2015 年度　修　士　論　文

Automatic Generation of Indoor Maps
using Crowdsourcing with Smartphones
スマートフォンを用いたクラウドソーシングによる
屋内地図の自動生成

稲 葉　　瞳
Inaba, Hitomi

東京大学大学院新領域創成科学研究科
社会文化環境学専攻

Abstract

Nowadays, crowdsourcing systems are popular. In OpenStreetMap(OSM), participants obtain position data and generate maps using crowdsourcing. Crowdsourcing provides a fun and relativley cheap way of creating large systems by offloading the work on the users. Global Positioning System(GPS) is often used to obtain positions outdoors, however, there is no standard system for indoor positioning. Also, indoor maps can only manually be generated based on participants log data. Thus, accurate indoor positioning systems and automated systems for tracing the position are needed. In this thesis, we propose a Pedestrian Dead Reckoning(PDR) based indoor positioning system that uses building features to improve the accuracy. Then, we propose an automated system for generating a "base-map" which is a map that shows entrances, junctions and corridors detected through crowdsourcing with smartphones. The system manages to find most of the commonly used junctions and corridors used in the experiments.

Acknowledgments

First, I would like to thank my Professor Sezaki of the University of Tokyo and Professor Gustafsson of Linkoping University for supervising my work. I would not have been able to write this thesis without you. Also, I would like to thank Parinaz for providing assistance with the Extended Kalman Filter and Professor Kobayashi for helping me with the structure of the thesis.

For helping me gather data for the experiment I would like to thank Lisa and Olle Westrin.

Finally, I would like to thank all Sezaki's lab members for their support during the time I spent on this thesis.

Contents

List of Figures

1

Introduction

1.1 Background and Motivation

Nowadays, a copious number of navigation systems are available and Location Based Systems (LBS) provide indispensable services for smartphone and car applications. Global Positioning System (GPS) is the most common positioning system for outdoors and Wi-Fi for indoors. For indoor positioning systems there are more systems available. Examples are e.g. Bluetooth, RFID and Indoor MEssaging System (IMES). IMES was invented to achieve seamless positioning for indoor positioning/navigation systems and it uses the same signal as GPS. However, IMES has some problems; it is costly and the handover time between IMES and GPS is long. Furthermore, while Wi-Fi and cellular based positioning systems do not require any changes in the receivers, IMES requires new receivers to be installed in the devices. Due to the problems above IMES is difficult to use as a standard.

We define indoor as, of course inside buildings, and urban canyons. An urban canyon is a place where GPS positions are inaccurate due to disturbance from the environment e.g. inside a tunnel or close to high buildings. Especially indoor environments, where the positioning accuracy is not accurate enough because of signal propagation, obstacles that interrupt signals and so on, GPS performs poorly. GPS also has problems such as multipath, propagation and spoofing.

Much research has been done to achieve a seamless positioning/navigation system to solve the problems above. Hybrid Positioning Systems which uses several different positioning technologies combined with Dead Reckoning (DR) to obtain positions have been proposed for so called Pedestrian Dead Reckoning (PDR) [10]. PDR integrates detection of step length, step detection and orientation estimation [7]. The result of PDR contains a lot of drift since the measurement devices cannot be held stationary. Therefore, some research apply filtering

to the sensor data and the map [9] and place the measurement device in a special place e.g. inside a shoe [7] or pocket [2].

In recent years, the use of Geographic Information Systems (GIS) has increased rapidly and GIS is applied to numerous situations e.g. several kinds of navigation systems, web map services, infrastructure management systems and urban planning. Therefore, it is important to have correct geographic information on the maps in real time. In navigation systems, web map services such as Google Maps and OpenStreetMap (OSM) are mainly used outdoors. Those maps contain much necessary information, however, information about traffic structures such as zebra-crossings and walking bridges as well as indoor related structures e.g. entrances, corridors and stairs are seldom available and is not accurately placed; that is to say, the quality of especially pedestrian navigation systems is not high enough. Using a crowdsourcing-based mapping project such as OSM and TomTom Maps, users can add and edit information manually, however, there is no guarantee that the information is correct. Some companies provide indoor maps of their premises e.g. universities and shopping malls therefore some indoor maps are available. However, it is difficult to create indoor maps using a crowdsourcing-based mapping approach since the obtained positions are not accurate. Although you can obtain positions from several indoor positioning systems, the convergence time is long; that is obtaining positions will not be in real time. In addition, if you start navigation from outdoors to indoors or vice versa, it does not work well because of the non-availability phase e.g. the areas where the indoor and outdoor positioning systems switches. From the above, detecting entrances can be applied to the non-availability phase to support handovers between systems. Detecting junctions can also be applied to real time mapping and navigation. In crowdsourcing-based mapping projects e.g. OSM, users create maps by taking GPS data and plot it over the base map. If we succeed to detect indoor structures automatically from people's trajectories, we can hopefully apply it to automatic generate structures and indoor maps.

1.2 Objective

As mentioned above, generating a map is mostly done manually by the participants. Therefore, the large goal of this thesis is to decrease the manual work. The goal is to remove the part where the participants has to manually plot the traces on the map. The map generation should be automatic. To achieve that, obtaining accurate positions and detecting indoor related structures e.g. entrances, corridors, junctions and rooms are required. Then, generating an indoor map automatically from the collected data is possible.

In this thesis, we aim to obtain accurate trajectories and detect structures, entrances, junctions and corridors which forms the base of the map and then, automatically create maps with that information. We will gather the information from multiple trajectories to simulate crowdsourcing.

Also, a limitation is that no expensive equipment should be used. Readily available smartphones will be used since they are cheap and everyone has al-

ready has one; that means, the availability is high and that is a prerequisite for crowdsourcing. If we succeed, it can be applied to help crowdsourcing-based maps e.g. OSM to automatically create indoor maps. This system can improve pedestrian positioning/navigation and handover between outdoors and indoors positioning systems. Also, this algorithm can also be applied outdoors e.g. new traffic structure detection to generate maps and observe people's mobility pattern in real time.

1.3 Contribution

The main contribution of this thesis is an automatically generated indoor map that shows junction positions and corridors. Entrances are also detected and put on the map. The map can be used by indoor positioning systems, OpenStreetMap and to put metadata on existing maps. In addition, during the map generation a heat map is created that shows where subjects walk the most. The heat map can be used to analyze how people walk through their buildings. Similar analysis can be done for junctions and entrances.

1.4 Thesis Structure

This section describes the major sections of the thesis.

1.4.1 Related Work

Work related to the proposed method is described in this section.

1.4.2 Basic Concept

Basic concepts that are required to be familiar with to understand this thesis are presented here. The concepts are, PDR, Kalman Filter and Extended Kalman Filter. Readers already familiar with these concepts can skip this section.

1.4.3 Method Proposal

The proposed method is described in this chapter. The method is divided into three sections described below.

Map Mining

In this section, we generated a binary map from a simplified map where it is easy to distinguish between indoors and outdoors. The purpose of the binary map is to simplify decisions where it is required to know whether a subject is indoor or outdoor.

Data Processing

We applied Pedestrian Dead Reckoning and filtered with an Extended Kalman Filter to create and smooth the trajectories from the subjects. We also proposed and applied a few conditions to increase the trajectories' accuracy. The conditions were taken from common characteristics of the experiment building and buildings in general. The trajectories were also adapted to the binary map for the following steps.

Map Generation

First, a non-parametric map was generated, which in this thesis is a heat map of the trajectories. Then, we detected nodes, defined as entrances and junctions, in order to generate a parametric map. Furthermore, we generated corridors from the detected nodes. The parametric map was then created by plotting the corridors on the binary map. A part of the parametric map is the topology information, node position etc., that can be used.

1.4.4 Evaluation Experiment

In this section, an experiment to evaluate our method is described. GPS, Accelerometer, Gyroscope, Magnetic and GPS accuracy data was collected in the B-building in Linkoping University. Two parameters's best values in terms of CDF-67% were calculated.

1.4.5 Result

The result from the various steps in the method is presented with figures and some brief comments.

1.4.6 Discussion

The results are discussed in more detail and the causes of some errors are explained together with possible mitigations.

1.4.7 Conclusion

The thesis is summarized and a section with future work is included.

2

Related Work

The description of related work is divided into two subsections, Data Processing and Map Generation, which are the major steps in the proposed method. Data Processing describes indoor positioning methods and various approaches to improve the accuracy of the acquired positions. It is a major part since map is generated from only the positioning data and the accuracy of that data is therefore essential.

The second major part Map Generation generates a map from the acquired positioning data and methods similar to the proposed method is described below.

2.1 Data Processing

Nowadays, numerous Pedestrian Dead Reckoning(PDR) research exists to detect one's location both outdoors and indoors. PDR uses sensors such as gyroscopes and accelerometers and does not require GPS coordinates, therefore it is suitable for indoor as well as outdoor environments. Other methods for obtaining position data in an indoor environment use Wi-Fi signal strength together with known fingerprints or radio frequency identification(RFID). The fingerprint based methods require that the fingerprints are manually collected before the system is possible to use. Furthermore, in buildings that lack Wi-Fi Access Points and other radio frequencies to measure such a system is not possible to use.

To obtain positions seamlessly, PDR can be applied as mentioned above. However, such systems obtain errors from the sensors for each position's estimation. Therefore, only using PDR is not suitable for long time positioning.

There are several methods that can be used to implement PDR. The basic one is explained in Section 3. PDR can be divided into several problems. Measuring the device's state, step detection, step length estimation and orientation estimation. As the device's state several methods exist such as putting the sen-

sors inside your pocket [2],[18], on your shoe [7] [19] and running mode [9] are proposed. Usually, these states are decided by how the sensor devices are held in general e.g. inside your pocket or held in one's hand. However, the authors of [2] applied the device's state to obtain accurate step detection and step length estimation based on the opening angle of the legs.

To make PDR more accurate, a magnetometer sensor can be used to improve the orientation estimation [9]. The magnetometer sensor can detect which direction that is north; that is to say, it functions in the same way as a compass. An accelerometer sensor is also required to get the direction from the magnetometer. The error in this direction will, in contrast to the angle estimation from the gyroscope, not increase due to bias errors. The gyroscope on the other hand is more accurate during short periods of time. The compass and gyroscope sensor values can therefore complement each other since they have good characteristics together. The magnetic compass will be able to reduce the gyroscope's bias error.

