Petra Hafner, Dipl.-Ing.

Development of a Positioning Filter for the Smartphone-based Navigation of Visually Impaired People

Doctoral Thesis

to achieve the university degree of
Doktor der technischen Wissenschaften
submitted to

Graz University of Technology

Supervisor
Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Manfred Wieser
Institute of Geodesy, TU Graz

Graz, August 2015

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present doctoral thesis.

Date

Signature

Acknowledgements

My sincere gratitude goes to my supervisor Manfred Wieser, who provided me the opportunity to work on this thesis and gave me inspiring hints. Without his support it would have not been possible to conduct this research.

Besides my supervisor, I would like to thank Betrand Merminod for his valuable comments during the final phase of my work, which helped a lot to improve this thesis.

Moreover, I want to thank my colleagues at the Institute of Geodesy, Working Group Navigation, for their fruitful discussions and with whom I had good times in particular during the daily lunch and coffee breaks.

Special thanks go to Thomas Moder, who supported me in every thesis-related concern and was furthermore always a friend with an open ear. I would like to say thank you to Karin Wisiol for her assistance in the final phase of my research.

Furthermore, I have to remark, it was a pleasure to work together with Alice Geiger, who gave me an insight into the life as visually impaired person.

I want to express my thankfulness to my family and friends, who supported me in pursuing my goals, encouraged me when it was necessary, and showed me how life should be besides the work.

But my biggest thanks goes to Daniel, for all his love and support.

Abstract

According to the Federal Ministry of Labor, Social Affairs, and Consumer Protection, approximately 3.9% of the Austrian population is suffering from permanent visual impairments. These people are faced with problems, others do not even think of. One example is the daily navigation through the urban environment, where traffic junctions and busy roads involve many risks and dangers if one of the sensory organs is impaired. Therefore, especially blind people urgently need accurate and reliable navigation systems to facilitate their daily routes and increase their safety.

Navigation in general comprises the three tasks positioning, routing, and guidance, whereby the routing and guidance performance strongly depends on the position accuracy. Therefore, the goal of the thesis is the development of a positioning tool which reliably provides accurate coordinates in in- and outdoor environments based on the built-in sensors of a smartphone. The core element of the positioning tool is a Rao-Blackwell Particle Filter (RBPF), which combines relative and absolute position information provided by positioning techniques like GNSS single point positioning with code measurements, RSSI-based WLAN fingerprinting, Pedestrian Dead Reckoning (PDR) with inertial data, and barometric height estimation. The position estimation is further supported by spatial constraints, which can be derived by integrating maps and a Pedestrian Activity Classification (PAC).

In outdoor environments, a special user-tailored map, which represents path preferences of blind persons, restricts the possible state space. Since the visual impairment leads to a different, more predictable movement, location probabilities can be defined. For example, blind persons most likely walk on pavements, cross the streets at crosswalks only, and follow tactile ground surface indicators. This knowledge can be used for the creation of a probability map, in which regions where the probability of the person's presence is higher are marked differently in contrast to regions of lower probability. Unfortunately, in indoor environments the movement of blind persons cannot algorithmically constrained in the same way due to the small-scaled infrastructure and the very individual navigation habits of blind pedestrians within buildings. Instead, an occupancy grid map representing the building layout and path verifications are integrated within the position estimation. Indoors, motion-related activities like going upstairs or downstairs implicitly include information that the pedestrian has to be located at a stairway. Hence, a PAC is employed within the RBPF to constrain the possible state space.

Due to the high working effort for the creation of the user-tailored map, a restricted test area had to be defined for the investigations. To avoid this for future applications, a Simultaneous Localization and Mapping (SLAM) approach is developed and investigated to make the RBPF applicable everywhere. The automatic map generation requires a more precise position than the GPS code solution which only provides accuracies in the range of some meters. For this reason, GNSS Precise Point Positioning (PPP) with accuracies in the decimeter level is employed instead. Since only the full operational capability of Galileo and COMPASS will provide enough satellites in urban canyons, the SLAM approach is treated in a simulation environment.

As a result of the developed RBPF, the position estimation in outdoor environments can be improved by about 47% to 69%, while the performance gain in indoor environments is in the range of 40% to 60%. Indoors, the accuracy of the across as well as the along track position component is within the sensing range of a blindman's stick. Outdoors, the achieved accuracy allows a localization at the correct pavement. Furthermore, the evaluation with simulated data reveals that SLAM provides the optimal framework for the derivation of location probabilities if a PPP solution is continuously provided.

Kurzfassung

Gemäß dem Bundesministerium für Arbeit, Soziales und Konsumentenschutz leiden in etwa 3.9% der österreichischen Bevölkerung an einer Sehbehinderung. Viele alltägliche Aufgaben können von dieser Gesellschaftsgruppe nur sehr schwer gemeistert werden. Ein Beispiel dafür ist die Navigation durch städtische Umgebungen, in der für sehbeeinträchtige Personen viele Gefahren lauern. Ein zuverlässiges, genaues Navigationssystem würde das Leben blinder Personen immens erleichtern und ihre Sicherheit im Alltag und vor allem im städtischen Verkehr erhöhen.

Die Navigation an sich besteht aus drei Komponenten, der Positionierung, dem Routing und der Zielführung. Je genauer die Positionierung ist, desto höher ist die Qualität der Routenberechnung und die entsprechende Zielführung. Das Ziel der Dissertation ist die Entwicklung eines Positionierungstools auf Basis der Sensoren eines Smartphones, welches sowohl in als auch außerhalb von Gebäuden Positionen mit hoher Genauigkeit und Zuverlässigkeit liefert. Das Kernstück des Positionierungstools, ein Rao-Blackwell Partikel Filter (RBPF), kombiniert dafür relative und absolute Positionsdaten. Diese Daten stammen aus einer Code-basierten GNSS Einzelpunktbestimmung, einem WLAN-Fingerprinting auf Basis von RSSI-Messungen, einem inertialen *Pedestrian Dead Reckoning* (PDR) und einer barometrischen Höhenbestimmung. Die Positionsbestimmung wird außerdem durch Karten und Informationen aus einer *Pedestrian Activity Classification* (PAC) räumlich eingeschränkt.

Eine blindenspezifische Karte, welche das Gehverhalten blinder Personen repräsentiert, wird außerhalb von Gebäuden zur Einschränkung des möglichen Zustandsraumes verwendet. Das ist daher möglich, da das Gehverhalten sehbeeinträchtigter Leute prädizierbarer ist als das nicht-blinder Personen, zum Beispiel, blinde Personen gehen bevorzugt auf Gehsteigen, kreuzen die Straßen nur über Zebrastreifen und folgen taktilen Leitstreifen. Dieses Wissen kann dazu verwendet werden, um räumliche Aufenthaltswahrscheinlichkeiten abzuleiten. Aufgrund der kleinräumigen Infrastruktur und des sehr individuellen Navigationsverhaltens von blinden Personen innerhalb von Gebäuden können solche Wahrscheinlichkeiten nicht für den Innenraum definiert werden. Stattdessen tragen sogenannte *Occupancy Grid Maps* und Pfadüberprüfungen zur Positionsbestimmung in Gebäuden bei. Da aktivitätsbezogene Informationen

implizit ortsrelevante Daten beinhalten (die Aktivität *Treppensteigen* reduziert den möglichen Aufenthaltsort auf eine Treppe), wird außerdem die im PAC detektierte Aktivität in die Positionsschätzung integriert.

Aufgrund des hohen Arbeitsaufwandes, der für die Erstellung der blindenspezifischen Karte notwendig ist, wird ein eingeschränktes Testgebiet für die Untersuchungen definiert. Um in Zukunft den entwickelten RBPF auch in anderen Gebieten einsetzen zu können, wird innerhalb der Dissertation ein *Simultaneous Localization and Mapping* (SLAM) Ansatz untersucht, um die Kartengenerierung zu umgehen. Die automatische Kartenerstellung erfordert jedoch eine GNSS-Positionslösung mit Genauigkeiten im Dezimeterbereich. Daher wird im Rahmen der SLAM-Methode *Precise Point Positioning* (PPP) anstatt der code-basierten Einzelpunktbestimmung verwendet. Da eine durchgehende PPP-Positionierung im städtischen Bereich erst bei voller Operationalität von Galileo und COMPASS möglich sein wird, wird der SLAM-Ansatz innerhalb einer Simulationsumgebung entwickelt.

Durch den Einsatz des entwickelten RBPF kann die Positionierung außerhalb von Gebäuden um 47% bis 69% und innerhalb um 40% bis 60% verbessert werden. In Gebäuden liegt die Positionsgenauigkeit somit im Tastbereich der blinden Person, während im Freien die Person am richtigen Gehsteig lokalisiert werden kann. Die Evaluierung auf Basis von simulierten Daten zeigt, dass die entwickelte SLAM-Methode das optimale Werkzeug für eine automatische Kartengenerierung ist.

Acronyms and abbreviations

AoA	Angle of Arrival
APOS	Austrian Positioning Service
AU	Activity Unit
CDF	Cumulative Distribution Function
CPU	Central Processing Unit
DBN	Dynamic Bayes Network
DGPS	Differential GPS
DOP	Dilution of Precision
ECEF	Earth Centered Earth Fixed
EGNOS	European Geostationary Navigation Overlay Service
EKF	Extended Kalman Filter
EPOSA	Echtzeit Positionierung Austria
ESA	European Space Agency
FAF	Floor Attenuation Factor
FOC	Full Operational Capability
GAGAN	GPS Aided GEO Augmented Navigation System
GEO	Geostationary Earth Orbit
GLONASS	Globalnaja Navigatsionnaja Sputnikovaja Sistema
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GUI	Graphical User Interface
HF	Histogram Filter
IGS	International GNSS Service
IGSO	Inclined Geosynchronous Satellite Orbits
IMU	Inertial Measurement Unit
INAS	Institute of Navigation, Graz University of Technology
INS	Inertial Navigation System
IRNSS	Indian Regional Navigational Satellite System
KF	Kalman Filter
LLF	Local Level Frame
LOP	Line Of Position

MAC	Media Access Control
MEMS	Micro-Electro-Mechanical System
MEO	Medium Earth Orbit
MFC	Main Frequency Component
MSAS	MTSAT Satellite Augmentation System
PAC	Pedestrian Activity Classification
PDF	Probability Density Function
PDR	Pedestrian Dead Reckoning
PF	Particle Filter
POI	Points of Interest
PPP	Precise Point Positioning
QZSS	Quasi Zenith Satellite System
RAM	Random Access Memory
RFID	Radio Frequency Identification
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
RTLS	Real-Time Location System
SBAS	Satellite Based Augmentation System
SDCM	System for Differential Corrections and Monitoring
SI	International System of Units
SIS	Sequential Importance Sampling
SLAM	Simultaneous Localization and Mapping
SOP	Surface of Position
SPP	Single Point Positioning
TDoA	Time Difference of Arrival
ToA	Time of Arrival
TU	University of Technology
UKF	Unscented Kalman Filter
UWB	Ultra Wide Band
WAAS	Wide Area Augmentation System
WAF	Wall Attenuation Factor
WEKA	Waikato Environment for Knowledge Analysis
WGS84	World Geodetic System 1984
WHO	World Health Organization
WiFi	Wireless Fidelity
WLAN	Wireless Local Area Network
ZUPT	Zero Velocity Update

Contents

1 Introduction

As stated in a yearly report of the World Health Organization (WHO), it is estimated that globally 285 million people are visually impaired, thereof 39 million are blind and 246 million have low vision (World Health Organization, 2014). Additionally, the WHO reported that about 90% of the visually impaired have a low income. Considering a first-world country like Austria, 3.9% of the population suffer from permanent visual impairments (Blinden- und Sehbehindertenverband Österreich, 2007) and therefore have problems with the daily navigation through their environment, especially through urban areas and unknown indoor environments. No other group of the human population depends more on a precise and permanent operational navigation system than visually impaired people. Unfortunately, common navigation systems deliver position accuracies in the range of several meters, which is not sufficient for a reliable guidance of blind users. Compared to conventional pedestrian navigation, the quality requirements on the position determination are much higher. The position must be accurate enough (at the two-meter level and below) so that objects and locations relevant for guidance instructions are reliably situated within the tactile range of the blindman's stick. To achieve these accuracy requirements and to guarantee a continuous temporal availability without interruptions, the fusion of complementary positioning sensors and methods is indispensable.

Before going into more detail, the main focus of the thesis has to be defined. As visualized in Figure 1.1, navigation comprises three general tasks: positioning, routing and guidance. Within positioning, the location of the user has to be estimated relative to a defined reference frame, while within routing, the optimal route to a chosen destination is determined to finally guide the user along this route. The focus of this thesis is the development of an accurate and reliable positioning tool being part of a navigation system especially tailored to visually impaired people. The positioning performance is of huge importance, since the determined coordinates are the base for reliable routing as well as for precise guidance. Considering visually impaired people, in addition to the announcement of guidance instructions, an effective obstacle warning has to be installed, which demands for positioning accuracies within the range of a blindman's stick. A blind or visually impaired person must rely on the guidance instructions generated by the system, since the person is unable to compensate any misinformation by visual faculty. Due to the lack of highly accurate position information, the guidance instructions of today's navigation systems can only give coarse warnings of obstacles and rough maneuver instructions.

routing
guidance
positioning

Figure 1.1: The three tasks of navigation.

Besides these facts, it has to be considered that most of the blind people have a small income and therefore, it is important to keep the cost of the navigation system low. Consequently, the goal of the developments within the thesis is to exploit all possibilities of filtering based on preprocessed measurement data of the built-in sensors of a common smartphone to provide outdoor as well as indoor positioning. By using the positioning and processing facilities of the smartphone, no additional device has to be financed and carried along. Even if a social insurance would pay for the costs of a special navigation system, the smartphone comes along with the advantage to have one device for communication and navigation.

The developed positioning tool uses as much smartphone sensors as possible. On the one hand, to have redundant information and on the other hand, to compensate the disadvantages of one sensor by the advantages of the others. Complementary and redundant data improves the positioning availability, accuracy, and consequently the reliability. In particular, the output of the smartphone-internal GNSS (Global Navigation Satellite System) receiver, the WiFi (Wireless Fidelity) scanner, and sensors like accelerometer, gyroscope, magnetometer, and barometer are used to derive absolute and relative position information.

All position information is combined by a particle filter integrating maps of the environments. For the position estimation, indoor and outdoor environments have to be treated differently concerning the involved sensors and the spatial constraints arising from the incorporated maps and building layouts. The indoor positioning is based on 3D coordinates obtained by a WLAN (Wireless Local Area Network) fingerprinting algorithm, relative position information estimated within a *Pedestrian Dead Reckoning* (PDR) approach using inertial and magnetometer data, and a relative height information delivered by a barometer. Outdoors, the WLAN solution is replaced by code-based GPS positions.

The innovative aspect of the implemented outdoor position solution is the use of spatial constraints in terms of a user-tailored raster map considering the requirements of blind people. Since the visual impairment leads to a different, more predictable movement compared to the movement of common pedestrians, streets and paths can be classified into different categories. For example, blind persons most likely walk on pavements, cross the streets at zebra crossings only, and follow tactile ground surface indicators at certain places. This knowledge is used for the creation of a special occupancy grid map, which can be integrated as spatial constraint within the filter process. As a particle filter approach is able to handle multivariate data and nonlinear and non-Gaussian processes, the particle filter is the optimal filter for incorporating spatial constraints, which cannot be described by any kind of probability distribution. By representing the estimate by particles, the full information content of a map can be used within the positioning process. In contrast, the parametric approach of the Kalman filter (representing the estimate by the first and second moment of a probability function) makes it difficult to integrate the complete map-information content because of the strict assumptions concerning the models' probabilities. Therefore, the particle filter is the chosen algorithmic framework for the integration of the position data and the user-tailored map. Common particle filters for outdoor environments only distinguish occupied and non-occupied regions (streets, buildings, obstacles etc.), while the presented approach subdivides the environment with regard to the needs and walking behaviors of blind persons.

The creation of the special occupancy grid map, which represents the accessibility of public areas and the path preferences of blind pedestrians, was done in cooperation with a blind consultant. Up to now, no adequate map is publicly available containing all essential information for the positioning of blind persons. Therefore, the map specifications and requirements, which have been defined together with the consultant, have acted as a basis for the surveying of the necessary data. Unfortunately, according to the consultant, in indoor environments the movement of blind persons cannot be algorithmically constrained in the same way. The structures are too small in size and the blind persons' habits are very individual. However, common occupancy grid maps based on ordinary building layouts are integrated within the particle filter to improve the system's performance. Additionally, a wall collision filter strongly increases the positioning reliability.

Besides the preprocessed sensor data and the spatial constraints given by the different raster maps for indoor and outdoor environments, the outcome of a *Pedestrian Activity Classification* (PAC), which identifies activities like *walking*, *going upstairs*, *going downstairs* or *standing*, is used as multiple support within the positioning process. In indoor environments, the activity provides additional information about the vertical displacement of the pedestrian, which can be used as complementary data for the height estimation. As not all of the state-of-the-art smartphones comprise barometers and the WLAN fingerprinting focuses on the horizontal position performance, this alternative height estimation approach is investigated.

The vertical displacement is gained by combining a step detection, the determined activity, and a standardized step height. If the actual activity is *going upstairs* or *going downstairs* and a step is detected, then the step height corresponds to the covered height. The activity is also used as support for the 2D position estimation. Pursuant to the activity, the possible state space can be probabilistically restricted. Considering for example a pedestrian going upstairs, the probability of being at a stairway has to be higher.

At the moment, the developed framework can only be tested and analyzed within a defined indoor and outdoor test area at the campus of Graz University of Technology. A restricted test area had to be defined, since the establishment of the user-tailored raster map is a very time- and cost-consuming process. Hence, as an alternative, a *Simultaneous Localization and Mapping* (SLAM) approach is employed within the thesis for the creation of the user-tailored map. The automatic map generation requires a more precise absolute positioning than the GPS code solution with accuracies in the range of some meters.

For this reason, multiple-GNSS *Precise Point Positioning* (PPP), which provides position accuracies in the decimeter level, is considered. Within the defined test area, different tests have already been performed, which showed that PPP could be used for the SLAM-based generation of the map if enough GNSS satellites (at least four GPS or five GPS and GLONASS satellites) are available within the urban canyons. At the moment this is not the case 24 hours a day due to the still insufficient number of satellites in orbit. Hence, in this thesis the SLAM approach is investigated in a simulation environment. However, with the full operational capability of the European Galileo and the Chinese COMPASS more than 100 satellites will be available, which would fulfill the SLAM requirements. Then, the developed SLAM could be applied to real data. Finally, the application of the SLAM approach for generating user-tailored maps for indoor environments is reviewed.

1.1 Research questions

One goal of the thesis is the development of a positioning filter for an accurate and reliable outdoor as well as indoor positioning. Therefore, the measurement data of the smartphone sensors are supported within the filter by information gained from outdoor and indoor maps. For the outdoor environment, a user-tailored raster map has to be developed, which is established for a restricted test area. To make the developed system applicable everywhere, another goal is the design of a SLAM algorithm which automatically generates a user-adjusted probability map of the environment. An effective SLAM could also be used for establishing probability maps for buildings. For the described goals, the following research questions can be formulated, which are answered within the framework of the thesis.

With regard to the development of the positioning filter tailored to blind pedestrians various questions arise:

- Does a user-tailored occupancy grid map improve the positioning accuracy?
- Which spatial information can be used for restricting the positioning indoors?
- Does a wall collision filter improve the positioning performance?
- Can height changes be determined by combining the PAC and PDR? Is this height information suitable for the floor detection in buildings?
- Which position can be defined as best estimate in case of a particle filter?
- How good is the outdoor and indoor positioning performance of the developed tool?

The design and implementation of a SLAM approach to establish a probability map of the environment leads to the following questions:

- Is SLAM suitable for generating a probability map for blind users?
- How good is the applicability of PPP in urban environments? Can PPP be used for the generation of a probability map based on SLAM?
- How well does the outdoor positioning based on the SLAM-derived probability map perform?
- Can SLAM also be exploited for establishing probability maps for indoor environments?

For answering the questions formulated above, the following research has been conducted:

- Design and implementation of a Rao-Blackwell particle filter for the position estimation in outdoor and indoor environments based on absolute and relative position information
- Definition of the requirements for the user-tailored raster map
- Modeling of the indoor environment
- Design and implementation of a wall collision filter
- Derivation of activity-related probabilities
- Design and implementation of a SLAM approach for the derivation of location probabilities representing the path preferences of blind pedestrians
- Establishment of a simulation environment for testing the SLAM

Following existing algorithms have been employed within the thesis:

- GNSS precise point positioning
- RSSI-based WLAN fingerprinting
- Pedestrian dead reckoning based on inertial data
- Barometric height estimation
- Pedestrian activity classification

1.2 Structure of the thesis

The content of the thesis is structured as follows:

Chapter 2 gives an overview of existing navigation aids and the current research in the field of navigation of blind pedestrians. Commercially available navigation devices as well as smartphone applications are investigated with respect to the special navigation needs of blind persons. Based on the current research, unsolved problems and possible improvements are discussed.

Chapter 3 focuses on filtering techniques belonging to the family of Bayes filters. First, the theory of the Bayes filter is explained, which is followed by the fundamentals of non-parametric and parametric Bayes filter representations. On this basis, three filters - the Kalman filter, the grid-based estimation and the particle filter are particularly highlighted.

Chapter 4 concentrates on the positioning with current available smartphones and its built-in sensors. The chapter is subdivided into several sensor specific sections, in which the sensor characteristics and positioning methods are discussed. The focus is put on methods which were used for the preprocessing of the raw sensor data within the thesis.

Chapter 5 describes the design of the positioning tool, which is based on a particle filter integrating absolute and relative position data, as well as spatial constraints obtained from maps of the environments. The establishment of the outdoor and indoor maps is covered in detail, and the design of the wall collision filter is specified. Additionally, the contribution of the activity information within the position estimation is investigated.

Chapter 6 demonstrates the performance of the positioning tool in outdoor and indoor environments based on defined performance measures.

Chapter 7 treats the generation of probability maps based on a SLAM approach. At first, the theoretical background and the architectural design of the algorithm are described, followed by investigations concerning the performance in outdoor environments. At last, the emphasis is set on the applicability of SLAM for generating indoor probability maps.

Chapter 8 summarizes the work and discusses the characteristics of the developed system. At the end, an outlook with respect to further ideas is given.

2 Motivation

Compared to pedestrians without any visual impairment, the requirements of blind persons concerning a navigation system are more demanding in terms of accuracy, availability, reliability, and usability (handling). All tasks of navigation, comprising positioning, routing, and guidance, are affected by the special needs of blind users and strongly differ from those of common navigation systems.

Considering positioning, the accuracy has to be at least in the range of a blindman's stick. The availability has to be guaranteed 100% of the time with 100% spatial coverage providing seamless transitions from outdoor to indoor environments and vice versa. Additionally, the reliability of the position information has to be guaranteed all the time or at least, the blind user has to be informed if the positioning requirements cannot be satisfied. As the positioning is the basis for routing and guidance, the demands on the accuracy, availability, and reliability are very high.

The routing of visually impaired people has to be adapted to the habits and needs of this special user group. Pathways with low accidental risks have to be favored over dangerous routes, e.g., pavements with bike parking areas or mailboxes want to be avoided by visually impaired people. In general, objects which are not in the sensing range of a blindman's stick or change temporarily in its volume and location have to be circumvented within the routing if alternative routes exist. Furthermore, ways with navigation infrastructure for blind persons (tactile ground surface indicators, traffic lights with acoustic signals, etc.) have to be preferred. Therefore, for an effective and secure routing, accurate positions have to be provided. The navigation of blind pedestrians along navigational aids such as ground surface indicators and the avoidance of hazardous objects require an accuracy, which enables at least the localization of the pedestrian at the correct pavement. Consequently, the routing component for blind pedestrians is more demanding compared to the routing of general pedestrians and requires a more accurate positioning as basis.

The guidance component should include, besides the common instructions for route guidance, an obstacle warning system and should support the blind person in any navigational purpose along the route. To successfully avoid collisions, the position accuracy has to be in a range, which allows the assignment of the pedestrian to the correct pavement and the announcement of any risks at the right time. The blind pedestrian can only trust the navigation announcements if the position accuracy enables an effective navigation.

The way how the navigation announcements are handed to the user is very controversial and up to now a topic of research. Different possibilities exist, ranging from vibrational to acoustic solutions. As proposed in Loomis et al. (2008), special virtual acoustic displays have been developed for providing auditory information to the user via earphones. This acoustic display supports the guidance by manipulating the auditory direction and the sound volume of the navigation announcements. Closer landmarks are announced with a higher sound level than more distant ones. Additionally, the direction of the sound corresponds to the direction which has to be steered. However, different studies approve that blind persons prefer non-acoustic solutions over acoustic ones, since the sense of hearing is preferably used for navigating through the daily life and for quickly noticing risks and dangers.

In the following, currently available navigation devices and recent developments are investigated concerning the usability for blind persons. Within section 2.1, different commercially available systems which are either developed as self-contained navigation device or implemented as an application for a supporting device (tablet, smartphone, etc.) are discussed. An overview of the ongoing research is given in section 2.2. Due to the primary focus of the thesis, the emphasis of the state-of-the-art investigation in the field of science is set on positioning. Finally within section 2.3, unsolved problems are outlined based on the state-of-the-art products and research. The focus hereby is especially set on the topic of positioning. The section is concluded with possible improvements for the identified problems.

2.1 Products for the navigation of blind persons

Present navigation systems for visually impaired pedestrians by all means comprise GNSS-based single point positioning, slightly adapted navigable maps for the purpose of route planning, and coarse guidance instructions mainly restricted to information about obstacles and Points of Interest (POI). In this context, a few current navigation systems shall be mentioned: *Breeze*, *Tormes*, *MyGo*, and *Wayfinder Access*. In former times, the focus of the research was the development of self-contained navigation systems. Because of the extensive positioning and computing possibilities which are offered by smartphones nowadays, the emphasis of the product development switched from the design and creation of special mobile devices to the software development of smartphone applications. In section 2.1.1, navigation aids, which were especially developed for the navigation of blind pedestrians and are still commercially available, are presented and compared. In section 2.1.2, the characteristics of different smartphone applications are discussed under the aspect of the navigation of blind people.

2.1.1 Mobile devices

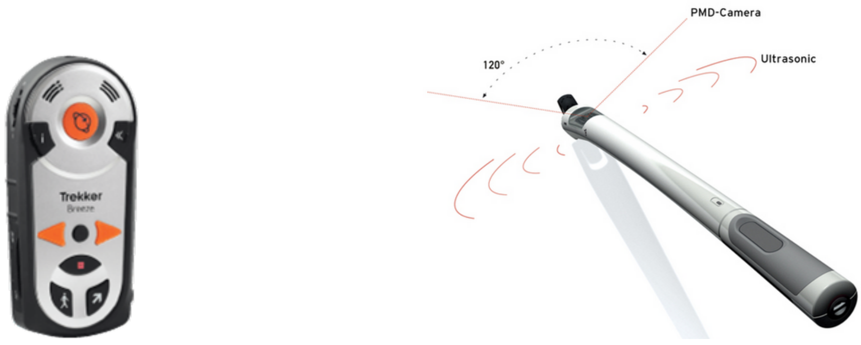
The first GPS-based navigation concepts for pedestrians appeared at the turn of the millennium. However, the operation of these devices was complicated and they had to be worn with a shoulder or neck strap. Products, which are simply applicable and helpful for the navigation of the blind and additionally low in weight and size, have been launched onto the market only during the last ten years. A short overview of the currently available navigation aids is given in the following.

Breeze, manufactured by the company *Trekker* (Humanware, 2015), is a handheld GPS device especially developed for the navigation of blind people in outdoor environments. Based on the estimated path, *Breeze* supports the blind pedestrian by voice instructions announcing the names of streets, intersections, and landmarks and gives also step-by-step instructions. In case of using public transport, *Breeze* indicates the names of the crossings to avoid missing the exit. Furthermore, *Breeze* offers a back-track function and the recording of tracks to use them for future guidance. Apart from the navigational functionality, the device itself was also especially designed for the use by blind people. It can be held in one hand and the user interface is kept as simple as possible, see Figure 2.1a. The disadvantage of *Breeze* is that no complementary sensor is integrated to support the GPS-based positioning in GPS-critical regions or indoors.

