation application on an iPhone 3GS. As the readings of the accelerometer are rather weak when holding the device in hand, the authors did some initial tests to cope with noise and found a reasonable cut-off frequency and sample rate to deal with. To test the accuracy of the step detection method a 50 step count test was performed (maximum error rate 6%) as well as a 200 meters walk with around 260 steps (maximum error rate 1.2%). To test the distance measurement, the system was calibrated to the step length of one test person and then distances of 5 meters, 10 meters and 50 meters were covered resulting in error rates of 8%, 8% and 2% respectively. In combination with magnetometer readings for heading estimation also a pre-test for Dead-Reckoning navigation was carried out resulting in an average error rate of 6.3%. This last test showed that the error grew with the distance walked which shows that an error correction has to be applied. The authors planned to continue the testing process with respect to the distance measurement and the navigation aspect. Moreover, they considered the integration of available indoor maps or an interaction with public Wi-Fi systems.

Apart from the future work presented in the publication the approach is closely related to what we want to show in this thesis although no navigational aspect will be considered. Special attention should be paid to the reliability of the magnetometer readings in indoor environments (cf. [46]). The assumption of constant stride length is also worth discussing as several others (e.g. [11, 25, 26]) claimed that a constant stride length cannot be assumed. Therefore, the use of several more sophisticated methods like linear combinations of a constant value and step frequency [38] could be examined. Also the fusion of the step frequency and accelerometer variations together with a constant [33, 55] could be of interest for our approach. Nonetheless, we are in a situation where the process of walking is the user's intended action so that we can assume that he can focus on his steps and can thereby take care of a constant step length.

PINwI – Pedestrian Indoor Navigation without Infrastructure

Löchtefeld et al. [40] presented an approach for indoor and outdoor navigation solely based on accelerometer and magnetometer readings in combination with an image as underlying map. The idea is based on the PhotoMap system [51] by Schöning et al. which was designed for outdoor usage and therefore based on GPS. The idea is to mark a point on an image of a map spontaneously taken by the user. After several steps, a second point has to be marked and the covered distance is used to calibrate the system by dividing the covered pixels on the map by the number of steps detected. As this only works for maps that are true in scale and the first real-world test showed that also stretched or compressed floor plans exist, a second approach was implemented that allows a calibration in x and y direction. After the calibration phase, the detected steps are used to move the user's position on the taken photo. An initial test showed that the magnetometer of the used Google Nexus One (Android OS) provides only very inaccurate values with deviations of more than 25°. However, a user test with seven participants using an iPhone 3GS showed positive results throughout. In order to be able to get rid of the present drift error due to the Dead-Reckoning approach, the user can reset his position when he can identify his correct position on the map, e.g. through room numbers.

The approach provides valuable insights into the problems that can occur in the context of indoor navigation such as incorrect magnetometer readings. As these problems seem to be device-dependent, it also shows that the general idea of indoor navigation with Dead-Reckoning is feasible. Another relevant aspect which has been shown is that the idea of a constant stride length can lead to good results but it is also evident that a solution for the drift problem has to be found. The proposed solution with manual position resets is not feasible if no underlying map is available.

Indoor Pedestrian Navigation System Using a Modern Smartphone

The Indoor Pedestrian Navigation System [53] presented by Serra et al. is based on a Dead-Reckoning approach using accelerometer and magnetometer. In order to show the position of the user, a map of the building is used. When a user enters a building, he has to scan a 2D barcode. This provides an initial position to the mobile application and also a link to the floor plan for downloading. The values taken from the accelerometer are used to detect the steps a user makes: every pair of negative and positive peaks in the values (zero-crossing) indicates a step. For the user's orientation the smartphone's magnetometer is used. Again, the initial value is set when the barcode is scanned. The movements of the user are visualized on the downloaded map. Experimental tests with a series of 20 runs of 40 steps on average showed an error rate of 3.8% for the step counter. According to the authors, the application based on both magnetometer and accelerometer readings "was able to detect accurately both orientation and displacement of an user in an indoor environment, for short runs (less than 100 m)."

The paper presents very promising results. It shows once more that the basic idea of Dead-Reckoning is feasible for indoor environments. Furthermore, an interesting finding is that the approach produces accurate results for runs of less than 100 meters which is sufficient for the task we want to solve as the typical outline of a room is less than 100 meters. Additionally, it is worth mentioning that no error correction is applied so far and also the use of the magnetometer is an additional source of error as already outlined earlier. With this in mind, even better results can be expected when error correction algorithms are applied or gyroscopes are used instead of magnetometers.

A robust dead-reckoning pedestrian tracking system with low cost sensors

The work by Jin et al. [26] initially describes three major drawbacks of using a standard smartphone as a device for Dead-Reckoning: the low cost sensors are more noisy which leads to faster error accumulation and moreover they offer lower refreshing rates. The third problem is the placement of a smartphone (e.g. in the user's pockets) so that no error correction technique like Zero Velocity Update (ZUPT) [44, 45] is possible. Hence, the authors proposed to use a step-based Dead-Reckoning approach instead of double integration raw sensor readings.

The main idea of the system is based on the fact that if two smartphones are carried by the same walking pedestrian, they have stable relative displacements with respect to the center of the moving user and only limited local random motions. For testing purposes, a Google Nexus One and a HTC Magic were used as both phones offer a triaxial accelerometer and a triaxial digital compass. Because of the arbitrary placement of the devices the sensor readings first need to be projected to a meaningful coordinate system. Then a low-pass filter is used for noise reduction both on the accelerometer and the digital compass. The next part of the algorithm provides the step detection: a step is detected when a valid local maximum and a valid local minimum occur in sequence in the accelerometer readings. An extremum is only considered valid if it occurs at least 150 milliseconds after the last extremum. As the stride length varies from step to step, Jin et al. use an estimation approach presented by Weinberg [59]. Because of the poor performance of the low cost sensors with an updating rate of less than 25 Hz on average, a heading detection scheme had to be used that performs trapezoidal-rule-based numerical integration over the projected acceleration readings. The overall pedestrian tracking task can then be formulated as a Maximum A Posteriori (MAP) sensor fusion problem (see [26] for further details). The overall approach was evaluated within an indoor area of $140m^2$ using two smartphones as independent sensor units. A third smartphone was used as manager device that is connected via Wi-Fi to the two others. All recorded sensor values were sent to the manager device for the position computation. A crucial part of the approach is the synchronization of the three devices for which a separate protocol was developed. Tests with the new sensor fusion approach showed that the error rate could be reduced by up to 73.7%.

The approach by Jin et al. provides a completely different approach to error correction. Opposed to other systems where several different sensors of one device are fused, here several devices are used for fusion. The very promising results show that the approach has potential but a disadvantage is the need for at least two sensing devices which is costly and therefore not widely available. Furthermore, additional overhead is needed to synchronize the different devices that are involved in the measurement process.

2.1.3 Discussion of the presented approaches

The following section provides a comparison of the systems presented above. Several aspects like the sensors used or the different approaches for distance and heading estimation will be discussed. Furthermore, we analyze which of the ideas that are currently only implemented in systems with special hardware requirements could be realizable on a smartphone as well.

Table 2.1 gives an overview on the different systems that rely on particular hardware. The first two systems, the Pedestrian Navigation System [11] and the Personal Odometry System [44–47], show the two main concepts for the estimation of the travelled distance. Whereas the first system tries to detect

System	**Sensors**	**Distance estimation approach**	**Heading estimation approach**
Pedestrian Navigation System [11]	Acc, Mag	Pedometer, neural network for step length estimation	Magnetometer-based, sensor fusion with GPS for error correction (Kalman filter)
Personal Odometry System [44–47]	Acc, Gyro	Double integration of accelerometer readings, Zero Velocity Update (ZUPT) [44, 45] for error correction	
Helmet-mounted PDR system [3]	Acc, Gyro, Mag	Pedometer, neural network for step length estimation (based on [28, 37])	Based on the IMU's yaw readings
Pedestrian Navigation Module [34]	Acc, Baro, Gyro, Mag	Pedometer, several underlying models for step length estimation	Sensor fusion of magnetometer and gyroscope readings (Kalman filter)
PDR system with Phone Location Awareness [54]	Acc, Gyro, Mag	Pedometer, step length estimation based on walking frequency and accelerometer values	(no information is given)

Table 2.1: Comparison of indoor tracking approaches with particular hardware (Acc=accelerometer, Baro=barometer, Gyro=gyroscope, Mag=magnetometer)

individual steps and to calculate the travelled distance by multiplication with the estimated step length (pedometer approach), the latter calculates the distance by double integration of the accelerometer readings. As this approach is likely to be subject to growing errors over time, special correction methods like Zero Velocity Update (ZUPT) [44, 45] have to be applied. As already mentioned, such an error correction is only realizable if the sensors are shoe-mounted. Therefore, it is not surprising that all other approaches rely on a pedometer approach. Several different methods were presented with regard to step length estimation. None of the systems rely on a constant step length. For calculating the actual step length techniques like neural networks or underlying physiological models are used.

For heading estimation also several ideas were presented. The approach of the Pedestrian Navigation System [11] is not applicable indoors as it relies on GPS for error correction. For the Helmet-mounted PDR system [3] and for the PDR system with Phone Location Awareness [54] no practical information is given. Therefore, only the Pedestrian Navigation Module [34] provides an idea that is of interest for us. The combination of gyroscope and magnetometer provides a good possibility to cope with magnetic disturbances which could occur indoors.

Remarkably, nearly all of the presented systems only use sensor types that are also contained in widely available consumer devices. With this in mind, it is reasonable to investigate which of the approaches could also be used with such a device. In contrast to the approaches that rely on particular hardware, Table 2.2 shows the systems that are usable with standard consumer devices. Furthermore, we also include our proposed solution to enable an easy comparison.

System	Sensors	Distance estimation approach	Heading estimation approach
DRec [15]	Acc, Mag	Pedometer, constant step length with offline calibration	Magnetometer-based
PINwI [40]	Acc, Mag	Pedometer, constant step length with online calibration	Magnetometer-based
Indoor Pedestrian Navigation System [53]	Acc, Mag	Pedometer, constant step length	Magnetometer-based
Robust DR Pedestrian Tracking System [26]	Acc, Mag (2 devices)	Pedometer, Weinberg algorithm [59] for step length estimation	Integration of accelerometer readings
Our approach	**Acc, Gyro, Mag**	**Pedometer, constant step length with online calibration**	**Sensor fusion with magnetometer and gyroscope readings**

Table 2.2: Comparison of indoor tracking approaches with consumer devices (Acc=accelerometer, Gyro=gyroscope, Mag=magnetometer)

Similar to most of the systems presented above, the smartphone-based systems rely on a pedometer approach for distance measurement. This is reasonable as other approaches would require special error correction methods to cope with errors increasing over time due to continuous integration. As already mentioned, common error correction methods for this rely on shoe-mounted sensors. Therefore, we will also follow the idea of a pedometer based distance estimation. The step detection will be based on the accelerometer readings.

As described above, none of the systems with particular hardware use a constant step length for the distance estimation. In contrast, most of the smartphone-based systems do. One reason for this could be the computational overhead such a dynamic estimation (e.g. with a neural network) produces. But also among the implementations that use a constant step lengths, different approaches exist: DRec [15] requires the user to enter his step length before using the application whereas the PINwI [40] app detects it during a calibration phase by itself. We will also rely on a constant step length as we can assume that a user can take care of this when mapping intentionally and will provide the user with a possibility to let the system detect the individual step length by itself.

For the heading estimation, most of the presented systems solely rely on the magnetometer readings and none of the already existing ones provide a sensor fusion approach. As this could be problematic indoors, for example due to electronic devices that disturb the magnetic readings, our approach will rely on sensor fusion with readings from the magnetometer as well as from the gyroscope. The gyroscopic values take care of short-term changes whereas the magnetometer readings provide long-term stable values. Such a sensor fusion is clearly only realizable if both sensor types are available. As there are still many smartphones that are not equipped with a gyroscope, we will also implement a fallback mechanism that is able to deal with situations where only readings from the magnetometer are available. Special attention must be paid to the computational restrictions of a smartphone on the one hand, but also on the other hand to the sensors which provide lower update rates and lower accuracy.

Summing up, our approach adopts the sensor fusion concept for heading estimation which is currently only used in systems with particular hardware. We will combine this with a pedometer-based distance estimation that relies on a constant step length as it is mostly done in systems using standard mobile devices.

2.2 Automated measurement with smartphones

The task of automated distance estimation has been examined with different technical concepts like ultra-wide-band radio [20] or laser scanners [35]. As these approaches require special hardware which is costly and also heavy and bulky, they are not feasible for a system which should only consist of a smartphone.

To our knowledge no scientific work discussing an automated room measurement approach with a smartphone has been published so far. Nonetheless, a commercial application called MagicPlan exists which provides a comparable approach. The app is available for iOS and is presented at the end of this chapter. For the Android OS no such app exists, but some others are available that provide at least some basic functionality for distance measurement. Advantages and shortcomings of these apps will be discussed in the next section.

Camera Arc Calculations

The app Camera Arc Calculations[21] offers the possibility to measure the height and width of objects with the help of the camera image. Furthermore, angular measurements can be carried out. A problem is the restricted accuracy of $0.1m$.

[21] `https://market.android.com/details?id=de.javaresearch.android.camCalc`, last accessed March 27, 2012

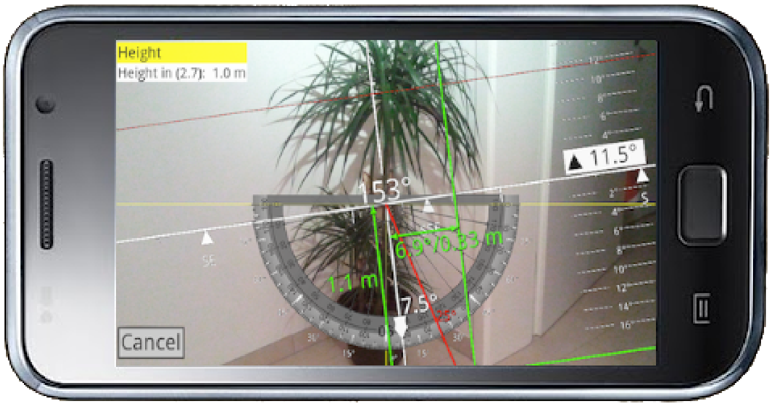


Figure 2.2: Camera Arc Calculations

Besides, the user interface seems overloaded because of several lines and overlapping and rapidly changing figures (seeFigure 2.2). The app only relies on the magnetic field sensor to detect the width of an object which can be considered as a major drawback considering the disturbances of the earth's magnetic field that can be expected indoors. Moreover as this app is designed for measuring only single objects, a room measurement task would suffer severe problems (e.g. no support for the alignment of measured walls to get a complete room). It is, however, possible to determine the room's angles in the corners, which at least gives the possibility to align the walls by hand and derive a floor plan out of the several measurements done with the app.

Smart Measure

With the help of Smart Measure[22] it is possible to measure the distance and height of an object based on trigonometric calculations. In the Pro edition it is also possible to determine the width of an object as well was its area. To be able to do so, it is only necessary to calibrate the app i.e. to define the height the smartphone is held at. The measuring process is simple and intuitive – the user interface provides one button for starting the computation and clearly shows the results (see Figure 2.3).

In the Pro edition the app could basically be used for the task of measuring a room but several drawbacks have to be considered: the app offers only an accuracy of up to $0.1m$ which could lead to errors of more than $0.5m$ per room. As the app is designed to measure single objects, no complete rooms can be covered at once. As a consequence, an angulation of the corners is not possible either.