The author of [9] created a magnetic deviation map to obtain accurate trajectory by using a magnetometer.

A hybrid positioning system consists of PDR, filtering and map filtering [10] [9]. After doing PDR, Kalman filter was used to fuse PDR and GPS in [9]. Accurate GPS values are observed only outdoors therefore it can only be applied outdoors. [10] uses their proposed angle-based localization and a map in the filtering. In the map filtering, they deployed vector maps that contain wall and door locations as well as accessible and inaccessible areas inside buildings to filter the PDR.

Handover positioning systems switches between positioning sensors depending on the location. In [6], the authors use GPS outdoors and Wi-Fi indoors to achieve seamless positioning. GPS is the most represented positioning system outdoors and Wi-Fi for indoors but other systems are also used in Handover positioning systems. A problem with handover positioning systems is the convergence time during the handover. Detecting when a user enters a building can be difficult and if it is detected too late the accuracy of the position will have reduced since the handover positioning system used inaccurate values from the outdoor system when the user was inside a building.

Usually, antennas (Access Points) send beacons in all directions. However, the authors used directional antennas which send signals in only one direction. Therefore you will obtain more accurate positions since it is possible to calculate from which direction where each found AP is in. The position can then be calculated by converting signal strength to distance from the antenna by knowing the signal's angle and your heading.

2.2 Map Generation

There are several approaches for generating road maps automatically. The authors of [1] detected road curves and lanes by clustering data from raw GPS data. From this data they then create road networks. Also the authors of [21] generate a minimal digitised road map using raw GPS data.

The proposed approaches can be applied outdoors but not indoors since for example the GPS is not accurate enough. Some systems are available to use indoors for positioning. However, there is no explicit system capable of obtaining positions in real time and generating indoor maps automatically. Generating a map using images is common, it was done by [13] and [20]. To generate an outdoor map, the authors of [20] detect the direction and find intersections by using two scanning laser range sensors. They then apply the result to create an autonomous navigation system. While the scanning laser range sensors method is common it is not useful to use in a crowdsourcing method since the laser scanner is expensive and heavy.

To generate an indoor map, corridor, junction and entrance detection is required. The author of [22] detects turning points and doors and marks them as intersections. Rooms and lobbies are classified as open areas. Each intersection and open area is then converted into a node. The corridors can be detected between the nodes. To find the locations of the nodes and ways Wi-Fi signal strength together with fingerprints was used.

3

Basic Concepts

In this section, we provide a basic introduction of the concepts that were used in this thesis. More details regarding the concepts is introduced later in this thesis as well.

3.1 Calculate Orientation

Orientation is calculated from the accelerometer *acc* and magnetometer *mag* sensors. The calculation is as follows.

$$
\begin{aligned}
Acc &= \sqrt{{acc_x}^2 + {acc_y}^2 + {acc_z}^2} \\
Mag &= \sqrt{{mag_x}^2 + {mag_y}^2 + {mag_z}^2} \\
h_x &= mag_y acc_z - mag_z acc_y \\
h_y &= mag_z acc_x - mag_x acc_z \\
h_z &= mag_x acc_y - mag_y acc_x \\
h &= \sqrt{{h_x}^2 + {h_y}^2 + {h_z}^2} \\
r_x &= acc_y h_z - acc_z h_y \\
r_y &= acc_z h_x - acc_x h_z \\
r_z &= acc_x h_y - acc_y h_x \\
r &= \sqrt{{r_x}^2 + {r_y}^2 + {r_z}^2} \\
\text{Orient} &= \arctan(h_x/h, r_x/r)
\end{aligned}
\tag{3.1}
$$

where x, y, z are the axis in each sensor and Orient is the calculated orientation.

3.2 Pedestrian Dead Reckoning

Dead Reckoning(DR) is a method to calculate the current position from a previously determined position by updating it with the the user's movement. To determine the movement, heading sensors such as gyroscopes, accelerometers and sensors that measure magnetic fields are used. Usually, this method is applied in urban canyons where the positions obtained from GPS lose accuracy. The model used in this thesis and the equations for updating the position are described in the following Motion Model and Position Update sections.

3.2.1 Motion Model

The orientation is calculated from the yaw rate from the gyroscope as shown in the following equation.

$$\begin{aligned} \mathrm{h}_k &= \mathrm{h}_{k-1} + \mathrm{Gyroscope}_z \Delta t \\ \Delta t &= \mathrm{Time}_k - \mathrm{Time}_{k-1} \end{aligned} \tag{3.2}$$

Each h_k consist of the sum of all previous gyroscope values times time plus an initial heading. Therefore, the error in h_k consists of the sum of the previous errors.

3.2.2 Position Update

The PDR equation which has been used in this thesis is presented below. It updates the State Vector x as shown in Equation 3.3.

$$\begin{aligned} x &= (X, Y, h)^T \\ X_k &= X_{k-1} + SL\cos(\mathrm{orientation}_k) \\ Y_k &= Y_{k-1} + SL\sin(\mathrm{orientation}_k) \end{aligned} \tag{3.3}$$

where SL is the user's step length. This position update has been used as time update in the filtering process below. To obtain the current position, X_k and Y_k, an estimation of SL is required. In this thesis, we decided the value in for each person, however, this can be calculated by using the estimated speed v and the time interval Δt between each detected step. The speed can be calculated from the GPS positions when the user is outside.

Multiplying SL by the user's heading gives the distance in each dimension from the previous point that the user has moved. The current position can then be estimated by adding the movement to the previous position. The disadvantage of DR is that the error in each estimated position is summed with the errors from the previous estimations. Therefore, the accuracy of the estimation will decrease for each estimation, that is to say, DR is not suited for long term positioning.

3.3 Kalman Filter

Kalman Filter(KF) is an algorithm that is used for estimation of time varying values e.g. one's position and velocity by measurement values which contain discrete errors. This algorithm is applied not only in positioning but also in numerous other fields such as navigation systems, computer vision, control of vehicles and data analysis of economics.

First, we describe the basic KF and define the state vector x in Equation 3.4, where $X_{k|j}$ is the state vector x at time k with measurement update at time j. A state vector consists of a position in the X-dimension, Y-dimension and a heading.

$$\begin{aligned} x &= (X, Y, h)^T \\ x_{k|k-1} &= A_k x_{k-1|k-1} + B_k u_k + w_k \end{aligned} \tag{3.4}$$

A is the state transition model matrix and it describes the relationships between the state variables in a time update. B is the control-input model for input u_k. w_k is the process noise with covariance Q_k. This process noise is assumed to be drawn from a zero mean multivariate normal distribution.

$$w_k \sim N(0, Q_k) \tag{3.5}$$

Measurement vector z of time k according to the state $x_{k|k}$ is,

$$z_k = C_k x_{k|k-1} + v_k$$

where C_k is the matrix that maps the state space into the measurement space. v_k is the process noise for the measurement and it has covariance R_k.

$$v_k \sim N(0, R_k) \tag{3.6}$$

We estimate the state values based on the state $x_{k|k}$ and the measurement z_k by repeatedly doing "Time update" and "Measurement update".

3.3.1 Time Update

In Time update, we estimate a new state vector $x_{k|k-1}$ and its covariance matrix $P_{k|k-1}$ from a previous state vector $x_{k|k-1}$. The equation for the Time update is shown in Equation 3.7.

$$\begin{aligned} x_{k|k-1} &= A_k x_{k-1|k-1} + B_k u_k \\ P_{k|k-1} &= A_k P_{k-1|k-1} A_k T + G_k Q_k G_k T \end{aligned} \tag{3.7}$$

3.3.2 Measurement Update

In Measurement Update, we fix the current position estimation, that was calculated in the time update, using the Kalman gain K to put a weight on the difference of the values from the Time update $x_{k|k-1}$ and the values from the sensor data or other secondary data T_k. The secondary data is received independently from the Time update.

$$\begin{aligned} K_k &= P_{k|k-1} C_k' inv(C_k P_{k|k-1} C_k' + R_k) \\ x_{k|k} &= x_{k|k-1} + K_k(T_k - C_k x_{k|k-1}) \end{aligned} \tag{3.8}$$

P is also updated in the Measurement update.

$$P_{k|k} = (I - K_k C_k) Pk|k - 1 \tag{3.9}$$

$x_{k|k}$ is the estimated value(output) of time k.

3.4 Extended Kalman Filter

Extended Kalman Filter(EKF) is a non-linear model of the basic Kalman Filter. This EKF linearizes the non-linear function f in the estimation. The EKF's state space model and measurement model equations are as follows,

$$\begin{aligned} x_{k+1} &= f(x_k, u_k, w_k), w_k \sim N(0, Q_k) \\ z_k &= x_k + v_k, v_k \sim N(0, R_k) \end{aligned} \tag{3.10}$$

where x_k is the state vector, u_k is a input. The function f is used to predict a new state from the previous estimate. w_k is the process noise with covariance Q_k. In the Measurement model equation z_k, v_k is the measurement noise with covariance R. We use Jacobian matrixes to linearize the model in each update.

3.4.1 Time Update

The Time Update equations are as follows,

$$\begin{aligned} x_{k+1} &= f(x_k, u_k, w_k) \\ P_{k+1} &= F_k P_k F_k T + Q_k \end{aligned} \tag{3.11}$$

where x_{k+1} is the time updated state, P_k is x_{k+1}'s covariance in the time update, F_k is the Jacobian matrix of $f(x_k, u_k, w_k)$ of x_k.

3.4.2 Measurement Update

The Measurement Update equation is as follows,

$$\begin{aligned} K_{k+1} &= P_{k+1} H_{k+1}^T (H_{k+1} P_{k+1} H_{k+1}^T + R_{k+1})^{-1} \\ x_{k+1} &= x_{k+1} + K_{k+1}(T_{k+1} - z_{k+1} x_{k+1}) \\ P_{k+1} &= (I - K_{k+1} H_{k+1}) P_{k+1} \end{aligned} \tag{3.12}$$

where H is Jacobian matrix of x_{k+1} in Equation 3.10. In the x_{k+1} equation, we multiply the Kalman gain K_{k+1} with $T_{k+1} - z_{k+1}$. This $T_{k+1} - z_{k+1}$ is the difference between the measurement model equation z_{k+1} and the real measurement state T_{k+1} that is taken from GPS values for Position X and Y and magnetic compass for Heading.