BrailleNote GPS of the *Sendero Group* (Sendero Group, 2014) offers in general the same functionalities as *Breeze* such as the announcement of nearby streets, intersections, etc. Paths can already be explored before the user starts going by a simulated navigation through the environment. There are two different guidance options comprising speech and Braille output.

Another similar product is *Tormes*, which was funded by the European Space Agency (ESA). Referred to ESA - Technology Transfer Programme (2003), *Tormes* is a computer with a Braille keyboard and GPS technology that gives voice instructions. *Tormes* incorporates corrections of the European Geostationary Navigation Overlay Service (EGNOS), thus it is providing accuracies in the range of 1 to 5 meters before applying a positioning filter. It weighs less than one kilo and can be carried over the shoulder.

As mentioned in Wired (2007), the product *Mygo* is a feature-laden rolling white cane, which uses a sensor-camera combination to screen the terrain in real-time (Figure 2.1b). Feedback to the user about the ground is given via a wireless headset. The product cannot be used as navigation aid, it is only used as assistance for the blind user. It was planned for the purpose of replacing a guide dog.



(a) *Breeze.*

(b) *Mygo.*

Figure 2.1: Commercially available navigation aids: *Breeze* is a handheld GPS device supporting the blind persons with guidance instructions in outdoor environments. *Mygo* is a feature-laden rolling white cane providing feedback about the terrain.

Except *Mygo*, all the above mentioned devices are designed for the navigation in outdoor environments. For the application within buildings, no commercially available device exists. As operable indoor solutions arised synchronously with the first smartphones, the developments are smartphone-based software products.

2.1.2 Smartphone applications

First solutions on mobile phones were developed for the operating systems Symbian and Windows Mobile. The positioning was purely based on GPS code solutions and the guidance as well as the operation of the phone was performed by a synthesized voice. *Wayfinder Access* was one of the first cell-phone-based GPS navigation software that can be used in combination with cell phone screen-reading software to assist a pedestrian in navigating through outdoor environments (American Foundation for the Blind, 2009). The software was originally developed for Symbian, second or third edition, but was extended for the installation on Android- and iOS-based phones. For the positioning, either the built-in GPS receiver or an external Bluetooth GPS receiver can be used. The advantage of an external receiver is the lower power consumption as well as the higher position accuracy. A similar product is *Mobile Geo* running on all Windows Mobile operating systems. *Mobile Geo* mainly differs from *Wayfinder* in the way how the map is accessed. While *Wayfinder* gets its map via the mobile network, *Mobile Geo* supports the storage of the maps on the phone.

The introduction of touchscreens was a step back for blind mobile phone users. However, with the launch of VoiceOver by Apple on iOS devices, a new era started for blind persons regarding the use of a smartphone as navigation aid (Apple Inc., 2015). Since then, the smartphone with its extensive positioning, computing, and communication possibilities is used as basis for the development of specialized navigation applications. VoiceOver verbally explains the contents of the screen, starting with the status of the battery, the name of the person calling, and ending with information about different apps. Thereby, the person just has to swipe over the screen to get the necessary information. In case of the communication with the user, the volume of currently played music is automatically reduced. Additionally, iOS devices are compatible with most of the common electronic Braille displays. Due to VoiceOver, iOS apple devices are the favored smartphones among blind users.

BlindSquare is a smartphone navigation application for blind users designed in collaboration with blind persons (BlindSquare, 2012). It is developed by the Finnish company *MIPSoft* for the use on iPhones and iPads. The application is based on open data sources and provides information about the address of the user's current location, the nearest street crossing, shops, and services, as well as assistance for the use of public transport. The positioning of *BlindSquare* is GPS-aided and therefore only applicable in outdoor environments. The operation of the app requires VoiceOver. The communication with the user for the purpose of guidance is voice-controlled. By shaking the smartphone, the user gets information about the actual location. Favored places and routes are saved to iCloud, consequently, all information is synchronized and can be used on all of the user's iOS-devices.

Ariadne GPS is another navigation application especially developed for blind persons, but offers less extensive navigation possibilities compared to *BlindSquare*. Searching public POIs or searching by address are only two missing options amongst others. Neither *Ariadne GPS* nor *BlindSquare* provide turn-by-turn guidance. *Myway Classic*, another smartphone application for aiding blind people, is mainly designed for the recording of walked routes, along which the person can be guided. *Google Maps* also provides navigation for pedestrians, but without any special guidance adaptions for visually impaired users. The mentioned applications differ concerning the need of an Internet connection. While *Ariadne GPS* and *Myway Classic* can be operated autonomously, the two others obtain the map information and the POIs from the Internet.

2.2 Research related to the navigation of blind persons

Outdoor positioning based on GNSS can be considered as well-solved (Mautz, 2012). Different approaches based on differential, relative, or precise point positioning exist, which provide position accuracies at the low decimeter or even centimeter level (Hofmann-Wellenhof et al., 2008; Groves, 2013). However, all these methods require raw GNSS code or phase measurements which are not available for smartphone users. Consequently, accurate smartphone-based outdoor solutions are realized by sensor fusion or the use of external Bluetooth GNSS receivers which transfer the position data to the mobile device.

I want to highlight two different projects, *POScity* and *PONTES*, which are relevant for the navigation of blind pedestrians in outdoor environments and are partly developed at Graz University of Technology, Institute of Geodesy, Working Group Navigation (former named Institute of Navigation). Within the project *PONTES* (Wieser et al., 2007), the Institute of Navigation worked in a close cooperation with the Styrian Association of the Blind and Visually Impaired People. Together, a navigation concept comprising the components positioning, routing, and guidance specially tailored to the blind people's use have been developed. The positioning is based on an integrated concept combining GPS positions with dead reckoning information. The experience gained in *PONTES* was the basis for the initiation of the project *POScity*, the focus of which is the improvement of the positioning.

Parts of the presented thesis have been realized in the course of *POScity* (Hafner et al., 2015). The focus of this project have been investigations regarding a particle filter incorporating 3D coordinates gained from a GNSS precise point positioning algorithm. The position estimation is supported by a user-tailored map, which allows the derivation of location probabilities. This user-tailored map represents the accessibility of areas and path preferences of blind pedestrians. The investigations have shown, that the fusion of PPP with a user-tailored probability map allows positioning with accuracies of a few decimeters. However, a geodetic GNSS receiver is necessary to provide these accuracies.

For the positioning within buildings, several indoor navigation systems have been developed over the past decade. For outdoor applications, GNSS provides positioning 24 hours a day with an appropriate accuracy and without special infrastructure (Hofmann-Wellenhof et al., 2008), while indoors no single technology covers the requirements for an accurate and reliable positioning (Mautz, 2012). Consequently, several indoor systems exist, which are based on different algorithmic approaches combining various sensor technologies. The current research treats positioning based on WLAN (Rajamäki et al., 2007) and Bluetooth networks (indoors, 2013), Radio Frequency Identification (RFID) (Ding et al., 2007; Chumkamon et al., 2008), dead reckoning sensors like accelerometer, gyroscope, magnetometer (Nilsson et al., 2014; Renaudin and Combettes, 2014), barometer, and cameras, etc.

The research in this field is very extensive and has a wide variety, because either the system does not provide the required accuracy, or special infrastructure has to be installed. Accurate systems are based on sensor fusion, e.g., Fischer et al. (2008) support the inertial-based pedestrian dead reckoning by absolute positions gained from ultrasound data, Rantakokko et al. (2011) combine PDR with Ultra Wide Band (UWB) ranging data, and Lópes-Salcedo et al. (2008) integrate measurements based on high-sensitivity GNSS and WLAN, but also data from an Inertial Measurement Unit (IMU) realized by Micro-Electro-Mechanical System (MEMS) technology. Fallah et al. (2013), Woodman (2010), and Mautz (2012) give a comprehensive overview of existing methods in the field of indoor positioning. In the following, some outstanding research projects are pointed out, which focus on positioning and navigation of blind pedestrians.

Noppa was a three-year pilot project funded by the Ministry of Transport and Communication of Finland (Virtanen and Koskinen, 2004). The aim of the project was to support blind pedestrians by public transport information and guidance in indoor and outdoor environments. As stated in Eltis - the urban mobility observatory (2008), *Noppa* was the first pedestrian navigation system offering real-time public transport information and speech recognition. For the outdoor navigation, code-based GPS, pedometer and compass data is used, while indoors, an integration of Bluetooth, WLAN, and compass/pedometer data is carried out. Therefore, a pocket PC in combination with a Bluetooth GPS receiver is employed. In the literature, no specifications concerning the position accuracy can be found.

LaureaPOP is an extended version of the system *Noppa* improving the indoor positioning (Rajamäki et al., 2007). *LaureaPOP* and *Noppa* are both designed for the use on a commercially available mobile phone. The system tries to convey the layout of the indoor facility to its user orally, including information about obstacles, the distance to destinations as well as navigation aids. The indoor localization is done with a WLAN positioning service of the Finnish company *Ekahau* (Ekahau, 2015). Since the year 2000, *Ekahau* has been operating in the field of designing, deploying, and maintaining WiFi networks. Based on their WiFi experience, they established a WiFi-based Real-Time Location System (RTLS), being one of the first offering such a service.

As proposed in Ran et al. (2004), *Drishti* is a wearable computer with a vocal communication interface and a wireless connection to a precise position measurement system. While outdoors Differential GPS is used, an ultrasound-based system provides the location indoors. The indoor positioning requires the installation of several ultrasound pilots within the building and two beacons which are mounted on the user's shoulder. Disregarding the inconvenient infrastructure, the ultrasound system convinces with its position accuracy of 22 cm (Ran et al., 2004). The transfer from outdoor to indoor is communicated to the system by a vocal command of the user.

The navigation system *PERCEPT* pursues a completely different solution. As mentioned in Ganz et al. (2012), it provides enhanced perception of the indoor environment by using RFID tags installed within the building. When entering a building, the person, who is equipped with a *PERCEPT* system, has to head for the so-called kiosk at the building entrance. There, the user can enter her/his intention, where she/he wants to go. Since the kiosk has an integrated RFID tag, the navigation system is able to scan the RFID tag which provides navigation related information. Based on this information, the user is guided to the intended destination.

Referred to Apostolopoulos et al. (2012) and Apostolopoulos et al. (2014), the project *Navatar* focuses on indoor positioning based on a particle filter estimating the location with dead reckoning sensor data and information about the presence of tactile landmarks. The blind person is regarded as sensor, documenting the appearance of landmarks like doors, floor transitions, water coolers, ramps, stairs, and elevators. The dead reckoning data is integrated as transition data within the update of the filter, while the reported landmarks represent the observations for the measurement update (details on positioning filters are given in chapter 3). By incorporating the tactile landmarks, the errors of dead reckoning can be mitigated. Additionally, a building map is integrated within the filter to restrict impossible pathways (e.g., wall or obstacle crossings). Within the investigations, different particle-filter-based schemes for automatically computing the step length of the user are treated. Since no sensor fusion is considered, the adaption to different step lengths within the filter is of huge importance as the location estimate is directly correlated to this parameter. Due to the step length estimation within the filter, no calibration/adaption of a user-depending step length model is required. By adapting the step length within the particle filter, the mean distance between the location estimation and the ground truth could be reduced from 2.31 m to 0.02 m. Details on the different performance investigations can be found in Apostolopoulos et al. (2014).

2.3 Unsolved problems and possible improvements

The outdoor positioning is well covered by GNSS-based techniques. Depending on the applied method, accuracies between a few meters up to a few centimeters can be reached. However, smartphones do not provide raw GNSS measurements which are necessary for advanced techniques yielding accuracies of decimeters/centimeters. Consequently, the smartphones' GNSS position accuracy is restricted to accuracies of a few meters. Therefore, in the near future, the smartphone providers have to be convinced to allow access to GNSS code and phase measurements. With the availability of code and phase measurements, PPP (see section 4.1.4) could easily be applied on smartphones. The necessary data (precise orbits and clock parameters) for providing coordinates with accuracies at the low decimeter level is globally valid and accessible in real-time by web-services like the International GNSS Service (IGS). Enabling PPP on smartphones would open the door for precise guidance instructions as well as reliable obstacle warnings.

In indoor environments, no technology fulfills all positioning requirements in terms of accuracy, reliability, and availability. Therefore, several approaches are topic of current research. Most of the methods either depend on infrastructure, which has to be installed and maintained, or do not satisfy the accuracy requirements for a reliable guidance. As proposed in Apostolopoulos et al. (2014), dead reckoning assisted by only a few absolute positions within an advanced Bayes filter approach, which also incorporates building layouts, enables accurate positioning without any extra infrastructure. However, within this method the user has to act as sensor reporting the presence of landmarks. A system without any user-driven feedback would let the blind person focus on the guidance instructions and is consequently favored.

The current research is showing that a sensor fusion is absolutely necessary to provide accuracy as well as reliability. In indoor as well as outdoor environments, the combination of complementary sensor data enables positioning 100% of the time. Indoor localization can considerably be improved by incorporating map information. Unfortunately, up to now no standard for building layouts has been established, although a standardization and an automatic disposal of maps at the building entrance would have a strong impact on the algorithmic solutions. At the moment, architectural blue prints without any standardization are the basis for generating maps which are used in the positioning process as well as for routing and guidance. By applying a non-parametric Bayes filter representation, the whole information content of the map can be exploited. Hereby, the particle filter has to be favored over grid-based approaches to keep the processing effort low.

Outdoors, particle filters are rarely used for the position estimation. Kalman filters are still state-of-the-art, because of the their compact parametric formulation, which is unbeatable concerning its computational costs. However, map information cannot be integrated within Kalman filters without any restrictions due to the Kalman filter's strict assumptions. The application of the Kalman filter demands for linear (or slightly non-linear) transition and observation models, and expects that the uncertainties of the involved data and models correspond to Gaussian distributions. These assumptions are violated if map information is integrated. Though, also outdoor maps contain information which improves the position accuracy. Within probabilistic Bayes approaches, maps, which are additional enriched by data relevant to blind people, can strongly contribute to the positioning through providing spatial constraints.

3 Bayesian filters

In navigation, Bayesian filters are used for positioning purposes estimating the state of a dynamic system based on a set of sensor measurements. The estimated quantities are not observed directly, however, the sensors provide partial information about the state. It has to be considered that the observations are affected by noise. Based on these noisy measurements, Bayesian filters provide a probabilistic estimation of the state of the dynamic system. Thereby, a probability distribution over the state x_t, defined as *belief* $bel(x_t)$, represents the estimate of the true state and the uncertainty in that estimate at every epoch. Bayesian filters sequentially determine such beliefs based on all available sensor data. The parameters which are estimated as well as the observation data are stochastic entities which are related by an analytical link (Haug, 2012). In general, the state contains all relevant information required to describe the system under investigation (Ristic et al., 2004). For example, considering the position and the spatial orientation as parameters describing the system, the state includes the three dimensional coordinates x, y and z, while the orientation of the object is represented by the orientation angles roll, pitch and yaw.

The framework of the Bayes filter provides the fusion of different sensor types and information, aiming at the integration of complementary data (Fox et al., 2003). In case of a sensor fusion, the measurement models have to be defined for each type of observation, since every sensor possesses its own characteristics, sources of errors and noise. However, for the positioning task within navigation, complementary sensors are preferred to compensate the disadvantages of one sensor by the advantages of the other one and vice versa (Woodman, 2010). More details on different sensors and their composition are given in chapter 4.

Based on the observations of one or more sensors, the belief $bel(x_t)$ of a dynamic system at a certain point in time t is given by

$$bel(x_t) = p(x_t|z_{1:t}), \tag{3.1}$$

representing the *conditional Probability Density Function* of the state x_t considering all available sensor measurements $z_{1:t}$ up to time t. More details on basic concepts in probability can be found in chapter A of the annex. Estimating the belief $bel(x_t)$ by a recursive Bayesian filter, two models, a dynamic model (also called prediction, state transition, or system model) and a measurement/observation model, are required. Both models together describe the dynamical stochastic system of the state which is declared as Dynamic Bayes Network (DBN). A general DBN is visualized in the left graph of Figure 3.1, which illustrates the temporal dependencies between the states of two consecutive epochs and the relationship between the estimated states $x_{1:t}$ and the observations $z_{1:t}$. Within the DBN, the dynamic model $p(x_t|x_{0:t-1})$ defines the object's dynamics and is responsible for the propagation of the belief from one epoch to the next, while the measurement model $p(z_t|x_{0:t}, z_{1:t-1})$ corresponds to the probability distribution representing the likelihood of the measurement at a certain point in the state space. Consequently, the dynamic model links two consecutive epochs, whereas the measurement model connects the state x_t with the observations z_t.

To make the realization of Bayesian filters tractable, the dynamic system is assumed to be a Markov chain of first order (Thrun et al., 2006). A Markov chain of first order expects that the state x_t is *complete* and consequently is the best predictor of the future. This means, that past states $x_{1:t-1}$ and previous measurements $z_{1:t-1}$ carry no additional information that would help predicting the future more accurately. As a result, the past and the future is independent if x_t is known. Considering the Markov assumption that the state x_t is complete, the state transition probability can be simplified to $p(x_t|x_{t-1})$, while the measurement probability is given by $p(z_t|x_t)$. The simplification of the DBN by assuming that the state is complete is illustrated in Figure 3.1.

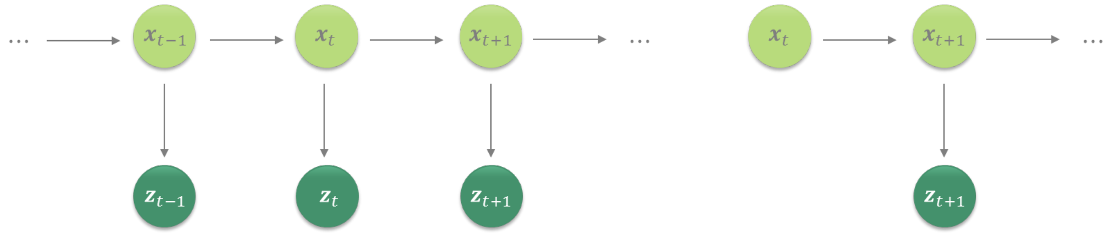


Figure 3.1: Illustration of a general DBN describing the dynamic stochastic system (left) and its simplified realization under the assumption that the state is complete (right).

As stated in Kremer (2005), the basis of Bayesian filtering is the *Bayes' rule*, which describes the relationship between two conditioned probabilities $p(x|y)$ and $p(y|x)$ based on two random variables X and Y. The Bayes' rule for the continuous case is defined by

$$p(x|y) = \frac{p(y|x)\, p(x)}{p(y)} = \frac{p(y|x)\, p(x)}{\int p(y|x)\, p(x)}. \tag{3.2}$$

In case of the discretization of the state space the integration is replaced by a sum

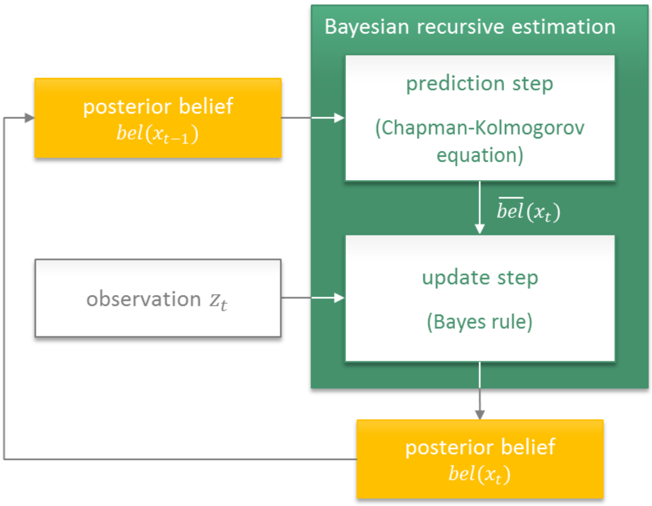
$$p(x|y) = \frac{p(y|x)\, p(x)}{\sum p(y|x)\, p(x)}. \tag{3.3}$$

Introducing the denominator $p(y)$ as a normalization constant η, equations (3.2) and (3.3) simplify to

$$p(x|y) = \eta\, p(y|x)\, p(x). \tag{3.4}$$

Details on the derivation of the Bayes' rule can be found in section A.1 of the annex.

Based on the Bayes' rule, the framework of the Bayes filter, which consists of a prediction and a measurement update step, can be defined. The general concept of Bayesian estimation is shown in Figure 3.2.



Bayesian recursive estimation
posterior belief $bel(x_{t-1})$
prediction step
(Chapman-Kolmogorov equation)
$\overline{bel}(x_t)$
observation z_t
update step
(Bayes rule)
posterior belief $bel(x_t)$

Figure 3.2: Principle concept of Bayesian estimation.

At first, for the tracking of the state through time, the dynamic model has to be applied to obtain a *prior belief* $\overline{bel}(x_t)$. Referring to Haug (2012) and Arulampalam et al. (2002), the link between the prior density defined by $p(x_t|z_{t-1})$ and the belief $bel(x_{t-1})$ of the previous epoch is provided by the Chapman-Kolmogorov equation:

$$\overline{bel}(x_t) = p(x_t|z_{t-1}) = \int p(x_t|x_{t-1})\, bel(x_{t-1})\, dx_{t-1}. \tag{3.5}$$

The belief $bel(x_{t-1})$ of the previous epoch implicitly includes the measurements up to time $t-1$. Therefore, the prior belief corresponds to the probability of the state x_t conditioned on the measurements up to time $t-1$.

Whenever a sensor measurement is available, the belief is updated by applying the Bayes' law incorporating the current measurements z_t. This step is called measurement update and performs the computation of the *posterior belief* $bel(x_t) = p(x_t|z_t)$ due to equation (3.2):

$$\begin{aligned} bel(x_t) &= \frac{p(z_t|x_t)\, p(x_t|z_{t-1})}{p(z_t|z_{t-1})} \\ &= \eta_t\, p(z_t|x_t)\, \overline{bel}(x_t), \end{aligned} \tag{3.6}$$

with the normalization factor $\eta_t = p(z_t|z_{t-1})$ (Haug, 2012; Doucet et al., 2010). The normalization factor η_t guarantees that the summation of the posterior belief over the whole state space results in 1.

Besides the state transition and measurement probability, the initial belief $bel(x_0) = p(x_0)$ of the state has to be defined. If prior knowledge is available, $bel(x_0)$ is initialized correspondingly, otherwise a uniform distribution is applied. Summarized, the posterior PDF of the state is determined with respect to all available information including the present measurements.

The dynamic system considered within this thesis is a pedestrian with visual impairments walking in indoor and outdoor environments. The observations incorporated within the update step are derived from smartphone sensor data. Considering visually impaired people, the possible pathways can be restricted compared to common pedestrians. For example, in outdoor environments it can be assumed that streets are only crossed at zebra crossings. This information can be additionally integrated besides the smartphone sensor data to estimate the posterior Probability Density Function (PDF). Figure 3.3 demonstrates how the information is probabilistically considered in the Bayes filter framework based on a concrete application.

Within this example, a blind person is moving along a street exhibiting three zebra crossings. The goal is to estimate the location of this person along the longitudinal axis of the street. By considering the movement restrictions of visually impaired persons, in case of crossing the street, the possible location can be probabilistically limited to crosswalks.

If there is no information about the initial position of the pedestrian, the belief $bel(x_0)$ is initialized uniformly over the state space, see graph (a) in Figure 3.3. If the person crosses the street, this will be done at a zebra crossing. Therefore, by integrating additional information such as the path restrictions of blind persons, the possible state space can be reduced to regions around crosswalks if the person traverses the street. The probability distribution $p(z_t|x_t)$ visualized with light green in graph (b) represents the geometric constraints in case of passing the street. The density function in dark green is the result of the update step, combining the uniform initial belief $bel(x_0)$ with the probability distribution based on the walking restrictions $p(z_t|x_t)$.

After crossing the street, the person continues moving along the street and crosses again at the second crosswalk. The prior belief $\overline{bel}(x_t)$ in graph (c) is obtained by applying the dynamic model $p(x_t|x_{t-1})$, which corresponds to a forward motion along the street. Every time the dynamic model is applied, the probability distribution gets spread due to the uncertainty of the motion. As soon as the person is crossing the street, the spatial constraints deliver the PDF in light green in graph (d). Multiplying the prior belief $\overline{bel}(x_t)$ by $p(z_t|x_t)$, most of the probability mass is centered around the true position as the dark green PDF in graph (d) demonstrates. The last graph (e) shows the prior belief after applying the dynamic model again. Although the uncertainty of the dynamic model causes a larger spreading of the probability distribution, the actual position can clearly be identified.

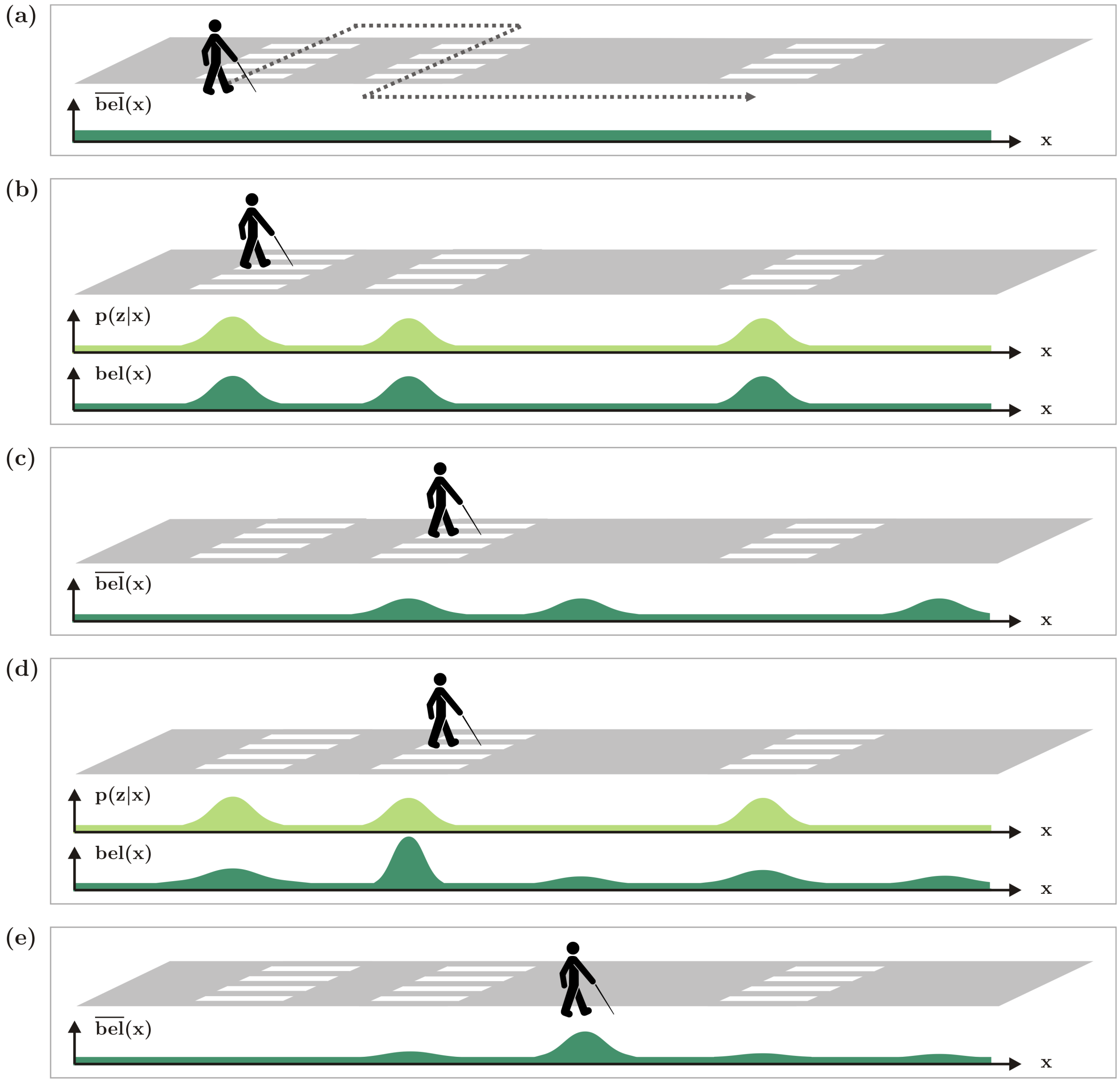


Figure 3.3: Illustration of the Bayes filter algorithm with respect to a blind pedestrian moving along the street.