In the Android market some more apps similar to the two presented here exist. All of them have the same disadvantage – namely the missing ability to capture

[22] `https://market.android.com/details?id=kr.sira.measure`, last accessed March 27, 2012

Figure 2.3: Smart Measure

more than single objects. However, the existence of all these apps and the number of their installations shows that a lot of users are highly interested in the assisted measurement of objects. As the apps could potentially be used to capture rooms, although the drawing of a floor plan has to be done manually, people probably use them to overcome the lack of more sophisticated apps on the Android OS like MagicPlan on iOS which is presented in the next section.

MagicPlan

MagicPlan is an app provided by Sensopia[23] to measure the size of a room, to draw a floor plan and to export it in several formats (DXF, PDF, JPEG as well as an interactive floor plan web site). As the app requires the functionalities of a gyroscope it is currently only supported on iPhone 4/4S, iPod (4th generation) and iPad 2. With the help of the app it is only necessary to take photos of the corners of each room (in a clockwise or alternatively counter-clockwise direction) and MagicPlan measures the dimensions of walls and doors, identifies the shape of the rooms and draws a floor plan for each room. If several rooms are captured, they can be aligned to capture a complete house. The app provides many more functions (e.g. furniture placement) but these are beyond the scope of this description. Figure 2.4 shows the marking of one of the room's corners during the capturing process of a room.

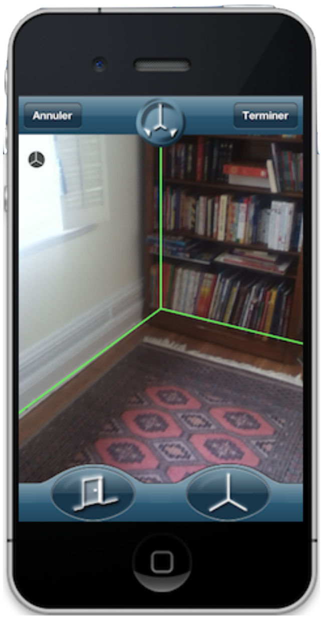


Figure 2.4: MagicPlan

[23] http://www.sensopia.com, last accessed March 27, 2012

Several demo videos on the app's website illustrate the manifold functionalities of MagicPlan. The large number of possibilities promises a quick and detailed capturing process. A lot of comments on the MagicPlan website show the great interest people have in an app that offers the measuring of a room or even a complete building.

2.2.1 Comparison of the presented smartphone apps

To conclude the chapter, we compare the presented smartphone apps with our approach. Table 2.3 shows the main features and illustrates which of them are supported by the respective approaches.

App	**Supported sensors**		**Angular measurements**	**Floor map generation**
	Magnetometer	Gyroscope		
Camera Arc Calculations	✓	×	✓	×
Smart Measure	✓	×	×	×
Magic Plan	×	✓	✓	✓
Our approach	✓	✓	✓	✓

Table 2.3: Comparison of smartphone apps for distance/size measurements

As the table shows, none of the existing apps provides both the possibility to take measurements with the help of a gyroscope as well as a method also working with smartphones that are not equipped with such a sensor. Furthermore, only the Camera Arc Calculations app and the Magic Plan app are able to measure angles and only the latter offers the possibility to draw maps of the captured environment. In contrast, our approach will use the gyroscopic readings if available but also offers a fallback mechanism for smartphones without a gyroscope. The measuring of angles is also supported as it is a prerequisite for floor map generation – a feature that is also available with our approach.

As a last remark it is worth mentioning that to our knowledge for none of the apps, neither the ones for the Android OS nor for MagicPlan, provable investigations on the accuracy or user acceptance are currently available. This makes it hard to compare the quality of the approaches with each other but also with other systems like the ones presented in the preceding section.

Chapter 3
Motivating study

As already outlined in Chapter 1 it is crucial for the success of a VGI project to address a large number of contributors. For this purpose, it is essential to have an understanding of people's interests when designing a VGI project. To raise our awareness of the users' needs, we conducted an introductory questionnaire.

3.1 Overview of the questionnaire

As our application requires Internet access, we opted for an online-only questionnaire. This offers the possibility to gather opinions from a large number of participants of different ages and backgrounds, but also ensures that the participants have at least some basic experience with the Internet or digital media in general. Furthermore, a digital questionnaire offers some features that allow for an increased comfort during participation, e.g. asking questions only after the user has given certain answers to previous questions. This is in principle also possible with a paper-based questionnaire, but then it is up to the user to find subsequent questions. Another solution to circumvent this problem would be to conduct interviews, but then an interviewer bias could potentially influence the results. Nonetheless, also the way the questions are formulated may already have a certain influence on the interviewee [17]. In addition to this, several advantages and shortcomings of online questionnaires are reported by Wright [61]. The automated question selection gave the possibility to ask well-directed questions according to the participant's knowledge of existing digital maps. The longest version of the questionnaire consisted of 21 topic-related questions (17 mandatory) whereas in the shortest version only 12 of these (9 mandatory) were asked. In addition, at the end of all versions, five mandatory questions concerning demographic aspects were asked. The questionnaire is reprinted in Appendix A.

We set up the questionnaire in English and German in order not to collect opinions only from German speaking people and so increased the number of possible participants. We distributed the link to the questionnaire through several different channels e.g. via email, social networks and among friends. In an introductory text, the participants were informed about the objective of the questionnaire and also that all given answers would only be used in anonymized form to ensure privacy. Additional explanatory texts were given prior to some questions e.g. a short explanation of the idea behind the planned project OpenIndoorMap (OIM). To be able to classify the answers given to questions targeting OIM, we also included several questions regarding the already existing OpenStreetMap project (OSM) for reasons of comparison. These questions were only presented to people who knew OSM beforehand. Already during the first weeks of the questionnaire, we noticed that only very few people knew OSM beforehand and therefore these specific questions were seldom asked. For this reason, we decided to publish the questionnaire again but this time we used a channel directly targeting active OSM users. To reach these users we announced the questionnaire via the German OSM mailing list "talk-de". We will now present some relevant results of the questionnaires separately for both groups. At the end of the chapter we will discuss which insights can be gained from the results.

3.2 Findings from common Internet users

We started the questionnaire on October 28, 2011 and in the next 14 weeks, 765 people participated – 509 of whom (381 male, 127 female) completed the questionnaire. The age distribution of the participants is illustrated in Figure 3.1.

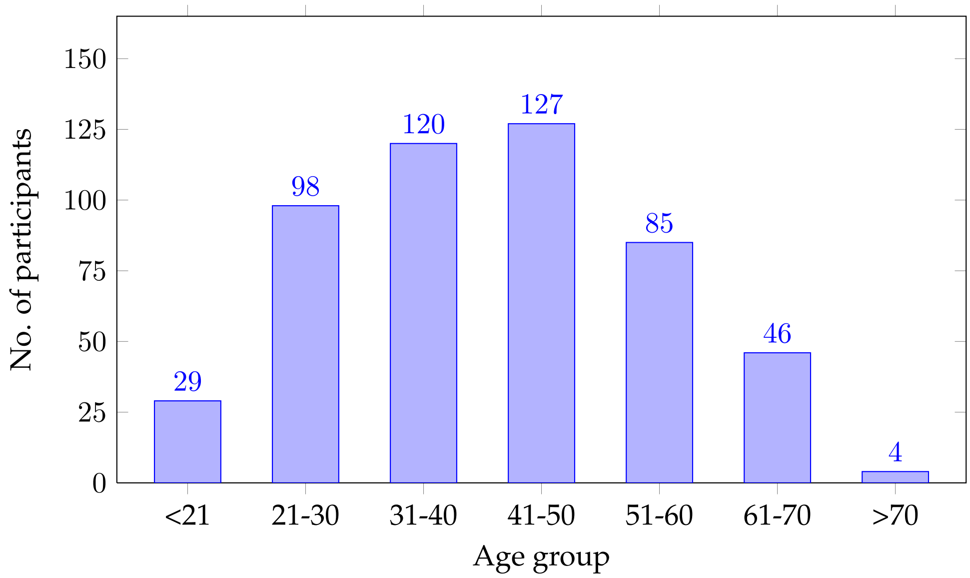


Figure 3.1: Age distribution of the participants

Most participants of the online questionnaire came from Germany, but also people from Australia, Austria, Belgium, Brazil, France, Italy, New Zealand, Switzerland, Thailand and the United Kingdom took part. The average completion time was 4:49 minutes (median 3:05 minutes).

First, people were asked in which manner they know or use some of the existing digital map services i.e. Google Maps, Yahoo Maps, Bing Maps and OpenStreetMap. In Figure 3.2 we illustrate the answers assigned to three categories: people who use a service on a regular basis, have used it at least once and those who have never used it before. Google Maps is the most frequently used service – 88% of the participants have used it before. Although OpenStreet comes next, only 23% already have used it. As mentioned above, this was the reason why we published the questionnaire again on the OSM mailing list. Nonetheless, compared to Yahoo Maps and Bing Maps, OpenStreetMap is used by a larger number of participants, especially if we consider the users who use it regularly.

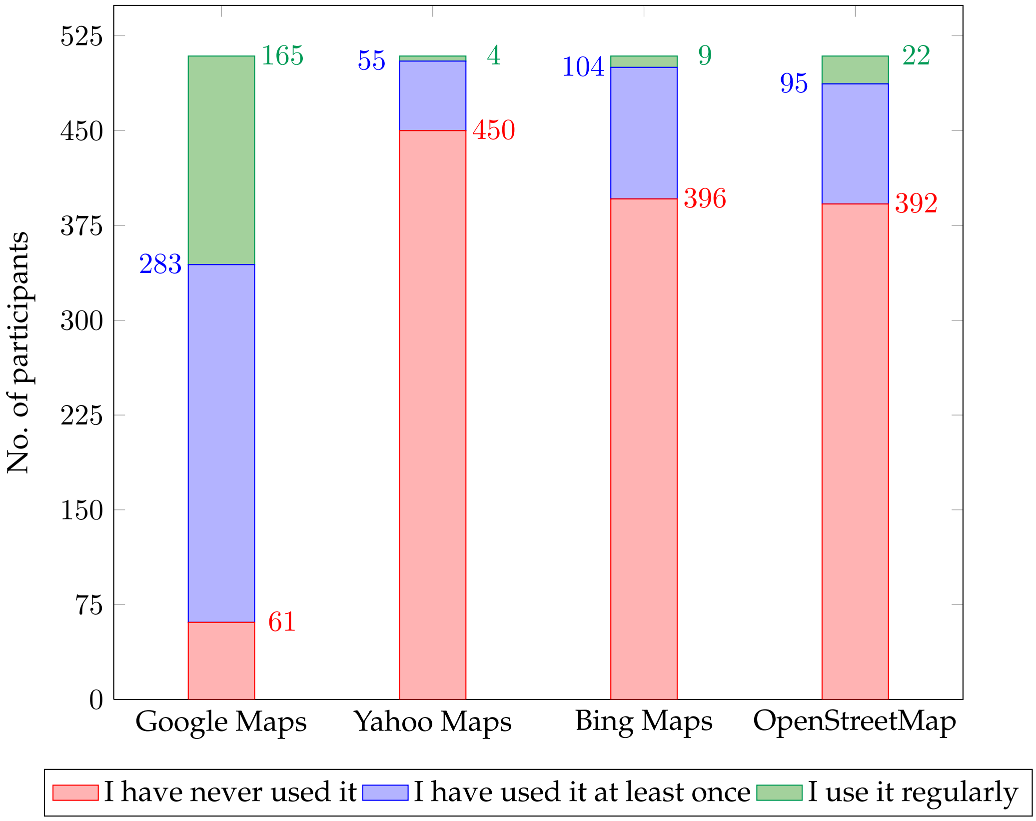


Figure 3.2: Clustered usage of existing digital map services

We also asked some questions to cover the usage of digital map services in general. 28.5% of the participants said that they had already wished to have such a service also for indoor environments. The participants were subsequently asked for which types of buildings they would like to have indoor maps. We decided to use a free text query for this to allow the mention of arbitrary buildings. In total 561 specific buildings or buildings types were mentioned. We analyzed the individual answers and clustered them in 18 categories. During this process specific buildings were assigned to the corresponding category, e.g. Frankfurt airport to airports. Furthermore, also different kinds of the same building type were merged in the same category e.g. city hall and tax office are both considered as administrative buildings. The clustered answers are illustrated in Figure 3.3. Some people took the opportunity of answering to a free text query to justify their answer. A reason often given when mentioning administrative buildings was their uniformity – all floors nearly look the same and therefore it is hard to find the correct destination.

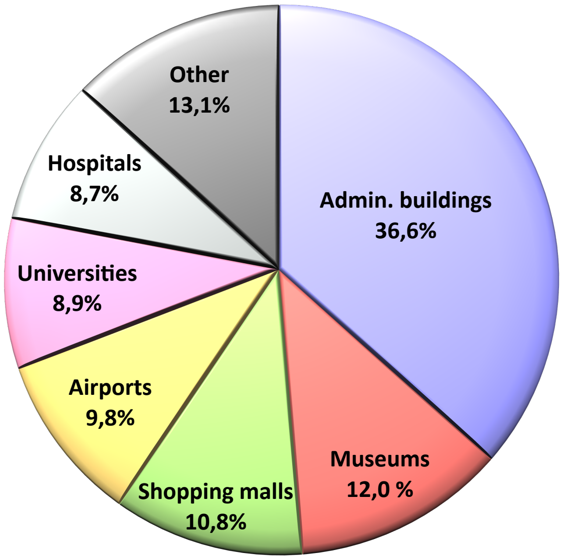
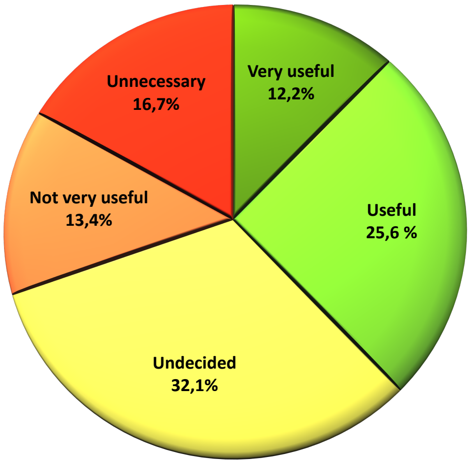


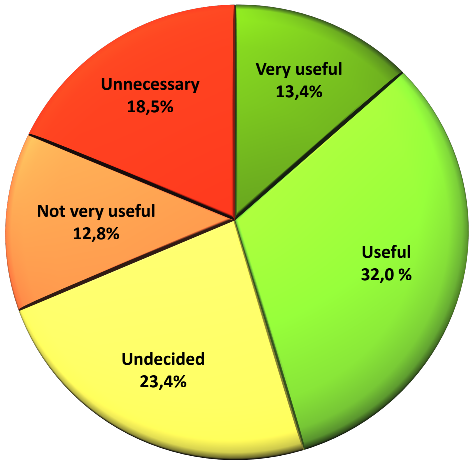
Figure 3.3: Buildings the participants would like to have indoor maps for

We also asked the participants if they would use a digital map service for indoor environments and 316 participants (62%) would do so. To round off the picture, people were also asked to state how useful they found the idea of a 3D map presentation. The same question was asked with respect to indoor navigation. The results are depicted in Figure 3.4. For both questions we can see that more than half of the participants were undecided or even found the feature not very useful or unnecessary. From these results we can draw the conclusion that both features could be valuable innovations, but they are not must-have features that have to be included in the first prototype.