The disadvantages of EKF is that the bias may increase in the estimated value because of the conversion of the average value. Also the calculation might become unstable against systems which has strong non-linearity.

4

Method Proposal

The proposed method is presented in this section. This section is divided into three major steps, Map Mining, Data Processing and Map Generation. These steps are described in their corresponding subsection below.

4.1 Map Mining

OpenStreetMap(OSM) was used to generate a simple binary image map. The map is considered simple since each metadata layer, e.g. names of roads and structures etc., were removed from the map before it was downloaded. The metadata was removed to make the generation of the binary map simpler since the metadata on the map complicated the color substitution described in Section 4.1.1. We picked the B-building in Linkoping University to use in the experiments in this thesis. The simplified map is shown in Figure 4.1 and such simplified maps can be downloaded from OSM's website.

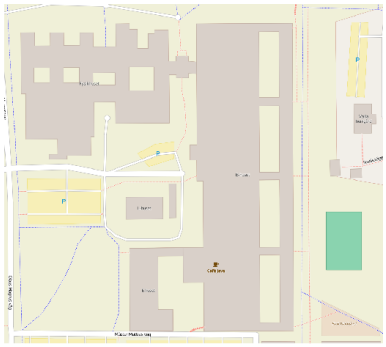

Figure 4.1: *Map without metadata*

4.1.1 Binary Map Generation

After downloading the map, we generated a binary map where white means indoor and black outdoor. It has been used multiple times to make sure that trajectories only inside the building is used. With the simplified map in Figure 4.1, we created the binary map by substituting the building colors with white and the rest with black. This gives a map where indoors is white and outdoors is black.

4.2 Data Processing

In this section, we will explain methods used to process the position data i.e. Pedestrian Dead Reckoning(PDR) and Fusioning of GPS and PDR to create trajectories. This chapter also describes how the junctions and corridors are detected from the trajectories.

4.2.1 Pedestrian Dead Reckoning

PDR can be divided into five sections which are "Device's State", "Step Detection", "Step Length Estimation", "Orientation Estimation" and "Position Update". To do precise PDR, researchers have done each section by using sensor data from people's behaviour and in many various situations, which was done by Kakiuchi [9], for example, chest mode and pocket mode. That is, the device is assumed to be in a common situation and the assumption is used to improve the accuracy as mentioned in Section 2.1. However, in this thesis, we assume that device is held horizontally and pointed in the walking direction. Therefore we do not consider the step "Device's State". Also, we merged the next three steps "Step Detection", "Step Length Estimation" and "Orientation Estimation" into one step that we call "Naive Dead Reckoning".

Naive Dead Reckoning

In this Naive Dead Reckoning algorithm, we used a band-pass Filter(BP-Filter), with cutoff frequencies picked to remove frequencies that are unlikely to be steps. The chosen cutoff frequencies were 0.5-4.0 Hz and were used to filter the data from the accelerometer sensor. Then, the filter's output was compared to a threshold, specific for each person and where the output was above the threshold a step was detected. The step detection's equation is as follows,

$$\sqrt{{acc_x}^2 + {acc_y}^2 + {acc_z}^2} \geq \text{th} \tag{4.1}$$

where acc_x, acc_y and acc_z are accelerometer in each dimension. The threshold th is decided before the experiment each person's steps lightly. Also step length is chosen by each persons step length. We assume that the subject walks normally with equally long steps.

To calculate the orientation, we only use the yaw from the gyroscope which is the z-axis. This was done to simplify the process since otherwise three axis

are needed to take into account. The estimation equation for the update of the orientation is as follows,

$$\text{orientation}_k = \text{orientation}_{k-1} + \text{Gyroscope}_z * \Delta t \tag{4.2}$$

where Δt is the time interval between two steps and Gyroscope_z is the yaw rate from the gyroscope. Each orientation_k consist of the sum of all previous gyroscope values. Therefore, the error in orientation_k also consists sum of the previous errors.

Position Update

The PDR equation which has been used in this thesis is presented below. It updates the position update State Vector x as shown in Equation 4.3.

$$\begin{aligned} x &= (X, Y, h)^T \\ X_k &= X_{k-1} + SL\cos(\text{orientation}_k) \\ Y_k &= Y_{k-1} + SL\sin(\text{orientation}_k) \end{aligned} \tag{4.3}$$

where SL is the user's step length. Later, this position update has been used as time update in the filtering process.

4.2.2 Merging

Since the accuracy of the positions received from the GPS receiver is low in an indoor environment such as inside buildings and urban canyons, relying on GPS alone is not enough for an accurate positioning system. Therefore, a second system is needed that can provide accurate positions indoors. In this thesis, the solution is to fuse the GPS values and PDR outdoors with an EKF and rely on only PDR, together with the magnetic compass, indoors with some conditions.

4.2.3 Fusing PDR with GPS

Extended Kalman Filter(EKF) was used to fuse PDR with GPS for the outdoor parts of the trajectories. The algorithm of this filtering is explained below.

First, the state vector x was defined as in Equation 4.4.

$$x = (X, Y, h)^T \tag{4.4}$$

where X is latitude in pixels and Y is longitude in pixels. Pixels were used as a coordinate system based on the picture of the map where the data was taken. h is the heading that was calculated by a fusion of gyroscope, accelerometer and magnetometer data.

The Extended Kalman Filter's state space model and measurement model equations are as follows,

$$\begin{aligned} x_{k+1} &= f(x_k, u_k, w_k) \\ z_k &= x_k + v_k \end{aligned} \tag{4.5}$$

where x_k is the state, u_k is the measured yaw rate and w_k is the process noise with covariance Q_k.

$$w_k \sim N(0, Q_k) \tag{4.6}$$

In the Measurement model equation z_k, v_k is the measurement noise with covariance R.

$$v_k \sim N(0, R_k) \tag{4.7}$$

The dynamics and covariance matrix which were used in this filtering implementation are as follows,

$$\begin{aligned}
X(k+1) &= X(k) + (SL + w_L(k))\cos(h(k)) \\
Y(k+1) &= Y(k) - (SL + w_L(k))\sin(h(k)) \\
h(k+1) &= h(k) + T(K)\omega_z(k) + w_z(k) \\
Q &= \begin{bmatrix} q_X & 0 & 0 \\ 0 & q_Y & 0 \\ 0 & 0 & q_h \end{bmatrix}, \\
R &= \begin{bmatrix} r_X & 0 & 0 \\ 0 & r_Y & 0 \\ 0 & 0 & r_h \end{bmatrix}, \\
F &= \begin{bmatrix} 1 & 0 & -SLsin(h(k)) \\ 0 & 1 & SLcos(h(k)) \\ 0 & 0 & 1 \end{bmatrix}, \\
H &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},
\end{aligned} \tag{4.8}$$

where ω_z is the yaw rate of gyroscope and T is the time interval in each time update. w_z is the process noise of the gyroscope, w_L is the process noise of step length and SL is the step length. F is the Jacobian matrix of the time update equations.

Time Update

The time update equations are as follows,

$$\begin{aligned} x_{k|k-1} &= (X, Y, h)^T \\ X(k+1) &= X(k) + (SL + w_L(k))\cos(h(k)) \\ Y(k+1) &= Y(k) - (SL + w_L(k))\sin(h(k)) \\ h(k+1) &= h(k) + T(k)\omega_z(k) + w_z(k) \\ P_{k|k-1} &= F_k P_{k-1|k-1} F_k^T + Q_k \end{aligned} \tag{4.9}$$

where ω_k is calculated as the sum of yaw rate in each step. x is the time updated state, P is x's covariance in the time update and T is the time interval between each time update.

Measurement Update

The Measurement Update equations are as follows,

$$\begin{aligned} K_k &= P_{k|k-1} H_k^T (H_k P_{k|k-1} H_k^T + R_k)^{-1} \\ x_{k|k} &= x_{k|k-1} + K_k(T_k - z_k x_{k|k-1}) \\ P_{k|k} &= (I - K_k H_k) P_{k|k-1} \end{aligned} \tag{4.10}$$

In the $x_{k|k}$ equation, we multiply the Kalman gain K with $T_k - Z_k$. This $T_k - z_k$ is the difference between the measurement model equation z_k and the real measurement state T_k that was taken from GPS values for Position X and Y and magnetometer for Heading.

4.2.4 Conditions

Since the fusion depends on whether the user is indoor or outdoor we have added conditions for when which measurement update is applied. The conditions are explained as follows.

Conditions for Outdoors

Regarding the filtering, Time Updates, which is the same as PDR, will be done step-wise. When an accurate GPS value is received a Measurement Update is done. We added three conditions that decides when we will do Measurement Updates using GPS coordinates to fix the position. First, if we are outside we will obtain GPS coordinates. Second, the color for the position on the binary map should be black. Third, the accuracy of the GPS values should be good. To check the accuracy, we have created an extended app that is called "Quat2Orient" to obtain the accuracy that the GPS receiver calculates. This app uses Android to collect GPS values with the belonging accuracy and also fuse magnetometer and accelerometer sensors values to create an orientation. We call it "Magnetic compass". The accuracy threshold for the GPS accuracy has been set to 5 which means

that only values with less than 5 meters in accuracy are considered accurate. This condition is applied to the whole processing step.

However, since we always obtain values from the magnetic compass and gyroscope, we have decided to do measurement updates for only the heading when we do not obtain accurate GPS values to make the trajectory more precise. This is applied to indoors as well. The equation of K_k and $x_{k|k}$ is the same as Equation 4.10, however, the matrix for position X and Y are set to zero since we only update the heading. In addition, we don't update $P_{k|k}$ in order to not affect the P of the next time update with the position X and Y.

Conditions for Indoors

In indoors, the color on the binary map should be white and we do PDR with the measurement update for the heading as described above.