As demonstrated in the example, the Bayes filter incorporating a dynamic and a measurement model provides the optimal framework for estimating the state of a dynamic system. The dynamic model aims at profiting from the information integrated in the previous epoch by concatenating the posterior belief $bel(x_{t-1})$ of the predecessor state x_{t-1} with the belief estimation of the successor state x_t.

Since the Bayes filter cannot be solved analytically over the whole state space, the algorithmic approach provides only a conceptual probabilistic framework for recursive state estimation (Arulampalam et al., 2002). As stated in Thrun et al. (2006), various practical realizations of the Bayes filter exist, being subject to different assumptions and approximations concerning the state transition probabilities, the measurement probabilities, and the initial belief. Looking for the optimal filter approach, parameters like computational efficiency, accuracy of the approximation, and ease of the implementation have to be deliberated. The computationally tractable realizations of the Bayes filter can be classified into two categories, the parametric and the non-parametric Bayesian filters. Figure 3.4 shows the hierarchy of the various Bayes filter realizations.

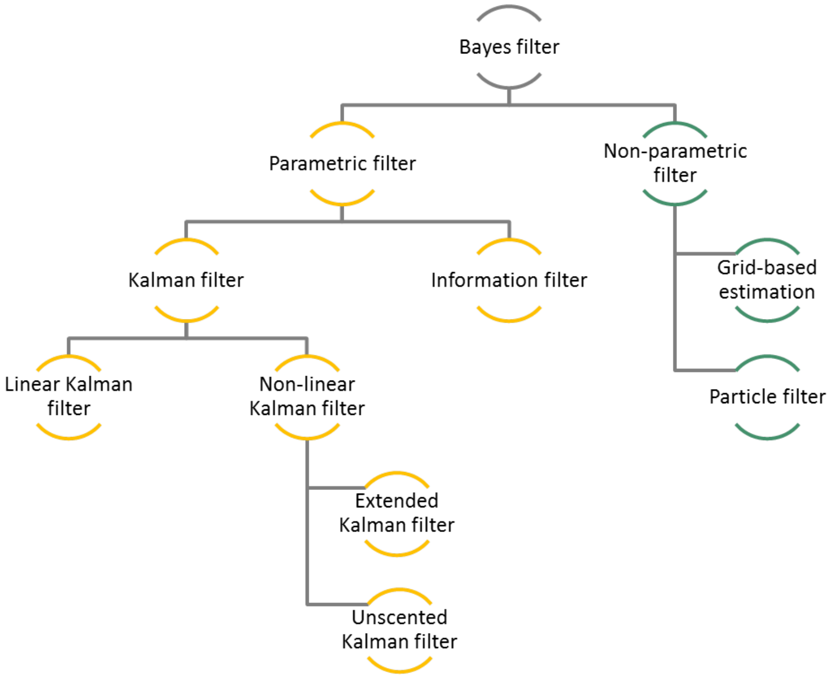


Figure 3.4: Hierarchy of Bayesian filter realizations.

Along the left-hand side the parametric filters are listed including different realizations of the *Kalman filter* and the *information filter*, while along the right-hand side the non-parametric filters comprising the *grid-based estimation* and the *particle filter* are shown. The special characteristics of these approaches are described in the following sections 3.1 and 3.2, while a tabular overview is given in Table 3.1. The overview demonstrates that the approaches vary in their assumptions and approximations concerning the linearity of the dynamic and the measurement model, the respective distributions, and the covered state space. The second column of Table 3.1 refers to the kind of models, which can be applied within the respective filter. The attribute *non-linear* means hereby, that both linear and non-linear models are employable.

Table 3.1: Characteristics of different Bayes filter implementations

Bayes filter	Models	Distribution	State space
Kalman filter	linear	unimodal (Gaussian)	continuous, infinite
extended	non-linear	unimodal (Gaussian)	continuous, infinite
unscented	non-linear	unimodal (Gaussian)	continuous, infinite
information filter	linear	unimodal (Gaussian)	discrete, infinite
grid-based estimation	non-linear	multimodal	discrete, finite
particle filter	non-linear	multimodal	discrete, infinite

The Kalman filter is restricted to dynamic systems with normal distributed uncertainties. It is applicable for the modeling of dynamic systems in a continuous and infinite state space. The huge drawback of this filter approach is the limited treatment of non-linear models. The information filter has the same assumptions and characteristics as the Kalman filter, but is more suitable for dynamic systems incorporating many different types of measurements (Thrun et al., 2006). In contrast, the grid-based estimation and the particle filter may be defined for linear as well as non-linear functions and can represent arbitrary, multimodal probability densities. While the particle filter is not limited to a finite state space, the grid-based estimation can only be applied for a restricted discrete state space.

The filters stated above can further be subcategorized into differently realized implementations, in which the assumptions and characteristics are slightly modified. Some of these realizations are already specified in Table 3.1. With the *extended* and *unscented Kalman filter* non-linear models may be handled. What remains, is the restriction to Gaussian distributions. The *Extended Kalman Filter* (EKF) belongs to the class of non-linear Kalman filters based on analytical linearization, while the *Unscented Kalman Filter* (UKF) extracts sigma points to circumvent the linearization of non-linear models (Figure 3.4). Nonetheless, their performance concerning non-linearity is still restricted, for details see section 3.1.1.

The Kalman filter, the grid-based estimation, and the particle filter, which are all very common in the field of positioning, are described in more detail in sections 3.1 and 3.2.

3.1 Parametric filters

Parametric representatives of the Bayes filter are approximations with respect to one special probability distribution, the Gaussian distribution. Consequently, the dynamic as well as the observation models correspond to Gaussian processes. Because of the assumptions concerning the model's distribution, these filters are, referred to Thrun et al. (2006), also called *Gaussian filters*. As stated in Haug (2012), within these Gaussian filters, the dynamic and measurement model as well as the estimated belief are represented by normal distributions, which are characterized by the mean μ and the variance σ^2. These parameters are the first and the second moment of a probability density distribution. Since the first two moments characterize the Gaussian density completely, a recursive propagation of these two parameters is suitable for optimally estimating the state of Gaussian processes (Haug, 2012).

The most famous parametric representative of the Bayes filter is the Kalman filter, first introduced in 1960. Kalman filters were the earliest tractable implementations of the Bayes filter defined for the continuous space (Groves, 2013). Besides the assumption of a Gaussian distribution, the dynamic and the measurement model have to be linear functions of the state x. As a consequence, the Kalman filter can only be applied on a limited class of dynamic systems. Extensions of the Kalman filter for non-linear models exist, but are still restricted to Gaussian distributions. Two different expanded filter approaches applicable to non-linear problems are treated in the subsections of section 3.1.1. These two are called extended and unscented Kalman filter and differ in the handling of the non-linear models. The EKF applies an analytical linearization, while the UKF is based on extracted sigma points, which are used for the non-linear transformations.

3.1.1 Kalman filter

Within the Kalman Filter (KF), the beliefs of the Bayes filter are represented by point estimates of a probability density function. The point estimates are the so-called moments of a probability density function, whereby the first and second moment correspond to its mean and variance (Thrun et al., 2006). Therefore, the probability density functions have to be defined for the dynamic and the measurement model. In case of the Kalman filter, they are assumed to be Gaussian. Under this assumption, all distribution functions appearing in the Bayes filter algorithm become Gaussian (Haug, 2012). Considering a normal distribution, all other moments except the first two are zero and hence, the Gaussian density is completely characterized by the mean and the variance.

The parametrization of a Gaussian distribution by its first and second moment is called *moments parametrization.* The first moment, the expectation of a random variable X is obtained by

$$E\left[X\right] = \int_{-\infty}^{\infty} x\, p(x)\, dx, \tag{3.7}$$

with $p(x) = p(X = x)$ representing the probability that the random variable X corresponds to the value x. The variance is defined as the expectation of the squared deviation from the mean, which is given by

$$\begin{aligned} Var\left[X\right] &= E\left[X - E\left[X\right]\right]^2 \\ &= \int_{-\infty}^{\infty} x^2\, p(x)\, dx - E\left[X\right]^2. \end{aligned} \tag{3.8}$$

The probability density function of the normal distribution $\mathcal{N}(x;\mu,\sigma^2)$ is given by the Gaussian function

$$p(x) = (2\pi\sigma^2)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\,\frac{(x-\mu)^2}{\sigma^2}\right), \tag{3.9}$$

where the mean μ is nothing else but the expectation $E\left[X\right]$ of equation (3.7) and σ^2 corresponds to the $Var\left[X\right]$ of equation (3.8). In equation (3.9), the mean μ, the variance σ^2, and the state x are expected to be scalar values. The normal distribution given by the Gaussian function is symmetric to its mean and has only a single maximum corresponding to a unimodal distribution.

In most of the state estimation problems, the state x is a multidimensional vector representing for example the horizontal coordinates and the orientation of an object. The normal distribution defined for a vector $\boldsymbol{x}$ is named *multivariate* and obtained by

$$p(\boldsymbol{x}) = det(2\pi\boldsymbol{\Sigma})^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\,\boldsymbol{\Sigma}^{-1}\,(\boldsymbol{x}-\boldsymbol{\mu})\right). \tag{3.10}$$

Considering a multivariate distribution, the state $\boldsymbol{x}$ as well as the mean $\boldsymbol{\mu}$ are vectors of dimension $(n \times 1)$, whereby $\boldsymbol{\mu}$ contains the corresponding mean values of $\boldsymbol{x}$. The covariance $\boldsymbol{\Sigma}$ is a quadratic matrix of size $(n \times n)$, being symmetric and positive semidefinite. The variances of the state's components are given in the main diagonal of the covariance $\boldsymbol{\Sigma}$, while the off-diagonal elements represent the covariances of the state entries.

Since the normal distribution is completely defined by the mean and the (co)variance, the recursive estimation of the first two moments within the Bayes filter framework represents the optimal estimation method for Gaussian processes (Haug, 2012). However, the beliefs can only be computed for a continuous state space, while a hybrid or discrete state space cannot be modeled by normal distributions. In addition to the Gaussian approximation, the Kalman filter approach needs linear dynamic and measurement models to guarantee that the posterior probability density function $p(\boldsymbol{x}_t|\boldsymbol{z}_t)$ estimated within the Kalman filter is also Gaussian (Khider, 2005). Therefore, the following three properties have to be fulfilled for the application of a Kalman filter:

1. The dynamic model $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ is a known linear function $f_t(\boldsymbol{x}_{t-1},\boldsymbol{\epsilon}_t)$, whose process noise $\boldsymbol{\epsilon}_t$ is drawn from a Gaussian distribution $\boldsymbol{\epsilon}_t \sim \mathcal{N}(\boldsymbol{0},\boldsymbol{Q}_t)$ with zero mean and covariance $\boldsymbol{Q_t}$. $\boldsymbol{Q}_t$ is the appropriate covariance matrix of the process noise $\boldsymbol{\epsilon}_t$.

2. The measurement model $p(\boldsymbol{z}_t|\boldsymbol{x}_t)$ is a known linear function $f_t(\boldsymbol{x}_t, \boldsymbol{\delta}_t)$, whose measurement noise $\boldsymbol{\delta}_t$ is drawn from a Gaussian distribution $\boldsymbol{\delta}_t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_t)$ with zero mean and covariance $\boldsymbol{R_t}$. $\boldsymbol{R}_t$ is the appropriate covariance matrix of the measurement noise $\boldsymbol{\delta}_t$.

3. The initial belief $p(\boldsymbol{x}_0)$ is known and corresponds to a normal distribution.

If these three assumptions regarding the linearity and the probability distribution of the dynamic and measurement model as well as the initialization hold, the estimated belief corresponds to a Gaussian density with mean $\boldsymbol{\mu}_t$ and covariance $\boldsymbol{\Sigma}_t$. The covariance $\boldsymbol{\Sigma}_t$ of the estimated state $\boldsymbol{x}_t$ represents the uncertainties in the Kalman filter state estimates and the degree of correlation between the errors in those estimates (Groves, 2013).

Considering the mentioned assumptions, the linear dynamic model can be defined by

$$\boldsymbol{x}_t = \boldsymbol{\Phi}_t \boldsymbol{x}_{t-1} + \boldsymbol{\epsilon}_t. \tag{3.11}$$

If $\boldsymbol{x}_t$ and $\boldsymbol{x}_{t-1}$ are state vectors of two consecutive epochs, the state transition matrix $\boldsymbol{\Phi}_t$ models the propagation of the state from one epoch to the next. The random variable $\boldsymbol{\epsilon}_t$ is the uncertainty of the modeled dynamics and is expected to be Gaussian. Following the Kalman filter's assumptions, the state transition probability $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ is then defined by a normal distribution $\boldsymbol{x}_t \sim \mathcal{N}(\boldsymbol{\Phi}_t \boldsymbol{x}_{t-1}, \boldsymbol{Q}_t)$ with mean $\boldsymbol{\Phi}_t \boldsymbol{x}_{t-1}$ and covariance $\boldsymbol{Q}_t$. The state transition probability, which is obtained by

$$p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}) = det(2\pi \boldsymbol{Q}_t)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}_t - \boldsymbol{\Phi}_t \boldsymbol{x}_{t-1})^T \boldsymbol{Q}_t^{-1} (\boldsymbol{x}_t - \boldsymbol{\Phi}_t \boldsymbol{x}_{t-1})\right), \tag{3.12}$$

demonstrates the interrelation between the probabilistic form of the Bayes filter and the Kalman filter realization.

The measurement function models the relationship between the state $\boldsymbol{x}_t$ and the current measurements $\boldsymbol{z}_t$. If the observation equation is linear, the analytical link between the state and the noisy measurements can be defined by

$$\boldsymbol{z}_t = \boldsymbol{H}_t \boldsymbol{x}_t + \boldsymbol{\delta}_t, \tag{3.13}$$

expecting that the so-called design matrix $\boldsymbol{H}_t$ is the ideal (noiseless) connection between $\boldsymbol{z}_t$ and $\boldsymbol{x}_t$ at time t. The measurement noise is assumed to be normal distributed and is given by the random variable $\boldsymbol{\delta}_t$. As a consequence of the assumptions and the linear observation model, the probability density of the measurement update corresponds to a normal distribution $\boldsymbol{z}_t \sim \mathcal{N}(\boldsymbol{H}_t \boldsymbol{x}_t, \boldsymbol{R}_t)$ with mean $\boldsymbol{H}_t \boldsymbol{x}_t$ and covariance $\boldsymbol{R}_t$:

$$p(\boldsymbol{z}_t|\boldsymbol{x}_t) = det(2\pi \boldsymbol{R}_t)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{z}_t - \boldsymbol{H}_t \boldsymbol{x}_t)^T \boldsymbol{R}_t^{-1} (\boldsymbol{z}_t - \boldsymbol{H}_t \boldsymbol{x}_t)\right). \tag{3.14}$$

Based on the defined assumptions, the three essential steps of the Kalman filter algorithm can be defined (Hofmann-Wellenhof et al., 2003):

Prediction step / time update: Transition of the state

$$\tilde{\boldsymbol{x}}_t = \boldsymbol{\Phi}_t \hat{\boldsymbol{x}}_{t-1} \tag{3.15}$$

$$\tilde{\boldsymbol{\Sigma}}_t = \boldsymbol{\Phi}_t \boldsymbol{\Sigma}_{t-1} \boldsymbol{\Phi}_t^T + \boldsymbol{Q}_t \tag{3.16}$$

Gain computation: Computation of the Kalman weight

$$\boldsymbol{K_t} = \tilde{\boldsymbol{\Sigma}}_t \boldsymbol{H}_t^T (\boldsymbol{H}_t \tilde{\boldsymbol{\Sigma}}_t \boldsymbol{H}_t^T + \boldsymbol{R}_t)^{-1} \tag{3.17}$$

Measurement update: Correction of the predicted state

$$\hat{\boldsymbol{x}}_t = \tilde{\boldsymbol{x}}_t + \boldsymbol{K}_t (\boldsymbol{z}_t - \boldsymbol{H}_t \tilde{\boldsymbol{x}}_t) \tag{3.18}$$

$$\boldsymbol{\Sigma}_t = (\boldsymbol{I} - \boldsymbol{K}_t \boldsymbol{H}_t) \tilde{\boldsymbol{\Sigma}}_t \tag{3.19}$$

These steps are performed recursively for the estimation of the trajectory which is represented by the mean $\boldsymbol{\mu}_t$, replaced by $\hat{\boldsymbol{x}}_t$ within the algorithm, and the covariance $\boldsymbol{\Sigma}_t$. After an initialization of the filter, the estimates are recursively updated by a weighted average of the previous estimate and the new estimate derived from the latest observations. All necessary steps, including the initialization of the filter, the prediction and the update step, are visualized in Figure 3.5.

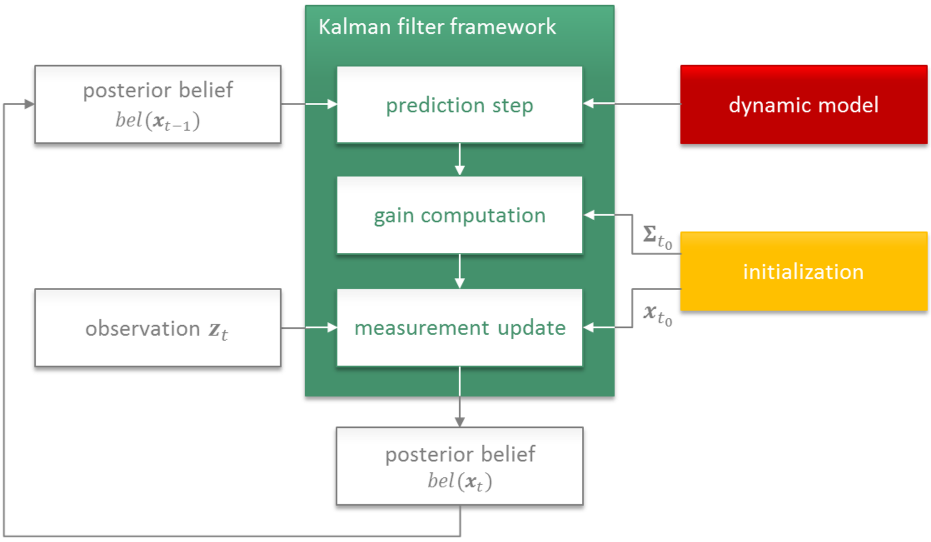


Figure 3.5: Concept of the Kalman filter.

The algorithm expects an initial estimate $p(\boldsymbol{x}_0)$ at time t_0 which has to be Gaussian. After the initialization of the belief, at every point t in time, first, a prior belief $\overline{bel}(\boldsymbol{x}_t)$ represented by the mean $\tilde{\boldsymbol{x}}_t$ and the covariance $\tilde{\boldsymbol{\Sigma}}_t$ is determined by applying the equations (3.15) and

(3.16) of the prediction step. For the propagation of the state, the transition matrix $\boldsymbol{\Phi}_t$ and the predecessor state $\hat{\boldsymbol{x}}_{t-1}$ are needed. The covariance of the prior belief $\overline{bel}(\boldsymbol{x}_t)$ depends on the posterior covariance $\boldsymbol{\Sigma}_{t-1}$, the transition matrix $\boldsymbol{\Phi}_t$, and the noise of the dynamic model which is given by $\boldsymbol{Q}_t$. The purpose of the prediction step is the propagation of the state and the covariance from the previous epoch to the time of the current measurements. In case of the absence of measurements, the Kalman filter still provides an estimated state and its error characteristics by applying the dynamic model (Groves, 2013).

The Kalman weight, computed within the gain computation step, defines how the measurements should be weighted in order to optimally determine the actual state $\boldsymbol{x}_t$ (Brown and Hwang, 1997). Therefore, the uncertainty of the predicted state which is given by $\tilde{\boldsymbol{\Sigma}}_t$, the uncertainty of the measurement model $\boldsymbol{R}_t$, and the design matrix $\boldsymbol{H}_t$ have to be provided. This Kalman weight specifies the influence of the current measurements $\boldsymbol{z}_t$ on the posterior belief. By applying equation (3.18) of the measurement update, the posterior belief of the state $\boldsymbol{x}_t$ is determined. Within this step, the Kalman weight $\boldsymbol{K}_t$ manipulates the estimation of the mean by adjusting the predicted mean by $\boldsymbol{z}_t - \boldsymbol{H}_t\tilde{\boldsymbol{x}}$ in proportion to $\boldsymbol{K_t}$. The difference between the actual measurement $\boldsymbol{z}_t$ and the expected measurement $\boldsymbol{H}_t\tilde{\boldsymbol{x}}_t$ is called *innovation*. Finally, the covariance of the posterior belief $\boldsymbol{\Sigma}_t$ is updated by equation (3.19) with respect to the current Kalman weight and the predicted covariance $\tilde{\boldsymbol{\Sigma}}_t$.

Figure 3.6 illustrates the prediction and measurement update within the Kalman filter in a probabilistic form, using again the example of localizing a blind pedestrian along the longitudinal axes of the street by incorporating a dynamic model and information about the location of the zebra crossings. Since the Kalman filter expects an initialization by a Gaussian density, the state $\boldsymbol{x}_{t_0}$ and the covariance $\boldsymbol{\Sigma}_{t_0}$ have to be defined. Consequently, the starting position is given by a unimodal distribution.

Graph (a) of Figure 3.6 shows the posterior belief at a certain point t in time. By applying the dynamic model, the state is transferred according to the defined motion which is illustrated in graph (b). The increase of the probability's spreading is caused by the uncertainty of the dynamic system. Incorporating the information of being at a crosswalk, one more assumption has to be made. The information must be assigned to a certain zebra crossing to get a unimodal Gaussian distribution centered at the location of the crosswalk. Because of the assumptions defined for the Kalman filter, multimodal information cannot be considered. By incorporating the information content of the map, which corresponds to the probability density function in light green in graph (c), the spreading of the posterior belief $bel(x)$ (dark green function in graph (c)) can be reduced, since the PDF of the measurement (light green function in graph (c)) is superimposed by the state transition probability (dark green function in graph (b)). As visualized in graph (d), by applying again the dynamic model, the probability density of the prior belief gets scattered.

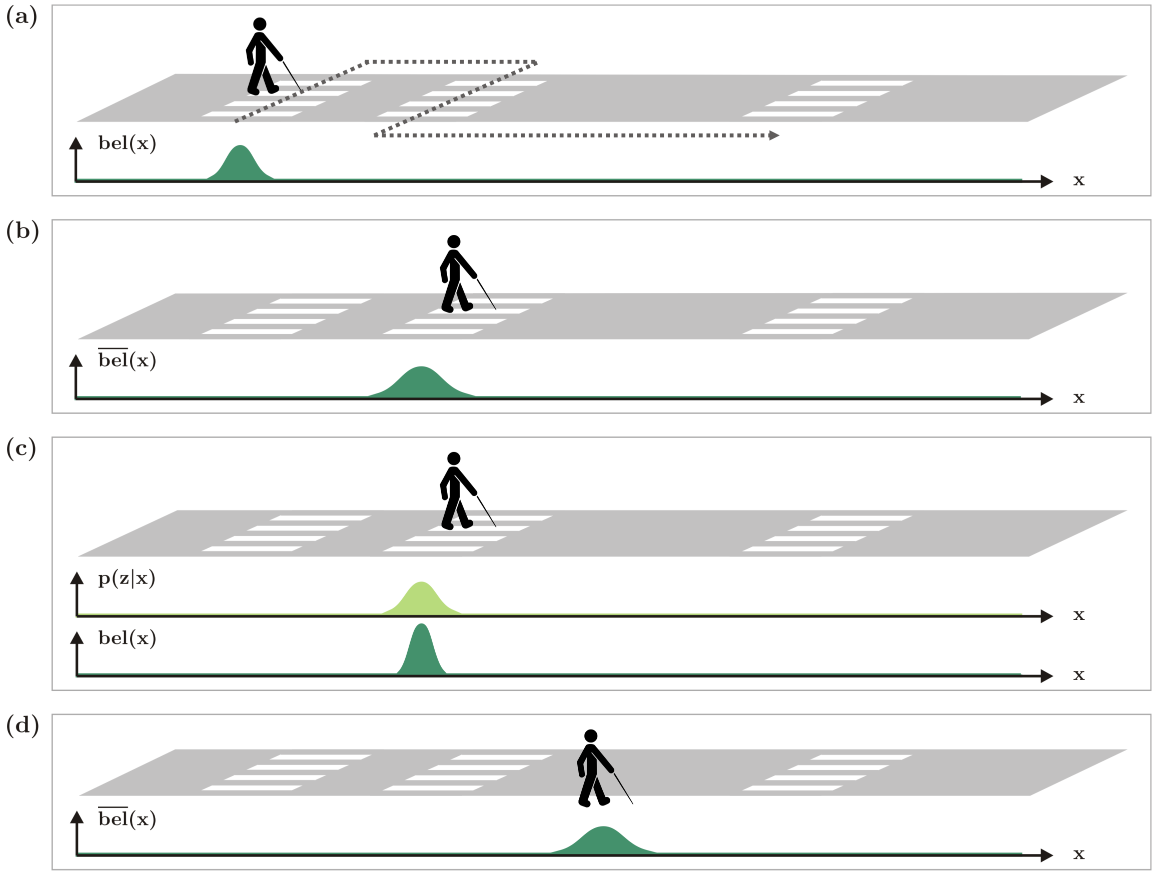


Figure 3.6: Illustration of the Kalman filter algorithm with respect to a blind pedestrian moving along the street. The probabilities are represented by Gaussian distributions.

As stated in Arulampalam et al. (2002), the Kalman filter is the optimal filter for linear models with appropriate Gaussian distributions. The advantage of Kalman filters compared to other filters are their efficiency in terms of memory and computation due to the parametrization of the probability density functions. All parameters are estimated in closed form. The dynamic and measurement model do not have to be evaluated over the entire state space. Hence, the Kalman filter is preferred for linear and near-linear systems, the beliefs of which are unimodal. However, they are only applicable for a restricted class of problems, which can be described by linear functions with normal distributed uncertainties.

The restrictive assumptions make the Kalman Filter inapplicable for many applications. As a consequence, expansions of the general algorithm have been developed. The extended Kalman filter allows the generalization of the theory to non-linear functions by linearizing the functions by a Taylor series expansion. The unscented Kalman filter applies a stochastic linearization based on extracted sample points. These sample points are used to propagate the parameters through the non-linear functions. The belief is then generated by a weighted linear regression process of these transformed sample points (Thrun et al., 2006). This process is called unscented transform and gives the filter the name.

Extended Kalman filter

The Extended Kalman Filter (EKF) is an extended version of the Kalman filter for non-linear problems. The state transition and the measurement probability can be modeled by non-linear functions which are linearized through a first-order Taylor series expansion. Therefore, the functions describing the state transition and the measurement model have to be differentiable functions. Referred to Haug (2012), through the linearization of the non-linear functions, the estimation of the state and the corresponding covariance can again be solved by applying a Kalman filter approach which is almost identical to that of the linear version. However, the EKF requires the computation of *Jacobian* matrices before performing the Kalman filter in a recursive manner. Although the Kalman filter's assumption concerning the linearity can be circumvented, the condition that the probability density functions correspond to Gaussian distributions remains.