As stated above, the dynamic characteristic of an online questionnaire was used to present questions only when appropriate. Therefore, the questions discussed in the following were only shown to participants who had used OSM at least once. This is reasonable as they cover aspects of the contribution to OSM. The participants were asked whether they had also contributed to OSM in the past



(a) Usefulness of 3D map presentations



(b) Usefulness of indoor navigation

Figure 3.4: Usefulness of additional features for digital map applications

i.e. added information to the database or maintained existing entries. Only 21.5% of the 127 interviewees who were shown this question also had done this. Their main reasons for contribution were a general interest in geographic information, the OSM project in particular, but also the completion of their own surroundings. Other factors like the popularity of the project or the influence of friends also contributing to OSM were chosen less often. Among the participants who are currently not contributing several said that they were not able to add anything new because the already collected information was complete and correct. Therefore, it is not very surprising that more than 20% of the non-contributors said that they would prefer working on a similar VGI project targeting indoor environments. We also asked active OSM contributors if they were also willing to work on a similar proposal for indoor maps. 66.7% answered this question positively. In total, 29.1% of the 127 interviewed participants would like to work on OIM. Compared to the 21.5% of the 127 participants contributing to OSM this is a promising result although only a small subset of all participants were asked these specific questions. Another noteworthy point are the answers people gave to the question why they would not contribute to OIM – one reason, mentioned by 13 participants, was that it they expected indoor mapping to be too complicated e.g. due to missing GPS signals. Again, some more specific questions were only asked when the participant stated that he was a potential OIM contributor. The vast majority (81%) would spend not more than 30 minutes per week for working on indoor maps. From the 22 participants that also own a mobile device, e.g. smartphone or tablet computer, only two said that they prefered working with it when capturing new areas, but another seven would work with a computer as well as with a mobile device and only one participant would only work with a computer. Similar results could be obtained for the question of editing already existing datasets. These findings make it seem reasonable to develop a mobile application, but it should also be kept in mind that a desktop application could be helpful to reach a higher number of contributors. As only 89% agreed to a software installation either on their mobile device or on their computer, also a web application could be considered helpful to reach additional contributors. In the course of this thesis, we will limit ourselves to the development of a mobile application. This enables on the one hand capturing on location, but on the other hand it also allows to support data editing tasks.

Finally, the participants had the possibility to enter their email address if they would like to receive further information on the topic of indoor maps and the idea of digitizing them. The email addresses were stored without any connection to the given answers to ensure privacy. 148 participants took the opportunity which shows an extended interest in the topic in general.

3.3 Findings from active OpenStreetMap users

As described above, we published the questionnaire again on the OSM mailing list because only few participants of the initial questionnaire had any experience with OSM. As we did not change the questions, we will shorten the presentation and especially point out differences between the two groups.

During a period of four weeks, 214 people began the questionnaire. In the following we will only consider the answers of the 149 participants (69.6%) who also completed it. Figure 3.5 shows the age distribution of the 142 men and 7 women who participated. The average completion time was 4:47 minutes (median 3:22 minutes).

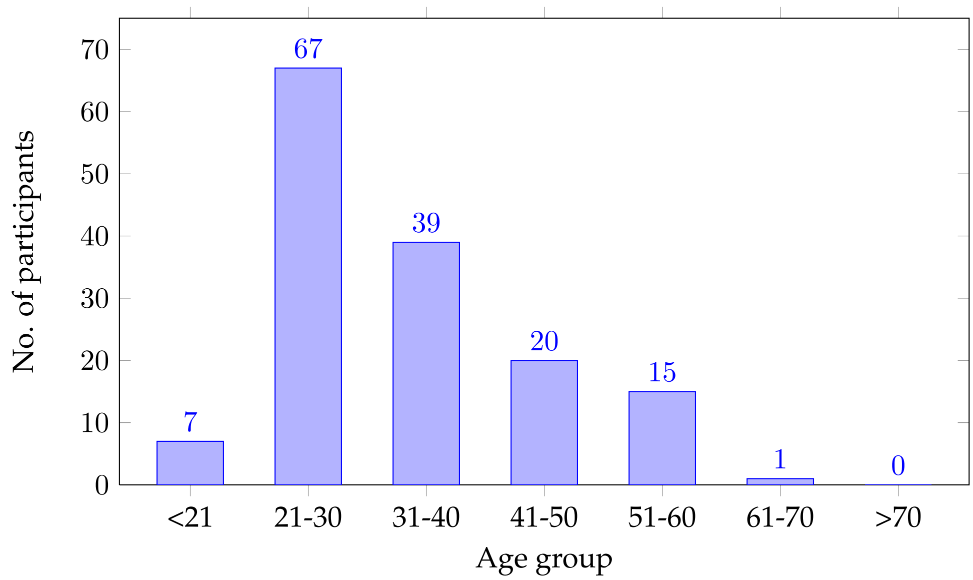


Figure 3.5: Age distribution of the participants

The observed age distribution differs from other surveys conducted in the field of OpenStreetMap (cf. [56, 57]) where about 22% of the participants were in the age group of 41 to 50. A reason for this could be the distribution via the mailing list where only a subset of the active mappers could be reached.

As expected all the participants have already used OpenStreetMap at least from time to time – 138 participants (92.6%) have even used it on a regular basis. Furthermore, 96.6% have already contributed something. The reasons for contribution were similar to the ones also reported above, but also the gathering of experience with a new technology played an important role. The majority (65%) of all participants found the idea of OpenIndoorMap useful or very useful – another 27.1% stated to be undecided. Only 7.9% perceived it as not very useful or even unnecessary. The question whether they would also contribute to OIM was answered positively by 74% of all participants. Again, it is remarkable that the most often named reason for not contributing is the apprehension of problems due to the lack of traditional mapping methods i.e. techniques based on GPS. For the questions regarding the time people would spend working on indoor maps, results similar to the ones reported above could be observed, but several users expressed their willingness to work even longer than two hours per week. A noticeable result is that only 40.2% of the participants own a mobile device (compared to 81.5% of the former participants that were asked this question). Nevertheless, the answers regarding the preferred device for adding or maintaining information were of the same kind and therefore the conclusions drawn above also hold for this user group.

As depicted in Figure 3.6 the buildings for which indoor maps should be available differed in their distribution from the ones mentioned by the participants of the first questionnaire but mostly the same buildings were of interest.

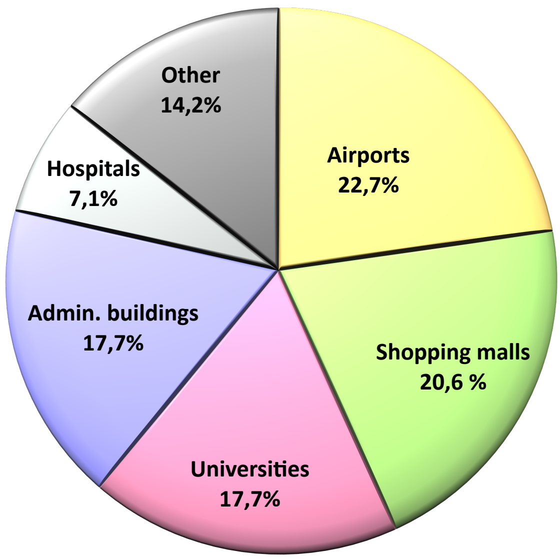
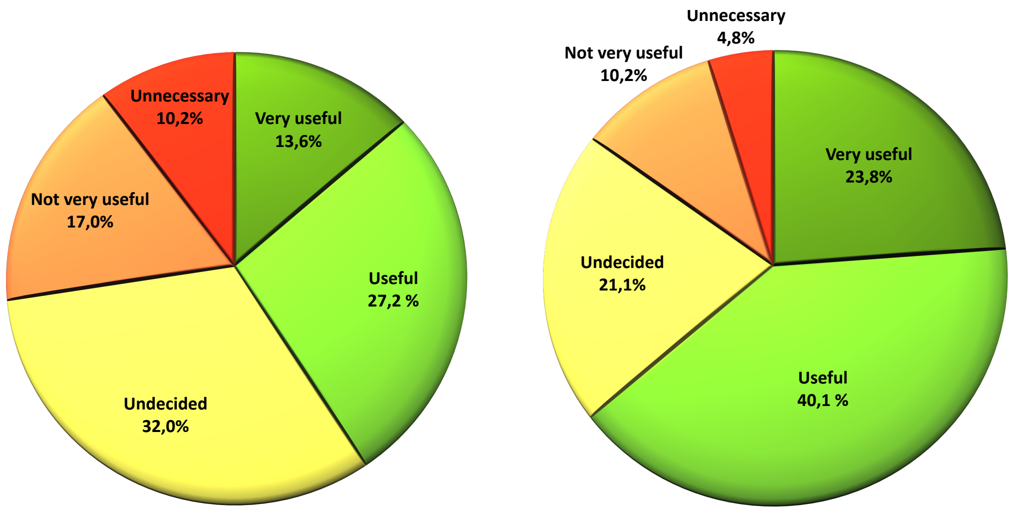


Figure 3.6: Buildings the participants would like to have indoor maps for

The question whether the participants had already wished to have an indoor map service was answered positively by 51.7%. A Chi-square test showed that this result differs significantly from the result gained from the first questionnaire ($\chi^2(1, N = 658) = 62.68, p < 0.01, \omega = 0.3$). A similar finding could be seen for the question whether the participants would use such a service. The result that 90% of the participants of the latter group would do so is significantly different from the result of the first group ($\chi^2(1, N = 658) = 20.3, p < 0.01, \omega = 0.18$).

For the question whether a 3D presentation of a map would be useful the answers were nearly the same compared to the one given by the participants of the first questionnaire (Figure 3.7 a)). For the usefulness of indoor navigation the situation is slightly different. As depicted in Figure 3.7 b), more people found the idea useful or very useful (63.9%) and only a small number of participants (4.8%) stated it as unnecessary.



(a) Usefulness of 3D map presentations (b) Usefulness of indoor navigation

Figure 3.7: Usefulness of additional features for digital map applications

As the participants were all subscribers of the OSM mailing list, we did not ask them for their email addresses as we already had a possibility to inform them about the progress of the project.

3.4 Implications of the gathered results

To conclude this chapter we will discuss some implications we can draw from the results of the two questionnaires we conducted. First of all we could see that people have a general interest in the field of indoor maps. This is beneficial for our project because, as outlined above, a large user base is necessary for a successful VGI project. Moreover, a lot of people showed their willingness to not only use such a project, but also to actively contribute. But we also saw that it could be hard for such a project to compete against the popularity of other services in the same field – especially Google Maps in this case. We also saw that several people worried about how to capture indoor environments as typical mapping methods are not available. Therefore, an appropriate VGI project has to provide solutions for this. Furthermore, we got to know several different types of buildings for which indoor maps would be useful – namely administrative buildings, museums, shopping malls, airports, universities and hospitals. We also asked whether people are interested in additional features such as a 3D map presentation and we saw from the results that it is not considered as a must-have feature by many users. Therefore, we find it reasonable to postpone the implementation of this feature. The situation is slightly different for the indoor navigation feature as more people were interested in this. Nonetheless, it would be out of scope to cover this topic in the course of this thesis. For more information and an already existing implementation for indoor navigation with the help of OSM data, see the Bachelor's thesis of Hubel [24].

Summing up, in the course of this thesis we will provide a mobile application with which it is possible to capture uninstrumented indoor environments without relying on external services e.g. GPS. We will thereby mainly focus on the map generation part itself, but we will also keep in mind that the collected data should be usable for further implementations as for example indoor navigation systems.

Chapter 4
Technical background

In this chapter we will describe the technical background necessary for the use of data obtained from different types of sensors that are nowadays often present in mobile consumer devices e.g. smartphones or tablet computers. We will analyze their functionality and illustrate advantages as well as some shortcomings of the respective types of sensors. Furthermore, we will introduce the concept of sensor fusion as a mechanism to overcome disadvantages of a certain type of sensor by combining it with a second type of sensor that does not have this specific issue.

4.1 Sensors in mobile devices

The increasing integration of sensors into off-the-shelf consumer devices such as smartphones [5] offers various new possibilities as a lot of people now use devices with sensing potential in their normal course. Typical hardware sensors that can be found are accelerometers, magnetometers and cameras. Newer and more expensive devices (e.g. Samsung Galaxy S II or Samsung Galaxy Nexus) also offer a gyroscope. Furthermore, the operating system can provide so-called virtual sensors that supply data derived from a combination of physical sensors. For example the Android OS offers a rotation vector sensor that provides information on the smartphone's orientation as a combination of an angle and an axis. The readings of this sensor are computed from accelerometer and magnetometer data, sometimes also data from the gyroscope is taken into account, but this depends on the specific vendor and model – the Samsung Galaxy S II uses for example only accelerometer and magnetometer even though a gyroscope is available.

In the next sections we will describe the different types of sensors in more detail and show results from accuracy tests of the specific sensors.

4.1.1 Accelerometer

An accelerometer is a sensor to measure proper acceleration. Schematically built up of a mass on springs, the accelerometer indicates the weight of the mass in the reference frame. A consequence of this construction is that the accelerometer shows a magnitude of about $9.81\ m/s^2$ when the device is lying on a table and clearly not accelerating. In contrast, the sensor does not show a magnitude, when the device is in gravitational free fall towards the Earth's center because there is no relative movement between the mass and the reference frame. In particular, the measured values are different from coordinate acceleration i.e. the change of the device's velocity in three-dimensional space. To be able to measure the real acceleration of the device, the force of gravity has to be eliminated. This is possible by applying a high-pass filter that lets high-frequency signals pass, but attenuates signals with frequencies lower than a cutoff frequency. On the contrary, a low-pass filter that attenuates high-frequency signals can be used to isolate the force of gravity.

A low-pass filter can also be applied to overcome noisy sensor readings and thereby get a smoothed signal. A drawback of this filtering approach is a delay in the sensor readings as depicted in Figure 4.1. The smartphone's rotation by about 80° in the low-pass filtered data (red) is only reached three seconds later than in the raw values (black).

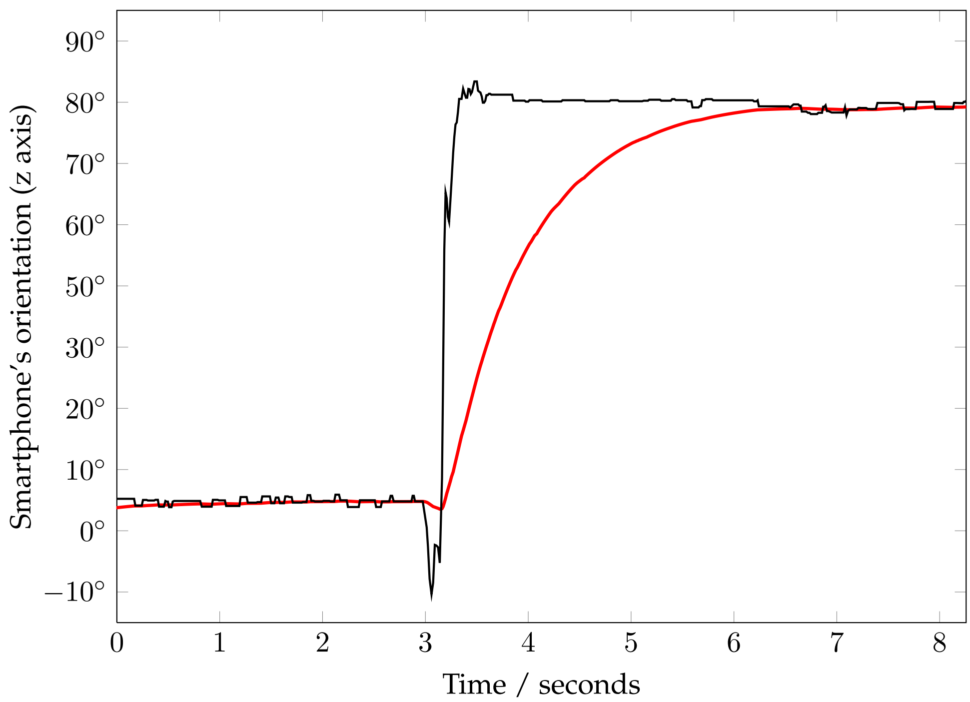


Figure 4.1: Raw accelerometer data (black) and low-pass filtered values (red)

4.1.2 Magnetometer

A magnetometer is a sensor to measure the strength and the direction of a magnetic field. A possible application is the determination of a device's orientation by measuring the Earth's magnetic field. Another application presented by Ketabdar et al. [29] was a detection mechanism for gestures made in the three-dimensional space around the device. A major disadvantage of using the information provided by a magnetometer is the sensor's lack of responsiveness regarding quick movements as well as its vulnerability to interference from external magnetic fields (cf. [42, 58]). In indoor environments especially the latter drawback could be problematic: walls can contain large metal packages and additionally, electronic devices such as computers or copiers are often present inside buildings, particularly in offices. Both conditions can be responsible for disturbances in the magnetic field that can be measured by a magnetometer.