The drift error is saved from the starting point due to the nature of the gyroscope, though the magnetic compass will compensate it. We have decided to reset the errors when we enter entrances to correct the orientation. The conditions is that if we switched color then the orientation is fixed to the correct orientation by assuming that the entrance is perpendicular to the building's side, which is a common property for entrances. Therefore we set the orientation as the closest multiple of 90 degrees of the current value.

A requirement is that we have to be able to detect entrances. The condition to detect entrances and reset error is described in following section. Errors from the gyroscope increases indoors since we cannot obtain accurate GPS values to fix the positions and the magnetic compass's accuracy is not perfect and is affected by magnetic fields. Therefore, since persons usually walk in straight lines we correct the orientation by assuming that the heading is a multiple of 45 or 90 degrees.

Regarding the building's structure features, turns and junctions are usually 90 degrees. In the experiment building of this thesis, there were only 90 degree turns. Therefore, we assumed that only 90 degrees turns are allowed. The method of resetting the error is that we examine the gyroscope's value and if it exceeds a threshold, we round the estimated heading to the closest multiple of 90 degrees. The gyroscope's threshold is set to 0.5 in this thesis. This value was decided by examining the behaviour of the gyroscope and magnetic compass.

However, the magnetic compass is easily disturbed by machines and reinforcing bars close to the sensors and we cannot always obtain accurate magnetometer values. Therefore a threshold for the magnetic field strength is used to remove values based on weak magnetic fields. The threshold is set to 30 μT since it is required to obtain accurate values in general. The magnetic field strength is calculated as follows,

$$\text{Magnetic Field Strength} = \sqrt{{Mag_x}^2 + {Mag_y}^2 + {Mag_z}^2} \tag{4.11}$$

4.3 Map Generation

We define "base map" as a simplified map which only contains entrances and junctions; that is, a map with structures that would not change often, not structures such as rooms and shops. In this thesis, only two dimensional maps are considered. Therefore, stairs and elevators are not considered and therefore not detected. We will generate a non-parametric and a parametric map of the base map. After generating such a map, it can be used to extend an existing map. For example, the generated indoor map could be overlayed on top of OSM to automatically create indoor maps.

4.3.1 Non-Parametric Map

The non-parametric map will be a heat map where the color of each position is decided by how many subjects that has been in this position. Two-dimensional normal distributions are applied when generating the heat map. The estimated position's accuracy degrades when the user is indoors since no accurate GPS values can be obtained. When the position is added to the non-parametric map not only the estimated position should be added, but the surrounding positions should be added with the height of the likelihood of the real position being in the current position. Two-dimensional normal distributions are used to represent the likelihoods.

Two-dimensional Normal Distribution

The outline of the two-dimensional normal distribution is as follows,

$$
\begin{aligned}
f(x,y) &= \frac{1}{2\pi\sigma_k^x\sigma_k^y} e^{\frac{-1}{2}-(\frac{x-\mu_k^x}{\sigma_k^x})^2+(\frac{y-\mu_k^y}{\sigma_k^y})^2} \\
\mu_k^x &= \mu_k^y = 0 \\
\sigma_k^x &= \sigma_{initE} + \sum_{k=2}^{L} w_k \\
\sigma_k^y &= \sigma_{initE} + \sum_{k=2}^{L} w_k
\end{aligned}
\tag{4.12}
$$

where μ_k^x and μ_k^y are mean and σ_k^x and σ_k^y are standard deviation each dimension. The initial σ is determined by the accuracy of the position obtained from the GPS.

The standard deviation σ_k increases for each step L by adding the process noise when the subject is indoors. The value of σ will be chosen from process noise. This represents that the accuracy in each position estimation indoors will decrease for each step. w_k is taken from where it can be calculated as the difference between the position estimated in time update and the real position. This method is naturally not applicable indoors since the accuracy of the GPS values

is poor. There is no correlation between the x and y dimensions. Therefore, the correlation function is considered to be 0.

Heat Map

A heat map can be generated using the two-dimensional normal distributions for each state. Non-traversed areas is set to blue color and the most trajected area is set to red with green in the middle. Possible corridors will be green or red. The method is as follows.

First, create a matrix with the same size as the map and initialize all the positions with zeros. Then, for each state, which corresponds to steps, add values from the two-dimensional normal distribution around the state's position with a radius of 2σ, the method for determining this radius is described in Section 7. The radius determines how much of each two-dimensional normal distribution that will be used.

After adding values from each state the positions that have high values in the matrix will have a color between red and green whereas the low values will be between blue and green. This heat map will contain noise as well. The noise is removed by removing all values that are under a threshold, this determination method is also described Section 7. In the heat map this means that all colors under the threshold will be changed to blue.

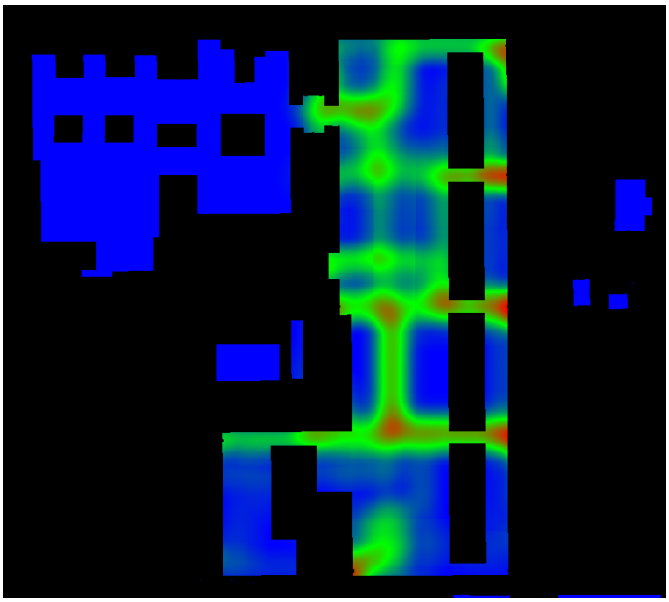

Figure 4.2: *Heat map*

4.3.2 Parametric Map

Our parametric map consists of nodes and paths that connect the nodes. Therefore, detecting the nodes and paths are required when generating a parametric map. In this thesis, nodes are entrances and junctions whereas the paths will be corridors.

4.3.3 Node Detection

The method for detecting nodes such as entrances and junctions is presented in this section. Both kinds of nodes are detected from the trajectories and then each node's exact location is detected by clustering.

Entrance Detection

The condition for when an entrance is detected and the error will be reset is as follows. When the subjects go from outdoor to indoor the color changes from black to white on the binary map. We decided that the first position with white color will be the entrance. To protect from errors the subjects also has to take several steps indoors after the entrance candidate in order for the candidate to be considered an entrance.

Entrances are only detected when the subjects transitions from outdoor to indoor. The reason is that the error in the estimation of the subject's position will be accurate in this instance but not when the subjects walks from indoor to outdoor.

Junction Detection

Junction detection is done by looking at gyroscope's behaviour. The Gyroscope sensor measures the speed at which the gyroscope is rotated around its axis. The unit is radians per second. That means that this gyroscope value can correct the values from the magnetic compass as described above. Because we assume that the phone is held still, the gyroscope's reliability is higher than the magnetic compass' due to nearby magnetic fields that disturb the sensors.

When the subject arrives at a junction and turns, there will be a spike in the gyroscope's output. By locating such spikes and saving the positions as junctions it is possible to detect junctions. To detect the spikes a threshold of 0.5 was used. A heat map similar to the one described for trajectories above in Section 4.3.1, was created but only for junctions and entrances. This heat map will show where junctions are, that is, positions where many subjects has turned. For this heat map, blue means nobody or few subjects turned here and the color gradually becomes red through green as more turns are concentrated in a small area. A junction heat map is shown in Figure 4.3.

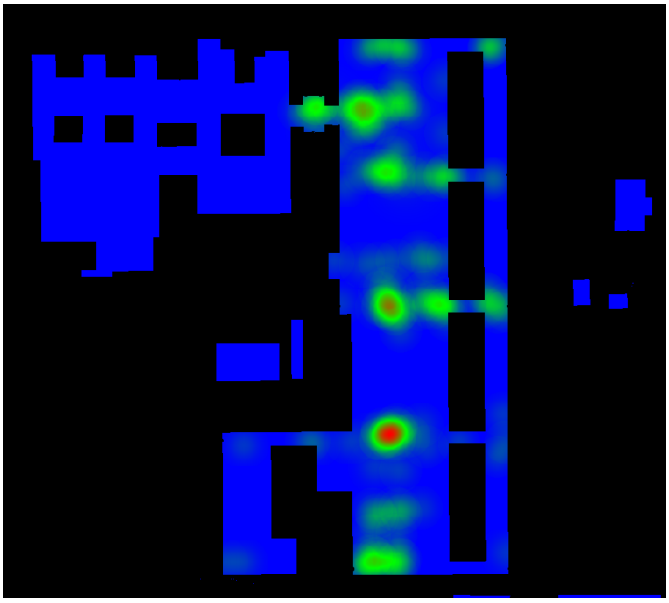

Figure 4.3: *Junction heat map*

After all trajectories has been used to detect nodes, the extraction of exactly one point for each node is required to determine the estimated node's position. To extract the point, we have decided to use a threshold of 0.1 to remove noise. As seen in Figure 4.3, each junction forms a cluster which we will call the node's cluster. After removing values with a faint green color to filter away noise, using the threshold described above, the hottest position in each cluster is chosen as the node's exact position. When a node is used to connect paths, the calculated position will be used. Clustering methods such as k-means cannot be used since it is not possible to know the number of nodes to detect.

4.3.4 Path Detection

We assume that path exist only between nodes; that means that paths are corridors and that corridors exist between an entrance and a junction or any combination of them. Therefore, corridors are generated by using the detected entrances and junctions extracted above.

The method used is to take all the turns for each trajectory and if adjacent turns, in the trajectory, are both inside two different junction clusters, a corridor will be created between those two junctions. Since there might be false positives and false negatives in the turn detection there are several checks for the corridors. The purpose of the checks are to remove false positives. First, no corridor is allowed to have a blue part in them. This can happen if a turn is not detected. The turn before and after the missed turn can then be connected if they are inside two junction clusters, creating a corridor where there should have been two. Also, the angle of the corridor is examined since we have restricted the angles of the

trajectory. A corridor is also not allowed to be outside the building since the scope for the automatic map generation is for indoor only.