Within the EKF, the dynamic model is given by a non-linear function of the state $\boldsymbol{x}_t$

$$\boldsymbol{x}_t = \boldsymbol{\varphi}(\boldsymbol{x}_{t-1}) + \boldsymbol{\epsilon}_t, \tag{3.20}$$

where the non-linear function $\boldsymbol{\varphi}(\boldsymbol{x}_{t-1})$ replaces the transition matrix $\boldsymbol{\Phi}_t$ in equation (3.11), and $\boldsymbol{\epsilon}_t$ is again a random process with zero mean described by the covariance of the system noise $\boldsymbol{Q}_t$. The measurement model is given by

$$\boldsymbol{z}_t = \boldsymbol{h}(\boldsymbol{x}_t) + \boldsymbol{\delta}_t, \tag{3.21}$$

with $\boldsymbol{h}(\boldsymbol{x}_t)$ being the non-linear function specifying the connection between the state $\boldsymbol{x}_t$ and the observations $\boldsymbol{z}_t$. The function $\boldsymbol{h}(\boldsymbol{x}_t)$ can be compared to the design matrix $\boldsymbol{H}_t$ in equation (3.13). $\boldsymbol{\delta}_t$ represents the measurement noise, which is normal distributed according to $\mathcal{N}(0, \boldsymbol{R}_t)$.

Because of the non-linear dynamic and measurement functions given in equation (3.20) and (3.21), a first-order approximation, which corresponds to a linearization of the function, has to be applied to obtain the transition matrix $\boldsymbol{\Phi}$ and the design matrix $\boldsymbol{H}$. This is done by expanding the non-linear functions in Taylor polynomials and maintaining only the linear terms (Haug, 2012). According to Zarchan and Musoff (2005), the transition matrix $\boldsymbol{\Phi}$ and the design matrix $\boldsymbol{H}$ are related to the non-linear functions by

$$\boldsymbol{\Phi} = \left.\frac{\partial \boldsymbol{\varphi}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}}, \tag{3.22}$$

$$\boldsymbol{H} = \left.\frac{\partial \boldsymbol{h}(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\tilde{\boldsymbol{x}}}. \tag{3.23}$$

Consequently, by linearizing the dynamic function with respect to $\hat{\boldsymbol{x}}$, the elements of the $n \times n$ transition matrix $\boldsymbol{\Phi}$ can be determined by

$$\boldsymbol{\Phi}_{j,q} = \left.\frac{\partial \boldsymbol{\varphi}_j(\boldsymbol{x})}{\partial \boldsymbol{x}_q}\right|_{\boldsymbol{x}=\hat{\boldsymbol{x}}}, \qquad \text{where} \begin{cases} j = 1, \ldots, n, \\ q = 1, \ldots, n. \end{cases} \tag{3.24}$$

The $l \times n$ elements of the design matrix $\boldsymbol{H}$ are gained by linearizing the observation functions with respect to the predicted state $\tilde{\boldsymbol{x}}$:

$$\boldsymbol{H}_{i,p} = \left.\frac{\partial \boldsymbol{h}_i(\boldsymbol{x})}{\partial \boldsymbol{x}_p}\right|_{\boldsymbol{x}=\tilde{\boldsymbol{x}}}, \qquad \text{where} \begin{cases} i = 1, \ldots, l, \\ p = 1, \ldots, n. \end{cases} \tag{3.25}$$

The transition matrix $\boldsymbol{\Phi}$ and the design matrix $\boldsymbol{H}$ are consequently the Jacobian matrix representations of the non-linear transition and observation models. As outlined in Thrun et al. (2006), the linearization process approximates a non-linear function $f(\boldsymbol{x})$ by a linear function that is tangent to $f(\boldsymbol{x})$ at the mean of the Gaussian distribution. Due to the linearization of the models, the estimated belief is no longer Gaussian distributed. The EKF is only able to calculate a Gaussian approximation to the true belief. Since the approximation represents a local linearization, the approximation result strongly depends on the non-linearity (especially at the approximation point) and the uncertainty of the models. It is of huge importance that the uncertainties of the beliefs are kept low to guarantee a transformation through the linearized function close to the approximation point.

Different methods exist for the linearization of non-linear functions. As mentioned above, the Taylor series expansion is applied within the EKF, which is given by:

$$\begin{aligned} f(x) = f(x_0) &+ \frac{1}{1!}\left.\frac{\partial f(x)}{\partial x}\right|_{x=x_0}(x - x_0) \\ &+ \frac{1}{2!}\left.\frac{\partial^2 f(x)}{\partial x^2}\right|_{x=x_0}(x - x_0)^2 + \frac{1}{3!}\left.\frac{\partial^3 f(x)}{\partial x^3}\right|_{x=x_0}(x - x_0)^3 + \ldots, \end{aligned} \tag{3.26}$$

with x_0 representing the approximation point or also called *Taylor point.* As stated in Hoepcke (1980), if the approximation point represents the state x appropriately, terms higher than order 2 become comparably small and can consequently be neglected. Within the EKF only the first two terms are used, and therefore, the choice of the approximation point is of great importance. As assuming a Gaussian distribution, the mean of the posterior belief $\boldsymbol{\mu}_{t-1}$ (corresponding to $\hat{\boldsymbol{x}}_{t-1}$ within the stated Kalman filter algorithm) most likely represents the previous state within the state transition function and is therefore chosen as Taylor point for the dynamic model (Thrun et al., 2006):

$$\boldsymbol{\varphi}(\boldsymbol{x}_t) = \boldsymbol{\varphi}(\hat{\boldsymbol{x}}_{t-1}) + \underbrace{\left.\frac{\partial \boldsymbol{\varphi}(\boldsymbol{x}_t)}{\partial \boldsymbol{x}_t}\right|_{\boldsymbol{x}_t=\hat{\boldsymbol{x}}_{t-1}}}_{\boldsymbol{\Phi}_t}(\boldsymbol{x}_t - \hat{\boldsymbol{x}}_{t-1}), \tag{3.27}$$

$$\boldsymbol{\varphi}(\boldsymbol{x}_t) = \boldsymbol{\varphi}(\hat{\boldsymbol{x}}_{t-1}) + \boldsymbol{\Phi}_t(\boldsymbol{x}_t - \hat{\boldsymbol{x}}_{t-1}). \tag{3.28}$$

For the linear approximation of the measurement function, the same linearization is utilized using the prior belief $\tilde{\boldsymbol{x}}_t$ as approximation point:

$$\boldsymbol{h}(\boldsymbol{x}_t) = \boldsymbol{h}(\tilde{\boldsymbol{x}}_t) + \underbrace{\left.\frac{\partial \boldsymbol{h}(\boldsymbol{x}_t)}{\partial \boldsymbol{x}_t}\right|_{\boldsymbol{x}_t=\tilde{\boldsymbol{x}}_t}}_{\boldsymbol{H}_t} (\boldsymbol{x}_t - \tilde{\boldsymbol{x}}_t), \tag{3.29}$$

$$\boldsymbol{h}(\boldsymbol{x}_t) = \boldsymbol{h}(\tilde{\boldsymbol{x}}_t) + \boldsymbol{H}_t(\boldsymbol{x}_t - \tilde{\boldsymbol{x}}_t). \tag{3.30}$$

As mentioned, the matrices $\boldsymbol{\Phi}_t$ and $\boldsymbol{H}_t$ in equation (3.28) and (3.30) are Jacobian matrices containing all first-order partial derivatives of the functions, as given in equation (3.24) and (3.25). Once the linearization of the models is done, the EKF algorithm (Hofmann-Wellenhof et al., 2003; Grewal and Andrews, 2008) is similar to the linear Kalman filter version:

Prediction step / time update: Transition of the state

$$\tilde{\boldsymbol{x}}_t = \boldsymbol{\varphi}_t(\hat{\boldsymbol{x}}_{t-1}) \tag{3.31}$$

$$\tilde{\boldsymbol{\Sigma}}_t = \boldsymbol{\Phi}_t \boldsymbol{\Sigma}_{t-1} \boldsymbol{\Phi}_t^T + \boldsymbol{Q}_t \tag{3.32}$$

Gain computation: Computation of the Kalman weight

$$\boldsymbol{K_t} = \tilde{\boldsymbol{\Sigma}}_t \boldsymbol{H}_t^T (\boldsymbol{H}_t \tilde{\boldsymbol{\Sigma}}_t \boldsymbol{H}_t^T + \boldsymbol{R}_t)^{-1} \tag{3.33}$$

Measurement update: Correction of the predicted state

$$\hat{\boldsymbol{x}}_t = \tilde{\boldsymbol{x}}_t + \boldsymbol{K}_t(\boldsymbol{z}_t - \boldsymbol{h}_k(\tilde{\boldsymbol{x}}_t)) \tag{3.34}$$

$$\boldsymbol{\Sigma}_t = (\boldsymbol{I} - \boldsymbol{K}_t \boldsymbol{H}_t) \tilde{\boldsymbol{\Sigma}}_t \tag{3.35}$$

The EKF differs from the common KF approach in equation (3.31) and (3.34) due to the non-linearity of the underlying models. All other equations are identical.

Because of the linearization, all incorporated models should have low uncertainties. Therefore, referred to Grewal and Andrews (2008) and Thrun et al. (2006), accurate sensors with high update rates should be preferred to keep the uncertainty of the state small. Additionally, if the initial belief is chosen inadequately and the linearization does not describe the system appropriately, convergence may not be obtained.

Unscented Kalman filter

According to Haug (2012), the UKF utilizes a numerical instead of an analytical linearization and consequently, the calculation of Jacobians is not required. Through approximating the probability density function by discrete points, a non-linear transformation can be applied. As a result, the Gaussian distribution can be represented more efficiently within the UKF compared to the EKF. Thereby, so-called *sigma points*, which are symmetrically arranged around the mean of the probability density function, are defined. These points are transformed through the non-linear functions and finally, based on these points, an estimation of the mean and the covariance of the posterior belief is carried out using a weighted sum. Following Grewal and Andrews (2008), this sigma point implementation is also called *unscented transform.*

For the unscented transform, sigma points, which are extracted from Gaussian distributions, are passed through the non-linear dynamic and measurement functions. These sigma points are chosen to be at the main axes of the covariance. The number of sigma points is defined with $2n+1$ depending on the dimension n of the state. One of the sigma points is the mean

$$\boldsymbol{\chi}^{[0]} = \mu, \tag{3.36}$$

whereas, the sigma points $\boldsymbol{\chi}^{[1:n]}$ are located symmetrically to $\boldsymbol{\chi}^{[0]}$

$$\begin{aligned} \boldsymbol{\chi}^{[i]} &= \mu + \left(\sqrt{(n+\lambda)\Sigma}\right)_i, \qquad \text{for } i = 1, \ldots, n \\ \boldsymbol{\chi}^{[i]} &= \mu - \left(\sqrt{(n+\lambda)\Sigma}\right)_{i-n}, \quad \text{for } i = n+1, \ldots, 2n \end{aligned} \tag{3.37}$$

and depend on the covariance Σ and a scalar factor λ. Referred to Thrun et al. (2006), this factor is given by $\lambda = \alpha^2(n+\kappa) - n$ containing two scaling parameters α and κ. Besides the covariance Σ, these two scaling parameters are responsible for the distance to the mean. The choice of sigma points with respect to the dimension $n = 1$, $n = 2$, and $n = 3$ is illustrated in Figure 3.7. One of the sigma points is located at the mean, while two sigma points per dimension are symmetrically arranged around the mean with a distance from the mean which is given by the scaling parameters α and κ, and the covariance Σ.

After extracting the sigma points, a transformation through the non-linear function f is performed:

$$\boldsymbol{\gamma}^{[i]} = f(\boldsymbol{\chi}^{[i]}). \tag{3.38}$$

Then, the transformed sigma points $\boldsymbol{\gamma}^{[i]}$ act as regression points for a statistical linear regression which is the basis for the computation of the mean μ and the covariance Σ. The linearization is calculated in a way that preserves the true mean and the true covariance of the

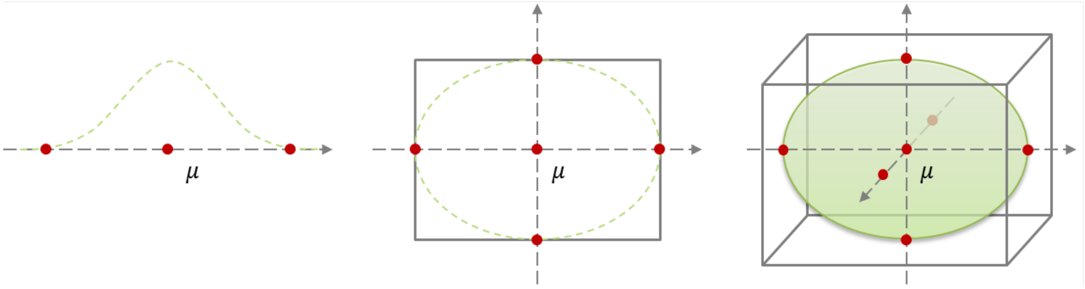


Figure 3.7: Sigma points for the one-dimensional (left), two-dimensional (center), and the three-dimensional case (right).

posterior distribution. Therefor, corresponding weights for the calculation of the mean and the covariance have to be determined. The weights $\omega_m^{[i]}$ are applied to the sigma points for the calculation of the mean μ, while $\omega_c^{[i]}$ are the weighting factors for the determination of the covariance Σ. The weights $\omega_m^{[i]}$ and $\omega_c^{[i]}$ for each sigma point are given by

$$\omega_m^{[0]} = \frac{\lambda}{n+\lambda}, \tag{3.39}$$

$$\omega_c^{[0]} = \frac{\lambda}{n+\lambda} + (1-\alpha^2+\beta), \tag{3.40}$$

$$\omega_m^{[i]} = \omega_c^{[i]} = \frac{1}{2(n+\lambda)}, \qquad \text{for } i = 1,\dots,2n \tag{3.41}$$

with β representing the similarity to a Gaussian distribution ($\beta = 2$ corresponds to the normal distribution). The mean μ' and the covariance Σ' of the Gaussian density of the mapped sigma points are obtained by

$$\mu' = \sum_{i=0}^{2n} \omega_m^{[i]} \gamma^{[i]}, \tag{3.42}$$

$$\Sigma' = \sum_{i=0}^{2n} \omega_c^{[i]} (\gamma^{[i]} - \mu')(\gamma^{[i]} - \mu')^T. \tag{3.43}$$

The UKF produces equal or better results than the EKF depending on the non-linearity of the dynamic and measurement functions and the spreading of the uncertainties. The reason therefor is, that the UKF considers terms that are of higher order than linear terms. Consequently, the UKF is more accurate than the EKF.

3.2 Non-parametric filters

Instead of approximating the Bayes filter by a parametric representation under the assumption of a Gaussian distribution, the non-parametric Bayes filters are realized by approximating the state space by a finite number of representations. Consequently, the state transition and measurement probabilities are only evaluated at discrete points in the space. The various non-parametric approaches of the Bayes filter differ in the way, how the state is approximated. Thereby, the quality of the approximation depends on the discretization of the space and the number of parameters which represent the posterior probability density function. As outlined in Thrun et al. (2006), the higher the number of state space representatives, the more the non-parametric technique tends to converge to the correct posterior. There are two main classes of non-parametric Bayes filter implementations:

- Grid-based estimation (histogram filter)
- Particle filter

Both estimation techniques are able to represent multimodal beliefs and do not need strong assumptions concerning the state transition and measurement model. The primary difference between these two filter approaches is the representation of the state space. While for the grid-based estimation the state space is divided into equally spaced grids, the particle filter is based on particles, which are located within the state space according to their probability. The following sections cover the algorithmic fundamentals and discuss the characteristics of each method. Since the grid-based estimation is restricted to the finite space, the particle filter is, because of its more diversified application, treated in more detail.

3.2.1 Grid-based estimation

As stated in Ristic et al. (2004), Woodman (2010), and Arulampalam et al. (2002), grid-based methods provide the optimal framework for estimating posteriors in a discrete and finite space. However, the grid-based estimation can also be applied for the continuous space by applying a discrete approximation of the probability functions (Burgard et al., 1996). Therefore, a decomposition of the state space into finitely many regions is performed. Commonly, an equally spaced grid is used to divide the possible state space into N discrete grid cells, whereby the centers of the grid cells are the discrete state representations. As a consequence, the estimated belief corresponds to discrete probabilities representing the likeliness of being in a certain grid. Following Fox et al. (2003), through the discretization of the state space, the integration within the state transition probability in equation (3.5) of the common Bayes filter is replaced by a summation. Applying a grid-based estimation, a fine-grained decomposition leads to smaller approximation errors, but requires a higher computational effort (Thrun et al., 2006).

The size of a grid cell has to be chosen with care, since it is fixed throughout the whole process. The recommended grid size strongly depends on the application, the structure of the environment, and the dynamic and measurement model. In particular, the motion of the tracked object and the uncertainty of the measurement data have to be considered for the definition of the grid size, since a strong correlation can be observed between these quantities and the approximation errors.

Within the grid-based estimation, the estimate is represented by discrete probability values

$$bel(\boldsymbol{x}_t) = \{p_{k,t} \,|\, k = 1, \dots, N\} \tag{3.44}$$

with $p_{k,t}$ providing the probability of the grid cell x_k at time t. Following Thrun et al. (2006), the state transition probability density for the discrete approximation is then

$$\overline{p}_{k,t} = \sum_i p(\boldsymbol{X}_t = \boldsymbol{x}_k | \boldsymbol{X}_{t-1} = \boldsymbol{x}_i) p_{i,t-1}, \tag{3.45}$$

whereby the variables $\boldsymbol{x}_k$ and $\boldsymbol{x}_i$ are individual states of the defined grid. The measurement update for the grid-based estimation provides the discrete probability $p_{k,t}$ of the grid cell's center $\boldsymbol{x}_k$ by

$$p_{k,t} = \eta_t \, p(\boldsymbol{z}_t | \boldsymbol{X}_t = \boldsymbol{x}_k) \, \overline{p}_{k,t}. \tag{3.46}$$

and corresponds to the correction step defined in equation (3.6) of the Bayes filter. The evaluation over each grid cell delivers the discrete probability distribution $\{p_{k,t}\}$. The conceptional design of the grid-based estimation is illustrated in Figure 3.8.

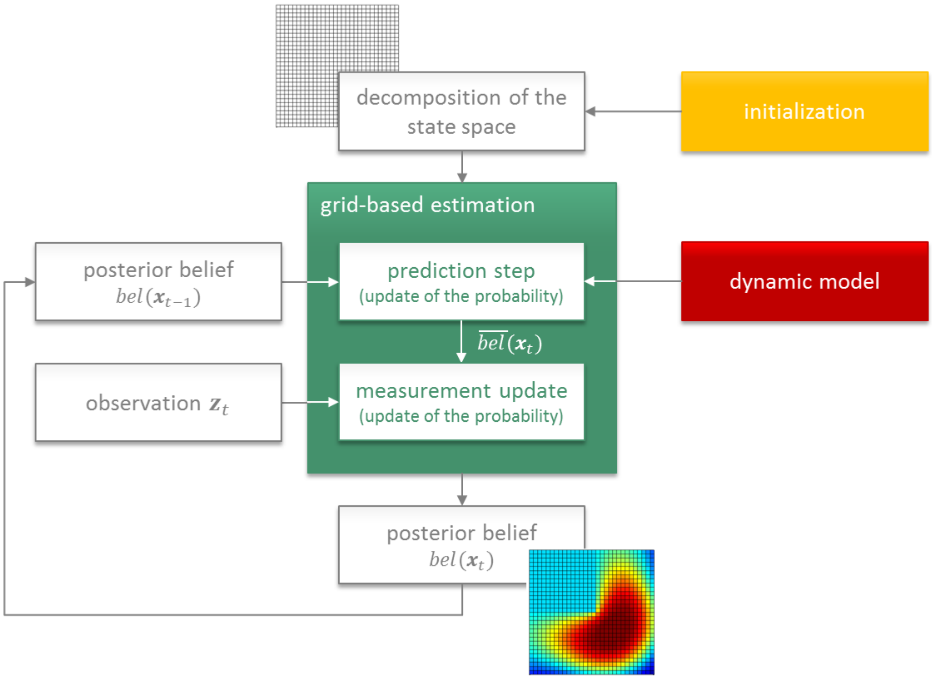

decomposition of the state space
initialization
grid-based estimation
posterior belief $bel(\boldsymbol{x}_{t-1})$
prediction step (update of the probability)
dynamic model
$\overline{bel}(\boldsymbol{x}_t)$
observation $\boldsymbol{z}_t$
measurement update (update of the probability)
posterior belief $bel(\boldsymbol{x}_t)$

Figure 3.8: Concept of the grid-based estimation.

In a first instance, the equally spaced grid representing the state space has to be initialized. A uniform distribution is suited for the initialization if no external information is available. Once the state space is decomposed and initialized, the estimation is done in a recursive manner executing first the prediction, which is followed by an update step. Within these steps, the probabilities are recomputed with respect to the appropriate models, however, the decomposition of the space remains the same. Figure 3.9 visualizes the probabilistic concept of the grid-based estimation based on the example of localizing a blind pedestrian. Again, the information of being at a crosswalk is incorporated in the estimation process.

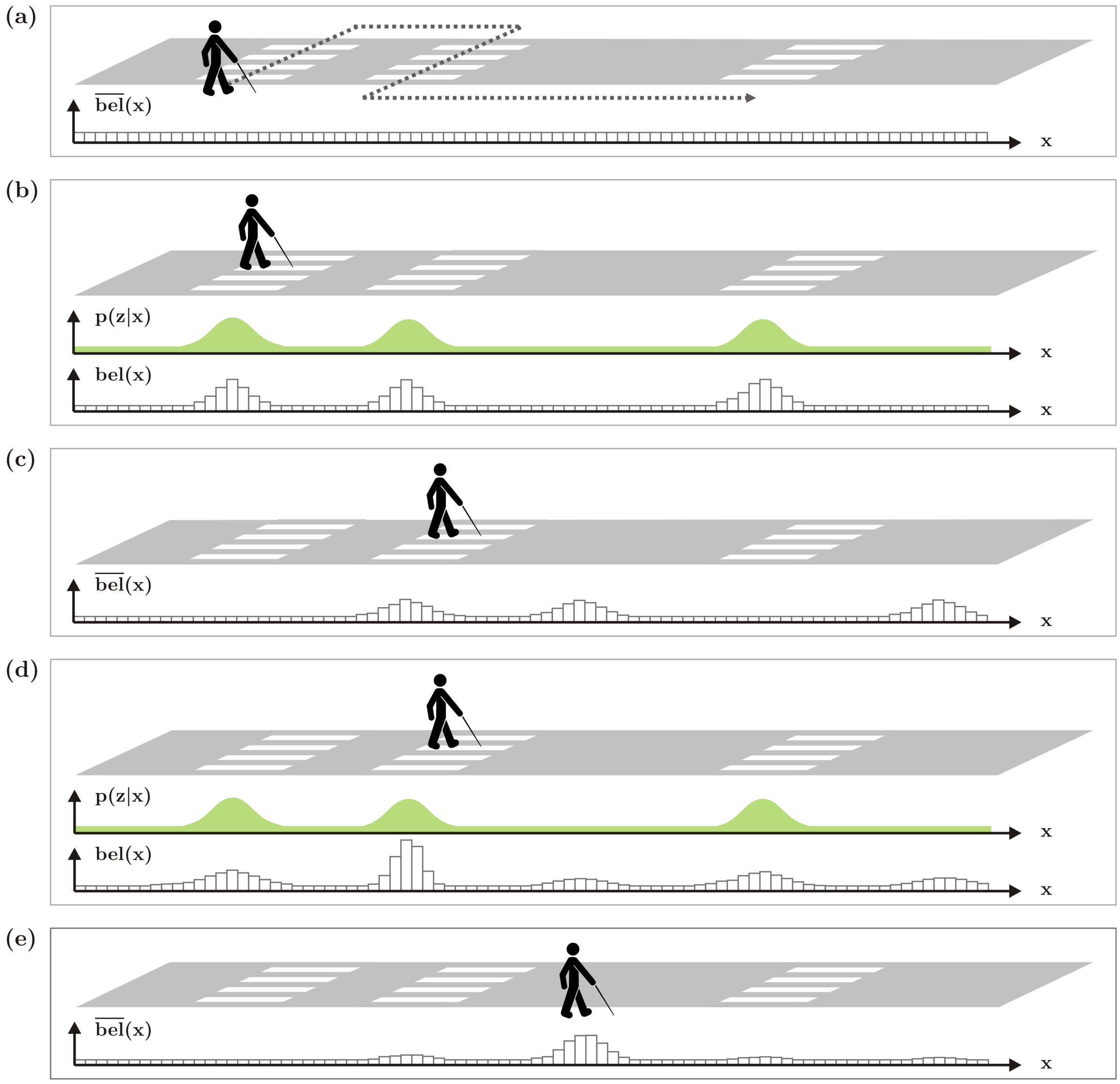

(a)
$\overline{\text{bel}}(x)$
x
(b)
p(z|x)
x
bel(x)
x
(c)
$\overline{\text{bel}}(x)$
x
(d)
p(z|x)
x
bel(x)
x
(e)
$\overline{\text{bel}}(x)$
x

Figure 3.9: Illustration of the grid-based estimation with respect to a blind person moving along the street. The probabilities are represented by discrete weights.

Instead of representing the space by a continuous probability function, the space is decomposed into a finite number of grids. Considering the one-dimensional case, a histogram with equally spaced classes is suitable for representing the location probabilities. For that reason, the grid-based estimation is also called histogram filter. As illustrated in Figure 3.9, the histogram classification for the prior and the posterior beliefs stays the same for every epoch. However, the probabilities change with regard to the state transition and measurement update.

Within the initialization of the filter, which is illustrated in graph (a) of Figure 3.9, the same probability is assigned to every grid cell. The map information is integrated as continuous density over the whole space, which is represented by the PDF in light green in graph (b) and graph (d). By updating the prior belief $\overline{bel}(x_0)$ by the map-based probabilities, a discretization takes place. The result of this step corresponds to the histogram in graph (b). Hence, the magnitude of a class represents the probability of a certain grid cell. The propagation with regard to the dynamic model, which is shown in graph (c), leads to a spreading of the probabilities caused by the uncertainty of the dynamic model.

As Figure 3.9 illustrates, the grid-based localization works similar to the general Bayes filter, but performs a discretization of the state space. The example points out, that multimodal distributions possessing more than one maximum can be algorithmically considered, which was not the case for the different approaches of the Kalman filter.

As mentioned in Fox et al. (2003), the main advantage of the grid-based estimation is the ability of handling arbitrary probability distributions and that there are no limitations concerning the propagation and measurement models. The discrete state representatives can be passed through propagation and measurement functions of any type. Additionally, due to the grid cell representation of the state space, map-based information can be integrated within the filter process. Since a map carries information which cannot be assigned to any distribution, a map integration would be impossible for the Kalman filter as it violates the Gaussian assumption. Furthermore, the grid-based estimation offers the possibility of using negative information, e.g., the person is not within the grid cell.

As outlined in Arulampalam et al. (2002), the drawback of this filter implementation is the necessary memory for N grid cells and the computational requirements for updating the state. For each update step, a summation over all N grid cells has to be carried out to calculate one individual weight. Consequently, one has to trade off the approximation errors coming with a coarse space decomposition against the computational costs correlated with a fine-grained grid.

Summarized, the recommendation concerning the use of the grid-based estimation tends to finite space applications with problems of low dimensionality, since the number of samples grows exponentially with the dimension of the state (Woodman, 2010).

3.2.2 Particle filter

In comparison to the grid-based localization, random samples called *particles* are used to represent the state space within the Particle Filter (PF). These particles are not equally spaced like the grid cells within the grid-based estimation. In contrast, they are distributed with respect to the belief's probability (Thrun et al., 2006). Consequently, the particle filter approximates the posterior belief $bel(x_t)$ by a set of N_s particles:

$$\chi_t = \left\langle \boldsymbol{x}_t^i \right\rangle, \qquad i = 1, ..., N_s. \tag{3.47}$$

For every sample $\boldsymbol{x}_t^i$ a so-called importance weight ω_t^i is determined within the measurement update. The importance weights thus represent the location probability of a particle. Based on these importance weights, the particles are redistributed in the space. As a consequence, the spatial distribution of the particles represents the probability distribution and respectively the belief $bel(\boldsymbol{x}_t)$. The new set of particles is the input for the state transition and the measurement update of the consecutive epoch. Figure 3.10 shows the individual steps of the particle filter resulting in an iterative process.