Figure 4.2 shows the sensor readings from the magnetic sensor integrated in a Samsung Galaxy S for a period of 10 seconds while the device was lying on a table inside an office. As can be seen, the values are unstable although the device was not moved. These fluctuations lead to deviations of the correct device heading of up to 10°.

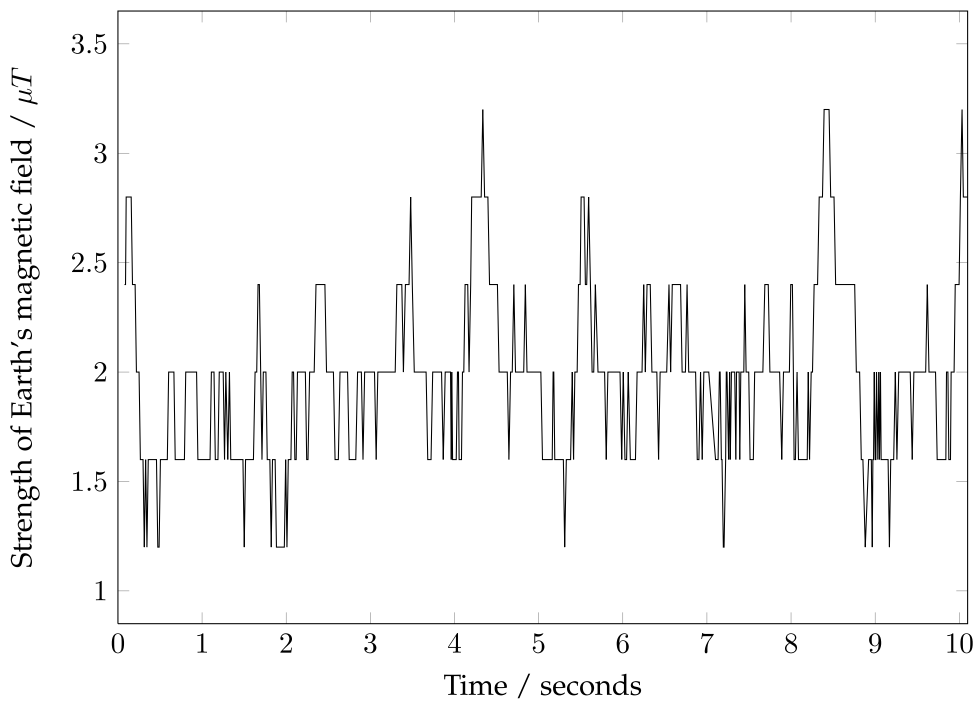
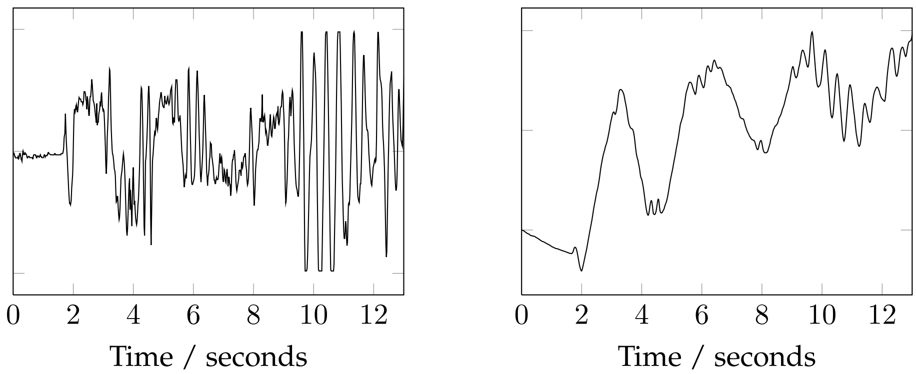


Figure 4.2: Sensor data from the magnetometer of a Samsung Galaxy S

4.1.3 Gyroscope

A gyroscope is a sensor to measure or maintain orientation based on the principle of angular momentum. Different types of gyroscopes such as mechanical or fibre optic exist as well as microchip-packaged **M**icro **E**lectro-**M**echanical **S**ystems (MEMS) devices that are present in consumer devices like smartphones. These chips make use of vibrating elements and the idea of the Foucault pendulum. Typical applications of gyroscopes include inertial navigation systems where digital compasses would not work or would not be accurate enough. When using a gyroscope a problem could be its so-called bias error that increases over time. Several factors such as changes in temperature or vibrations influence this bias error. Due to this, consumer-grade gyroscopes drift by up to 1° per second [16].

Apart from this, an additional kind of drift occurs due to the way the sensor readings are used. As the gyroscope provides measurements of angular rotation, the values read are mostly integrated over time to calculate a rotation describing the orientation change during the considered time span. A side-effect of this process is a conversion from sensor noise to drift as depicted in Figure 4.3. The left side shows some intentionally created, noisy sensor data and on the right, the integrated values over time are depicted. The signal is much smoother, but also has an observable drift.



(a) Raw sensor data (b) Integrated sensor data

Figure 4.3: Correlation between noisy sensor data and drift due to integration

4.1.4 Camera

Another sensor often found in mobile devices is a camera that enables the capture of images and videos. With the help of the gathered information a technique called ego-motion or visual odometry [48] becomes possible. With this approach it is possible to track an object's movements by analyzing image data it acquires during motion. A typical field of application is the position estimation for autonomous robots, e.g. the NASA Mars Exploration Rovers used such an approach [41]. The functionality of the technique can be described as follows: in a first step, features have to extracted from the captured images. Often Harris corners [23] are used for this purpose (e.g. in [43]), but as criticized by Konolige et al. [31] they are not always well-suited depending on the type of the underlying scene e.g for outdoors. Therefore, Agrawal et al. proposed a multiscale feature called CenSurE [1] for this purpose. In the next step the extracted features have to be matched across consecutive images or video frames and, based on this information, an optical flow field has to be constructed. The optical flow field should be analyzed afterwards for possible tracking errors and in order to remove outliers [30]. After these preparatory steps the estimation of the camera motion from the optical flow can take place (cf. [7] for more details). As shown by Campbell et al. [6], the visual odometry approach can yield good results. They tested their implementation with a camera mounted on an autonomous robot and reached an error rate of about 3% indoors on carpet, about 5% in an outdoor environment on grass and ice and about 6% on asphalt.

Nonetheless, the implementation of such a visual odometry approach would be beyond the scope of this thesis. We will therefore consider techniques that rely on accelerometer, magnetometer and gyroscope.

4.2 Sensor fusion

As we have seen in the preceding sections, the different types of sensors also have disadvantages which make it difficult to rely on only one kind of sensor. Hence, the idea arises to combine different sensors to compensate the disadvantages of one particular type of sensor with the advantages of the others. This process of combining sensor data from different data sources so that the resulting combination is somehow better than the results the individual sensors could provide is known as sensor fusion. Basically, two different possibilities to fuse several sensor values exist. The first alternative handles the combination directly in the hardware and provides only one new type of sensor to the overlying software, in this way completely hiding the process of sensor fusion. This approach offers two big advantages: first of all, the sensor fusion can be done very fast as it happens directly on a hardware chip especially designed for this purpose. The second advantage addresses the later user of the sensor: as the implementation details are hidden in the hardware chip, no one has to take care of the individual combination but can use the sensor data "out-of-the-box". But in contrast, this could also be a seen as a drawback if some different type of combination would be better for a specific situation. Another drawback of this approach is the fact that such specialized sensors are much more expensive and therefore, at least nowadays, not widely available.

Instead of the combination on the hardware layer, the sensor fusion could also be done in the software. In this case every sensor is individually accessible and it is up to the overlying software to combine the single sensor values. This is costeffective as widely available, cheap single-purpose sensors such as accelerometers or gyroscopes can still be used. Therefore, this approach is also applicable with standard consumer hardware. As the individual sensor readings are now available on the software layer, the operating system is able to do sensor fusion and provide so-called virtual sensors to the programmer via its API. The Android OS for example provides a rotation vector sensor to get information on the current orientation of the smartphone. The data this sensor provides is mostly a combination of readings obtained from the accelerometer, the magnetometer and in rare cases, also gyroscopic data is considered if available. This kind of sensor fusion is very comfortable for the programmer as the fusion process itself is completely hidden inside the operating system – from the programmer's perspective, there is no difference between a standard sensor that is available as hardware component and a virtual sensor provided by the operating system. But there is also a downside of this comfort: the programmer has no influence on the actual way the sensor data is fused. For example, on most devices the rotation vector is computed by only considering accelerometer and magnetometer readings – even if a gyroscope is available. Especially for our purposes of an indoor usage it would be beneficial to additionally consider the gyroscopic data to cope with magnetic disturbances. Therefore, it is up to the programmer to do the sensor fusion manually to be able to control the different data sources and the way in which they are combined.

A sensor fusion algorithm that is often used is the Kalman filter. The main idea is to estimate the difference between the current estimated sensor output and a second reference value. The computed error value can then be taken into account to correct the estimation. Thereby, different parameters of the Kalman filter can be used to adjust the influence of the correction part. For more details on the Kalman filter see [27, 60]. When applying such a sensor fusion algorithm on a mobile device, it is important to keep in mind that there are certain restrictions on computational power. Furthermore, attention should be paid to the fact that expensive computations can reduce battery life. Therefore, we will not investigate the Kalman filter any further because it requires extensive computations. Instead, we will consider a more simplified algorithm, however, still providing a reliable way for fusing different types of sensors. The main idea of the so-called complementary filter is to combine information obtained from a gyroscope together with rotation data obtained from a magnetometer and an accelerometer. The high-pass filtered data from the fast responding sensor i.e. the gyroscope handles short-term changes and the low-pass filtered rotation data from the other sensors provides long-term stable readings and eliminates the gyroscopic drift.

Chapter 5
Implementation

This chapter describes the implementation of two different approaches for capturing indoor rooms without the need to instrument the environment beforehand. Both approaches make use of a standard Android smartphone equipped with sensor technology presented in the previous chapter.

For both approaches it is crucial to know the smartphone's orientation. Therefore, we will first describe how the smartphone's sensors can be used to calculate the current orientation and will afterwards present in which way the obtained information is used for the two capturing algorithms. In Section 5.2.1 we will present a pedometer-based approach for room capturing followed by a trigonometry-based approach in Section 5.2.2. We conclude the chapter with some initial tests to access the quality of the proposed algorithms.

5.1 Orientation calculation

For the detection of the smartphone's orientation, we will consider two different approaches to be able to provide a solution that can be used on a large number of mobile devices, e.g. smartphones. The first approach targets those users whose mobile device is not equipped with a gyroscope and therefore relies only on the readings from accelerometer and magnetometer. As presented in Chapter 4 the magnetometer is susceptible to environmental impacts and often provides inaccurate data. To cope with this, the second implementation makes use of a gyroscope as an additional sensor to provide more reliable data. As at present only a few publicly available mobile devices provide a gyroscope, we offer this implementation only as a second alternative. In the next two sections we will provide details of the implementations of both approaches in Android.

5.1.1 Two-sensor approach

The implementation of our orientation calculation approach relying on the readings of accelerometer and magnetometer uses several methods the Android API provides. As presented in Listing 5.1, first, the values of the accelerometer and the magnetometer are read and stored in their respective variables (lines 2 and 6). If for both sensors a value is available, the values are used to calculate a 3x3 rotation matrix (see Equation 5.1) for the device's orientation with the help of the API provided function SensorManager.getRotationMatrix() (line 14). Afterwards the equations in (5.2) provided by Craig [14] enable the calculation of the three individual Euler angles that led to the device's current orientation (lines 16-20). The special cases when $\beta = \pm\pi/2$ need not be considered as no valid measurements would result from such a device orientation.

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \tag{5.1}$$

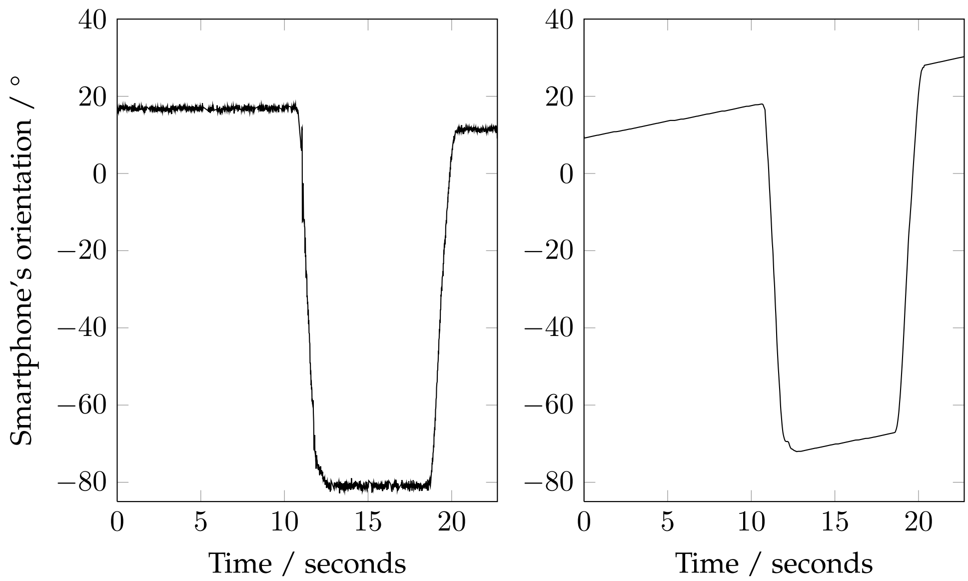
$$\begin{aligned} \beta &= \mathrm{Atan2}\left(-r_{31}, \sqrt{r_{11}^2 + r_{21}^2}\right) \\ \alpha &= \mathrm{Atan2}\left(\frac{r_{21}}{\cos(\beta)}, \frac{r_{11}}{\cos(\beta)}\right) \\ \gamma &= \mathrm{Atan2}\left(\frac{r_{32}}{\cos(\beta)}, \frac{r_{33}}{\cos(\beta)}\right) \end{aligned} \tag{5.2}$$

```
1  if (event.sensor.getType() == Sensor.TYPE_ACCELEROMETER) {
2    gravity = event.values.clone();
3  }
4
5  if (event.sensor.getType() == Sensor.TYPE_MAGNETIC_FIELD) {
6    geomagnetic = event.values.clone();
7  }
8
9  if (gravity != null && geomagnetic != null) {
10
11   float R[] = new float[9];
12   float I[] = new float[9];
13
14   SensorManager.getRotationMatrix(R, I, gravity, geomagnetic);
15
16   pitch = Math.atan2(-R[6], Math.sqrt(R[0] * R[0] + R[3] * R[3]));
17
18   yaw = Math.atan2(R[3] / Math.cos(pitch), R[0] / Math.cos(pitch));
19
20   roll = Math.atan2(R[7] / Math.cosh(pitch), R[8] / Math.cos(pitch));
21
22 }
```

Listing 5.1: Angle calculations based on accelerometer and magnetic field sensor

5.1.2 Three-sensor approach

As stated in Chapter 4, a problem when using the magnetometer indoors are the unstable sensor readings (see Figure 5.1(a)), e.g. due to electronic devices. Therefore, the usage of the gyroscope seems reasonable as it does not encounter this problem. But as described in Chapter 4, the gyroscope also has a disadvantage – namely the drift the sensor values accumulate over time (see Figure 5.1(b)). This means if we consider the readings of the gyroscope for a longer period of time, they will become incorrect. Therefore, we cannot solely rely on them.