5

Evaluation Experiment

In this chapter, we will describe the evaluation experiment of the method which is presented in Section 4. The experiment environment and data collection details are presented in the following sections.

5.1 Experiment Environment

The data collection was done in the B-building in Linkoping University. Figure 5.1 shows an indoor/outdoor map of the experiment place taken from Google Maps.

Figure 5.1: *Map of the B-building in Linkoping University*

5.2 Data Collection Applications

Data was collected using two Android applications called "Sensor Fusion app" together with "Quat2Orient". "Sensor Fusion app" was developed by Professor Fredrik Gustafsson to take and store log data from several sensors. "Quat2Orient" was used for fusing magnetometer and accelerometer sensor values to calculate the user's orientation. This application obtains the accuracy value of GPS and merge this data with the stored data from "Sensor Fusion app". The GPS accuracy is obtained at the same time as GPS values are logged in "Sensor Fusion app" and the calculation of orientation is done by an Android function. The basic calculation of the orientation from accelerometer and magnetometer sensor data is written in Section 3.

5.2.1 Devices and Sensors

The Android cellphones Samsung Galaxy S5 and Oneplus One were used to take sensor data in the experiments. GPS, Wi-fi, Gyroscope, Magnetometer, Accelerometer, Light, Pressure and Bluetooth sensor data were taken with the sampling frequency 100Hz. However, the Light, Pressure and Bluetooth sensor data was not used in this thesis.

5.2.2 Data Collection Method

We have collected multiple trajectories to simulate crowdsourcing. The three subjects walked outdoor to indoor; that is, the subjects obtained accurate GPS coordinates outside before going into the building. The accurate GPS positions are required to have a good estimation of the subject's position before it enters the building. Since the position estimation used indoors will not improve the positions accuracy, due to the summation of errors described above, it is essential to have an accurate estimation before the subject enters the building.

The subjects paths were decided to spread the trajectories throughout the building. However, the trajectories were not spread evenly since some corridors were used more than others because of the layout of the building.

While the Subjects walked they tried to hold the phone as horizontal as possible and take steps with the same step length. The phones were held in this position since only values from one of the gyro's axis were used. Before the data session started the phone's magnetometer was calibrated by moving the phone in an 8-shaped pattern.

6

Result

The result from Evaluation Experiment is presented in each step. The steps are Map Mining, Data Processing and Map Generation.

6.1 Map Mining

The generated binary map is shown in Figure 6.1.

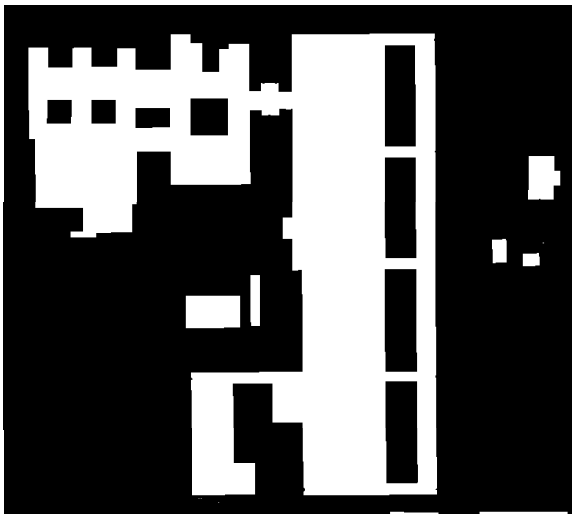

Figure 6.1: *Binary map*

As a result of this algorithm, the simple map was transformed into a binary

map where it is trivial to determine where buildings are located. This algorithm cannot be applied to a map that has a copious amount of metadata since we need to decide which color pixel to substitute and some metadata might share those colors. In a map with metadata there are some colors that are used both indoors and outdoors. For example, the street and building names are both written on the streets and on the buildings. This increases the complexity of generating an accurate binary map.

6.2 Data Processing

77 trajectories were taken in crowdsourcing way. However, 11 trajectories did not get any GPS data and therefore only 66 trajectories were used for the data processing. The subsections below shows the results in each step. Also, since the number of trajectories are high there are several trajectories that are the same but taken at different times. Figure 6.2 shows an example of how a subject walked in the trajectory.

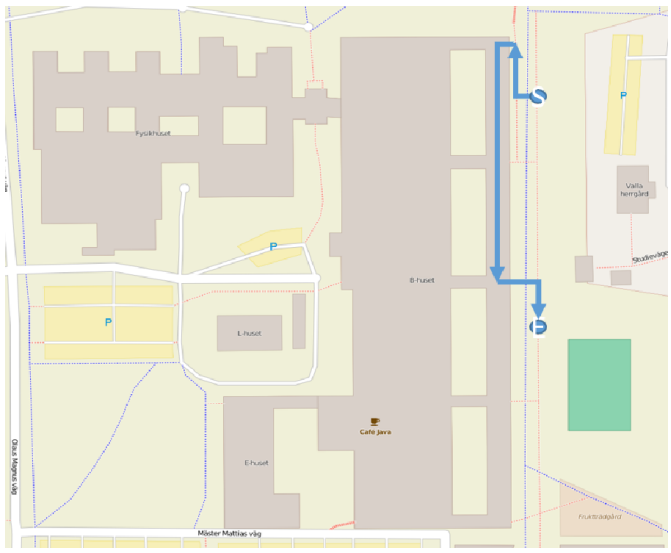

Figure 6.2: *Trajectory*

6.2.1 PDR

The PDR result is shown in Figure 6.3. The initial heading has been picked manually as well as the gyroscope's bias. The initial positions was taken from the first GPS value and shows how important it is for this system to have accurate initial conditions.

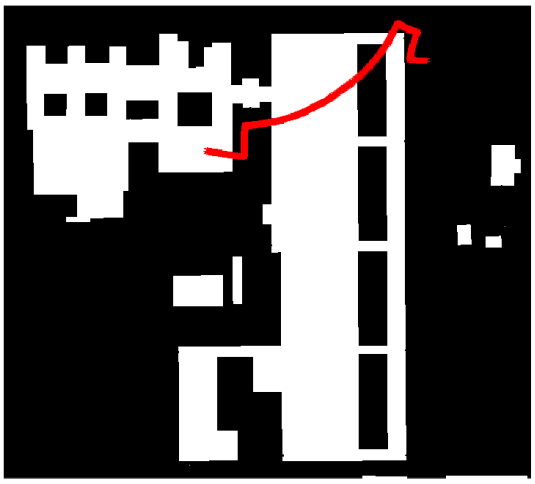

Figure 6.3: *PDR*

The bias and process noise has been saved in every step and it is easy to see that it is increasing: that is, the gyroscope alone is not reliable. Therefore, we consider the magnetic compass to obtain better results which are shown in the next section.

6.2.2 PDR+Magnetic Compass

The result of PDR is used with the magnetic compass is shown below,

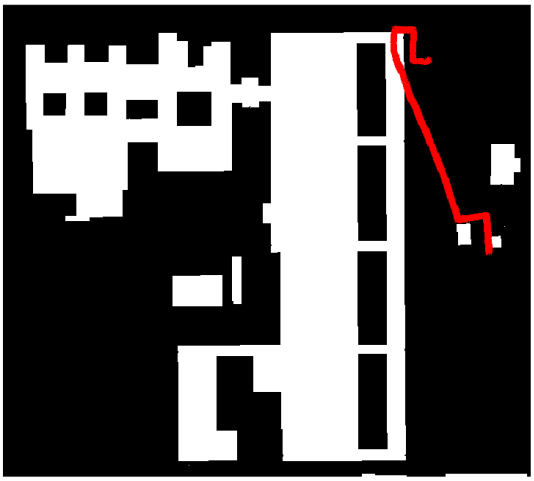

Figure 6.4: *PDR with magnetic compass*

To detect orientation by using both gyroscope and magnetic compass works much better than only suing PDR as in Figure 6.3. The bias error has been cancelled. However, the values from the magnetic compass is not reliable inside the buildings, and in this case also outside, in the presence of magnetic field indoors, therefore the trajectory got affected by those fields and the result is still inaccurate. Also, the process noise has been saved gradually to the end of the trajectory as well as the fact that the bias error was not perfectly cancelled.

6.2.3 Fusing PDR with GPS(EKF)

To improve the results above, EKF has been adopted to fuse PDR with GPS sensor data.

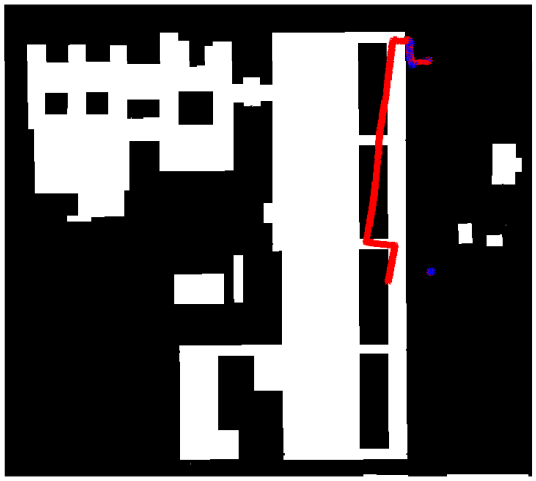

Figure 6.5: *Fusing PDR with GPS(EKF)*

The red line is how the positions are estimated by our algorithm and the blue lines are the positions received from the GPS. We assume that the GPS value is accurate enough since the experiment field has a high sky factor and the accuracy of the GPS is set to be within 5 meters. In this case, the filtered result looks good where we obtained GPS values outdoors since it is following the GPS values and smoothed by the filter. However, the result from indoors is not as promising since some bias and process noise still has been saved, not much as in PDR, although the trajectory shape is promising. If we are indoors, we will not consider GPS values in this algorithm. Therefore the GPS result is sparse in the result. The next EKF contains conditions to improve the result.

6.2.4 EKF with Conditions

The conditions described in Section 4.2.4 were applied with EKF and the result is shown as follows.