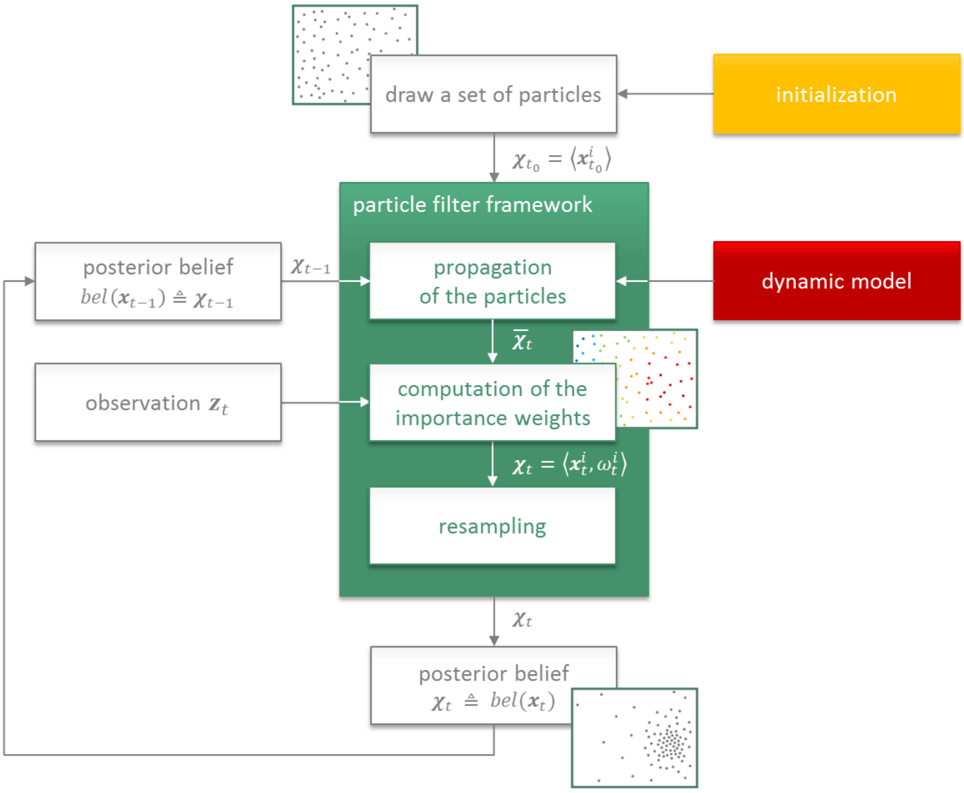

draw a set of particles
initialization
$\chi_{t_0} = \langle \boldsymbol{x}_{t_0}^i \rangle$
particle filter framework
posterior belief
$bel(\boldsymbol{x}_{t-1}) \triangleq \chi_{t-1}$
χ_{t-1}
propagation of the particles
dynamic model
$\overline{\chi}_t$
observation $\boldsymbol{z}_t$
computation of the importance weights
$\chi_t = \langle \boldsymbol{x}_t^i, \omega_t^i \rangle$
resampling
χ_t
posterior belief
$\chi_t \triangleq bel(\boldsymbol{x}_t)$

Figure 3.10: Concept of the particle filter.

At the beginning of the iteration, the particles have to be initialized. Either they are randomly distributed over the whole state space, or they may follow an appropriate distribution based on prior knowledge about the state. Regardless which kind of initialization is chosen, all initialized particles $\boldsymbol{\chi}_{t_0}$ have a uniform weight $\omega^i_{t_0} = 0$. After the initialization, the propagation of the sample set from one epoch to the next has to be carried out by applying the dynamic model. Compared to previous mentioned Bayes filter approaches, no restrictions concerning the transformation equations exist. Particle filters are qualified to handle linear as well as non-linear functions.

The set of particles $\overline{\boldsymbol{\chi}}_t$ obtained from the temporal transformation represents the prior belief $\overline{bel}(\boldsymbol{x}_t)$. As stated in Thrun et al. (2006), the transformed sample set is the basis for the computation of the importance weights ω^i_t, which are determined by incorporating the current measurements $\boldsymbol{z}_t$ by evaluating the observation model $p(\boldsymbol{z}_t|\boldsymbol{x}_t)$. According to the computed importance weights ω^i_t, a new set of equally weighted particles is generated within the *resampling* or *importance sampling*. Thereby, the resampling transforms the particle set into another set of samples by applying the Bayes rule on the prior belief $\overline{bel}(\boldsymbol{x}_t)$ as a function of the importance weight ω^i_t. Consequently, the new particle set is distributed according to $bel(\boldsymbol{x}_t) = \eta\, p(\boldsymbol{z}_t|\boldsymbol{x}_t)\, \overline{bel}(\boldsymbol{x}_t)$. Particle filter approaches which involve a resampling are known as *sequential importance sampling with resampling.* Details on the resampling can be found in the subsection on importance sampling.

The spatial distribution of the resampled particles represents the posterior belief $bel(\boldsymbol{x}_t)$. However, the posterior belief $bel(\boldsymbol{x}_t)$ can be derived from the spatial distribution of the resampled particles or by evaluating the weighted particle set determined within in the measurement update. Nevertheless, for the iterative state estimation it is of huge importance to resample the particles, since the performance of the filter strongly depends on the spatial disposition of the samples. As proposed in Thrun et al. (2006), the importance sampling is a probabilistic implementation of the Darwinian idea of *survival of the fittest*, focusing the computational resources to regions in the state space where it is required most.

Figure 3.11 illustrates the concept of the particle filter algorithm showing the example of the blind pedestrian moving along the street and crossing the street at zebra crossings. In the first step, when there is no information about the position of the pedestrian, the particles are randomly distributed over the whole state space and equally weighted, see graph (a).

The information about being at the zebra crossing is incorporated within the measurement update. The appropriate probability corresponds to the light green PDF in graph (b). The incorporation of the densities which represent the location probabilities of the crosswalks results in an importance weight for each particle. In graph (b), the particles equipped with importance weights correspond to the posterior $bel(\boldsymbol{x})$. Based on the importance weights, the particles are resampled in the space. The higher the importance weight, the more particles are positioned in this region. After the resampling of the particle set, the transition probability

$p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ is applied to propagate the particles from one epoch to the next. Graph (c) shows the resampled and already transformed particles, representing the prior belief $\overline{bel}(\boldsymbol{x})$. The propagated particles then act as input for the next measurement update. Graph (e) demonstrates that after a few iterations, most of the particles are located close to the true state. The example illustrates, that the particle filter automatically focuses the computation to regions where it is most needed.

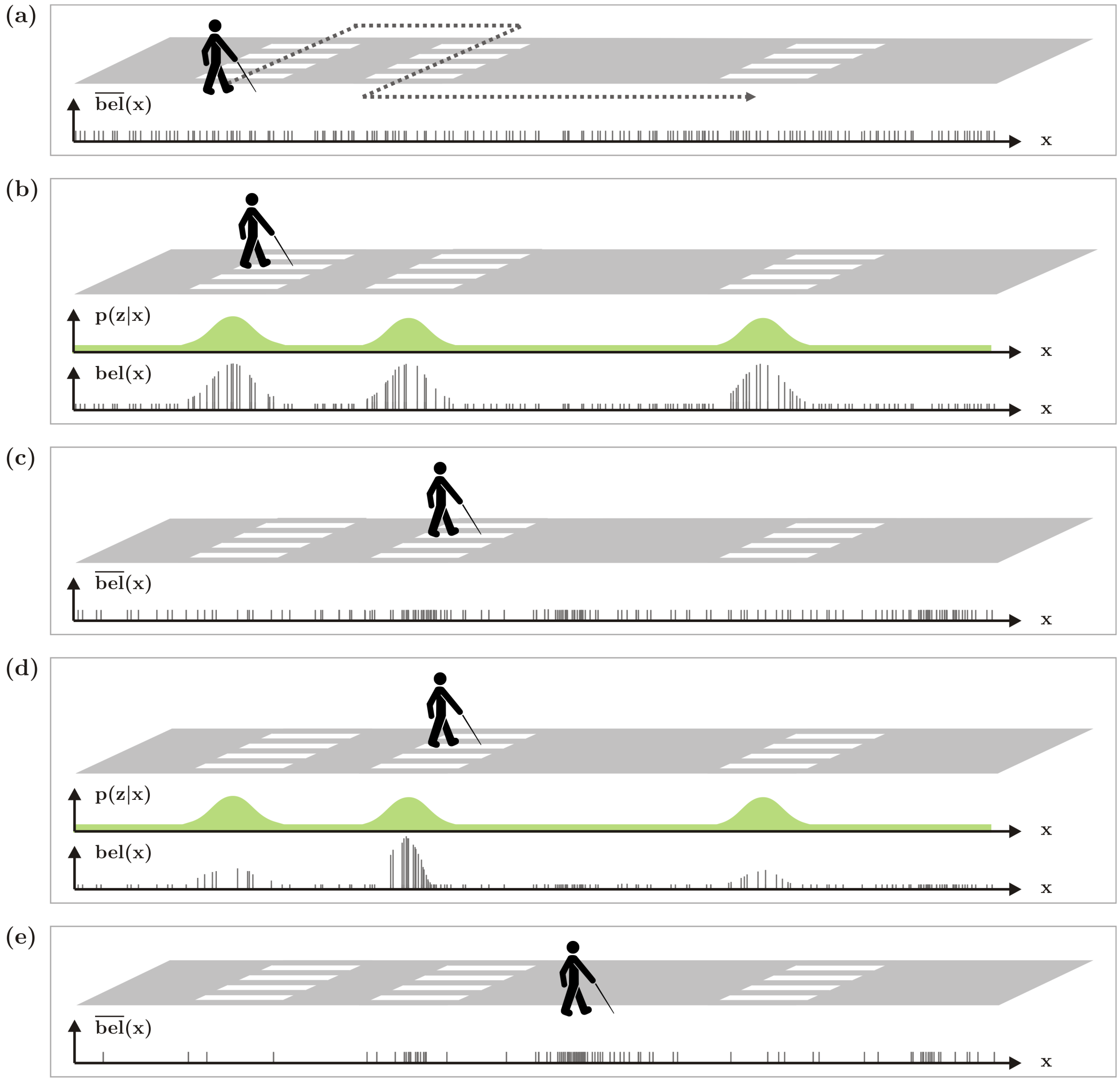

(a)
$\overline{\text{bel}}$(x)
x
(b)
p(z|x)
x
bel(x)
x
(c)
$\overline{\text{bel}}$(x)
x
(d)
p(z|x)
x
bel(x)
x
(e)
$\overline{\text{bel}}$(x)
x

Figure 3.11: Illustration of the particle filter with respect to a blind pedestrian moving along the street. The location probabilities are represented by the spatial distribution of the particles.

Due to the fact, that the particle filter offers the possibility to locate the particles in regions of higher probability, a better approximation result compared to the grid-based localization can be achieved with less computational effort. The number of spatial representatives in regions of low probability can be reduced, while the computation can be intensified in regions with higher probability. However, the more particles are generated, the better is the approximation of the posterior probability of the true state. A higher number of particles increases the probability that the posterior belief of the particle filter converges to the true one. Since the particle filter initializes the particles randomly, the output is not reproducible.

As stated in Fox et al. (2003), the advantage of the particle filter is its ability of representing arbitrary probability densities. The filter can converge to the true posterior even in non-Gaussian, non-linear dynamic systems. In contrast to the grid-based localization, the state space is not restricted to finite and discrete characteristics. The disadvantage of this filter approach is the limitation to problems of low dimensions. The computational costs grow with the dimension of the state and the number of particles.

Importance sampling

The goal of the particle filter is to generate particles which represent the posterior belief $bel(x_t)$ appropriately. The problem is, that the PDF $p(x)$ of the posterior belief is unknown. Consequently, the particles have to be drawn from a different but known PDF $q(x)$, the *proposal distribution*. As a consequence, a weighting factor $\omega(x)$ has to be determined to account for the mismatch between the *target distribution* $p(x)$ and the proposal distribution $q(x)$ (Thrun et al., 2006). As proposed in Haug (2012), the target distribution $p(x)$ is proportional to the PDF $q(x)$ at each value of x, so $q(x)$ is a scaled version of $p(x)$. The weighting factor $\omega(x)$ is the unnormalized ratio between these two densities and is defined by

$$\omega(x) = \frac{p(x)}{q(x)}. \tag{3.48}$$

The concept of importance sampling is required if the expectation $E\left[f(x)\right]$ of the random variable $f(x)$ needs to be determined, but the PDF $p(x)$ is unknown. The expectation $E\left[f(x)\right]$ based on the PDF $p(x)$ is defined by

$$E\left[f(x)\right] = \int f(x)\, p(x)\, dx. \tag{3.49}$$

As the probability distribution $p(x)$ is unknown, a known *proposal* distribution $q(x)$ is used to express the expectation of X by

$$E\left[f(x)\right] = \int f(x)\, \frac{p(x)}{q(x)}\, q(x)\, dx. \tag{3.50}$$

By introducing the weighting factor $\omega(x)$ for the quotient $p(x)/q(x)$, the expectation now depends on $q(x)$:

$$E\left[f(x)\right] = \int f(x)\,\omega(x)\,q(x)\,dx. \tag{3.51}$$

Figure 3.12 illustrates the determination of the importance factor $\omega(x)$ for the one-dimensional case based on a multimodal distribution $p(x)$ and a Gaussian distribution $q(x)$. It can be seen, that the target distribution $p(x)$ is proportional to $q(x)$ by $\omega(x)$.

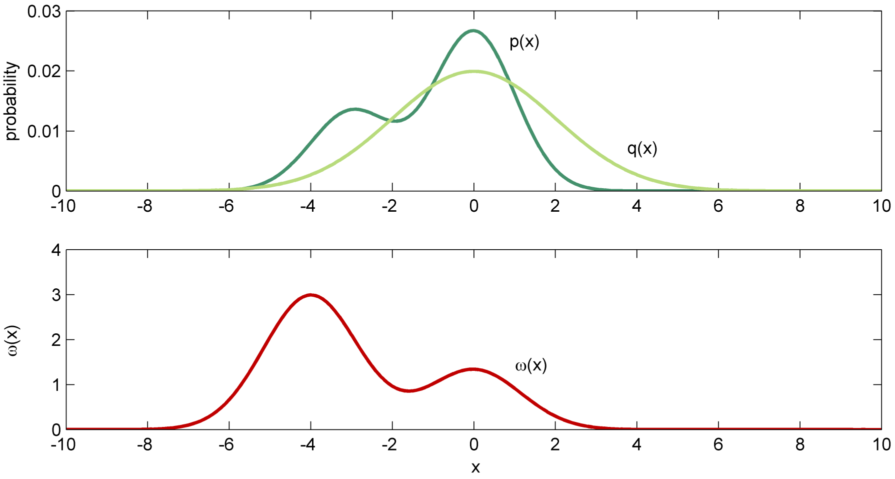


Figure 3.12: Illustration of the creation of the importance factor $\omega(x)$.

Considering a first-order Markov process whose posterior probability is given by $p(x_t|z_t)$, the expectation $E\left[f(x_t)\right]$ can be written as

$$E\left[f(x_t)\right] = \int f(x_t)\,p(x_t|z_t)\,dx_t. \tag{3.52}$$

Since the probability $p(x_t|z_t)$ is unkown, the expectation of $f(x_t)$ consequently has to be expressed by another probability function $q(x_t|z_t)$. The probability function $q(x_t|z_t)$ is known and is therefore suitable for sampling:

$$\begin{aligned} E\left[f(x_t)\right] &= \int f(x_t)\,\frac{p(x_t|z_t)}{q(x_t|z_t)}\,q(x_t|z_t)\,dx_t \\ &= \int f(x_t)\,\omega(x_t)\,q(x_t|z_t)\,dx_t, \end{aligned} \tag{3.53}$$

with

$$\omega(x_t) = \frac{p(x_t|z_t)}{q(x_t|z_t)}. \tag{3.54}$$

Applying now the Bayes' rule for $p(x_t|z_t)$, the weight $\omega(x_t)$ can be written as

$$\omega(x_t) = \frac{p(z_t|x_t)\,p(x_t|z_{t-1})}{p(z_t|z_{t-1})}\,\frac{1}{q(x_t|z_t)}. \tag{3.55}$$

As $p(z_t|z_{t-1})$ is a normalization term, $\omega(x_t)$ can be simplified to

$$\omega(x_t) = \eta_t\,\frac{p(z_t|x_t)\,p(x_t|z_{t-1})}{q(x_t|z_t)}. \tag{3.56}$$

Introducing $\omega(x_t)$ into equation (3.53) leads to

$$\begin{aligned} E\left[f(x_t)\right] &= \int f(x_t)\,\eta_t\,\frac{p(z_t|x_t)\,p(x_t|z_{t-1})}{q(x_t|z_t)}\,q(x_t|z_t)\,dx_t \\ &= \int f(x_t)\,\eta_t\,p(z_t|x_t)\,p(x_t|z_{t-1})\,dx_t. \end{aligned} \tag{3.57}$$

Comparing the two equations (3.52) and (3.57), it is obvious that

$$\begin{aligned} p(x_t|z_t) &= \eta_t\,p(z_t|x_t)\,p(x_t|z_{t-1}) \\ &= \eta_t\,p(z_t|x_t)\,\overline{bel}(x_t) \end{aligned} \tag{3.58}$$

corresponds to the measurement update of the general Bayes filter. The derivation of the formula of the measurement update shows, that the resampling based on importance weights enables an appropriate particle sampling even if the target distribution is unknown. Due to the resampling, the samples will be focused on regions of higher probability.

However, resampling leads to a loss of diversity in the particle population which is called *degeneracy problem*. The particles are only generated at the initialization of the filter and therefore, the variance of the particle set decreases with every resampling. As a consequence, one has to balance the risk of losing diversity with the circumstance of having many samples wasted in regions of low probability.

As stated in Haug (2012), a suitable measure for the degeneracy of the particles is the effective sample size N_{eff} which can be obtained by

$$N_{\mathit{eff}} = \frac{1}{\sum_{i=1}^{N_s}\left(\omega_t^i\right)^2}, \tag{3.59}$$

with N_s representing the number of particles. The smaller the effective sample size N_{eff} compared to the number of particles N_s, the higher is the degeneracy. $N_{\mathit{eff}} = 1$ corresponds to the case, that only one particle has a non-zero weight, while $N_{\mathit{eff}} \approx N_s$ indicates very little degeneracy (Haug, 2012).

There are several strategies to control the variance reduction. Two prominent methods are listed here:

– Reduction of the frequency of the resampling depending on the effective sample size N_{eff} (Thrun et al., 2006): This method maintains the importance weights in memory as long as no resampling is applied. Between two consecutive resampling steps the weights are updated by $\omega_t^i = p(z_t|x_t^i)\,\omega_{t-1}^i$. In case of a resampling, the importance factors become equally weighted again.

– Regularization based on a kernel density estimate (Haug, 2012): After each resampling step, the probability density of the particles is estimated. Based on the determined PDF, the particles are resampled once more, however, some samples are randomly distributed for improving the spreading over the state space.

Another problem caused by importance sampling, is the *particle deprivation*. The particle deprivation happens, if the number of particles is too small or the distribution is not appropriate to cover all regions with high likelihoods. This means that no particle is in the vicinity of the correct state. This problem can be circumvented by adding a small number of randomly generated particles to the sample set after each resampling process.

Rao-Blackwellization

Since the particle filter comes along with high computational efforts, high-dimensional estimation problems are hard to treat within a Sequential Importance Sampling (SIS) particle filter. Decreasing the computational load by reducing the number of particles deteriorates the accuracy of the estimation. Therefore, Haug (2012) proposes to use a *Rao-Blackwellization*, if the dynamic process can be classified into a Gaussian part and a non-Gaussian part. The idea of the Rao-Blackwell Particle Filter (RBPF) is to split the estimation process into a Kalman filter and a particle filter process. The posterior density of the Gaussian and linear components of the state are estimated within the Kalman filter, while the particle filter focuses on the processing of the non-Gaussian state components. As a consequence, by applying the Rao-Blackwell method the dimensionality of the particle filter can be reduced, nonetheless each component of the state can be estimated optimally.

3.3 Comparison of the different filters

This section introduced the Bayes filter with its parametric and non-parametric representatives. The Kalman filter with all of its expansions (extended and unscented Kalman filter) belongs to the parametric class of Bayes filters, while the particle filter and the grid-based estimation (histogram filter) are non-parametric Bayes filters.

In case that the dynamic and observation model are linear and correspond to the Gaussian class of distributions, the linear Kalman filter is the optimal estimation method. As outlined in Haug (2012), for Gaussian models with a slight to medium non-linearity, the extended Kalman filter is the right choice. The EKF approximates the non-linear functions by a first-order Taylor polynomial leading to a Kalman filter of the same form as the linear Kalman filter. In contrast, the unscented Kalman filter uses sigma points for the non-linear transformation. The mean and covariance estimate are determined by a weighted sum of the transformed points. This procedure produces a better estimate than the one gained by the EKF. Furthermore, no explicit calculation of Jacobians is required. Through the extraction of transformation points and the direct use of non-linear functions, the UKF performs better for high-order non-linearities compared to the EKF. However, the computational load of the UKF version increases with the dimension of the state, since the number of sigma points depends on the dimension of the state vector. Nonetheless, the UKF needs much less computational effort than the grid-based estimation and the particle filter.

Further expansions of the Kalman filter exist, but were not treated within this section. One example is the Monte Carlo Kalman Filter, which approximates the Gaussian density by a set of transformed Monte Carlo samples. This filter is similar to the unscented version of the KF, but the approximation performs even better because of the higher number of sample points representing the state space.

Kalman filters are, because of the moments representation, very time and memory efficient. However, no Kalman filter implementation determines the posterior probability appropriately, if the involved densities are non-Gaussian, multivariate, or unknown. Different non-parametric methods for handling arbitrary densities within the estimation process have been developed over the past decades. The basis for these methods is the extraction of samples which represent the state space. Due to the sample representation, the integrals of the Bayes filter can be replaced by weighted sums and therefore the processing becomes feasible. The samples can either be extracted as equally spaced grid (grid-based estimation) based on known densities, or, if the density is unknown, by importance sampling (particle filter). The advantage of non-parametric approaches, in which the state space is represented by samples, is the ability of integrating map information and state constraints.

The grid-based estimation is based on samples which represent an equally spaced grid. For the application of the grid-based estimation the state space has to be finite and discrete. For every iteration step, the dynamic and measurement model is used to update the posterior probability. As the decomposition of the state remains the same, the posterior belief is represented by the different probable grid points.

The particle filter decomposes the state space by considering the posterior probabilities, which means that the particles are distributed according to their probability. Thus, the computation is focused to regions where it is most required. As a consequence, less samples are needed for an appropriate state estimation. However, as stated in section 3.2.2, for the particle filter implementation, problems like degeneracy and particle deprivation have to be considered.

In Figure 3.13, the initialization and posterior density representation of the Kalman filter, the grid-based estimation, and the particle filter are opposed to each other.

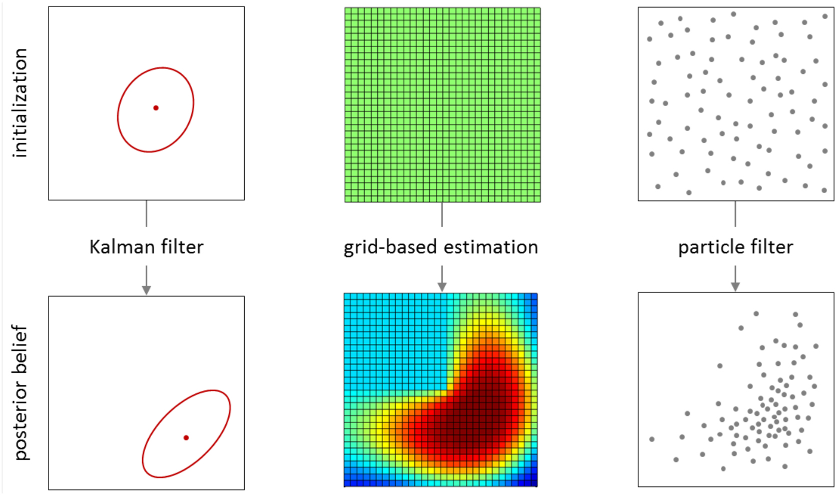


Figure 3.13: Initialization of the different Bayes filter implementations and the representation of the posterior belief.

Since the Kalman filter requires an initialization which conforms a Gaussian distribution, the filter has to be initialized by a mean and an appropriate covariance. The grid-based estimation is based on an equally spaced decomposition of the state space initialized with the same probability for each grid, while the particle filter randomly distributes equally weighted particles. The posterior probability is either given by a mean and a covariance (Kalman filter), probabilities over the equally spaced grid (grid-based estimation), or particles, the distribution of which corresponds to the posterior probability (particle filter). Since the extracted example is a two-dimensional problem, the covariance of the Kalman filter can be illustrated as an error ellipse with the mean as center.

Table 3.2 summarizes the properties of the different Bayes filter implementations with regard to the practical application for indoor and outdoor positioning. The computational requirements of the non-parametric approaches are higher compared to Kalman filter implementations, but their ability of handling non-linear models and arbitrary distributions predominates the computational drawback. Additionally, the grid-based (histogram) filter as well as the particle filter are perfectly suited for the integration of map information and other state space constraints (speed limits, boundaries, etc.). The Kalman filter cannot benefit from map information or spatial constraints due to its Gaussian assumptions. Unfortunately, the grid-based estimation is limited regarding the applicability for a global localization which is caused through the finite space representation. Consequently, the particle filter has to be favored for outdoor and indoor applications if the processing demands of the application are not too high. If some components of the state fulfill the Kalman filter assumptions, the Rao-Blackwell method is recommended to reduce the computational burden.

Table 3.2: Comparison of the different Bayes filter implementations, Kalman filter (KF), extended Kalman filter (EKF), unscented Kalman filter (UKF), grid-based estimation / histogram filter (HF), particle filter (PF), with respect to practical applications.

	KF	EKF	UKF	HF	PF
multi-modal distributions	-	-	-	++	++
non-linear functions	-	+	++	++	++
computational load	++	++	+	-	-
global localization	++	++	++	-	++
map integration	-	-	-	++	++

4 Smartphone-based positioning

As navigation systems specially tailored to the requirements of blind people are no mass market products, a custom-built development would be directly reflected in the costs of the navigation system. Therefore, the goal of the investigations was to keep the costs of the developed system low by using existing positioning possibilities which can be found in the sensors of a smartphone. Besides many built-in sensors for positioning, smartphones provide a lot of opportunities for the communication with the blind person (vibrations, speech). Additionally, smartphones feature low power consumption and high processing potentials and can be used anywhere at anytime. Consequently, smartphones with their comprehensive possibilities in the sense of positioning and computing seem to be the optimal device for the positioning/navigation of blind persons.

Within this thesis, complementary sensors with different characteristics are used to guarantee an accurate and reliable positioning with 100% temporal and spatial coverage. By using complementary positioning systems and principles, the disadvantages of one system can be compensated by the advantages of the others and vice versa. Current smartphones offer already a wide range of different sensors enabling a position estimation based on complementary principles. Modern smartphones comprise technologies like GNSS, WLAN, accelerometers, gyroscopes, magnetometers, and barometers, which may contribute to the determination of the current location, movement, and orientation of an object. The integration of the measurements of all the mentioned systems and sensors requires appropriate filtering algorithms (see chapter 3), which finally enable a reliable and continuous three-dimensional positioning in outdoor and indoor environments.

Outdoors, positioning can be performed via well-established GNSS techniques with accuracies ranging from a few decimeters up to some meters (Hofmann-Wellenhof et al., 2008; Sanz et al., 2013). However, conducting positioning outdoors with GNSS only does not satisfy the requirements of a reliable guidance especially of blind users. Compared to conventional pedestrian navigation, the quality requirements regarding availability, reliability, and redundancy are much higher. Accurate coordinates have to be provided continuously without any interruption, so that objects and locations relevant for guidance instructions can be localized by the blindman's stick.