(a) Data from accelerometer / magnetometer (b) Data from gyroscope (integrated)

Figure 5.1: Unreliable values from magnetometer, accelerometer and gyroscope

A possible solution is the use of sensor fusion with the help of a complementary filter to combine them with readings from the magnetometer and the accelerometer (cf. Section 4.2). In this way, the readings from the magnetometer and accelerometer ensure long-term stable values without drift whereas the readings from the gyroscope allow for movements in the short-term and eliminate the noise the other sensor readings would show. The relevant source code for determining the angle change by integration of the gyroscopic readings as well as its combination with the orientation values computed with the data from accelerometer and magnetometer (cf. Listing 5.1) is shown in Listing 5.2. First, the time period between the last sensor event and the current one is determined (line 3) and then used for integrating the measured values from the gyroscope to obtain the angular change made since the last measurement (lines 4-6). The sensor fusion (lines 13-15) applies a low-pass filter to the data from accelerometer and magnetometer by slowly adding these values and combines them with the gyroscopic values computed above.

```
if (event.sensor.getType() == Sensor.TYPE_GYROSCOPE) {
    if (event.timestamp != 0) {
        float dT = (event.timestamp - lastTimestamp) / 1000000000.f;
        gyroAngle[0] = event.values[0] * dT;
        gyroAngle[1] = event.values[1] * dT;
        gyroAngle[2] = event.values[2] * dT;
    }
    lastTimestamp = event.timestamp;
}

[...]

filtered[0] = 0.98 * (filtered[0] + gyroAngle[0]) + 0.02 * pitch;
filtered[1] = 0.98 * (filtered[1] + gyroAngle[1]) + 0.02 * yaw;
filtered[2] = 0.98 * (filtered[2] + gyroAngle[2]) + 0.02 * roll;
```

Listing 5.2: Sensor fusion with the help of a complementary filter

In contrast to the graphs presented above, Figure 5.2 shows sensor readings without noise and gyroscopic drift due to sensor fusion with the help of a complementary filter.

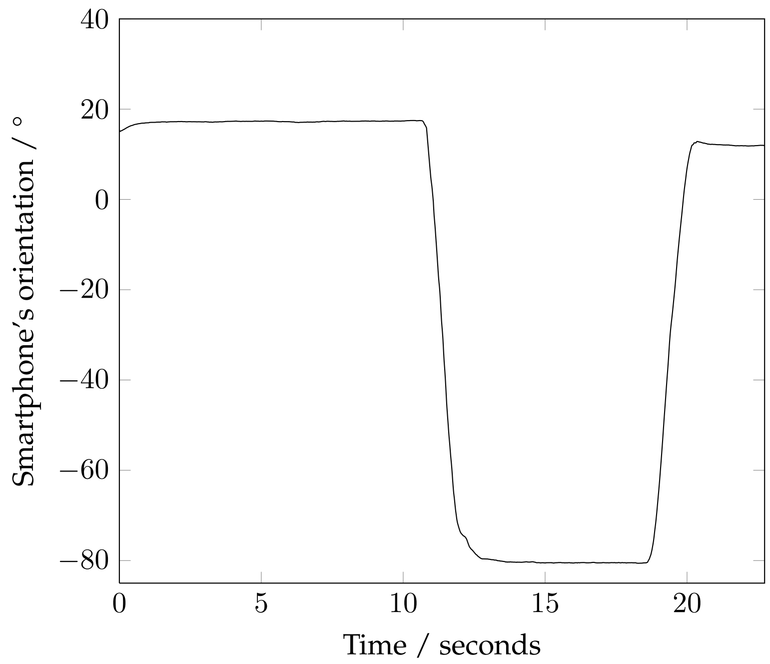


Figure 5.2: Reliable sensor data due to sensor fusion

Both orientation calculation algorithms are encapsulated in separate classes implementing Android's `SensorEventListener` interface and each providing the calculated values to the rest of our application via a self-defined interface (`IOrientationListener`).

5.2 Walls' dimensions calculation

To be able to capture a large variety of rooms we offer two different implementations of our walls' dimensions calculation. The first approach is especially designed for larger rooms offering the possibility to walk along their walls. The idea is to allow a user to move along the room's outer walls while his mobile device captures his movements. From this information the shape of the room can be retrieved. To keep track of the user's movements in an uninstrumented environment, a pedometer-based approach is employed. The details on this implementation will be given in the next section.

For smaller rooms such as offices in which a lot of furniture is standing on the walls the approach presented above is not applicable. Therefore we also developed a second approach that especially focus on smaller rooms that can not be captured by walking along the walls. The main idea of this approach is to determine a room's dimensions by the help of trigonometric calculations based on the orientation data obtained from the user's smartphone. Details on this approach will be given in Section 5.2.2, but first we will focus on the calculations for larger rooms.

5.2.1 Pedometer-based approach

The idea behind this approach is to detect the individual steps a user makes and if in addition also the step lengths and the user's orientation are known, his movements can be reconstructed. As outlined in Chapter 2 we will assume a fixed step length for our calculations as we can think that a user takes care of his step length if he is intentionally walking along the walls of a room. As we already presented two algorithms for orientation calculation, we will focus on the step detection algorithm in this section.

For the step detection, information from the smartphone's accelerometer can be used as it shows clear peaks when the user makes a step and holds the device in front of him. Figure 5.3 shows the readings obtained during a 14 seconds test run consisting of 20 steps with an average velocity of 90 steps per minute (21 steps were detected, the incorrectly detected step is marked with a red square). To detect the individual steps, a thresholding is applied first in order to remove noisy readings. Afterwards each pair of a high peak (acceleration $> 11\ m/s^2$) and a following low peak (acceleration $< 8\ m/s^2$) is considered as a step (illustrated with a square in the graph).

The implementation of the step detection algorithm is encapsulated in a class following the observer pattern [18] to inform registered observers when a step is detected. Whenever a step is detected, the observers can obtain the user's current orientation from the orientation calculation algorithm and subsequently the user's movements along the walls of a building can be tracked and thereby a floor plan can be created.

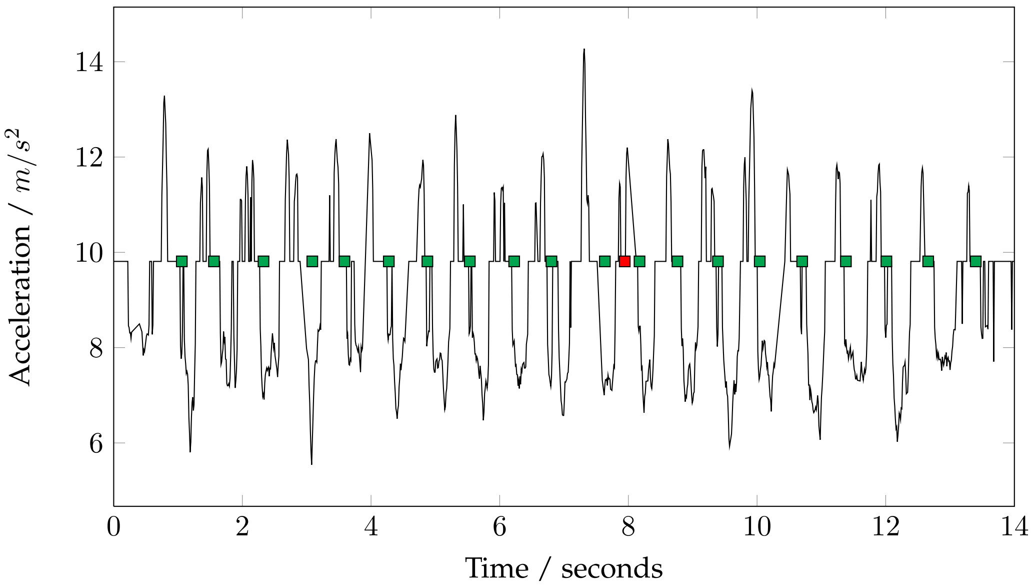


Figure 5.3: Accelerometer readings with illustration of the detected steps

To test the functionality and the accuracy of the step-detection we conducted a small test consisting of three straight walks with different number of steps and compared the number of recorded steps with the number of steps really made. Each test was conducted three times and the results are shown in Table 5.1. As it could be seen, the step detection provides fairly accurate results for all three test cases.

	15 steps	30 steps	50 steps
Run 1	15	28	53
Run 2	13	31	48
Run 3	15	29	49

Table 5.1: Test runs to measure accuracy of the step detection algorithm

5.2.2 Trigonometry-based approach

The second room capture approach targets situations where the application of the first approach is not possible. Typical situations for this could be rooms where obstacles stand on the walls. The main idea behind the second concept is to use the smartphone's orientation to calculate a room's dimensions with the help of trigonometric functions. We will therefore use the algorithms presented at the beginning of this chapter to calculate the smartphone's orientation. In the next section, we will present the mathematical background behind the idea.

Mathematical background

The task of retrieving the dimensions of a room can be divided into several subtasks – to be specific, measuring the walls of the room individually. For the following, a wall is defined as the distance between two corner points. A corner point is defined as any point of interest for the capturing. Common corner points are the corners of a room as well as the left and right border points of a door. Figure 5.4 shows a simple-shaped room and the different points of interest (marked with A - F) as well as the resulting walls (marked with 1 - 6).

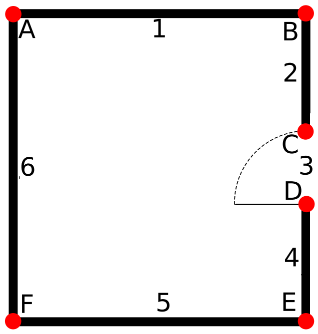


Figure 5.4: Corner points and walls

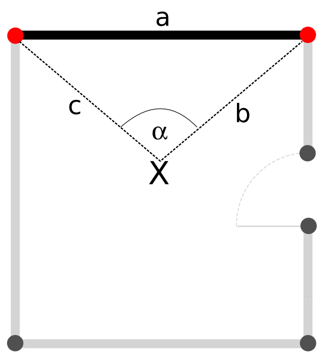


Figure 5.5: Annotated top view

The first subtask of the capturing process would be to capture a single wall respectively its size with the help of its two corner points, e.g. wall 1 with its corner points A and B. This goal can be achieved with the information illustrated in Figure 5.5. Given the angle α and the length of the segments b and c respectively, it is possible to calculate the length of segment a which is the length of wall 1. The necessary formula for this calculations is the law of cosine, solved for a (Equation 5.3).

$$a = \sqrt{b^2 + c^2 - 2bc \cos \alpha} \tag{5.3}$$

This changes the initial problem to the determination of the length of the segments b and c as well as the size of the angle α. As depicted in Figure 5.6, we can calculate w.l.o.g. c with the help of Equation 5.4.

$$c = h / \tan(90° - \gamma) \tag{5.4}$$

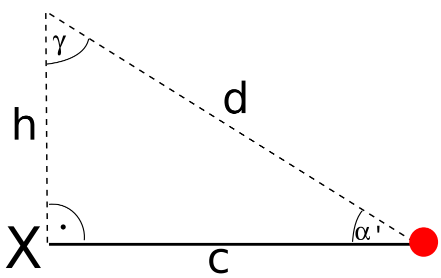


Figure 5.6: Annotated side view

To be able to apply Equation 5.4 we need two pieces of information: the length of segment h as well as the angle γ. If we consider the user is standing at point X, holding his smartphone at height h and tilting it at angle γ, we have all necessary information by hand. As we can determine the angle γ with the help of our orientation calculation, the only information the user has to provide is the height, he is holding his device at. With the same considerations we can calculate the length of segment b if the user targets his phone towards corner point B. To be able to calculate the length of segment a (wall 1), we now only need to know the angle α which we can get from the orientation calculation as it tracks the smartphone's orientation change when the user turns from point A to point B.

All in all, the calculations for a single wall can be divided into three triangle calculations for which we only have to know the height at which the user is holding the device as we can obtain all other necessary information from our orientation calculation.

5.3 Initial tests and further improvements

As both concepts for wall length calculations rely on information about the smartphone's orientation provided by the two algorithms presented at the beginning of this chapter, we will now provide some insights into the accuracy of these algorithms. Therefore, we have carried out some basic tests with the trigonometry-based approach for walls' dimensions calculation as the accuracy of the results is directly related to the accuracy of the underlying orientation calculation algorithm. We will therefore examine several tests with and without the usage of a gyroscope to be able to best both algorithms.

For the first assignment a single wall of 2 meters length was to be measured from a distance of around 2.7 meters. The angle at the user's measuring point between the two corner points was about 40° and the respective lengths to the corner points were 2.8 meters (distance 1) and 3 meters (distance 2). To test the implementation without gyroscope we used a Samsung Galaxy S and a HTC Rhyme and conducted the test three times with each device. Table 5.2 shows the results for the test runs.

Samsung Galaxy S	Test run 1	Test run 2	Test run 3
Distance 1 (2.8m)	2.64m	3.07m	2.94m
Distance 2 (3.0m)	2.81m	3.21m	2.89m
Angle (40°)	29°	32°	31°
Calculated distance	**1.37m**	**1.74m**	**1.56m**

HTC Rhyme	Test run 1	Test run 2	Test run 3
Distance 1 (2.8m)	2.77m	2.73m	2.6m
Distance 2 (3.0m)	3.02m	2.87m	2.72m
Angle (40°)	29°	21°	32°
Calculated distance	**1.47m**	**1.03m**	**1.47m**

Table 5.2: Measured values with uncalibrated smartphones

As already stated in Section 4.1.2 the values from the magnetometer are very unreliable which explains the inaccurate measurements of the angle between the two corner points. Remarkably, the magnetometer error in vertical direction was much smaller for nearly all measurements which leads to fairly good estimations of distance 1 and distance 2. However, the calculated distance of the wall segment is very inaccurate due to large errors in the measurement of the horizontal orientation change. Therefore, we explored how we could improve the values from the magnetometer and we found out that the Samsung Galaxy S provides the possibility to do so with the help of a special sensor calibration tool. Hence, we conducted the same test again with the Samsung Galaxy S, but calibrated the magnetometer before each test run. The results are presented in Table 5.3.

Samsung Galaxy S	Test run 1	Test run 2	Test run 3
Distance 1 (2.8m)	2.8m	3.02m	2.77m
Distance 2 (3.0m)	2.98m	2.79m	3.01m
Angle (40°)	35°	37°	33°
Calculated distance	**1.74m**	**1.86m**	**1.66m**

Table 5.3: Measured values with calibrated smartphone

As can be seen from the measured angles as well as from the calculated distances, the results were much more accurate which clearly shows that the inaccuracies of the first test runs were directly related to the improperly calibrated magnetometer. To test the implementation of the second approach which uses sensor fusion with gyroscopic data, we repeated the same test with a Samsung Galaxy Nexus which also offers a gyroscope. The results are reported in Table 5.4. As expected, the results of the measurements with the help of gyroscopic data were much more accurate than the ones we could achieve before.