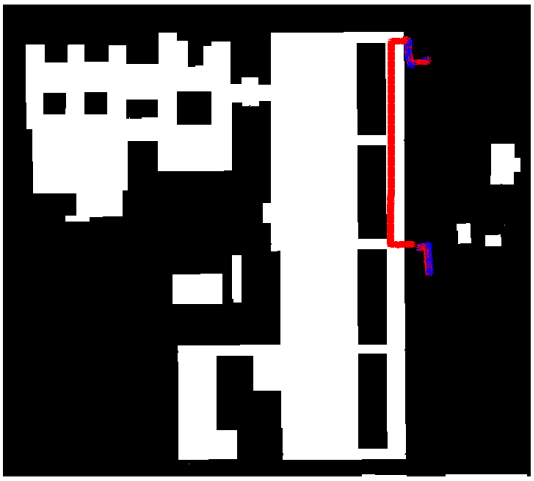

Figure 6.6: *Fusing PDR with GPS(EKF) with some conditions*

Using the structure's features can be applied to obtain a better result. With all of these additions we created a filtered trajectory that was accurate enough for the purposes of this thesis.

6.3 Map Generation

The result of Map Generation is presented as follows. Two maps, a non-parametric map and a parametric map, was generated together with node and path detection. The results for the parametric map is presented in several subsections after Section 6.3.1.

6.3.1 Non-Parametric Map

The non-parametric map is shown in Figure 6.7. All states are plotted as described in Section 4.3.1.

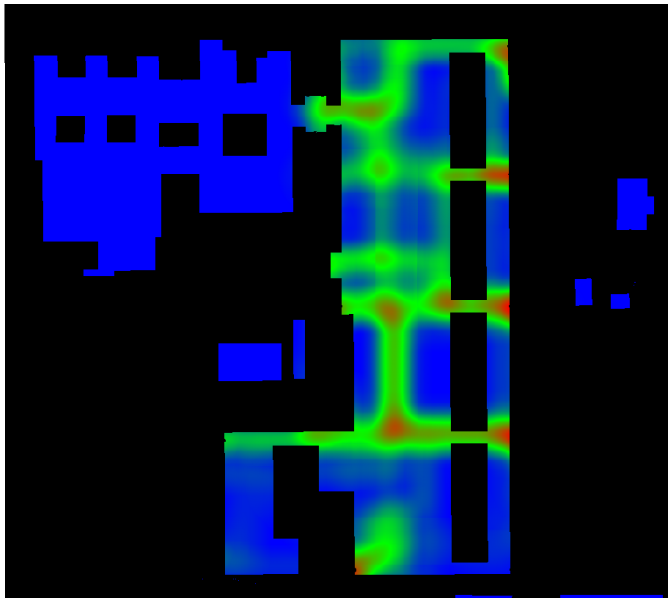

Figure 6.7: *Non-parametric map*

The result of the non-parametric map is very promising. Almost all walked corridors are visible. The missing corridors are in the bottom left corner and between the four entrances on the right side. The reason is that the subjects walked through those corridors very few times compared with the others. The large green corridor at the middle bottom of the building should consist of four corridors in a square, but their corridors have merged into one.

6.3.2 Node Detection

The result of Node detection is shown following in the subsection. For the calculations below 2σ and threshold 0.1 was used as parameters. The reason those values were chosen will be discussed in the following chapter.

Node Detection in Each Trajectory

The result below is created from the trajectories and the non-parametric map above. Figure 6.8 shows the detected entrance and junctions from a trajectory. The entrances were detected by examining the gyroscope behaviour and the binary map.

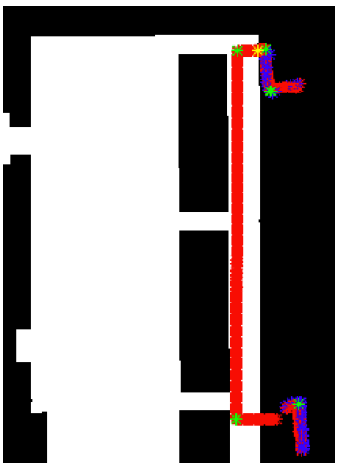

Figure 6.8: *Entrance and junction detection from a trajectory*

The yellow star in the Figure 6.8 is the detected entrance and the green stars are the detected turns. We do not consider turns outside, therefore the detected turns outside will be discarded. Each entrance and junction, except those outside, will be used when creating the entrance and junction heat map.

Entrance Detection

Figure 6.9 detected entrances that were detected as described in Section 4.

Figure 6.9: *Entrance detection heat map*

The accuracy of the detected entrances is very high. All of the entrances that were used frequently has successfully been detected and with good accuracy. This accuracy is important since the heading is fixed when the subject walks into a building. An error in the entrance detection could therefore result in large errors for the rest of the trajectory. For future work the position could also be fixed, to the entrance's position, when the subjects enters.

Junction Detection

The heat map of all detected junctions is shown in Figure 6.10.

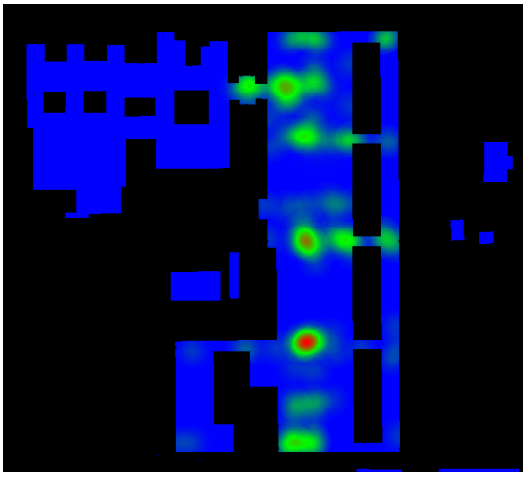

Figure 6.10: *Junction detection heat map*

This heat map was used when detecting which junction that the turns in the trajectories belong to. Most junctions were detected but some are very faint and was later removed since they were under the threshold.

Node Detection

We merged entrances and junctions into one to generate the paths. The heat maps above, Figure 6.9 and Figure 6.10, have been filtered with the threshold 0.1 and it shows the position of the nodes. The result is shown in Figure 6.11.

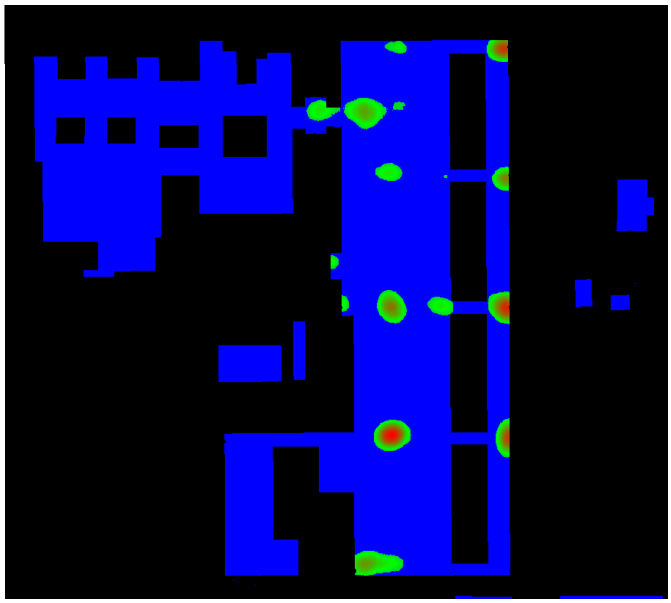

Figure 6.11: *Junction detection binary map*

From this heat map, the nodes will be chosen as the hottest point in each cluster. A red star in Figure 6.12 means the extracted position for each cluster. Comparing with an indoor map from Google Maps shown in Figure 5.1, you can see that the 15 detected junctions were very accurately detected in terms of position. However, the total number of junctions that should have been detected is 38. The number of junctions detected and position accuracy strongly depends on the cutoff σ and the threshold used for removing noise in the heat maps. This determination of threshold and cutoff σ is discussed in the following chapter.

Figure 6.12: *Detected structures with threshold 0.1*

However, the detection of junctions and entrances work very well among all clusters. A reason that relatively few junctions were detected is because junctions are detected when subjects turn in them, not when simply walking through them. The red triangle marks such a junction. It is a junction where the subjects has mostly walked straight through but has not turned in. That is why it is not detected. These junctions become false negatives and are difficult to detect since they are very similar to corridors. This means that there is a risk that junctions where subjects turn very seldom in becomes false negatives like the one marked on the map.

6.3.3 Path Detection

Using the method described in Section 4 paths were generated. It shows the detected corridors as red lines on the map. Figure 6.13 shows the parametric map that was generated with 2σ and a threshold of 0.1.

Figure 6.13: *Parametric map*

Not all corridors are detected and one of the corridors is not straight, but in general the corridors that were walked most in were detected.

7

Discussion

7.1 Map Mining

The result is very solid, the binary map is accurate and can successfully be used to determine whether a participant is indoor or outdoor. The proposed method for generating a binary map was tested on other buildings as well and with the same level of success. The main problem that was encountered first was before we removed the metadata. Some of the metadata shared inside and outside colors which made it difficult to create an accurate binary map.

7.2 Data Processing

The PDR result does not consider GPS outdoors and therefore it only depends on sensor values from the smartphone and no received signals; that is to say, it strongly depends on how the subjects behave e.g. how the smartphone is held. For instance, if someone walks towards the subject, the subject needs to walk around them and then the sensors obtain not-expected values. That tends to lead to extra step and turn detections.

In indoor environments, subjects hold their phone as stable as possible that means the gyroscopes obtain reliable data compared with the magnetic compass due to magnetic fields issues. Furthermore, the detected angle change from the gyroscope should be the same as the change in the magnetic compass. Taking this fact into account in the filtering can improve the filter and therefore improve the accuracy of the estimated trajectory.

The author of [9] created a magnetic deviation map to increase the accuracy of trajectories. However, in our approach, we have a threshold for the magnetic field strength that is used in order to use only accurate values from the sensor. The result below with accurate magnetic compass values does show large improvements compared with PDR by comparing with the result from Figure 7.1.