To fulfill the accuracy requirements, either a sophisticated GNSS positioning like PPP or relative positioning (see section 4.1), or an integrated concept of position determination is indispensable. Within the integrated concept, the GNSS signals are augmented by different sensors (accelerometers, gyroscopes, magnetometers, etc.) or signals of opportunities (Chen, 2012b). Signals of opportunities are signals which are originally not intended for positioning purposes and comprise amongst others radio frequency signals from WLAN and Bluetooth networks. In outdoor environments, the integrated concept for pedestrian navigation is typically based on GNSS and inertial sensors. Since GNSS strongly depends on an open view to the sky, an integrated system has to be favored over a GNSS-only solution if, as it is the case for the positioning of blind people, a continuous positioning is of huge importance.

Nowadays, different sophisticated GNSS positioning algorithms exist which enable kinematic positioning with accuracies in the range of a few decimeters or even centimeters with GNSS data only. These methods are based on raw code and phase measurements and require, depending on the method, additional data like pseudorange corrections, measurements of a reference station, or precise orbit and clock products (section 4.1), which are provided by different online services. Nevertheless, accurate kinematic GNSS positioning is not possible with smartphones. The reason is simple. Current smartphones do not offer possibilities to observe the raw GNSS code and phase measurements. Therefore, only preprocessed coordinates based on Single Point Positioning (SPP) methods with code measurements are available. Consequently, the user has no influence on the processing method, and the position accuracy is equal to the accuracy of common code solutions, which is in the range of some meters.

Inside buildings, there is no GNSS equivalent, which enables positioning with an accuracy in the range of some meters 24 hours a day and is furthermore independent of especially installed infrastructure. Inertial sensors need no additional infrastructure, but have the huge drawback of drifting. That means that systematic errors are accumulated and therefore, the sensors have no long-time stability. In contrast, radio-based techniques require appropriate infrastructure, which has to be installed and maintained. Furthermore, the accuracy of radio-based methods is affected by obstructions and multipath. Consequently, ubiquitous positioning in indoor environments demands for integrated concepts based on complementary systems and sensors.

Based on the above mentioned facts, it can be summarized, that neither outdoors nor indoors, one single technology is capable to provide accurate position solutions at any circumstances at all times. As a consequence, the positioning tool, which is developed within the thesis, is based on complementary sensors. The core element of the positioning tool is a Rao-Blackwell particle filter (see section 3.2.2), which combines preprocessed GNSS, WLAN, accelerometer, gyroscope, magnetometer, and barometer data. The way from the raw sensor measurements through the preprocessing steps to the input of the filter is illustrated in Figure 4.1. Although the preprocessing of the measurement data is not part of the thesis, the principles of the different techniques are given in the course of this section, as these are the basis for the development of the filter. The thesis itself focuses on the optimal design and implementation of a positioning filter being part of a navigation system for visually impaired people. The filter combines the already preprocessed data, which are GNSS-based and WLAN-based coordinates, velocity and heading information estimated within a PDR algorithm, barometric height data, and an activity determined within the PAC. In the following, details on the filter's input data and the necessary processing techniques are given.

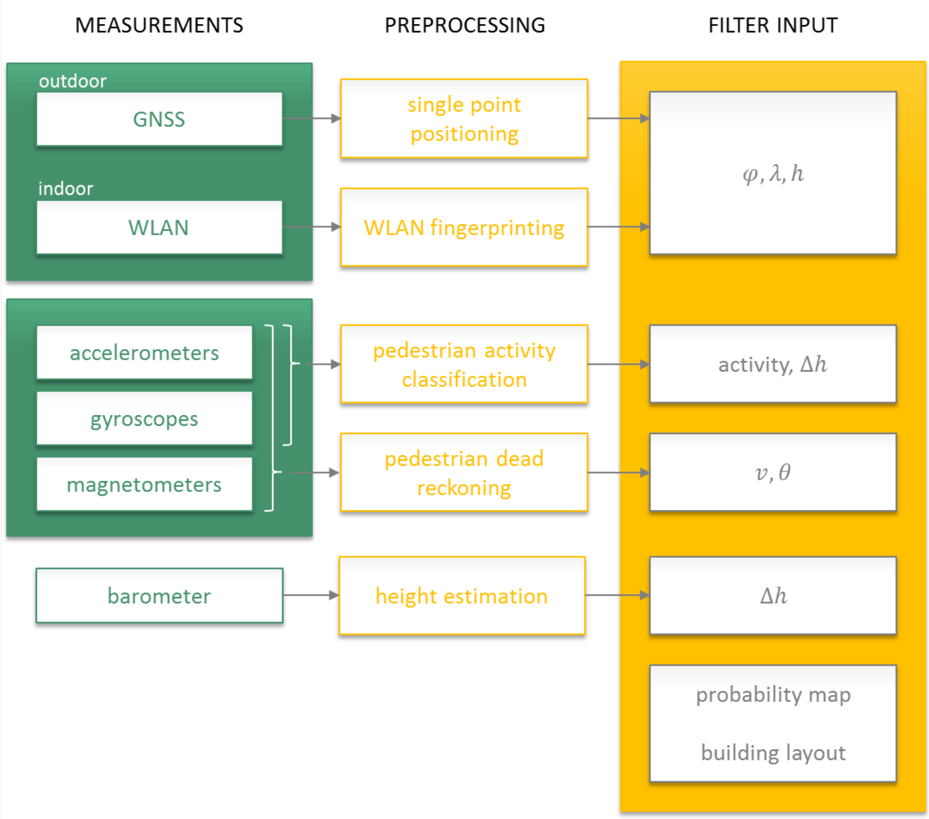


Figure 4.1: Input data of the positioning filter determined in preprocessing algorithms based on the smartphone sensor measurements.

While outdoors, the absolute positions are obtained from the internal GPS SPP software of the smartphone, indoors, WLAN fingerprinting provides the 3D coordinates $[\varphi, \lambda, h]$. Outdoors as well as indoors, heading and velocity data is used as support for the position estimation to improve the availability, reliability, and accuracy. The heading θ and the velocity v are computed in a PDR algorithm, which incorporates accelerometer, gyroscope, and magnetometer observations. As the deployed WLAN fingerprinting technique is focused on the two-dimensional positioning, and the geometric constellation given by the GNSS satellites disadvantages the height accuracy, neither outdoors nor indoors the height is determined appropriately. As a consequence, the height estimation further integrates a change in height Δh which is based on the barometer data.

Besides absolute and relative position information, the activity (*walking*, *standing*, *going upstairs*, and *going downstairs*) of the person acts as input for the filter. A PAC algorithm based on accelerometer and gyroscope data provides the current activity, which can be integrated within the filter in several ways. On the one hand, an additional relative height Δh and spatial constraints for the filter can be derived, and, on the other hand, the model for the step length estimation within the PDR can be tuned. The last input for the filter are different kinds of maps acting as environmental constraints within the filter. While the different sensor-related filter inputs are discussed in the course of this section, the incorporation of map information is described in detail in section 5.1.1 and 5.2.1.

The following sections treat the fundamentals, observables, and positioning algorithms of the sensors of common smartphones. GNSS-based and WLAN-based positioning are discussed in section 4.1 and 4.2. Details on inertial navigation based on accelerometer, gyroscope, and magnetometer data are given in section 4.3, while the barometric-based height estimation is treated in section 4.4. The principles of the PAC and its contribution to the height estimation are explained in section 4.5, while the incorporation of the activity as spatial constraint within the filter is shown in section 5.2.3.

4.1 GNSS positioning

The Global Navigation Satellite System (GNSS) provides 3D coordinates for the filter in outdoor environments. This chapter focuses on the satellite systems, which are affiliated with GNSS and the main principles of satellite-based positioning. Furthermore, details on different positioning methods like single point, differential, relative and precise point positioning are given. Although at the moment smartphones enable only code-based single point positioning, the more sophisticated techniques are explained as well, since these methods are considered for the SLAM-based probability-map generation discussed in chapter 7. Furthermore, the GNSS positioning on smartphones will be improved as soon as the reading of raw GNSS signals is available.

As stated in Sanz et al. (2013), the GNSS provides 3D coordinates with a global coverage and during 24 hours a day. Therefore, satellites orbiting the Earth are continuously transmitting signals and allow the determination of the user position. The term GNSS comprises all satellite systems contributing to global position determination. Within GNSS, fully operational, planned, or prospectively upcoming satellite positioning systems have to be distinguished. With 24 satellites orbiting the Earth in 1995, the US Global Positioning System (GPS) was the first fully operational GNSS system followed by the Russian GLONASS (Globalnaja Navigatsionnaja Sputnikovaja Sistema) in 1996. While GPS maintained the full operational status permanently till now, the number of operable GLONASS satellites dropped continuously because of missing satellite re-launches due to the reduced space funding. In 2001, the number of operational GLONASS satellites reached its lowest level with a constellation of only six remaining space vehicles, which made a continuous GLONASS-only positioning with global coverage impossible. Referred to the Information and Analysis Center for Positioning, Navigation and Timing (2014), in the year 2001, the Russian government decided the modernization of GLONASS which resulted in a full operability with 24 satellites in 2011. From 2011 up to now, GLONASS can be used as self-contained satellite navigation system independently from support of any other system (Information and Analysis Center for Positioning, Navigation and Timing, 2014).

The upcoming Galileo system is the European GNSS contribution, which can be used in an interoperable way together with GPS and GLONASS. Galileo has not reached the Full Operational Capability (FOC) up to now, however, according to the Information and Analysis Center for Positioning, Navigation and Timing (2014), the in-orbit validation phase with four operational satellites and the installation of the appropriate ground infrastructure was finished in 2012. In 2015, 18 satellites were expected to be in operation, thus Galileo would reach the FOC status (European Space Agency, 2012). As stated by the European Space Agency (2014b) and the International GNSS Service (2014b), the fifth and sixth satellites were launched 2014, but did not reach the nominal orbit because of launching problems. 2015, only six further satellite launches are planned (GPS World, 2015), consequently, the FOC status will not be reached in 2015. However, until 2020, 30 satellites should be in space, guaranteeing a visibility of six satellites everywhere on Earth.

According to the Information and Analysis Center for Positioning, Navigation and Timing (2014), Beidou (also known under the name COMPASS), the Chinese GNSS contribution, is currently under development and will provide global coverage in 2020. As stated by the China Satellite Navigation Office (2013), the Beidou constellation has been planned to include 35 satellites, comprising five Geostationary Earth Orbit (GEO) satellites and 30 non-GEO satellites. Three of the non-GEO satellites are in Inclined Geosynchronous Satellite Orbits (IGSO), while the rest is in a Medium Earth Orbit (MEO). The five GEO satellites will support the positioning by correction and integrity data, while the three IGSO satellites improve the signal availability in GNSS-critical areas (urban canyons, highly obstructed areas, etc.). As stated by the International GNSS Service (2014b), up to now, the Beidou constellation consists of five operable GEO, five IGSO and six MEO satellites.

Besides these four global systems, also regional satellite systems are installed to provide independent positioning or to support the global satellite positioning systems. The Japanese Quasi Zenith Satellite System (QZSS) is not intended as stand-alone system, but should rather complement the global GNSS in particular in the region of Japan (European Space Agency, 2014a). Corresponding to the Japanese Aerospace Exploration Agency (2014) and Chen (2012b), three satellites being visible in Japan at very high elevation angles and transmitting GPS compatible signals are planned to improve the GPS performance. The QZSS signals increase the number of available satellite signals most notably in urban areas of Japan and Asian Pacific regions, what consequently enables continuous positioning at anytime by combining GPS and QZSS signals. Furthermore, QZSS aims at improving the position accuracy by transmitting appropriate correction data. According to the European Space Agency (2014a), the first QZSS satellite named *Michibiki* was brought into orbit in 2010, while the others are scheduled to be launched before the end of 2017.

In contrast to QZSS, the Indian Regional Navigational Satellite System (IRNSS) is a regional independent satellite navigation system enabling positioning in a surrounding region of 500 km around India. As stated in Majithiya et al. (2011), the planned system should consist of seven satellites (three geostationary and four geosynchronous satellites). IRNSS is expected to provide 3D coordinates with an accuracy of about 10 m in central India and up to an accuracy of 20 m over the Indian Ocean region. The aim of IRNSS is to improve the Dilution of Precision (DOP) value and the number of visible satellites, leading finally to an increased accuracy. As stated by the European Space Agency (2014a), the first satellite was successfully launched in 2013, while the launch of the remaining six is planned to be between 2015 and 2016.

The global and regional satellite systems are additionally supported by a so-called Satellite Based Augmentation System (SBAS). SBAS is a generic term for different region-based augmentation systems which are realized by geostationary satellites. They are continuously visible at certain areas and broadcasting augmentation data comprising correction data and integrity information (European Space Agency, 2014a; European GNSS Agency, 2014). Figure 4.2 displays the existing and prospective augmentation systems and their area of operation. There are three systems which are already operational. The American Wide Area Augmentation System (WAAS), the Japanese MSAS (MTSAT Satellite Augmentation System) and the European EGNOS (European Geostationary Navigation Overlay Service) provide correction data for the visualized areas in Figure 4.2. According to the International GNSS Service (2014b), the Indian GAGAN (GPS Aided GEO Augmented Navigation System) and the Russian SDCM (System for Differential Corrections and Monitoring) are in a test phase. Additionally, South and Central America, Malaysia, Africa, and South Korea have separately announced to start their own SBAS implementations.

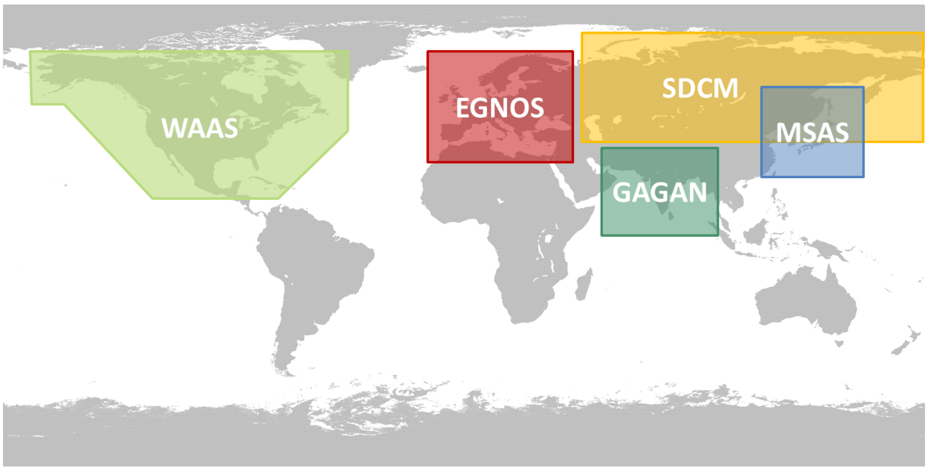


Figure 4.2: Existing SBAS and their covering area.

By combining different global and regional systems, the number of available GNSS satellites can be strongly increased, which leads to improved DOP values and enhanced integrity. The integration of different GNSS is of high importance especially in urban and GNSS-denied areas, since for satellite-based positioning at least observations of four satellites are necessary. According to Chen (2012b), in the case that all GNSS systems will be operational, more than 100 satellites will be in space, which will increase the positioning availability in GNSS-critical areas significantly.

4.1.1 Principles of GNSS positioning

The GNSS satellites are continuously transmitting radio frequency signals consisting of three components - the carrier, the ranging code, and the navigation data. The ranging code allows the determination of the travel time of the signal, while the navigation data provides the orbital parameters for the computation of the satellite positions (Sanz et al., 2013). The ranging code as well as the navigation data is modulated onto the carrier signal. GPS, GLONASS, Galileo, and Beidou are all providing several carriers with different frequencies (Hofmann-Wellenhof et al., 2008; Groves, 2013).

The basic GNSS observation for positioning is the travel time Δt of the signal transmitted by the satellite. The travel time Δt can be converted to the traveled distance by multiplying the travel time with the travel speed, which is the speed of light (Sanz et al., 2013). As stated in Hofmann-Wellenhof et al. (2008), by observing distances between the user and at least three satellites, the position of a user can theoretically be determined since the satellites coordinates are given by the orbital parameters. However, as GNSS is based on a *one-way-concept*, two different clocks are involved - one at the receiver and one at the satellite site. Because of the non-synchronization of the receiver and satellite clocks, the ranging code does not represent the geometric distance between the satellite and the receiver. Instead, so-called *pseudoranges* are observed. Therefore, an additional parameter representing the clock error has to be estimated besides the receiver coordinates to get a reliable solution. However, it has to be mentioned that the clocks of the satellites within one GNSS are synchronized by atomic standards. The remaining offset, which corresponds to the synchronization residual, is given in the navigation data (Hofmann-Wellenhof et al., 2008).

The principal equation of GNSS positioning at the observation epoch t is given by

$$R_r^s(t) = \rho_r^s(t) - [\delta t_r(t) - \delta t^s(t)]\, c, \tag{4.1}$$

with

$$\rho_r^s(t) = \sqrt{[X^s(t) - X_r(t)]^2 + [Y^s(t) - Y_r(t)]^2 + [Z^s(t) - Z_r(t)]^2}, \tag{4.2}$$

where R_r^s represents the measured pseudorange, and ρ_r^s is the geometric distance between the satellite s and the receiver r. The terms δt_r and δt^s are the receiver and satellite clock error, and c is the speed of light. The coordinates of the satellites and the receiver are specified by $[X, Y, Z]^s$ and $[X, Y, Z]_r$ respectively. Since the navigation data delivers $[X, Y, Z]^s$ of each satellite and the satellite clock error δt^s, four simultaneously observed pseudoranges allow the estimation of the unknown parameters $[X, Y, Z]_r$ and δt_r, which is realized within a parameter adjustment (Bauer, 2011). The concept for positioning with several GNSS is similar, however, one additional parameter (for each additional system) has to be estimated representing the time difference between the GNSS time systems.

The accuracy of the Single Point Positioning (SPP) technique based on ranging code is in the range of several meters (Hofmann-Wellenhof et al., 2008). Applying more advanced techniques, the accuracy can be improved up to a few centimeters. However, these techniques need a more detailed differentiation of the parameters in equation (4.1), considering the different GNSS error sources. The path of the GNSS signal through the atmosphere, or more precisely, the ionosphere and the troposphere, is influenced by refraction. Additionally, the orbit and clock parameters given in the navigation data are predicted values and are therefore restricted in their accuracy. At the receiver site, multipath and antenna phase center variations are common error sources. All these errors may be classified with regard to three main sources of errors. As categorized in Table 4.1, the errors are either related to the satellite or receiver site, or are caused by the propagation medium. Besides these influences, further impacts induced by, e.g., relativistic and tidal effects exist, but will not be considered further because of their small magnitudes.

Considering the error sources of Table 4.1, the observation equation for the ranging code can be defined as a function of the geometric range and various terms which represent the error influences on the distance:

$$R_r^s(t) = \rho_r^s(t) + \Delta\rho_r^s(t) + \Delta\rho^s(t) + \Delta\rho_r(t). \tag{4.3}$$

The terms $\Delta\rho_r^s$, $\Delta\rho^s$, and $\Delta\rho_r$ correspond to the errors dedicated to the propagation medium, the satellite, and the receiver respectively (Hofmann-Wellenhof et al., 2008).

Table 4.1: Classification of the error sources of satellite-based positioning

satellite	clock error
	orbital error
propagation medium	ionospheric refraction
	tropospheric refraction
receiver	clock error
	multipath
	antenna phase center variation

Differential positioning, which is also denoted as *Differential GPS* (DGPS) in literature, reduces the influence of satellite- and propagation-based errors by applying correction data, which is provided by so-called reference stations or satellite-based augmentation systems. Within relative positioning, differencing techniques are used to account for error sources like the atmosphere, imprecise orbit and clock parameters and additional unmodeled effects. While for differential as well as relative positioning the data of several receivers are needed, PPP benefits from precise orbit and clock products provided by various GNSS services. The accuracy of DGPS is in the range of a few meters, while with relative positioning or PPP accuracies of a few decimeters or even centimeters can be reached.

Due to the promising accuracies of relative and precise point positioning, these techniques are perfect candidates for the SLAM-based map generation. However, DGPS and PPP require less data for the improvement of the position accuracy, which is advantageous in the case of smartphone applications. Comparing all properties of these three techniques, PPP turns out to be the optimal candidate for the SLAM-based probability map generation if raw GNSS measurements are available. While the following sections on differential, relative and precise point positioning provide more details on the different positioning techniques, chapter 7 treats the applicability of these techniques for the SLAM algorithm.

4.1.2 Differential positioning

For differential positioning at least two GNSS receivers are needed, whereby the coordinates of one static receiver (master/reference station) have to be known, while the coordinates of the rover need to be determined. The known coordinates and the static measurements at the master station are used for the computation of corrections for the pseudoranges. These corrections are modeled as *pseudorange corrections* and *range rate corrections* and are transmitted to the rover (Hofmann-Wellenhof et al., 2008). By applying these corrections at the rover, atmospheric influences as well as satellite orbit and clock errors can be reduced, which consequently improves the position accuracy to 1 to 5 meters. The condition for DPGS is the availability of at least the same four satellites at both receivers.

The pseudorange corrections can be determined by introducing the known coordinates of the master station into equation (4.3). The pseudorange correction PRC^s for one particular satellite s at a reference epoch t_0 is therefore defined by

$$\begin{aligned} PRC^s(t_0) &= \rho^s_{master}(t_0) - R^s_{master}(t_0) \\ &= -\Delta\rho^s_{master}(t_0) - \Delta\rho^s(t_0) - \Delta\rho_{master}(t_0). \end{aligned} \tag{4.4}$$

In addition to the pseudorange correction $PRC^s(t_0)$, a range rate correction $RRC^s(t_0)$ is determined to model the temporal change of the pseudorange correction, which allows finally the computation of the pseudorange correction for the observation epoch t:

$$PRC^s(t) = PRC^s(t_0) + RRC^s(t_0)(t - t_0). \tag{4.5}$$

The application of the correction on the measurements of the rover by

$$\begin{aligned} R^s_{rover}(t)_{corr} &= R^s_{rover}(t) + PRC^s(t) \\ &= \rho^s_{rover}(t) + [\Delta\rho^s_{rover}(t) - \Delta\rho^s_{master}(t)] + [\Delta\rho_{rover}(t) - \Delta\rho_{master}(t)] \end{aligned} \tag{4.6}$$

eliminates the satellite-based errors $\rho^s(t)$ and reduces the amplitude of the error term caused by the propagation medium $\rho^s_{rover}(t)$. The smaller the distance between the master and rover station, the more similar is the atmospheric effect. Consequently, the closer the stations, the better is the reduction of the error $\Delta\rho^s_{rover}(t)$. Therefore, the DGPS accuracy strongly depends on the distance to the master station, which is recommended to be less than 150 km (Hofmann-Wellenhof et al., 2008). The accuracy can be further improved by installing a net of reference stations and performing an interpolation of the correction data. As stated in Groves (2013), DGPS can be realized by self-installed reference stations, various services based on networks of reference stations, or by satellite-based augmentation systems. Examples for differential positioning services in Austria are *EPOSA* (EPOSA, 2015) and *APOS* (BEV - Bundesamt für Eich- und Vermessungswesen, 2015).

4.1.3 Relative positioning

Applying relative positioning techniques, accuracies of some centimeters can be achieved for the kinematic case. However, for reaching such accuracies the carrier phase has to be used as observation instead of the ranging code (Hofmann-Wellenhof et al., 2008). Using phase measurements, for each satellite an additional time-constant parameter, the phase ambiguity N, has to be estimated, since only the fractional phase Φ of the signal can be observed. The phase-based distance ϕ between the receiver r and the satellite s is

$$\phi_r^s(t) = \Phi_r^s(t) + N^s. \tag{4.7}$$

The observation equation of the phase measurement consequently includes an additional parameter for the phase ambiguity:

$$\Phi_r^s(t) + N^s = \rho_r^s(t) + (\delta t_r(t) - \delta t^s(t))c. \tag{4.8}$$

Similar to DPGS, at least two receivers are needed, whereby one has to be static, located at a known position and furthermore, simultaneous measurements from both receivers to the same satellites must be available. The simultaneous measurements are used to build so-called single, double, and triple differences which correspond to combinations of measurements of different receivers to different satellites at different epochs (Groves, 2013). By subtracting the phase measurements, receiver-based and satellite-based errors are eliminated, while the reduction of the atmospheric impact strongly depends on the length of the baseline between the receivers (Hofmann-Wellenhof et al., 2008). The high accuracy improvement cannot only be accredited to the differencing but also to the use of phase measurements, which are more accurate than code measurements.

4.1.4 Precise point positioning

Precise Point Positioning (PPP) in comparison to differential and relative positioning needs no reference stations and therefore only one receiver. The PPP technique allows accurate point positioning with accuracies in the range of a few decimeters (kinematic positioning) or even centimeters (static positioning) by applying precise orbit and clock parameters instead of the parameters given in the broadcasted navigation data. For this reason, PPP needs an accurate measurement modeling.

As stated in Huber et al. (2011), the huge advantage of PPP is the independency from local reference stations, since the corrections and parameters are globally valid, and consequently, the operational range is not restricted. Furthermore, PPP needs no simultaneous measurements and can be performed with code or phase measurements or in combination. However, for high-accuracy positioning, the use of phase data is indispensable (Sanz et al., 2013). Referring to Hofmann-Wellenhof et al. (2008), using the phase data of two carriers, the ionospheric influence can be reduced by an ionosphere-free linear combination, while with a single frequency, additional information is needed to reduce the impact of the ionosphere. Neglecting the ionospheric influence would degrade the accuracy by several meters.

To achieve accuracies in the range of a few decimeters in case of kinematic applications, PPP demands for a very accurate measurement modeling that includes all sources of errors. Figure 4.3 shows the main GNSS error sources, which have to be either modeled or eliminated within PPP, whereby the multipath is an effect of GNSS which is location-dependent and can unfortunately not be corrected by PPP. However, PPP covers the modeling/elimination of precise satellite orbits and clocks, atmospheric effects, as well as antenna biases. Details on the processing are given in the following paragraphs.

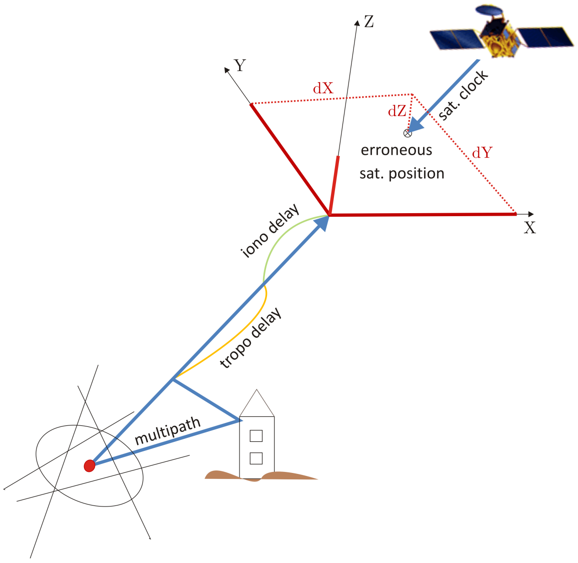


Figure 4.3: Overview of the main GNSS error sources (Huber et al., 2011).

Precise satellite orbits and clocks are provided by services like the International GNSS Service (IGS) and are applied instead of the broadcasted parameters. The accuracy of the products depends on the time span between the observation and the data provision. Detailed information about all available IGS products can be found on the IGS website (International GNSS Service, 2014a). The accuracy of ultra-rapid orbits, which are available in real time, is specified with 5 cm, while the broadcast ephemerides are in the range of 1 m. The accuracy of the satellite clocks improves by 2 ns if precise clock data is used.

Atmospheric effects have to be modeled or reduced. If dual-frequency data are given, an ionosphere-free linear combination is performed, while for the processing of single-frequency data TEC maps of the IGS are used to reduce the influence of the ionosphere. The tropospheric refraction can be modeled by different models, Sanz et al. (2013) recommends the mapping of Niell.

Antenna biases of the satellites and the receivers have to be considered, which are documented in ANTEX files, also available at the IGS.

As stated in Sanz et al. (2013) and Huber (2015), to reach cm-level accuracy for static positioning and dm-level or better for kinematic positioning, additional influences like relativistic effects, differential code biases, carrier phase wind-up effects, etc., have to be taken into account, and the ambiguities of the phase measurements have to be fixed.