Samsung Galaxy Nexus	Test run 1	Test run 2	Test run 3
Distance 1 (2.8m)	2.83m	2.76m	2.79m
Distance 2 (3.0m)	2.97m	3.01m	3.04m
Angle (40°)	37°	37°	38°
Calculated distance	**1.88m**	**1.85m**	**1.91m**

Table 5.4: Measured values with a gyroscope-enabled smartphone

To sum up we could see that inaccuracies in the readings of the magnetometer occur and lead to inaccurate estimations of the wall length we want to measure. Nonetheless, we could also notice that a prior calibration of the magnetometer enables much better values that are at least in some cases of the same quality as those we could reach with our sensor fusion approach. Finally, our tests also showed that the implemented sensor fusion approach works very well and provides accurate data. Based on this data, the calculations of the wall size have an error rate of less than 1%.

In addition to the technical results we could gather with these initial tests, two conceptional problems became obvious. An often occurring problem was the measuring of corner points that were covered, e.g. by furniture, which makes accurate targeting of them hard. Therefore, we implemented a solution that offers the possibility to measure the corner points also on the ceiling level as the points of interest are seldom covered there. Figure 5.7 depicts the idea:

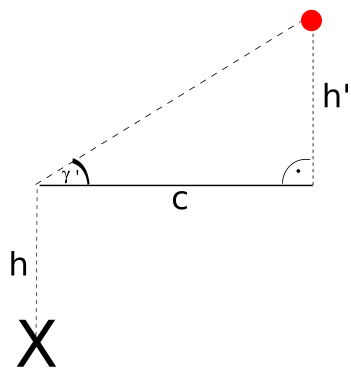


Figure 5.7: Calculations on ceiling level

If the user is standing at position X and holds his device at height h, height h' is the difference to the room's height. The smartphone's tilt information $\gamma\,'$ can be obtained from our orientation calculation algorithms. We are basically interested in h' to determine c with the formula given in Equation 5.5. The determination of h' must only be done once per room (or even once per floor or building if the room height does not change). The necessary calculations can be done via a two-step measurement where the same point on a wall has to be measured at floor level and at ceiling level. This is for example possible at a room's door as there typically both measurement points are uncovered. The idea is to use the normal measurement method presented in Figure 5.6 to determine the user's distance c to the measurement point and then use this distance in a rearranged formula (Equation 5.6) of the ceiling measurement process to determine h'.

$$c = h'/\tan\gamma\,' \tag{5.5}$$
$$h' = c \cdot \tan\gamma\,' \tag{5.6}$$

As a benefit of this approach, it is also possible to derive a room's height by just two measurements.

A second problem was the way measurements were initiated. To enable a user to align his smartphone accordingly a cross line is overlaid on the camera image and to initiate a measurement, it is necessary to activate one of the buttons in the user interface (see Figure 5.8). Touching the device sometimes led to incorrect measurements as the smartphone moves when a button is touched. Therefore, a solution had to be found that enables the initiation of a measurement without touching the device.

We decided to use a speech recognition approach as it offers a natural way of interaction between the user and the application. The Android OS offers a mechanism for speech recognition through an activity that can be started, handles the complete speech processing part and afterwards delivers a of recognized words. A big drawback of this method is that this speech processing activity pauses the requesting activity and, furthermore, shows a dialog indicating that speech recognition takes place. These two issues make the approach useless for us as we need a clear view on the camera image to align the phone and the

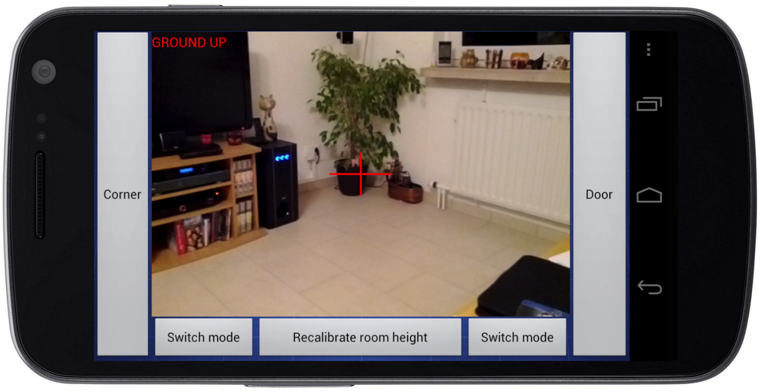


Figure 5.8: User interface for the camera-based approach

activity has to be active to take care of the sensor readings. Therefore, we decided to directly use the underlying `SpeechRecognizer` and `RecognitionListener` to perform the speech recognition part. With this approach, it is possible to do the processing in the background without showing a dialog and without pausing or leaving the main activity. The relevant parts of the speech recognition are shown in Listing 5.3. All recognized inputs are returned via the `RecognitionListener`'s `onResults` method (line 10-14) and can be handled further (line 12). Afterwards the recognition is restarted (line 13) to receive the next command.

```
1  private void startVoiceRecognition() {
2
3      Intent iRec = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
4      iRec.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
           RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
5      iRec.putExtra(RecognizerIntent.EXTRA_CALLING_PACKAGE,
           getClass().getPackage().getName());
6      SpeechRecognizer recognizer = SpeechRecognizer.
         createSpeechRecognizer(this.getApplicationContext());
7      RecognitionListener listener = new RecognitionListener() {
8
9          @Override
10         public void onResults(Bundle results) {
11             ArrayList<String> voiceResults = results.getStringArrayList
                       (SpeechRecognizer.RESULTS_RECOGNITION);
12             handleVoiceInput(voiceResults);
13             startVoiceRecognition();
14         }
15     }
16     recognizer.setRecognitionListener(listener);
17     recognizer.startListening(iRec);
18 }
```

Listing 5.3: Speech recognition with `SpeechRecognizer` and `RecognitionListener`

In Chapter 6 we will present a typical workflow for capturing a floor of a building with the help of the algorithms presented in this chapter.

Chapter 6
Usage of the OIM toolkit

In this chapter we will provide some insights into the work with our OIM toolkit. In the next section we will illustrate the typical workflow for capturing a building. In Section 6.2 we will give some more details on the usage of the trigonometry-based approach and afterwards, we will present some results from an actual capturing done with the OIM toolkit.

6.1 Typical workflow

Figure 6.1 presents the typical workflow for capturing one or several rooms of a building. At startup meta data such as the name of the building and the floor level have to be entered. Afterwards the user can choose to capture a room and has to enter an identifier for this room. Then he can pick one of the two capturing methods presented in Chapter 5. Subsequently, the capturing process begins.

For the pedometer-based approach this means that the user can start walking along the walls of the room to be captured and whenever he reaches a door, he presses the corresponding button in the user interface to indicate that the following steps will cover a door. When reaching the other side of the door, the user can reset the capturing mode to wall capturing again. For creating a floor plan including all captured rooms it is necessary to know which rooms are adjacent. To gain this information we show a dialog to the user whenever he captures a door and allow him to define a unique identifier for this door. During the capturing of an adjacent room, the same door will be captured again and by selecting the previously defined identifier, the necessary information which of the rooms are adjacent can be collected. When a room is completely captured, a visualization of this individual room is shown and the user has the possibility to accept the result or to reject it, e.g. due to measurement errors.

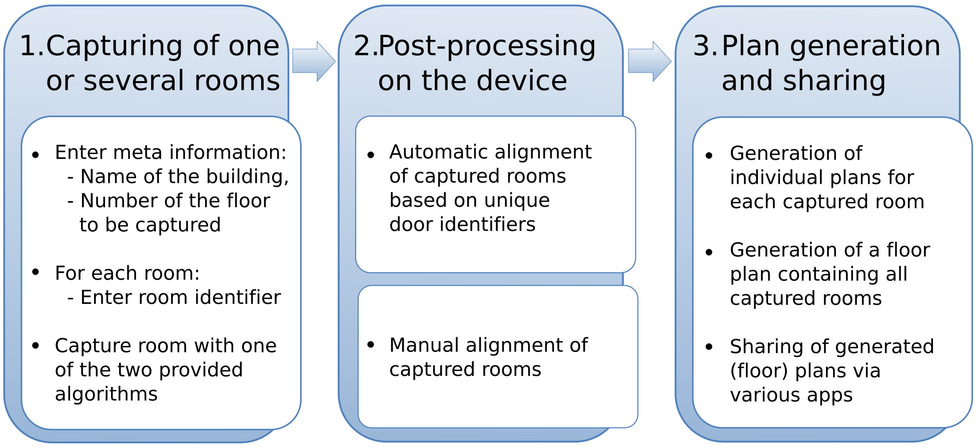


Figure 6.1: Typical workflow for building capturing

The steps presented above have to be repeated for every room to be captured. If a user decides to capture a room using the trigonometry-based approach more interaction possibilities are offered to him that allow for an accurate capturing also if for example a room's corner is hidden behind furniture. We will present the different steps of this approach in more detail in Section 6.2.

After the user ends the capturing process, two options are offered for post-processing directly on the mobile device. If the user decides to use our provided automatic room alignment than nothing additional has to be done by him, as all necessary information, i.e. which rooms are adjacent, has already been gathered through the unique identifiers of the rooms' doors. In contrast, the user also has the possibility to align the captured rooms by himself. After the alignment process is completed, several options for plan generation and plan sharing are offered. The user has on the one hand the possibility to let the toolkit generate individual plans for every captured room or on the other hand, he can request a complete floor plan where the aligned rooms are presented together. The generated plans can then be shared via Android's `ACTION_SEND` intent which enables to send data to any app that declared to be able to handle it. With the help of this standardized interface, we can offer a lot of sharing possibilities to our users, e.g. uploading the captured rooms to Facebook or sending them via email.

6.2 Interactions in the trigonometry-based approach

Figure 6.2 illustrates the different decisions and actions that are of importance during the capturing of a room with the help of the trigonometry-based approach. We distinguish between decisions the application can do on its own (colored in beige) and decisions the user has to take (colored in blue). If a user starts capturing a room he can start in an arbitrary corner point, i.e. a room's corner or a border point of a door. If it is one of the room's corners, the question is whether

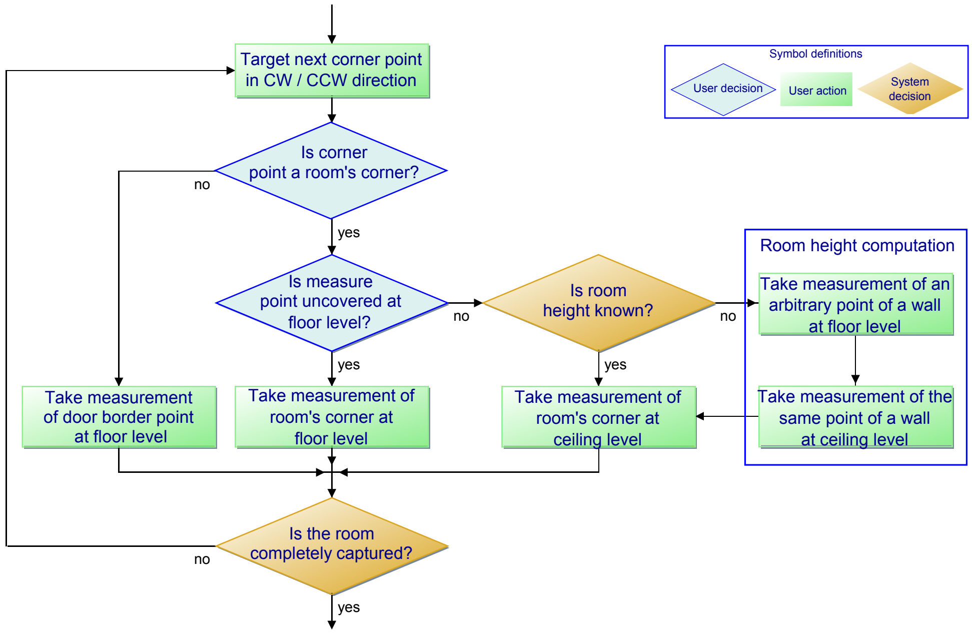


Figure 6.2: Interactions in the trigonometry-based approach

it can be captured on the floor level or if it has to be captured on the ceiling level e.g. because of furniture. In the latter case, the room height has to be known. If the application already knows this information, the capturing at the ceiling level is directly possible. Otherwise the user is first asked to follow the two steps of the room height computation and can than later on capture the original point of interest. For the other cases where the corner point can be measured at the floor level, no additional steps are necessary but the measuring step itself. After each measurement the application checks wether the room is captured completely i.e. the last captured point equals the first one in this room or not. If there are still corners to capture, the process starts at the beginning when the user targets the next corner point in clockwise or counter-clockwise direction. If all corner points of a room are captured, the next steps are again equal to those described in the previous section.

In the next section we will provide the results of capturing several rooms following the typical workflow outlined in Section 6.1 with the trigonometry-based approach (cf. Figure 6.2).

6.3 Example floor plans of captured rooms

To assess the quality of our indoor mapping algorithms under real operating conditions we decided to capture several rooms for which the original architect's plan (see Figure 6.3 for a simplified version) was available to enable a direct comparison. We mapped seven out of ten possible rooms to capture – the remaining

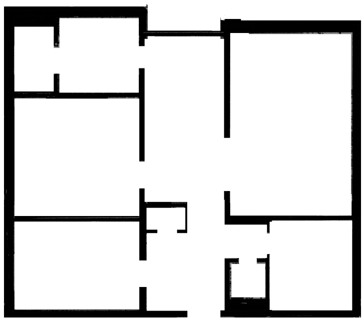


Figure 6.3: Simplified version of the original architect's plan

ones, a small bathroom and two store-rooms were too small for being captured with one of our approaches. Nonetheless, their dimensions can be derived from the dimensions of the adjacent rooms. As all rooms are relatively small in size and moreover often furniture is present on the walls, we decided to use the trigonometry-based approach for capturing the rooms. In the following pictures we will exemplary illustrate the shape of two rooms we captured in comparison to the original shapes derived from the official architect's plan. Blue segments stand for walls whereas red segments illustrate doors.

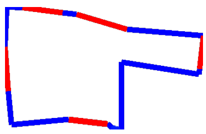


Figure 6.4: Captured corridor

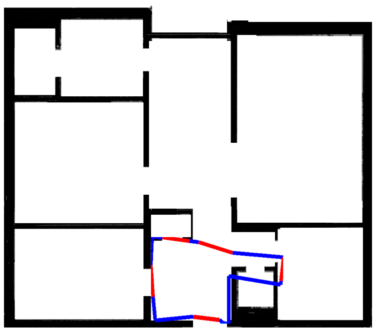


Figure 6.5: Corridor (overlay)

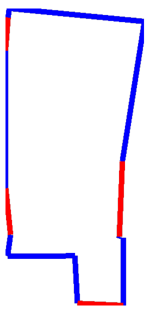


Figure 6.6: Captured dining room

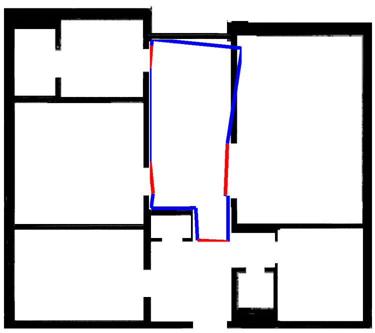


Figure 6.7: Dining room (overlay)

From the given examples in Figures 6.4-6.7 we can see that the capturing algorithm based on trigonometric calculations performs quite good. In the first example we see a slight inclination towards the bottom of the floor plan for some of the measurement points whereas in the second example only one point in the upper right corner has a little offset. In Figure 6.8 we see an overlay with all rooms we have captured in this test case and when we remove the underlying architect's plan, the structure of the flat is still clearly visible (Figure 6.9).