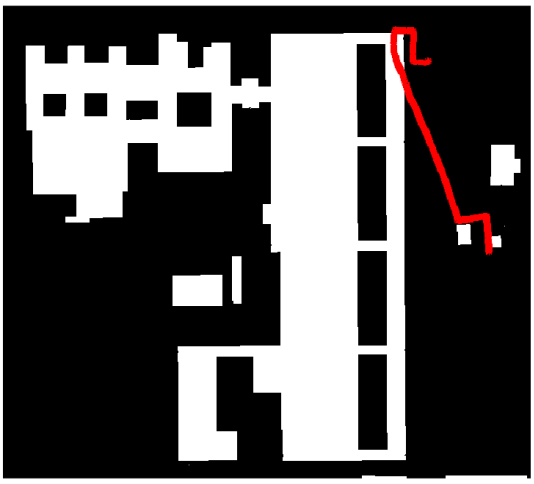

Figure 7.1: *PDR with magnetic compass*

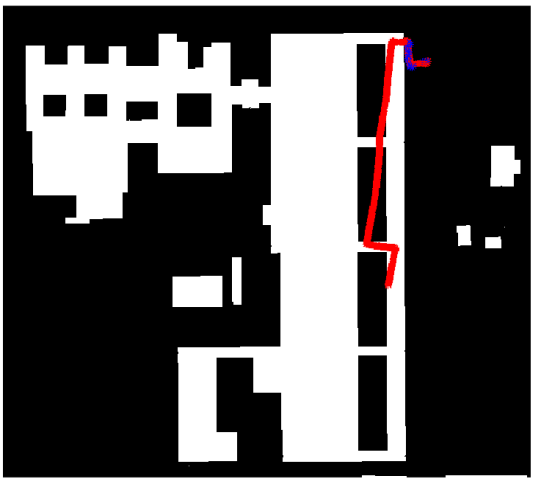

Figure 7.2: *PDR with accurate magnetic compass*

Although adding the magnetic compass to the PDR, which is the same as EKF for indoors, the error has been saved gradually. Especially indoors due to the magnetic field issue as mentioned above. Therefore, considering the structure's features as a method of resetting the heading in entrances yields a better filter. However, this algorithm will naturally not work with structures designed differently since indoor structures such as entrances and junctions will not have the same shape e.g. curved turns and non-perpendicular entrances. To prevent it, the authors of [10] deployed vector maps that contain wall and door locations as well as accessible and inaccessible areas inside buildings to filter the PDR. We also had accessible and inaccessible areas in the building used in the experiment in our research. Without entrance and walls information vectors, we did not succeed to fit the trajectories inside without conditions. Except resetting errors when passing through an entrance, we have not used the topology of the map to improve the positioning. Detecting entrances and junctions and the generation of corridors enables such improvements.

The results from EKF and EKF with conditions in the algorithms above strongly depends on the GPS coordinate outdoors. If you take this trajectory in an urban canyon, it is difficult to obtain accurate GPS values. Since the GPS values are essential for detecting entrances and detecting entrances are essential for getting a good initial setting for the indoor trajectory, the proposed method is not suitable in such settings.

However, this experiment field has a high sky factor as mentioned above, only adding a threshold for accuracy of GPS is enough. Regarding handovers, the handover's convergence time can be reduced since the positions of the entrances can be detected faster and after several subjects have entered in each entrance the handover can be predicted before the subject enters the building.

7.3 Map Generation

The result of each step of the map generation is discussed separately below.

7.3.1 Non-Parametric Map

A large amount of green color can be seen in the result in Figure 6.7. Subjects walked around in almost all of the corridors in the experiment building. However, due to reconstruction and some doors being locked not all of them have been walked in. Basically, this non-parametric map(heat map) shows where and how much subjects walked in different places on the map. This result affects the following structure detection and generation of the parametric map since the heat map's color shows the probability for a corridor to exist there.

A problem with the non-parametric map is that the corridors are very wide. The problem comes from that σ is large due to the process noise. A large σ increases the flatness of the two-dimensional normal distribution and that is the cause of the wide corridors. The impact of the large σ can be seen in the middle of the bottom where four corridors in square has become just one corridor.

A smaller σ would have yielded a better result in this case. However, reducing σ is no trivial task. The initial value is taken from the accuracy of the GPS, which means that the accuracy of the positions from the GPS has to be increased in this case. Provided that the initial σ could be decreased, the impact on the non-parametric map would be significant. Naturally, this applies to both of the dimensions of σ.

7.3.2 Parametric Map

This section is divided into three sections. First, we will discuss how we decided σ and threshold to generate heat maps and for the node and path detection.

Deciding σ and Threshold

To achieve CEP-67%, we considered which multiple of the standard deviation σ for the two-dimensional normal distributions should be used as the radius in the heat map generation. The radius decided how much of the two-dimensional normal distribution that will be used. From the centre of the position, all positions inside the radius around the centre will be increased with the value from the two-dimensional normal distribution. The factor for σ was chosen from 1, 2 and 3. In general, a radius of 1σ means that 68.27% of the distribution will be used, 2σ means 95.45% and 3σ means 99.73% statistically. The relationship between the radius and σ is that the radius will be two times the factor times σ.

The threshold for removing noise from the heat maps was chosen from 0.0, 0.05, 0.1 and 0.15. We evaluated all combinations of σ and thresholds by creating a Cumulative Distribution Function(CDF) and counting the number of detected junctions. The requirement for this research was that CEP-67% should be fulfilled and the Error distance should be less or equal to 2.89 m. The requirement on the Error distance was determined from the shortest width of the corridors in the experiment building. The table below shows the evaluation result. It shows the number of detected junctions and if any CDF requirement was fulfilled for each combination of σ and threshold.

	0.0		0.05		0.1		0.15	
±1σ		26		26	50%(2.76m)	15	50%(2.70m)	6
±2σ	50%(2.25m)	8	50%(2.70m)	16	67%(2.76m)	15	50%(2.70m)	8
±3σ		4	50%(2.55m)	17	67%(2.76m)	13	50%(2.70m)	8

Figure 7.3: *Performance for different combinations of σ and threshold*

The table above shows that only 2σ with threshold 0.1 and 3σ with threshold 0.1 fulfilled the requirements. The result with 2σ with threshold 0.1 has detected the most number of junctions. Therefore, we have decided 2σ with threshold 0.1 has the best performance and it has been used in the generation of the figures above.

Node Detection

As discussed in the Section 4, the trajectory and structure detection in this algorithm strongly depends on the accuracy of the GPS coordinates outside, the subject's behaviour and the experiment's conditions. If subject's behaviour was unstable, for example that the subject did not hold the phone steady and horizontal, the system tends to detect false positive turns.

In addition, if the values obtained from the GPS are inaccurate, the entrance detection will be very inaccurate. The accuracy of the whole trajectory depends on the accuracy of the position detected when the subject enters. Therefore, having poor accuracy when the user passes the entrance will yield a inaccurate trajectory. The cause to this problem is that, without GPS values, the position cannot be corrected through a measurement update. The heading can still be corrected but it will not correct the position. However, this problem can be mitigated by the fusion of the gyroscope and magnetic compass since it will determine in which direction the user is walking and thereby reduce the impact of inaccurate GPS coordinates.

Once the user entered the building three things were especially important. First, the subjects need to hold the phone as stable as possible in order for the gyroscope to obtain reliable data. Second, the magnetic compass needs to be calibrated. Finally, strong magnetic fields should be avoided. The heading can still be corrected nicely even if the magnetic compass obtains noisy data. Since the gyroscope measure radians per second the gyroscopes output should be equal to the change in the magnetic compass. By exploiting this fact it is possible to filter away sudden changes in the magnetic compass that was not detected by the gyroscope. Therefore, the trajectories shape and detection of junctions and turns can be improved by exploiting this fact.

In the merged structure detection, we conclude that a radius of 2σ with the threshold 0.1 yields the best result. This was determined by using CDF-67% for the evaluation. The impact of the radius is that a small radius means that only positions that are close to the estimation will be increased on the heat map. Increasing the radius will increase the noise added to the map while decreasing the radius increases the risk that the real position is not inside the radius.

The threshold has a similar function. It determines how much of the data that will be considered noise. A high threshold means that only the hottest positions on the map will be left after the values are removed. A suitable value for the threshold strongly depends on the standard deviation of the two-dimensional normal distributions. If the standard deviation is small the distribution will be flat. This means that the positions impact will be spread over a large area of the heat maps. A too small threshold would in this situation leaves lots of noise from the flat distribution which causes two major problems. First, the non-parametric map will be inaccurate. The corridors will get wider and there will be less blue color and instead more faint green color. Second and more important, the number of false positives in the corridor detection will likely increase. One part of the removal of false positives was that the whole path of the corridor should not be blue in any position. The faint green is not blue so this detection of false positives

would not work well due to the flatness of the distribution together with the low threshold. With a very flat two-dimensional normal distribution and a very small threshold the whole building would be non-blue.

Path Detection

Figure 7.4: *Parametric map*

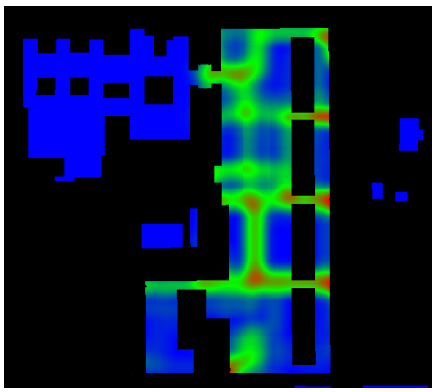

Figure 7.5: *Non-parametric map*

Figure 7.4 and 7.5 shows both types of generated maps in this thesis. The red corridors in the parametric map are the detected corridors while the other colors represents missing corridors that will be discussed here.

First, focus on the two green corridors. The reason that these two corridors are not detected is due to the faintness of the green color in the non-parametric map. Very few subjects have walked those paths and as a result the non-parametric map will be blue in the middle of the corridor. Since one of the conditions is that no blue should be in the corridor those two corridors are not chosen.

Second, note that the blue corridor is not detected even though the area is green in the non-parametric map. The cause of this missing corridor is that many subjects exited the building in the left end of the corridor. Since exits are not detected, and therefore not added as a junction to the trajectory, the last corridor that the subject walked through will not be detected in that trajectory. When the subjects entered through the left side of the corridor and turned where the blue and red meet, the corridor can be detected. It will only be detected if the detected turn is inside the junction's cluster which did not happen in this experiment. In this case, the corridor's color is also deceptively strong. Some of that comes from trajectories where the subject walked straight through the blue and red junction, since there was no turn the red and blue junction will not be detected.