Because of the globally valid models and corrections, PPP is favored over differential and relative positioning for the SLAM-based probability map generation. While DGPS does not meet the accuracy requirements for the map generation, relative positioning needs measurement data of a second receiver and comes along with higher computational efforts.

4.2 WLAN positioning

Most mobile devices (laptops, tablets, smartphones, etc.) have integrated WLAN network interface cards (Kushki et al., 2012), which enable the use of the Wireless Local Area Network (WLAN) infrastructure for position estimation purposes. Therefore, radio signals which are exchanged between the Wireless Local Area Network (WLAN) access points and the mobile device are observed. The positioning techniques reach from basic *cell-of-origin* approaches, *trilateration*, *angulation* to the most common applied technique *fingerprinting*. WLAN provides position accuracies in the range of some meters and can be applied indoors as well as outdoors.

WLAN access points periodically broadcast beacon signals including a time stamp, path loss information, and supported data rates. For the position determination, the strength of these signals, which is called Received Signal Strength (RSS), is measured. Besides this parameter, WLAN signal features like Angle of Arrival (AoA), Time of Arrival (ToA), and Time Difference of Arrival (TDoA) may be used for positioning (Kushki et al., 2012). Depending on the measured features, different techniques are applied to get the user's location. In Table 4.2, all WLAN observables and the corresponding positioning methods are listed.

Table 4.2: WLAN measurements and possible positioning methods

Measurements	Positioning methods
Received Signal Strength	cell-ID, enhanced cell-ID, trilateration, fingerprinting
Angle of Arrival	enhanced cell-ID, triangulation
Time of Arrival	trilateration, fingerprinting
Time Difference of Arrival	hyperbolic trilateration

According to Chen et al. (2012), the observables AoA, ToA, and TDoA require additional or modified hardware, whereas the RSS can be derived from existing observables including the Media Access Control (MAC) address and the Received Signal Strength Indicator (RSSI). While the MAC address unambiguously identifies the access point, the RSSI is an indication of the power level received by the antenna. Since this thesis focuses on positioning with common smartphones without any additional hardware, this section mostly treats RSS-techniques, while the other methods are only described briefly.

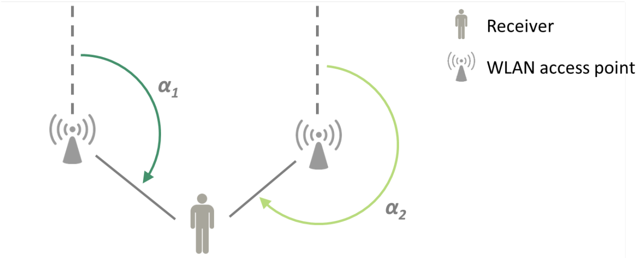
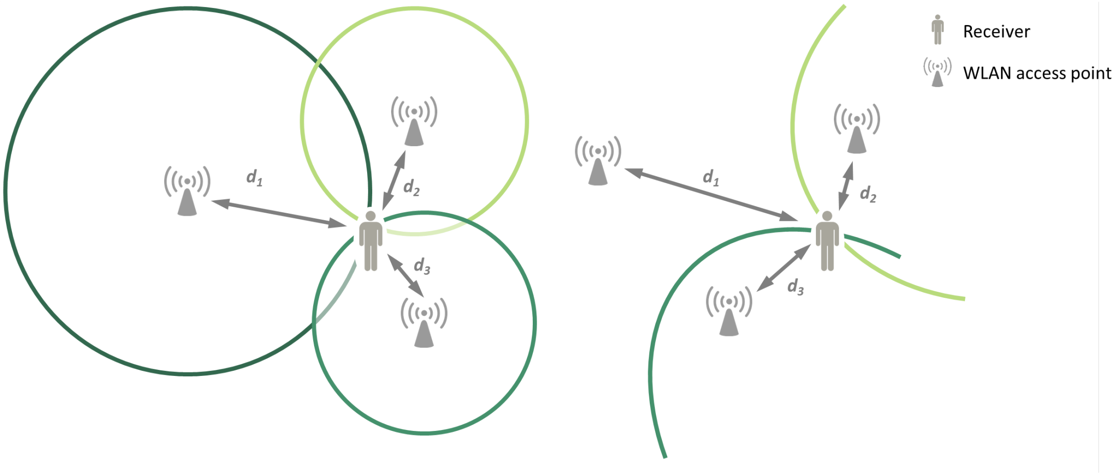


Figure 4.4: Angulation based on AoA observations.

As shown in Figure 4.4, with two AoA measurements, the 2D coordinates of the user can be determined by angulation. The principle of angulation is the computation of the intersection point of the appropriate Lines Of Position (LOP). In case of AoA measurements, the LOP corresponds to a straight line. This technique requires a special antenna or antenna arrays (Chen et al., 2012).



(a) Lateration based on ToA observations.

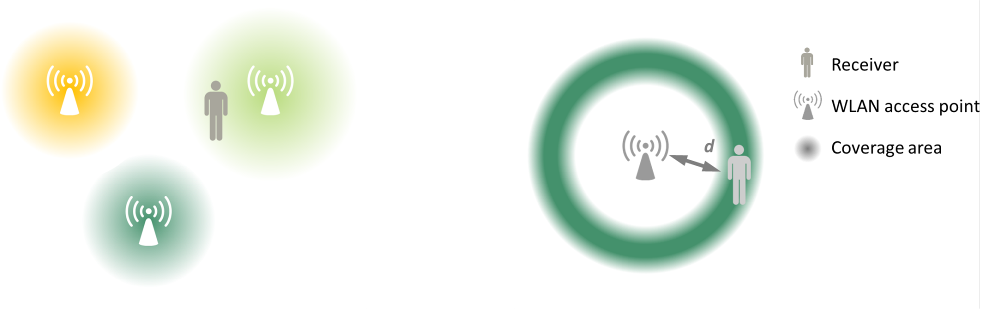
(b) Hyperbolic positioning based on TDoA observations.

Figure 4.5: Positioning with ToA and TDoA observations.

The ToA corresponds to the travel time of the radio signal, which can be converted to a distance if the propagation speed is known. As shown in Figure 4.5a, lateration with three distances provides the coordinates of the user. By performing hyperbolic lateration with TDoA measurements, which can be derived from ToA observations, the influence of errors can be reduced. Hyperbolic positioning needs at least three ToA measurements providing two TDoA observables, see Figure 4.5b. For TDoA as well as ToA positioning, a synchronization at least among the transmitters is needed, which may be realized with a GNSS receiver or a location measurement unit (Pahlavan and Krishnamurthy, 2013).

Based on the RSSI and the MAC address of every observable access point, cell-of-origin, multilateration, and fingerprinting techniques can be applied. As stated in Chen et al. (2012), the cell-of-origin or cell-ID method was the first RSSI-based positioning implementation. The principle of this method is, that the user coordinates correspond to that of the nearest WLAN access point (Figure 4.6a), which is the access point with the strongest RSSI. The position accuracy is equal to the radius of the circular signal coverage error.

The accuracy can be improved by using the actual signal strength, which can be converted to a distance, in addition to the assigned cell. The Surface of Position (SOP) can consequently be reduced to a doughnut, see Figure 4.6b. The combination of the cell-ID and an additional observation is denoted enhanced cell-ID positioning (Chen, 2012a; Bensky, 2008). Using a sectorizational antenna, the direction of the signal can be restricted to a certain sector which further minimizes the SOP.



(a) Cell-of-origin technique.

(b) Doughnut representing the SOP of enhanced Cell-ID positioning.

Figure 4.6: Positioning based on information about the cell-ID.

As mentioned in Pahlavan and Krishnamurthy (2013), when performing multilateration, the RSSI values have to be converted to distances, which requires an appropriate path-loss or radio propagation model (Rayleigh fading model, Rician distribution model, etc.). These models are often defined experimentally. However, walls and obstacles within buildings degrade the quality of the determined distance, since the propagation does not correspond to a line-of-sight propagation. By using the Floor Attenuation Factor (FAF) propagation model, the effects of indoor obstacles can be slightly treated by adopting the Wall Attenuation Factor (WAF) with respect to the present indoor environment. The WAF is empirically determined and depends on the building layout and the construction materials. However, even if building structures are considered within the path loss model, dynamic indoor environments, moving humans, and multipath effects downgrade the accuracy.

Referred to Chen et al. (2012), fingerprinting methods can be used if the measurements are correlated with the location of the user. Within fingerprinting, two different phases, the training and the positioning phase, have to be distinguished. During the training phase, location-dependent measurements at so-called reference points have to be collected. Therefore, as Figure 4.7 shows, the space is divided into cells, whereby the center of the cells are the reference points with known coordinates. Both, the coordinates of the reference points and the corresponding RSSI measurements are called a fingerprint and are stored in a fingerprinting database (radio map).

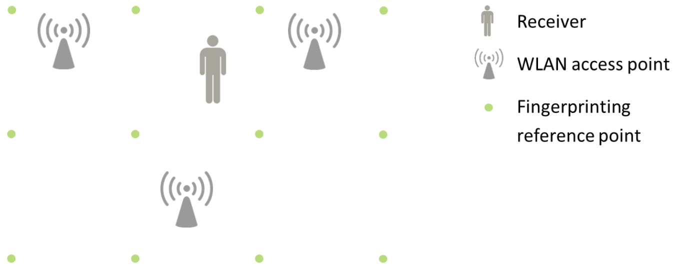


Figure 4.7: Fingerprinting-based positioning.

The concept of the positioning phase is illustrated in Figure 4.8. The goal is to find the reference point with those RSSI observations, that match the real-time RSSI measurements best. The position of the user corresponds therefore to the reference point with the maximum correlation, which can be determined by computing the signal distance considering all i WLAN access points:

$$d(k) = \left(\sum_{i=1}^{n} |s_i - f_i(k)|^q \right)^{(1/q)}, \tag{4.9}$$

with $f_i(k)$ being the collected and stored RSSI tuple at the reference point k and s_i being the measured RSSI data. For WLAN fingerprinting, the Manhattan distance ($q = 1$) and the Euclidean distance ($q = 2$) turned out be the most efficient. These computed signal distances act as basis for finding the reference point with the maximum likelihood, which corresponds, in case of observing RSSI values, to the reference point with the minimum signal distance $d(k)$. The nearest neighbor $\hat{\boldsymbol{x}}_{ML}$ in signal space can therefore be found by

$$\hat{\boldsymbol{x}}_{ML} = \underset{\boldsymbol{x}_k}{arg\,min}\, d(k). \tag{4.10}$$

As this maximum likelihood estimator chooses one of the reference points as the estimated position, the accuracy of this technique strongly depends on the density of reference points. Instead, using the average of K reference points with largest likelihoods reduces the dependency on the cell decomposition. The K-nearest-neighbor estimator is mathematically defined as

$$\hat{\boldsymbol{x}}_{KNN} = \frac{1}{K} \sum_{i=1}^{K} \boldsymbol{x}(i), \tag{4.11}$$

with $\boldsymbol{x}(i)$ representing the K reference points with the largest likelihoods. Considering further the likelihood of each reference point, the performance can be improved by applying a weighted K-nearest-neighbor estimator

$$\hat{x}_{KNN} = \frac{1}{\sum_{i=1}^{K} \omega_i} \sum_{i=1}^{K} \omega_i \boldsymbol{x}(i), \tag{4.12}$$

with

$$\omega_i = f(d(i)). \tag{4.13}$$

The weight ω_i is hereby a function of the signal distance, which is given in equation (4.9).

More advanced fingerprinting methods are probabilistic approaches which are based on the Bayes theorem. The fingerprint of such an approach is a set of probability density functions and the estimated position corresponds to the maximum conditional probability. However, the Bayesian approaches cause higher processing efforts and require more memory. Consequently, maximum likelihood or K-nearest-neighbor estimators have to be preferred for smartphones. The WLAN positions incorporated within the developed RBPF are based on a K-nearest-neighbor fingerprinting with $K = 4$.

The design of the radio map including the number of reference points and training samples strongly influences the positioning performance. Additionally, the attenuation caused by the human body degrades the estimation accuracy which can be improved by collecting samples in several directions during the training phase. Due to the missing standardization of the RSSI values of different devices, the radio map generated by one device is not necessarily optimal for another device.

The advantage of the fingerprinting technique is that the environment and therefore the signal propagation is already modeled within the training data set, and consequently, no assumptions regarding the path loss have to be made. The disadvantage of this technique is the time- and thus cost-consuming training phase. Additionally, the radio map has to be up-to-date and thus maintained due to environmental changes.

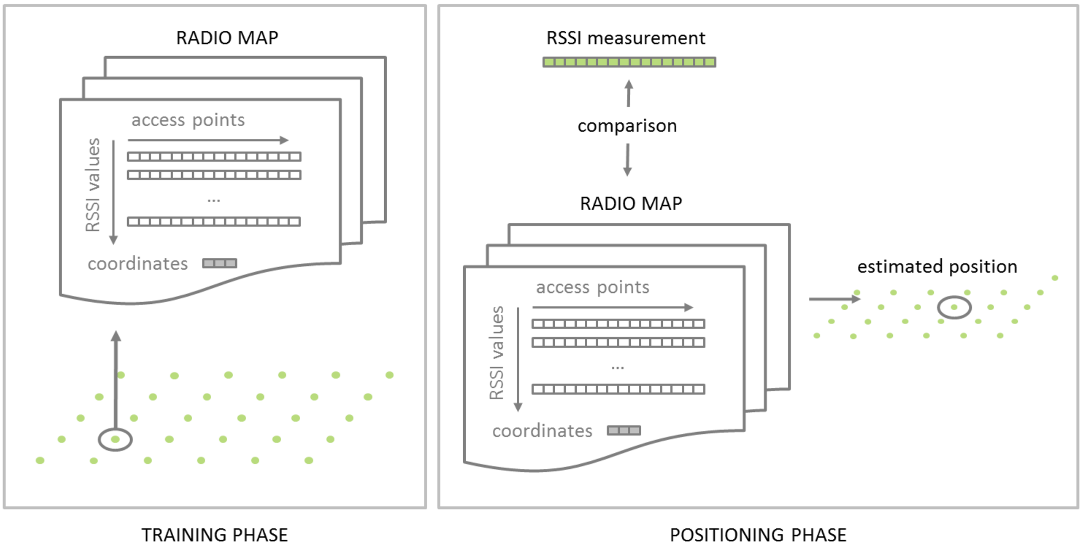


Figure 4.8: The two phases of fingerprinting-based positioning.

Typical fingerprinting accuracies are 2 to 5 meters, which approximately corresponds to the outdoor performance of GNSS SPP. As stated in Woodman (2010), more accurate systems with a reported accuracy of 1.4 m for 95% of the time exist, however they are based on a high density of access points, which are especially installed for navigation purposes. Considering the positioning of blind people, realistic WLAN environments based on already installed access points have to be investigated. Therefore, an accuracy in the range of the above mentioned 2 to 5 meters is assumed within the thesis. Based on these assumptions, WLAN fingerprinting can be treated as indoor equivalent to GNSS with respect to the kind of position information and the corresponding accuracy.

4.3 Inertial Navigation

Within the thesis, preprocessed inertial data delivers information about the relative motion of the pedestrian. Hence, this chapter deals with the main principles of inertial navigation, including the measurement types and the processing techniques. The focus is hereby set on smartphone sensors and PDR algorithms.

An Inertial Measurement Unit (IMU) measures the accelerations and angular rates, which can be used to estimate the changes in position, velocity, and attitude. An IMU consists of three orthogonally arranged accelerometers and gyroscopes measuring the accelerations a and angular rates ω in three directions:

$$a = [a_x, a_y, a_z]^T, \tag{4.14}$$

$$\omega = [\omega_x, \omega_y, \omega_z]^T. \tag{4.15}$$

While the accelerometers determine the specific force which acts on the IMU, the gyroscopes measure the angular rates, both with respect to the inertial frame (Woodman, 2010). The current pose (position and orientation of an object) can be determined by combining the previous position and the actual estimates for the velocity vector and the three-dimensional orientation (attitude), both derived from the IMU data. The method of determining absolute positions based on relative position information is denoted *Dead Reckoning* (Renaudin, 2014). However, the initial position and orientation have to be known for subsequently estimating the trajectory.

Modern low-cost IMUs have additionally embedded magnetometers (digital compass) consisting of three perpendicular sensors, which are detecting the magnetic field's strength (Mautz, 2012). A magnetometer measures the magnetic flux densities, given in units of Tesla (T), to determine the direction to the magnetic north pole. Especially in the low-cost sector, this information acts as support for the gyroscopes in determining the heading of the person.

Since the inertial measurements are given relative to the sensor reference frame (body frame), the angular motion of the sensor frame has to be monitored to allow a transformation of the specific force to a reference frame, which is suitable for the application. An inertial measuring system, the sensors of which are rigidly attached with the moving body of the system, is called *strapdown inertial system* (Titterton and Weston, 2004). Such a system demands, because of the strapdown configuration, a transformation of the measurements into an appropriate reference frame which could be a Local Level Frame (LLF) or an Earth Centered Earth Fixed (ECEF) frame (Groves, 2013). The corresponding transformations from the body frame to the LLF and the ECEF frame respectively can be found in Hofmann-Wellenhof et al. (2003), chapter 3, and Groves (2013), chapter 2.

The current rotation angles for the transformation can be determined by an integration of the angular rates. Based on these determined rotation angles, the accelerations can be transformed into the chosen reference system. The integration of the accelerations $\ddot{\boldsymbol{x}}$ given in the inertial frame with respect to time yields the change in velocity, which further gives the actual velocity $\dot{\boldsymbol{x}}$ if information about the initial velocity $\dot{\boldsymbol{x}}_A$ is available:

$$\dot{\boldsymbol{x}}(t) = \int_{t_A}^{t} \ddot{\boldsymbol{x}}(\tau) d\tau + \dot{\boldsymbol{x}}_A. \tag{4.16}$$

The initial velocity $\dot{\boldsymbol{x}}_A$ corresponds to the velocity of the previous epoch, or in a first instance, the velocity at the starting point. The current position $\boldsymbol{x}(t)$ can be determined by integrating the velocity vector $\dot{\boldsymbol{x}}(t)$ under consideration of the previous position $\boldsymbol{x}_A$:

$$\boldsymbol{x}(t) = \int_{t_A}^{t} \dot{\boldsymbol{x}}(\tau) d\tau + \boldsymbol{x}_A. \tag{4.17}$$

In equation (4.16) and (4.17), τ represents the integration variable with respect to time.

As stated in Titterton and Weston (2004), the accelerometers measure the specific force, which includes besides the object's accelerations, accelerations caused by the gravitational forces of the Earth. This fact has to be accounted for to get the acceleration of the device. The main principle of inertial navigation is visualized as block diagram in Figure 4.9. It includes the determination of the actual attitude for the transformation of the accelerations into the appropriate coordinate frame, the gravity compensation, and the double integration for the final estimation of the position. The diagram represents the main functions of a strapdown inertial navigation system.

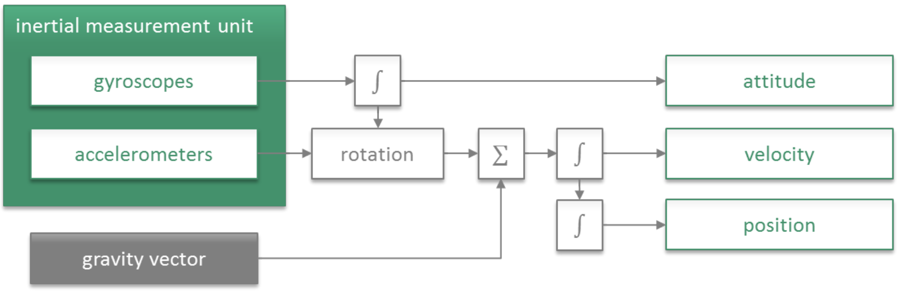


Figure 4.9: The concept of strapdown inertial navigation.

Since an IMU measures the changes in orientation and position only, the accuracy of the estimated parameters is degraded with time because of the accumulation of systematic sensor errors. Without any additional information from external sensors, the accuracy of the estimation gets continuously worse. Furthermore, the quality of the position strongly depends on an accurate parameter initialization. However, as stated in Mautz (2012), performing inertial navigation in combination with an external sensor, which continuously updates the position and orientation, gives more precise motion information compared to an interpolation between the trajectory points determined without inertial data.

The inertial system of a smartphone is realized by Micro-Electro-Mechanical System (MEMS) sensors, which are very small in size, weight, and its cost, but come with low quality sensor characteristics. Gyroscope and accelerometer sensors can be classified with respect to their quality, which is mainly effected by the different inertial sensor technologies (Renaudin, 2014). As stated in Woodman (2010), different error sources like noise, bias, bias instability, calibration errors, and temperature instability are the basis for separating the sensors into three quality grades: the navigation grade, the tactical grade, and the low-cost sensors (sorted with decreasing quality). The MEMS (Micro-Electro-Mechanical System) sensors of smartphones belong to the class of low-cost sensors. The navigation and tactical grade sensors are still to large in size and are therefore not applicable for pedestrian navigation. Due to the large amounts of drift and noise caused by low-cost sensors, positions with meter-level accuracy cannot be provided even over a period of only some minutes. Therefore, in the field of pedestrian navigation, different algorithms have been introduced which exploit the biomechanical and physical characteristics of the human stride to compensate for the bad error characteristics of the sensors. These algorithms are called Pedestrian Dead Reckoning and are divided into inertial-based (constrained) and step-based approaches.

The inertial-based approach is based on the concept of strapdown inertial navigation, where the accelerations are integrated two times along its path to estimate the relative position. Within the inertial-based PDR approach, the rapid accumulation of measurement errors is prevented by using the standstill phases during a step for the correction of the Inertial Navigation System (INS). Within this phase, the foot is put on ground, and consequently no accelerations occur and the velocity is zero. As stated in Renaudin (2014) and Woodman (2010), this information can be used as Zero Velocity Update (ZUPT). To detect the ground phases correctly, this method makes a foot-mounted IMU indispensable.

While the inertial-based PDR approach focuses on the incorporation of ZUPTs during the standstill phases of a stride, the objective of the step-based approach is to use the inertial data to detect a step, determine the step's length, and estimate the direction of the step. This method is robust enough to deliver relative positions based on inertial data of current smartphones, regardless if the smartphone is handheld or carried in a pocket. Since step-based algorithms can be applied for a variety of sensor positions, the input of the RBPF is based on a step-based Pedestrian Dead Reckoning (PDR) approach. In the following section 4.3.1 the characteristics and the algorithmic concept of the algorithm employed in this thesis are discussed.

4.3.1 Step-based pedestrian dead reckoning

Step-based PDR algorithms are separated into three steps: step detection, step length estimation, and heading estimation. Every time a step is detected, the position $\boldsymbol{x}(t) = [n(t), e(t)]^T$ is updated by the estimated step length $sl(t)$ and direction $\theta(t)$, as long as the coordinates of the previous epoch $t-1$ are known:

$$n(t) = n(t-1) + sl(t) \cdot \cos\theta(t), \tag{4.18}$$

$$e(t) = e(t-1) + sl(t) \cdot \sin\theta(t). \tag{4.19}$$

Figure 4.10 visualizes the concept of relative positioning based on step-based PDR. The vectors $\Delta\boldsymbol{x}_{t-1}$ and $\Delta\boldsymbol{x}_t$ in Figure 4.10 represent the relative change in the horizontal position, while their norms $\|\Delta\boldsymbol{x}_{t-1}\|$ and $\|\Delta\boldsymbol{x}_t\|$ correspond to the appropriate step lengths $sl(t-1)$ and $sl(t)$.

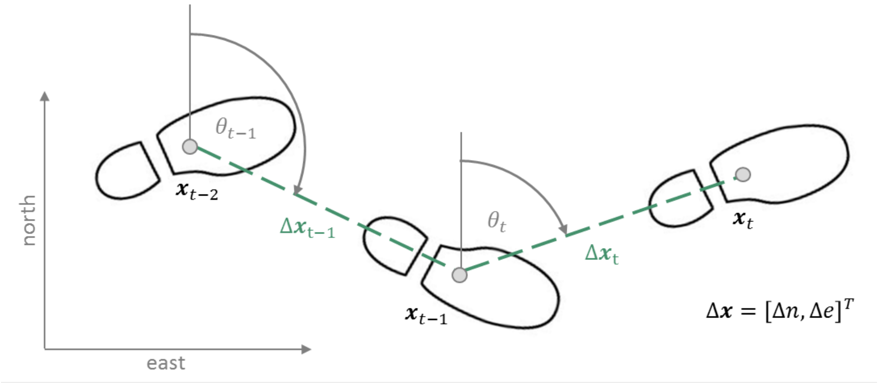


Figure 4.10: The concept of step-based PDR.

In contrast to the inertial-based approach, the traveled distance is determined by a signal segmentation instead of double integrating the acceleration data. This makes the step-based approach eligible for different sensor positions like waist, foot, hand, backpack, etc. However, different sensor locations result in different signal amplitudes and characteristics, which consequently lead to various step-based PDR approaches. The input data of the RBPF is based on a PDR approach, which assumes that the person carries the smartphone in the hand. Thus, the applied methods focus on the signal characteristics of hand-held sensors. In the following, the concept of the employed PDR is described. More details can be found in Moder et al. (2015), Hafner et al. (2014b), and Aichhorn et al. (2013).

Step detection: The correct identification of a step is the basis for the step-based PDR. Only if a step is detected, the estimated step-length and heading are used for a position update. Steps can be identified by applying different methods ranging from simple peak detection, zero-crossings techniques, stance-phase detection to more advanced approaches of segments (pattern) identification. Within the applied method, the step detection is done by identifying the step's cycle pattern as proposed in Park and Suh (2010), whereby the segmentation is based on the total acceleration $a = \sqrt{a_x^2 + a_y^2 + a_z^2}$ and empirically defined thresholds. After detecting a step, the elapsed time between two standstill phases has to be determined, since this time span t_{step} is required as input for the applied step length model. Therefore, a peak detection based on the total accelerations is performed.

Step length estimation: The length of a step is typically determined with a model which is either a constant/quasi-constant, a linear, a non-linear, or an artificial intelligence model (Chen et al., 2011). The model parameters have to be determined in a training phase or are defined empirically. Within the applied PDR approach, the traveled distance is based on an empirical model defined in Chen et al. (2011) and is given by

$$sl = \left[0.7 + a\left(h - 1.75\right) + b \frac{\left(\frac{1}{t_{step}} - 1.79\right) h}{1.75} \right] c, \tag{4.20}$$

with sl being the step length, h the height of the pedestrian, and t_{step} the step's time span. The parameters a, b, and c are coefficients. While c is a personal factor, which represents the person's particular walking style and has to be trained either on-line (with external sensor data, e.g., GNSS) or in advance, the other two are given ($a = 0.371$ and $b = 0.227$). To keep the processing at the smartphone as simple as possible, c was determined in advance with respect to the actual test person.

Heading estimation: For the heading estimation, the orientation of the sensors relative to the person has to be either known or calibrated. The applied approach is based on the assumption that the smartphone is carried handheld in walking direction, and therefore the orientation is constant. While for the step detection and the step length estimation only the data of the accelerometers are considered, a robust heading computation requires data of all three sensor groups (accelerometer, gyroscope, magnetometer). Especially in indoor environments, a magnetometer-only heading solution strongly suffers from magnetic deviations making the heading unreliable for brief intervals, which consequently reduces the magnetometer's applicability as standalone heading sensor. In contrast, an angular-rate-based heading is subject to drift caused by the accumulation of errors. Since the quality of the MEMS gyroscopes is not good enough, the incorporation of the digital compass is absolutely necessary and therefore, a combination of both data within an appropriate filter has to be preferred (Chen, 2012b). Thereby, the magnetometers provide absolute heading information, while the gyroscopes deliver relative information. The accelerometer data is needed for analytically leveling the three-axis magnetometer. The approach developed at the Institute of Geodesy, Working Group Navigation is outlined in Hafner et al. (2014b).

4.4 Barometer-based height information

Modern smartphones have built-in barometers to detect changes in height, which can be used as support for the height estimation based on WLAN or GNSS (Kuusniemi et al., 2012). In indoor environments, the height is an important factor, since the incorporation of a floor map within the position estimation requires floor level information. The WLAN height does not meet the requirements for a reliable detection of floor transitions, since the height cannot be modeled sufficiently by fingerprints. Outdoors, the GNSS height is the coordinate component which can be determined worst due to the constellation of the satellites. Consequently, a sensor fusion is performed for the height determination.