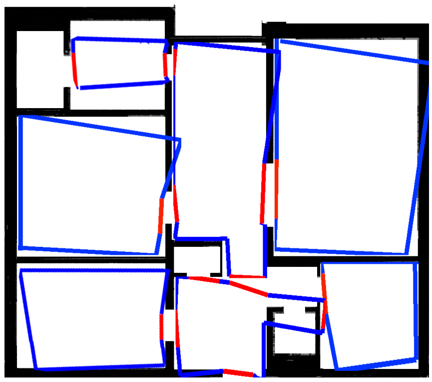


Figure 6.8: Captured rooms (overlay)

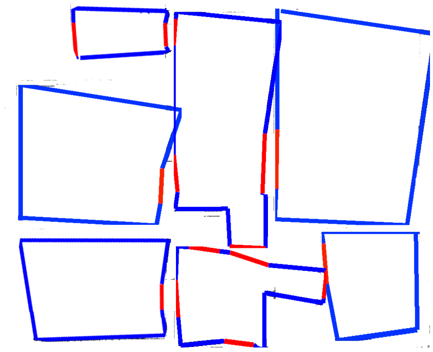


Figure 6.9: Captured rooms

Chapter 7
Conclusion

In the following section we will provide an overview on what we have done in the course of this thesis. Afterwards an outlook will show possible future work based on the realization of this thesis.

7.1 Summary

Motivated by the positive results of an online questionnaire on the public interest in digital indoor maps we conducted, we presented different algorithms that allow the capture of uninstrumented indoor environments with the help of a standard smartphone and we illustrated how the captured data can be utilized.

To base the work on meaningful data, we conducted an online questionnaire to investigate the public interest in digital indoor maps. More than 650 participants in two iterations (standard Internet users and active OpenStreetMap users) completed the questionnaire. For both groups it was obvious that the participants had a certain interest in the topic of digital indoor maps and that most of them would use a corresponding service if there was one. We also learned that there are some types of buildings, e.g. larger buildings such as administrative buildings, shopping malls, airports or universities for which people would like to have indoor maps for. Another finding was that there are significantly more OpenStreetMap users that already wished to have indoor maps and that also significantly more of them would use an indoor map service if it was available. Moreover we found out that many participants were willing to contribute to a system transferring the idea of OpenStreetMap to the inside of buildings. Finally, this online questionnaire showed that a reason for not contributing was that the participants expected this task to be too difficult because of missing positioning systems like GPS indoors.

One possible solution to address this problem is the usage of instrumented environments, but this would require a set-up phase and potentially expensive hardware e.g. RFID tags or IR beacons. In contrast, we investigated technical possibilities to enable reliable indoor mapping solely based on off-the-shelf consumer devices, i.e. smartphones or tablet computers as many people already own such a device and no prior instrumentation has to be done. We therefore considered advantages and shortcomings of different types of sensors which are often available in these devices, e.g. an accelerometer, a magnetometer and sometimes also a gyroscope. As we had expected, the negative aspects of smartphones' sensors were often amplified compared to other devices using the same type of sensors. We decided to further investigate how the disadvantage of a type of sensor could be overcome. It turned out that the combination of different types of sensors (sensor fusion) is a promising approach to cope with the sensors' individual drawbacks. To implement the sensor fusion we used a so-called complementary filter which can not only cope with noisy readings from otherwise stable orientation data obtained from magnetometer and accelerometer (orientation data), but can also cope with a fast and precisely reacting gyroscopic sensor which, however, over time accumulates drift – especially as the raw sensor values have to be integrated over time. As we had anticipated, the combination of the two different signals using a low-pass filter for the orientation data and a high-pass filter for the gyroscopic information provides less noisy data that is both long-term stable and fast reacting.

As we wanted to provide a solution that is applicable to a large variety of rooms, we implemented two different approaches: the first targets large rooms with only a few pieces of furniture, e.g. halls or corridors, whereas the second aims at capturing smaller rooms, e.g. offices with, typically, desks or cabinets. The idea behind the first approach is to keep track of a user's movements, allowing to capture the structure of a building if the user walks along the walls. The distance the user has covered is calculated with the help of a pedometer approach which counts the user's individual steps. By multiplying the number of steps with the user's step length, the distance can be calculated. For the orientation estimation we implemented two different algorithms and it depends on the smartphone's hardware which of them is used. For the majority of devices which do not have a gyroscope, the basic version detects the orientation with the help of accelerometer and magnetometer. As especially the latter is susceptible to interferences e.g. due to electronic devices, the algorithm can provide inaccurate data indoors. Therefore, whenever a gyroscope is available, the second algorithm employing sensor fusion is used. As the application of this pedometer approach is clearly not possible if desks or other furniture is standing on the walls, we proposed a second algorithm which determines the dimensions of the room with the help of trigonometric calculations. As it is crucial for this approach to know the exact orientation of the device, we also employed our two orientation estimation algorithms presented above and use the one with the gyroscope whenever possible.

With the help of the implemented solutions, it is possible to capture a single room, but also a complete floor can be captured in one single run because a complete floor can be seen as a number of single rooms. To reconstruct the floor plan, we implemented an approach based on assigning identifiers to doors during the capturing process. If the user captures a door again (when mapping the room on the other side), we can identify that these two captured rooms are adjacent and connected by the door twice captured. We illustrate the functionality of this approach by producing a floor plan image containing all the rooms captured during a single run.

7.2 Outlook

In the following we will give some ideas for next steps that could be taken in the course of the further development of the idea of OpenIndoorMap.

First of all, a large-scale evaluation should be undertaken to test whether the interaction concepts are well-defined and usable by a large number of people without the need of providing them with additional explanations beforehand. An evaluation could also furnish valuable insights with respect to the technical aspects of the different algorithms. One of the interesting questions to be considered would be for example whether the step detection algorithm also works well if other users with different walking patterns are tracked. In addition, more rooms should be captured during such an evaluation to identify possible shortcomings or aspects that were overlooked during the implementation of the prototype version. After the integration of the gathered feedback, a port to other operating systems for mobile devices should be considered to be able to address potentially more users. A good addition would be iOS as many people own an iPhone and moreover all devices since the iPhone 4 are equipped with a gyroscope which offers the possibility to use the more reliable orientation detection algorithm.

It should also be investigated whether additional objects to be captured have to be integrated. If more than one floor should be mapped at a time, it is for example necessary to implement the possibility to model vertical connections such as stairs or elevators. The detection of the number of stairs would also be possible by using the pedometer approach for the capturing. Depending on the application of the captured data, it could also be of interest to map windows when capturing a room, e.g. for a 3D visualization of the captured building. In this context, it could also be of interest to compute the height of a building as well as the height of the individual floors. Such a height estimation is for example also possible with a smartphone that is equipped with a barometric sensor.

Moreover, additional capturing methods could be integrated in the application. An example for an additional method is a visual odometry approach in which the device's movements can be tracked with the help of calculations based on an optical flow field which is constructed from extracted features in consecutive images or video frames.

Another open question is in which way the collected data could be used further. In addition to the graphical output already implemented, different output formats could be useful. One possibility could be to tag the captured data based on the 3D Building Ontology designed by Götz and Zipf [22], but also other tagging schemes are currently discussed within the OpenStreetMap community[24]. The tagged data can then be written to an output file that can be imported in the offline OpenStreetMap editor JOSM[25] to be able to add the collected information to the publicly accessible OpenStreetMap database.

[24] http://wiki.openstreetmap.org/wiki/Indoor_Mapping#Tagging, last accessed March 27, 2012
[25] http://josm.openstreetmap.de/, last accessed March 27, 2012

Appendix A
Online questionnaire

In this chapter we present the online questionnaire that was provided to the participants of our motivating study. The questionnaire was structured in sections each labeled with a caption indicating the topic the questions will cover. This structure is maintained here. Depending on their former answers, not all questions were shown to all participants. This is indicated by footnotes here. Questions that were not forced to be answered are printed in italic.

Welcome to this questionnaire

As part of a Master's thesis at Saarland University, Germany some data about the usage of existing digital map services should be collected.

Furthermore, similar data should be gathered for a hypothetical system that deals with indoor maps. Such a system will be developed as part of the Master's thesis.

To complete this task, your help is needed.

All data will be collected anonymously. It will took 5-10 minutes to answer the questions. Sent answers cannot be edited afterwards.

At the end of the questionnaire, your feedback regarding the questionnaire is appreciated.

If you have any further questions, feel free to contact me:
osmSurvey@frederickerber.de

Usage of existing map services

Which of the following options applies for you to the service "Google Maps"?

I use it regularly	I use it from time to time	I already used it	I know it but never used it	I never heard of it
○	○	○	○	○

Which of the following options applies for you to the service "Yahoo Maps"?

I use it regularly	I use it from time to time	I already used it	I know it but never used it	I never heard of it
○	○	○	○	○

Which of the following options applies for you to the service "Bing Maps"?

I use it regularly	I use it from time to time	I already used it	I know it but never used it	I never heard of it
○	○	○	○	○

Which of the following options applies for you to the service "OpenStreetMap"?

I use it regularly	I use it from time to time	I already used it	I know it but never used it	I never heard of it
○	○	○	○	○

Collaboration on OpenStreetMap[1]

Have you already edited contents on OpenStreetMap or added new map data?

- ◯ Yes
- ◯ No

What are the reasons for your contribution?[2]

- ☐ General interest in geo information
- ☐ Special interest in OpenStreetMap
- ☐ Friends are also working on OpenStreetMap
- ☐ Autonomous work in a big project
- ☐ Popularity of the project
- ☐ Completion of own surrounding
- ☐ Gather experiences with new technology

Are there additional reasons for your contribution?[2]

Can you give some comments why you did not contributed things by now?[3]

Would you prefer working on a similar project that focusses on indoor maps (e.g. for airports or museums)?[3]

- ◯ Yes
- ◯ No

[1] Shown only if the previous question has been answered with "I already used it", "I use it from time to time" or "I use it regularly"
[2] Shown only if the first question on this page has been answered positively
[3] Shown only if the first question on this page has been answered negatively

OpenIndoorMap - Digital indoor maps[4]

OpenStreetMap is almost completely limited to outdoor map data. It would be beneficial, especially for large buildings like malls or airports, to have detailed indoor maps.

This idea should be examined with the project "OpenIndoorMap". Based on already collected data (building outlines) additional information like room boundaries and floor number should be integrated as easily as possible.

How useful would you find the idea of OpenIndoorMap?

Very useful	Useful	Undecided	Not very useful	Unnecessary
◯	◯	◯	◯	◯

Would you also work on OpenIndoorMap?
◯ Yes
◯ No

Why would you not want to contribute something to OpenIndoorMap?[5]

The following five questions were only shown if the question "Would you also work on OpenIndoorMap" has been answered positively

How long would you like to work on OpenIndoorMap (approx. – per week)[6]
◯ Less than 30 minutes
◯ Between 30 minutes and 1 hour
◯ Between 1 hour and 2 hours
◯ Between 2 hours and 3 hours
◯ Between 3 hours and 5 hours
◯ More than 5 hours

[4] Shown only if the first or the last question on the previous page has been answered positively
[5] Shown only if the previous question has been answered negatively
[6] Shown as drop-down field

Do you have a smartphone or tablet computer?

- ◯ Yes
- ◯ No

Which way of adding maps would you prefer?[7]

Only with a computer	Mostly with a computer	With a computer as well as with a mobile device	Mostly with a mobile device	Only with a mobile device
◯	◯	◯	◯	◯

Which way of editing maps would you prefer?[7]

Only with a computer	Mostly with a computer	With a computer as well as with a mobile device	Mostly with a mobile device	Only with a mobile device
◯	◯	◯	◯	◯

Would you install software on your computer / mobile device for editing reasons?

- ◯ Yes
- ◯ No

[7] Shown only if the participant has stated to have a smartphone or tablet computer

Usage of digital map services

How often do you use digital map services for street maps?[8]

- ◯ Less than 1× a month
- ◯ 1× – 5× a month
- ◯ 2× – 5× a week
- ◯ 6× – 10× a week
- ◯ More than 10× a week

Have you already wished to also have a digital map service for indoor purposes?

- ◯ Yes
- ◯ No

For which buildings would you like to have indoor maps?

Would you use a service that provides indoor maps?

- ◯ Yes
- ◯ No

How often would you use digital map services for indoor maps?[9]

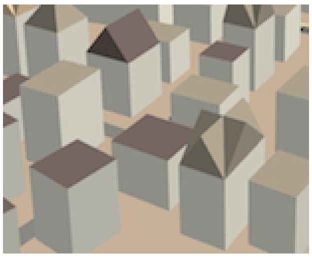
- ◯ Less than 1× a month
- ◯ 1× – 5× a month
- ◯ 2× – 5× a week
- ◯ 6× – 10× a week
- ◯ More than 10× a week

Why would you not use such a system?[10]

[8] Shown as drop-down field
[9] Shown only if the previous question has been answered positively, shown as drop-down field
[10] Shown only if the second to last question has been answered negatively

3D presentation and indoor navigation

A drawback of traditional maps is the fact that they are just two-dimensional. By displaying a map on a computer or mobile device, this issue can be circumvented and a view like the following is possible:



Source: http://www.osm-3d.org/screenshots.en.htm

How useful would you find a 3D indoor maps instead of a 2D map?

Very useful	Useful	Undecided	Not very useful	Unnecessary
◯	◯	◯	◯	◯

Despite the help of maps it is sometimes difficult to find a certain room inside a building. In such situations indoor navigation systems, like the ones known from cars, could be helpful.

How useful would you find an indoor navigation mechanism (e.g. for mobile devices)?

Very useful	Useful	Undecided	Not very useful	Unnecessary
◯	◯	◯	◯	◯

Do you have additional ideas/comments regarding the project "OpenIndoorMap"?

General data

Please provide your gender

◯ Male

◯ Female

How old are you?[11]
◯ Under 20 years
◯ Between 20 and 29 years
◯ Between 30 and 39 years
◯ Between 40 and 49 years
◯ Between 50 and 59 years
◯ Between 60 and 69 years
◯ Older than 70 years

From which country are you from?

How big is the city you live in?[11]
◯ Up to 5,000 residents
◯ 5,000 – 30,000 residents
◯ 30,000 – 100,000 residents
◯ 100,000 – 500,000 residents
◯ 500,000 – 1,000,000 residents
◯ More than 1,000,000 residents

How open-minded about new technology do you think you are?

Very open-minded	Open-minded	Undecided	Skeptic	Very skeptic
◯	◯	◯	◯	◯

Do you have any comments regarding the questionnaire?

[11] Shown as drop-down field

Bibliography

[1] AGRAWAL, MOTILAL; KONOLIGE, KURT AND BLAS, MORTEN R. CenSurE: Center Surround Extremas for Realtime Feature Detection and Matching. In *ECCV (4)* (2008), David A. Forsyth, Philip H. S. Torr and Andrew Zisserman, Eds., vol. 5305 of *Lecture Notes in Computer Science*, pp. 102–115.

[2] ALLEN, GREG. SWAT team blames Gehry architecture for delay in trapping Cleveland shooter. http://greg.org/archive/2003/05/10/swat_team_blames_gehry_architecture_for_delay_in_trapping_cleveland_shooter.html, 2003. [Last accessed, March 3, 2012].