Last, compare the area in the dashed orange rectangle with the non-parametric map and the junction heat map in Figure 7.6.

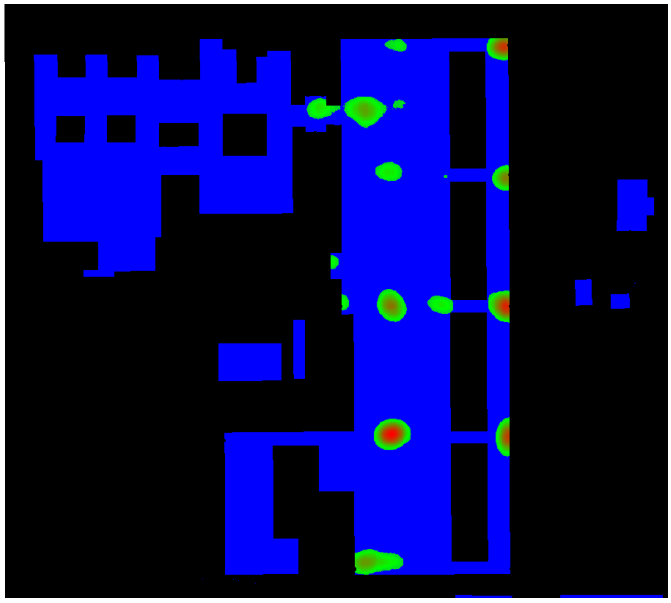

Figure 7.6: *Filtered junction heat map*

There should actually be four corridors shaped as a square inside that box. Due to the large σ the corridors are only detected as one large green cluster. A similar result is seen in the junction heat map. Instead of four junctions only one will be detected since all of them are in the same cluster (the other two were removed in the filter). The trajectories that walked to the blue and red junction in Figure 7.4 walked through one of the junctions that were filtered away. Another problem in this particular case is that the subjects walked too few times in these corridors to make them visible in the non-parametric map.

8

Conclusion

In this thesis, we proposed a method for automatically generating indoor maps using crowdsourcing with smartphones. The requirement of the proposed method was that the node detection would fulfil CEP-67% with the error distance $2.89m$. The data was taken from an experiment that simulated crowdsourcing with smartphones. To achieve this, we split the method into three large sections.

First, we did Map Mining. This was important for this thesis since it enabled us to easily distinguish between outdoors and indoors. To do that, we used an original map that was taken from an existing map service, OSM in this thesis. Then we can tell whether you are indoor or outdoor in a simple way by creating a binary map where white means indoor and black means outdoor. The map was created by substituting colors.

Second, we aimed to create accurate trajectories from the data from the participants phones. The trajectories could then be applied to detect nodes such as entrances and junctions. PDR together with some added conditions taken from a building's general features were used. The condition was that we assumed that the building's turns are always a multiple of 90 degrees. In addition, we also added the magnetic field strength as a condition. The reason is that numerous machines which has strong magnetic field are inside general building. That affect to the magnetic sensor data. However, gyroscope sensor can aid the error since we assume that holding smartphone still vertical to you and horizontal towards the ground.

Last, in the Map Generation, we generated a non-parametric map and a parametric map by detecting nodes and paths. The non-parametric map was created as a Heat map and it was created by adding the users' positions to the map using a two-dimensional normal distribution that used the positions' estimated accuracies to determine the shape of the two-dimensional normal distribution. The non-parametric map can be created and updated in real time. Node detection was

done by clustering all detected junctions and entrances. The result of the node detection was a heat map which showed the nodes' most likely positions. From each cluster the hottest point was chosen as that node's position. Corridors were created between the nodes where the participants had walked and the detected corridors formed the parametric map.

8.1 Suggested Future Work

There are several approaches that can be done to improve the automatic map generation.

The generated map could be used to extend another map. For example, the generated map could be uploaded to OSM.

In data processing, more building features can be considered. For example, you can give threshold to the trajectory length in each dimension by considering the building width and depth.

From the point of positioning, Wi-Fi can be applied since Wi-Fi is the most common positioning tool indoors and fingerprinting could be done automatically and APs could be located in a similar way as junctions' positions are created by a heat map. Radio antennas can be considered as well since Ericsson has released a new antenna for indoors and it is therefore possible to obtain accurate positions indoors.

To decrease the σ a bit, using the GPS accuracy of the last accurate position before the participant enters a building could be used as the initial σ instead of taking the max allowed accuracy.

We started saving statistics for how often which corridors were used to be able to calculate probabilities for how persons would walk. Unfortunately, due to lack of time we could not finish it. Finishing this part could yield many new use cases. For example, helping shopkeepers analyse consumers walking patterns to improve the shopping experience.

When a participant exited a building, the exit could have been matched to one of the existing entrances. This would improve the corridor detection since more the last corridors that the user walked through would have been detected as well.

The used structure feature where only 90 degree turns are considered was applied, however, other degree turns can be applied as well as future work.

Bibliography

[1] Lili Cao and John Krumm. From gps traces to a routable road map. *17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 3–12, 2009. Cited on page 6.

[2] Extefania Munoz Diaz and Ana Luz Mendiguchia Gonzalez. Step detector and step length estimator for an inertial pocket navigation system. *Indoor Positioning and Indoor Navigation (IPIN)*, 2014. Cited on pages 2 and 6.

[3] Eric Foxlin. Intertial head-tracker sensor fusion by a complementary separate-bias kalman filter. *Virtual Reality Annual International Symposium 1996., Proceeding of the IEEE 1996*, pages 185–194, 267, 1996. Not cited.

[4] David Gundlegård. Estimation and accuracy. 2014. URL `http://webstaff.itn.liu.se/~davgu/tnk106/seminar_8.pdf`. Not cited.

[5] Fredrik Gustafsson. *Statistical Sensor Fusion*. Studentlitteratur, 2:1 edition, 2012. Not cited.

[6] René Hansen, Rico Wind, Christian S. Jensen, and Bent Thomsen. Seamless indoor/outdoor positioning handover for location-based services in streamspin. *10th Inthernational Conference on Mobile Data Management: Systems, Services and Middleware*, pages 267–272, 2009. Cited on page 6.

[7] A.R. Jiménez, F. Seco, C. Prieto, and J. Guevara. A comparison of pedestrian dead-reckoning algorithms using a low-cost mems imu. *6th IEEE International Symposium on Intelligent Signal Processing*, pages 37–42, 2009. Cited on pages 1, 2, and 6.

[8] Noriaki Kakiuchi and Shunsuke Kamijo. Pedestrian dead reckoning for mobile phones through walking and running mode recognition. *16th International IEEE Annual Conference on Intelligent Transportation Systems (ITSC 2013)*, pages 261–267, 2013. Not cited.

[9] Noriaki Kakiuchi and Shunsuke Kamijo. Pedestrian dead reckoning for mobile phones using magnetic diviation map. *IEICE TRANSACTIONS on Fun-*

damentals of Electronics, Communications and Computer Sciences, (1):313–322, 2015. Cited on pages 2, 6, 16, and 44.

[10] P. Kemppi, T. Rautiainen, V. Ranki, F. Belloni, and J. Pajunen. Hybrid positioning system combining anglebased localization, pedestrian dead reckoning and map filtering. *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 162–270, 2010. Cited on pages 1, 6, and 45.

[11] Tsuyoshi Kinoshita and Kazunori Miyata. Relative positioning system for pedestrian using smart-phone sensor and bluetooth. *IPSJ SIG technical report*, (35):181–188, 2012. Not cited.

[12] Kay Kitazawa, Yusuke Konishi, and Ryosuke Shibasaki. A method of map matching for personal positioning systems. *The 21st Asian Conference on Remote Sensing*, 2000. URL `http://www.casa.ucl.ac.uk/kay/paper/acrs001205_e.pdf`. Not cited.

[13] Karl Kluge. Extracting road curvature and orientation from image edge points without perceptual grouping in to features. *Intelligent Vehicles '94 Symposium*, pages 109–114, 1994. Cited on page 7.

[14] Mille Millnerta, Fredrik Gustafsson, and Lennart Ljung. *Signal Processing*. Studentlitteratur, 1:3 edition, 2011. Not cited.

[15] Ling Pei, Ruizhi Chen, Yuewei Chen, Helena Leppäkoski, and Arto Perttula. Indoor/outdoor seamless positioning technologies integrated on smart phone. *1st International Conference on Advances in Satellite and Space Communications*, pages 141–145, 2009. Not cited.

[16] Toshihiro Shingai, Kenji Nakayama, Akihiro Hirano, Hideki Tanino, Kiyoshi Tsurumi, and Hiroshige Asada. On gyro-gain estimation in dead reckoning navigation system using kalman filter. *22nd SIP SYMPOSIUM*, pages 395–400, 2007. Not cited.

[17] Mikio Shishido. Quaternion calculation by matlab. *MSS Technical Report*, 19:44–49, 2008. Not cited.

[18] Ulrich Steinhoff and Bernt Schiele. Dead reckoning from the pocket - an experimental study. *IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 162–270, 2010. Cited on page 6.

[19] Ross Grote Stirling. Development of a pedestrian navigation system using shoe mounted sensors. *Master Thesis*, 2004. URL `http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.72.4381&rep=rep1&type=pdf`. Cited on page 6.

[20] Atsushi Watanabe, Shigeru Bando, Kazuhiro Shinada, and Shinichi Yuta. Road-following-based navigation in park and pedestrian street with finding intersection and orientation detection. *Duckburg Journal of Commerce*, 30(3):271–279, 2012. Cited on page 7.

[21] Stewart Worrall and Eduardo Nebot. Automated process for generating digitised maps through gps data compression. *Australasian Conference on Robotics and Automation (ACRA)*, pages 13–18, 2007. Cited on page 6.

[22] Kaiqing Zhabg, Hong Hu, Wenhan Dai, Yuan Shen, and Moe Z. Win. Indoor localization algolithm for smartphones. URL `http://arxiv.org/pdf/1503.07628v1.pdf`. Cited on page 7.