[3] BEAUREGARD, STÉPHANE. A helmet-mounted pedestrian dead reckoning system. In *Proceedings of the 3rd International Forum on Applied Wearable Computing (IFAWC)* (Bremen, Germany, 2006), pp. 1–11.

[4] BUTZ, ANDREAS; BAUS, JÖRG AND KRÜGER, ANTONIO. Augmenting Buildings with Infrared Information. In *Proceedings of the International Symposium on Augmented Reality (ISAR)* (Munich, Germany, 2000), pp. 93–96.

[5] CAMPBELL, ANDREW T.; EISENMAN, SHANE B.; LANE, NICHOLAS D.; MILUZZO, EMILIANO; PETERSON, RONALD A.; LU, HONG; ZHENG, XIAO; MUSOLESI, MIRCO; FODOR, KRISTÓF AND AHN, GAHNG-SEOP. The Rise of People-Centric Sensing. *IEEE Internet Computing 12* (2008), pp. 12–21.

[6] CAMPBELL, JASON; SUKTHANKAR, RAHUL AND NOURBAKHSH, ILLAH. Techniques for evaluating optical flow for visual odometry in extreme terrain. In *Proceedings of Intelligent Robots and Systems (IROS)* (Sendai, Japan, 2004), pp. 3704–3711.

[7] CAMPBELL, JASON; SUKTHANKAR, RAHUL; NOURBAKHSH, ILLAH AND PAHWA, AROON. A Robust Visual Odometry and Precipice Detection System Using Consumer-grade Monocular Vision. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (Barcelona, Spain, 2005), pp. 3421–3427.

[8] CHO, SEONG Y. Design of a pedestrian navigation system and the error compensation using RHEKF filter. PhD thesis, Department of Control and Instrumentation Engineering, Kwangwoon University, Korea, 2004.

[9] CHO, SEONG Y.; LEE, KI W.; PARK, CHAN G. AND LEE, JANG G. A Personal Navigation System Using Low-Cost MEMS / GPS / Fluxgate. *ION 59th Annual Meeting/CIGTF 22nd Guidance Test Symposium* (2003), pp. 122–127.

[10] CHO, SEONG Y. AND PARK, CHAN G. Tilt Compensation Algorithm for 2-Axis Magnetic Compass. *Electronic Letters 39*, 22 (2003), pp. 1589–1590.

[11] CHO, SEONG Y. AND PARK, CHAN G. MEMS Based Pedestrian Navigation System. *The Journal of Navigation 59*, 01 (2006), pp. 135–163.

[12] CHO, SEONG Y.; PARK, CHAN G. AND JEE, GYU I. Measurement system of walking distance using low-cost accelerometers. In *Proceedings of the 4th Asian Control Conference (ASCC)* (Singapore, 2002), pp. 1799–1803.

[13] COLLIN, JUSSI; MEZENTSEV, OLEG AND LACHAPELLE, GÉRARD. Indoor Positioning System Using Accelerometry and High Accuracy Heading Sensors. In *Proceedings of the 16th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS/GNSS)* (Portland, USA, 2003), pp. 1164–1170.

[14] CRAIG, JOHN J. *Introduction to Robotics: Mechanics and Control*, 2nd ed. Addison-Wesley Longman Publishing Co., 1989.

[15] DEKEL, AMNON AND SCHILLER, ELAD. DRec: exploring indoor navigation with an un-augmented smart phone. In *Proceedings of the 12th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI)* (Lisbon, Portugal, 2010), pp. 393–394.

[16] EL-SHEIMY, NASER. Emerging MEMS IMU and Its Impact on Mapping Applications. In *Proceedings of the 52th Photogrammetric Week* (Stuttgart, Germany, 2009), pp. 203–216.

[17] FERBER, ROBERT AND WALES, HUGH G. Detection and Correction of Interviewer Bias. *Public Opinion Quarterly 16*, 1 (1952), pp. 107–127.

[18] GAMMA, ERICH; HELM, RICHARD; JOHNSON, RALPH E. AND VLISSIDES, JOHN. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1995.

[19] GOLDEN, STUART A. AND BATEMAN, STEVEN S. Sensor Measurements for Wi-Fi Location with Emphasis on Time-of-Arrival Ranging. *Mobile Computing, IEEE Transactions on 6*, 10 (2007), pp. 1185–1198.

[20] GONZÁLEZ-JIMÉNEZ, JAVIER; BLANCO, JOSÉ-LUIS; GALINDO, CIPRIANO; ORTIZ-DE GALISTEO, ANTONIO; FERNÁNDEZ-MADRIGAL, JUAN-ANTONIO; MORENO, FRANCISCO-ANGEL AND MARTÍNEZ, JORGE. Mobile robot localization based on Ultra-Wide-Band ranging: A particle filter approach. *Robotics and Autonomous Systems 57* (2009), pp. 496–507.

[21] GOODCHILD, MICHAEL F. Citizens as sensors: the world of volunteered geography. *GeoJournal 69*, 4 (2007), pp. 211–221.

[22] GÖTZ, MARCUS AND ZIPF, ALEXANDER. Extending OpenStreetMap to Indoor Environments: Bringing Volunteered Geographic Information to the Next Level. In *Urban and Regional Data Management: Udms Annual 2011* (2011), Massimo Rumor, Sisi Zlatanova and Hugo LeDoux, Eds., pp. 47–58.

[23] HARRIS, CHRISS AND STEPHENS, MIKE. A Combined Corner and Edge Detection. In *Proceedings of the 4th Alvey Vision Conference* (Manchester, UK, 1988), pp. 147–151.

[24] HUBEL, ANDREAS. Webbrowser based indoor navigation for mobile devices based on OpenStreetMap. Bachelor's thesis, Department of Computer Science, Technische Universität München, Germany, 2011.

[25] JIMENEZ, ANTONIO R.; SECO, FERNANDO; PRIETO, CARLOS AND GUEVARA, JORGE. A comparison of Pedestrian Dead-Reckoning algorithms using a low-cost MEMS IMU. In *Proceedings of the IEEE International Symposium on Intelligent Signal Processing (WISP)* (Budapest, Hungary, 2009), pp. 37–42.

[26] JIN, YUNYE; TOH, HONG-SONG; SOH, WEE-SENG AND WONG, WAI-CHOONG. A robust dead-reckoning pedestrian tracking system with low cost sensors. In *Proceedings of the 9th International Conference on Pervasive Computing and Communications (PerCom)* (Seattle, USA, 2011), pp. 222–230.

[27] KALMAN, RUDOLPH E. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME – Journal of Basic Engineering 82*, Series D (1960), pp. 35–45.

[28] KÄPPI, JANI; SAARINEN, JUKKA AND SYRJRINNE, JARI. Mems-IMU Based Pedestrian Navigator for Handheld devices. In *Proceedings of the 14th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS)* (Salt Lake City, USA, 2001), pp. 1369–1373.

[29] KETABDAR, HAMED; YÜKSEL, KAMER ALI AND ROSHANDEL, MEHRAN. MagiTact: interaction with mobile devices based on compass (magnetic) sensor. In *Proceedings of the 15th international conference on Intelligent user interfaces (IUI)* (Hong Kong, China, 2010), pp. 413–414.

[30] KITT, BERND; GEIGER, ANDREAS AND LATEGAHN, HENNING. Visual Odometry based on Stereo Image Sequences with RANSAC-based Outlier Rejection Scheme. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)* (San Diego, USA, 2010), pp. 486–492.

[31] KONOLIGE, KURT; AGRAWAL, MOTILAL AND SOLÀ, JOAN. Large scale visual odometry for rough terrain. In *Proceedings of the 13th International Symposium on Robotics Research (ISRR)* (Hiroshima, Japan, 2007), pp. 201–221.

[32] KOUROGI, MASAKATSU AND KURATA, TAKESHI. Personal positioning based on walking locomotion analysis with self-contained sensors and a wearable camera. In *Proceedings of the 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)* (Tokyo, Japan, 2003), pp. 103–112.

[33] LADETTO, QUENTIN. On foot navigation: continuous step calibration using both complementary recursive prediction and adaptive Kalman filtering. In *Proceedings of the 13th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS)* (Salt Lake City, USA, 2000), pp. 1735–1740.

[34] LADETTO, QUENTIN AND MERMINOD, BERTRAND. In Step with INS – Navigation for the Blind, Tracking Emergency Crews. *GPS World*, 10 (2002), pp. 30–38.

[35] LEE, KYEONG-HWAN AND EHSANI, REZA. Comparison of two 2D laser scanners for sensing object distances, shapes, and surface patterns. *Computers and Electronics in Agriculture 60* (2008), pp. 250–262.

[36] LEE, SEON W. AND MASE, KENJI. Recognition of walking behaviors for pedestrian navigation. In *Proceedings of the IEEE International Conference on Control Applications (CCA)* (Mexico City, Mexico, 2001), pp. 1152–1155.

[37] LEPPÄKOSKI, HELENA; KÄPPI, JANI; SYRJÄRINNE, JARI AND TAKALA, JARMO. Error Analysis of Step Length Estimation in Pedestrian Dead Reckoning. In *Proceedings of the 15th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS)* (Portland, USA, 2002), pp. 1136–1142.

[38] LEVI, ROBERT AND JUDD, THOMAS. Dead Reckoning Navigational System Using Accelerometer to Measure Foot Impacts, 1996. U.S. Patent Number 5,583,776.

[39] LIU, YUE; WANG, YONGTIAN; YAN, DAYUAN AND ZHOU, YA. DPSD algorithm for AC magnetic tracking system. In *Proceedings of the IEEE Symposium on Virtual Environments, Human-Computer Interfaces and Measurement Systems, (VECIMS)* (Boston, USA, 2004), pp. 101–106.

[40] LÖCHTEFELD, MARKUS; GEHRING, SVEN; SCHÖNING, JOHANNES AND KRÜGER, ANTONIO. PINwI: pedestrian indoor navigation without infrastructure. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries (NordiCHI)* (Reykjavik, Iceland, 2010), pp. 731–734.

[41] MAIMONE, MARK; CHENG, YANG AND MATTHIES, LARRY. Two years of Visual Odometry on the Mars Exploration Rovers. *Journal of Field Robotics 24*, 3 (2007), pp. 169–186.

[42] NASIRI, STEVE; SACHS, DAVID AND MAIA, MICHAEL. Selection and integration of MEMS-based motion processing in consumer apps. http://invensense.com/mems/gyro/documents/whitepapers/Selection-and-integration-of-MEMS-based-motion-processing-in-consumer-apps-070809-EE-Times.pdf, 2009. [Last accessed, January 3, 2012].

[43] NISTÉR, DAVID; NARODITSKY, OLEG AND BERGEN, JAMES. Visual odometry. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (Washington, USA, 2004), pp. 652–659.

[44] OJEDA, LAURO AND BORENSTEIN, JOHANN. Non-GPS Navigation for Emergency Responders. In *International Joint Topical Meeting: "Sharing Solutions for Emergencies and Hazardous Environments"* (Salt Lake City, USA, 2006).

[45] OJEDA, LAURO AND BORENSTEIN, JOHANN. Non-GPS Navigation for Security Personnel and First Responders. *The Journal of Navigation 60*, 03 (2007), pp. 391–407.

[46] OJEDA, LAURO AND BORENSTEIN, JOHANN. Non-GPS navigation with the personal dead-reckoning system. In *Unmanned Systems Technology IX* (Orlando, USA, 2007), Grant R. Gerhart, Douglas W. Gage and Charles M. Shoemaker, Eds., vol. 6561 of *Proceedings of Spie*.

[47] OJEDA, LAURO AND BORENSTEIN, JOHANN. Personal Dead-reckoning System for GPS-denied Environments. In *Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR)* (Rome, Italy, 2007), pp. 1–6.

[48] OLSON, CLARK F.; MATTHIES, LARRY H.; SCHOPPERS, MARCEL AND MAIMONE, MARK W. Stereo Ego-motion Improvements for Robust Rover Navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)* (Seoul, Korea, 2001), pp. 1099–1104.

[49] PRIYANTHA, NISSANKA B.; CHAKRABORTY, ANIT AND BALAKRISHNAN, HARI. The Cricket location-support system. In *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MobiCom)* (Boston, USA, 2000), pp. 32–43.

[50] RANDELL, CLIFF; DJIALLIS, CHRIS AND MULLER, HENK L. Personal Position Measurement Using Dead Reckoning. In *Proceedings of the 7th International Symposium on Wearable Computers (ISWC)* (White Plains, USA, 2003), pp. 166–173.

[51] SCHÖNING, JOHANNES; KRÜGER, ANTONIO; CHEVERST, KEITH; ROHS, MICHAEL; LÖCHTEFELD, MARKUS AND TAHER, FAISAL. Photomap: Using Spontaneously taken Images of Public Maps for Pedestrian Navigation Tasks on Mobile Devices. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI)* (Bonn, Germany, 2009), pp. 105–114.

[52] SCHWARTZ, TIM. The Always Best Positioned Paradigm for Mobile Indoor Applications. PhD thesis, Department of Computer Science, Saarland University, Germany, 2012.

[53] SERRA, ALBERTO; CARBONI, DAVIDE AND MAROTTO, VALENTINA. Indoor pedestrian navigation system using a modern smartphone. In *Proceedings of the 12th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI)* (Lisbon, Portugal, 2010), pp. 397–398.

[54] SHIN, SEUNG H.; LEE, MIN S.; PARK, CHAN G. AND HONG, HYON S. Pedestrian Dead Reckoning System with Phone Location Awareness Algorithm. *Electrical Engineering* (2010), pp. 97–101.

[55] SHIN, SEUNG H.; PARK, CHAN G.; HONG, HYON S. AND LEE, JAE M. MEMS-Based Personal Navigator Equipped on the User's Body. In *Proceedings of the 18th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS/GNSS)* (Long Beach, USA, 2005), pp. 1998–2002.

[56] STARK, HANS-JÖRG. Umfrage zur Motivation von Freiwilligen im Engagement in Open Geo- Data Projekten. In *Tagungsband der Anwenderkonferenz für Freie und Open Source Software für Geoinformationssysteme (FOSSGIS)* (Osnabrück, Germany, 2010), pp. 173–177.

[57] STARK, HANS-JÖRG. Warum Herr Schmidt in OSM mitmacht, Frau Müller hingegen nicht – Empirische Untersuchung der Motivation von Teilnehmenden bei der freiwilligen Erfassung von Geodaten, 2011. *Invited talk at GeoInformatik 2011*.

[58] VAN TONDER, BRADLEY AND WESSON, JANET. The impact of sensor fusion on tilt interaction in a mobile map-based application. In *Proceedings of the South African Institute of Computer Scientists and Information Technologists Conference on Knowledge, Innovation and Leadership in a Diverse, Multidisciplinary Environment (SAICSIT)* (Cape Town, South Africa, 2011), pp. 249–258.

[59] WEINBERG, HARVEY. Using the ADXL202 in Pedometer and Personal Navigation Applications. *AN-602 Application Note, Analog Devices* (2002).

[60] WELCH, GREG AND BISHOP, GARY. An Introduction to the Kalman Filter. Tech. Rep. 95-041, University of North Carolina, Chapel Hill, USA, 1995.

[61] WRIGHT, KEVIN B. Researching Internet-Based Populations: Advantages and Disadvantages of Online Survey Research, Online Questionnaire Authoring Software Packages, and Web Survey Services. *Journal of Computer-Mediated Communication 10*, 3, article 11 (2005).

[62] YAMASAKI, RYOTA; OGINO, ATSUSHI; TAMAKI, TSUYOSHI; UTA, TAKAKI; MATSUZAWA, NAOTO AND KATO, TAKESHI. TDOA location system for IEEE 802.11b WLAN. In *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC)* (New Orleans, USA, 2005), pp. 2338–2343.