As mentioned in Groves (2013), the barometer measures the ambient air pressure. As the air pressure decreases with increasing height, the air pressure can be used to determine the change in height by applying a standard atmospheric model, which is given by

$$\Delta h_{baro} = \frac{T_s}{k_T} \left[\frac{p_b}{p_s}^{-\frac{Rk_T}{g_0}} - 1 \right] , \tag{4.21}$$

with p_b being the measured air pressure, p_s and T_s the pressure and temperature at a defined height, $k_T = 6.5 \times 10^{-3}\,\mathrm{K\,m^{-1}}$ the atmospheric temperature gradient, $R = 287.1\,\mathrm{J\,kg^{-1}\,K^{-1}}$ the gas constant, and $g_0 = 9.80665\,\mathrm{m\,s^{-2}}$ the average acceleration due to the gravity at the surface of the Earth. For p_s and T_s either standard parameters for the sea level can be used or parameters measured at the trajectory's starting point need to be determined. By combining the barometric relative height with an external height information, absolute heights can be gained. The model given in equation (4.21) is one possibility amongst others to determine changes in height. Another model can be found in Bevermeier et al. (2010). The unit for pressure corresponding to the International System of Units (SI) is pascal (PA), but hPA or millibar are more commonly used for the atmosphere pressure.

The accuracy of the estimated height change is in the range of a few meters, depending on the atmospheric stability and the period of time the barometer is used. Li et al. (2013) reported, that for a period of 10 minutes the variations of a smartphone barometer are less than 0.2 hPa which corresponds to 1 to 2 meters in height. Investigations within our research group revealed that the accuracy over a shorter period of time is better, enabling the detection of floor transitions with a success rate of 100%. Nevertheless, a sensor fusion is indispensable, since on the one hand, weather fronts can cause barometric height errors, and on the other hand, in indoor environments meter-level variations can occur due to ventilation or door/window opening (Groves, 2013). However, if the air is stable, the barometric height is more reliable than the GNSS or WLAN height (Li et al., 2013).

4.5 Pedestrian activity classification

Since barometers are not standard in state-of-the-art smartphones and WLAN fingerprinting delivers poor height information, but the detection of the correct floor level is very important for providing the appropriate floor plan, an alternative or back-up system had to be found for the height estimation in indoor environments. A Pedestrian Activity Classification (PAC) algorithm detecting the activities going up- or downstairs can deliver this information. Additionally, the PAC can assist a PDR algorithm by reducing errors within the step length estimation (Moder et al., 2015). Furthermore, the current activity delivers spatial constraints for the estimation of the horizontal coordinates (section 5.2.3) and restricts, by a combined evaluation of the activity and the location, floor level changes to areas, where a transition between different floors is possible.

In general, the human activity recognition is based on machine learning techniques. The PAC algorithm has to be trained with a set of labeled data including all classes of activity (Moder et al., 2014a). Based on the training data, which consists of selected features and the corresponding classes, the learning algorithm generates a classifier. After the training process, the classifier is able to identify the class based on an input vector containing feature values (Wisiol, 2014). As mentioned in Frank et al. (2010), different methods like k-nearest neighbor approaches, support vector machines, artificial neural networks, decision trees, or Bayesian techniques can be applied for the activity classification.

The PAC algorithm, which is the basis for the activity determination within this work, was implemented in the framework of a master thesis (Wisiol, 2014) at the Institute of Navigation, Graz University of Technology (now Institute of Geodesy). Within the PAC, accelerometer and gyroscope measurements are used for the feature computation. In general, the PAC is a method for monitoring any kind of activity of a person. For the support of the positioning within this thesis, the following activities have been defined for the recognition within the Pedestrian Activity Classification (PAC):

- walking,
- standing,
- going upstairs,
- going downstairs.

For training the algorithm, a data set with a duration of about 20 minutes was collected by a pedestrian carrying the smartphone handheld in walking direction. During these 20 minutes, the test person performed all four defined activities. The time series was labeled correctly before training the classifier.

For the robust identification of the activities *going upstairs* and *going downstairs* the observed accelerations have to be transformed from the body frame of the sensors to the local level frame to separate the horizontal and vertical components. For the transformation, the orientation angles roll and pitch are used. The vertical acceleration component in the LLF is the most relevant information for recognizing *going upstairs* and *going downstairs*. Consequently, besides the norm of both measurement groups, the horizontal and vertical component of the accelerations act as input for the feature computation. The horizontal component $a_{horiz_{LLF}}$ and the vertical component $a_{vert_{LLF}}$ are given by

$$a_{horiz_{LLF}} = \sqrt{a_{n_{LLF}}^2 + a_{e_{LLF}}^2}, \tag{4.22}$$

$$a_{vert_{LLF}} = a_{u_{LLF}}, \tag{4.23}$$

where $a_{n_{LLF}}$ and $a_{e_{LLF}}$ are the accelerations in north and east direction in the LLF, while $a_{u_{LLF}}$ represents the acceleration along the vertical axis. The total acceleration a_{total} and the total angular velocity ω_{total} are computed by the norm over the acceleration vector $[a_x, a_y, a_z]^T$ and angular rate vector $[\omega_x, \omega_y, \omega_z]^T$ measured in the sensor frame:

$$\begin{aligned} a_{total} &= \|a\| \\ &= \sqrt{a_x^2 + a_y^2 + a_z^2}, \end{aligned} \tag{4.24}$$

$$\begin{aligned} \omega_{total} &= \|\omega\| \\ &= \sqrt{\omega_x^2 + \omega_y^2 + \omega_z^2}. \end{aligned} \tag{4.25}$$

Suitable features for training the classifier are statistical values as the maximum, mean, standard deviation, and root mean square, or less established parameters like the interquartile range, Main Frequency Component (MFC), amplitude of the MFC, arc tangent of the ratio of the vertical and horizontal component, and the Activity Unit (AU). These features represent parameters of the time and frequency domain. As demonstrated in Wisiol (2014), a window size of 32 samples (with 25 overlapping samples) is best qualified to detect the defined activities. This means, that 32 samples are taken for the determination of the current activity. Then, the window moves 7 samples ahead for taking the next 32 samples. Thereby, an overlapping of 25 samples is given for two consecutive iterations.

As the smartphone used in this thesis offers a maximum measurement frequency of 28 Hz for the accelerometers and gyroscopes, 32 samples cover a period of 1.14 s which is approximately the duration of one double step. Since the activities are all related to walking, the

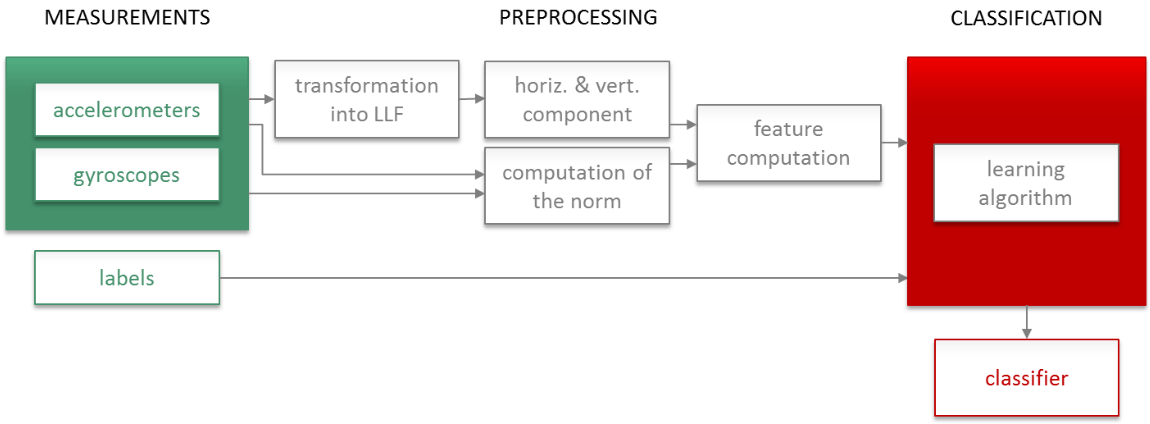


Figure 4.11: Training phase for the extraction of a classifier: The inertial data transformed to the LLF acts as input for the feature computation. The training of the classifier happens within a learning algorithm based on the extracted features.

performance of the PAC based on 32 samples turned out to be best. However, the best qualified number of samples depends on the application and the measurement frequency of the device.

In Figure 4.11 the training phase of the PAC is illustrated as block diagram. The output of the training phase is the classifier (red box), which is the basis for the activity recognition. The Waikato Environment for Knowledge Analysis (WEKA) software was used as classification and evaluation tool. As stated in Hall et al. (2009), WEKA is an open source software which enables the application of state-of-the-art machine learning techniques on a variety of data formats. The software can be used on different platforms and has many preprocessing tools and data visualization facilities. As mentioned in Wisiol (2014), the built-in classifier J48, which belongs to the Decision Tree C4.5 classifiers, was used for training the classifier based on the training data set.

Wisiol (2014) showed, that the classifier performs best with personal training data. Classifiers can also be built with impersonal (the training data is of a different person) or hybrid training data (combination of personal and impersonal data). However, although the impersonal and hybrid training data lead to a worse PAC performance, the activities can be detected correctly with a success rate of at least 93% (Wisiol, 2014).

The evaluation of the classifier is done by a 10-fold cross validation. The validation revealed that *standing* can be detected best with a success rate of 100%, since the extracted feature values differentiate from the walked activities. The recall rate for *walking* and *going down-*

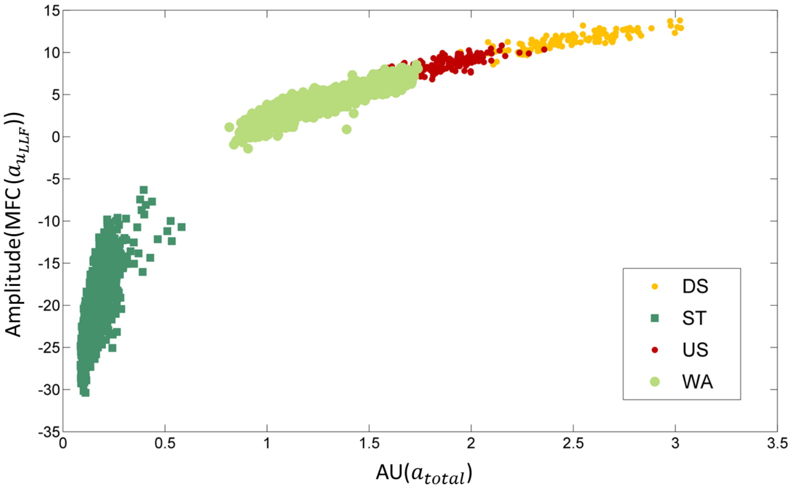


Figure 4.12: Comparison of two features (MFC of the vertical acceleration in the LLF versus the AU of the total acceleration) extracted from accelerations, which include the activities *going downstairs* (DS), *standing* (ST), *going upstairs* (US), and *walking* (WA).

stairs is about 99%, while *going upstairs* shows a worse result with only 94% due to feature similarities to *walking* (Moder et al., 2014b). The overall success rate of the PAC is about 99%, whereby personal training data is used.

Figure 4.12 shows the separation of the different activities based on two different features, the AU of the total acceleration and the amplitude of the MFC of the vertical acceleration. While *standing* is clearly separated from the three others, a fluent transition between *going upstairs*, *going downstairs*, and *walking* can be observed.

4.5.1 PAC-based height estimation

In indoor environments, the detection of the current floor, which is needed for Bayesian filters incorporating navigable maps, fails without a reliable height information and makes robust position estimation impossible. Therefore, besides the barometric height estimation, alternative approaches are investigated to determine the current floor. By detecting the activities *going upstairs* and *going downstairs* and assuming a standardized stair height, an additional height information can be obtained.

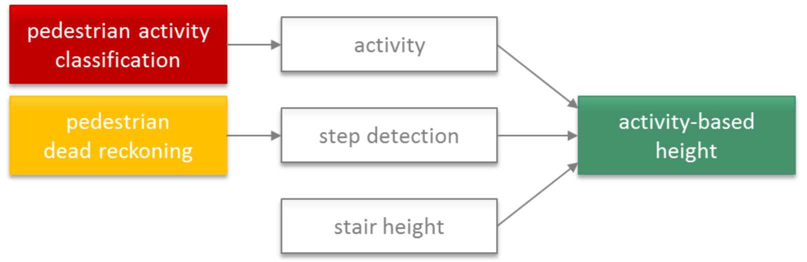


Figure 4.13: Computation of the height based on the results of the PAC and the PDR, and the standardized heights of the stairs.

In detail, a vertical displacement is computed by combining the information of the step detection gained from the PDR, the activity of the PAC, and the standardized step height (Figure 4.13). If the actual activity is *going upstairs* or *going downstairs* and a step is detected, then the height of one stair corresponds to the covered height. This can be done, since the stair height in general is the same in most of the buildings.

5 Filter design for the positioning of visually impaired people

The focus of the thesis is the design and development of a navigation filter for the purpose of positioning against the background of the navigation of blind persons. As mentioned in previous chapters, the position determination is based on today's smartphone sensors, comprising GNSS, WLAN, inertial, magnetometer, and barometer technologies. All of these data types are preprocessed within appropriate algorithms, which provide absolute and relative position information as input for the positioning filter, see chapter 4. The smartphone's internal GPS chip provides a GPS SPP code solution, while in indoor environments a WLAN fingerprinting algorithm, which is described in section 4.2, is applied for the determination of 3D coordinates. The WLAN and GNSS coordinates refer both to the global World Geodetic System 1984 (WGS84) reference frame. The proposed PDR method (section 4.3.1) delivers the change in distance and heading based on accelerometer, gyroscope, and magnetometer data, while the barometer gives information about the change in height (section 4.4).

In addition, information about the current activity is used as spatial constraint within the filtering process and in combination with the output of the PDR to determine a relative height. The estimation process is further supported by different kinds of map information. For the position determination in outdoor environments, a user-tailored raster map is used to derive location-based probabilities corresponding to the movement restrictions of blind persons. Unfortunately, as stated by blind people themselves, such location-based probabilities cannot be defined for indoor environments due to the small-scaled environment and the very individual navigation habits of blind persons within buildings. However, a map of the building layout is incorporated within the filter. Furthermore, a wall collision filter proofs the plausibility of room and floor changes within buildings.

Based on the investigations concerning the applicability of the different Bayes filter implementations (chapter 3), the filter design for the outdoor and indoor positioning is defined. The main goal was to fulfill the quality requirements of an reliable and accurate navigation system especially tailored to the use of visually impaired persons. To benefit from all the mentioned input data, a particle filter approach was chosen as core element of the positioning tool. To keep the processing costs as low as possible and to provide the correct floor plan in indoor environments, the height estimation is done in an outsourced Kalman filter. Consequently, a Rao-Blackwell particle filter, which estimates the 2D coordinates within a particle filter and the height within a Kalman filter, is chosen to be the optimal filter for this application. The conceptual design of the developed RBPF is illustrated in Figure 5.1, which is the basic framework for the position estimation in outdoor and indoor environments. As for the outdoor and indoor positioning different maps and observations are involved, two separate, but similar, filter approaches had to be developed.

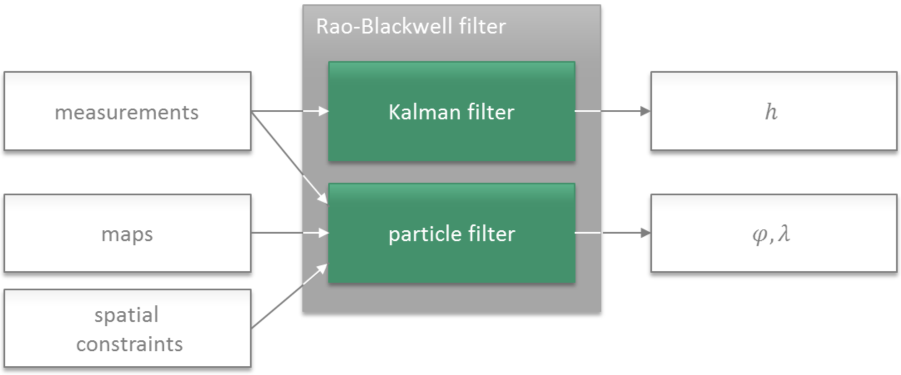


Figure 5.1: Principle design of the positioning filter which corresponds to a RBPF.

The outdoor RBPF is based on GNSS, PDR, and barometer data, and profits from the user-tailored raster map. The activity recognition is only used for the correction of the step length within the PDR in case of going upstairs or downstairs (Moder et al., 2015), because outdoors, the defined activities are not suited to derive spatial constraints for the position determination.

WLAN, PDR, barometer, and PAC data are combined within the indoor RBPF, which is further supported by information gained from a building layout. Since indoors the use of the correct floor plan depends on information about the current floor level, a floor detection algorithm has to be installed within the height estimation process. Moreover, to proof the plausibility of the particle propagation, a wall collision filter, which verifies room and floor changes, is part of the designed filter. As the recognized activities provide spatial constraints for indoor environments, these are considered as well.

Consequently, two different Rao-Blackwell filters, one for outdoors and one for indoors, are designed. The algorithmic concepts, the map integration, as well as the derived spatial constraints are treated separately for the different environments within section 5.1 and 5.2. In chapter 6, the performance of the filters is investigated.

The positioning tool was developed with the software package MATLAB R2013a and realized as Graphical User Interface (GUI) to investigate different filter settings. During the development of the tools and functions, the real-time applicability of the positioning was always kept in mind, so that at the end of the development and validation phase, a transfer of the software concept onto a smartphone is no problem. The implemented GUI, the main features of which are shown in Figure 5.2, provides amongst others the following options:

- Import of preprocessed sensor data
- Simulation of measurement data based on a reference trajectory
- Application of different navigation filters with various adjustment possibilities
- Visualization and analysis of the filter results
- Generation of movies showing the performance of the filter

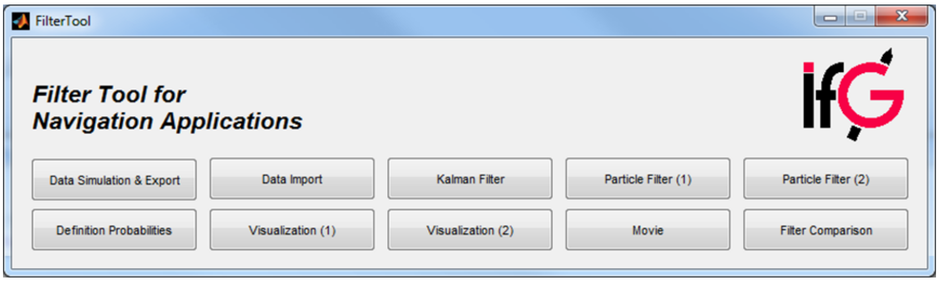


Figure 5.2: Main features of the GUI, which is called *Filter Tool for Navigation Applications*.

As device for the investigations, a smartphone Samsung Galaxy Nexus, manufactured in 2011, is used. The operating system of this smartphone is Android 4.0. The smartphone features an internal GNSS receiver, a WLAN scanner, a barometer, as well as three-axis accelerometers, gyroscopes, and magnetometers. The operation of the smartphone is done via a touch-screen. At the Institute of Geodesy, Working Group Navigation (TU Graz), special smartphone applications have been developed for measuring the sensor data, which is recorded in text files. These files are the basis for the processing, which is done on an Intel i7 desktop CPU with a processor speed of 3.2 GHz and 12 GB RAM. During the measurements, the smartphone is handheld by the pedestrian and points in walking direction.

5.1 Outdoor positioning

As mentioned in the previous sections, for the positioning in outdoor environments a user-tailored raster map has been generated, which is incorporated within the state estimation. Based on the user-tailored raster map, map-based probabilities can be derived, which represent the walking habits and path preferences of blind pedestrians. These probabilities are integrated in the state estimation process to derive map-based importance weights contributing to the 2D positioning. Thereby, the position accuracy and reliability can be improved.

In Figure 5.3, the concept of the Rao-Blackwell positioning filter for the outdoor environment is visualized. The particle filter estimates the 2D coordinates, while a Kalman filter is responsible for the height determination.

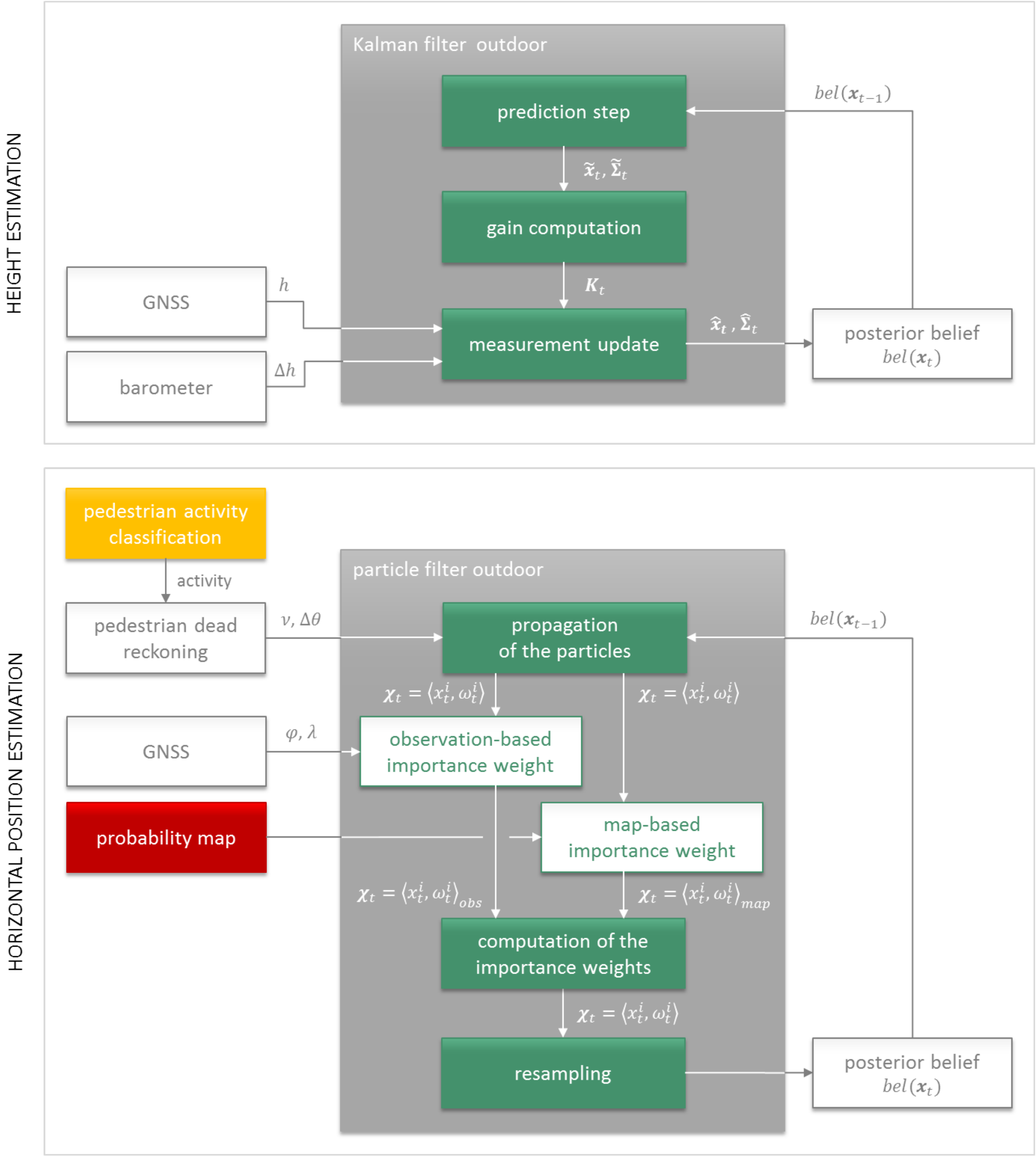

HEIGHT ESTIMATION
Kalman filter outdoor
prediction step
$bel(\boldsymbol{x}_{t-1})$
$\tilde{\boldsymbol{x}}_t, \tilde{\boldsymbol{\Sigma}}_t$
gain computation
$\boldsymbol{K}_t$
GNSS
h
barometer
Δh
measurement update
$\hat{\boldsymbol{x}}_t, \hat{\boldsymbol{\Sigma}}_t$
posterior belief
$bel(\boldsymbol{x}_t)$
HORIZONTAL POSITION ESTIMATION
pedestrian activity classification
activity
pedestrian dead reckoning
$v, \Delta\theta$
particle filter outdoor
propagation of the particles
$bel(\boldsymbol{x}_{t-1})$
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
GNSS
φ, λ
observation-based importance weight
probability map
map-based importance weight
$\chi_t = \langle x_t^i, \omega_t^i \rangle_{obs}$
$\chi_t = \langle x_t^i, \omega_t^i \rangle_{map}$
computation of the importance weights
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
resampling
posterior belief
$bel(\boldsymbol{x}_t)$

Figure 5.3: Rao-Blackwell estimation process for the positioning in outdoor environments. The height is determined within a Kalman filter, while the estimation of the horizontal position is performed within a particle filter.

As Figure 5.3 illustrates, the Kalman filter estimates the height based on GNSS and barometer data. The height component of the GNSS solution and the barometer-based relative height are used together with their corresponding covariances within the measurement update, given by equation (3.18) and (3.19), to estimate a posterior belief of the current height. In case of the Kalman filter, the posterior belief $bel(\boldsymbol{x}_t)$ is represented by the mean $\hat{\boldsymbol{x}}_t$ and the covariance $\hat{\boldsymbol{\Sigma}}_t$ of the estimated height (compare section 3.1.1). The developed Kalman filter is described in detail in section 5.1.2.

The particle filter integrates all data, which are relevant for the estimation of the horizontal coordinates. Besides the GNSS coordinates and the PDR data (velocity and heading), the probability map is defined as input for the particle filter. The PDR data, which describes the motion of the pedestrian, is used for the propagation of the particles. The predicted particle set $\boldsymbol{\chi}_t$ corresponds to the prior belief $\overline{bel}(\boldsymbol{x}_t)$, which acts as basis for the derivation of the importance weights within the measurement update. Hereby, the GNSS observations as well as the probability map deliver separate importance weights, which are combined to get the posterior probability of each particle. These probabilities are used to resample the particles in the state space to focus the processing in regions where the probabilities are highest, see section 3.2.2. Details on the implemented particle filter approach are given in section 5.1.3, while the generation of the user-tailored raster map and the derivation of the corresponding probabilities are described in section 5.1.1.

5.1.1 Maps tailored to blind people

For the derivation of spatial constraints, which represent the navigation habits of blind pedestrians, special occupancy grid maps, representing all relevant elements of the scenery, have to be established. Considering blind persons, regions where the person can move without any restrictions have to distinguished from those with dangers and obstacles. Visually impaired people preferably walk on pavements, cross the streets only at zebra crossings, and avoid unratable objects like bike parking areas or shared cycle- and pathways. Furthermore, pathways, which provide navigation assistance like tactile ground surface indicators or acoustic traffic lights, are favored. Based on these walking and navigation habits appropriate location-based probabilities can be derived acting as geometric constraints within the positioning filter.

For the incorporation of the map within the particle filter, it is of great importance, that the map information is provided as raster map. Although a raster map needs a higher storage capacity compared to a vector map, it is better suited for the integration within the particle filter as geometric operations can be circumvented. A georeferenced raster map enables a fast verification of the pedestrian's location within the map if the grid size is known. The map coordinates deliver then the corresponding location probabilities. Furthermore, by classifying the features of the map grid cells, information about the grid cell's characteristic (representing a zebra crossing, pavement, etc.) can be provided.

Within the thesis, a concept had to be developed, which enables an easy and fast retrieval of the relevant map information based on a georeferenced raster map. Figure 5.4 shows the design process of the map including all sources of input data, which comprise a vector dataset, the preferred pathways of blind persons, and data which is relevant for blind people and missing in the dataset.

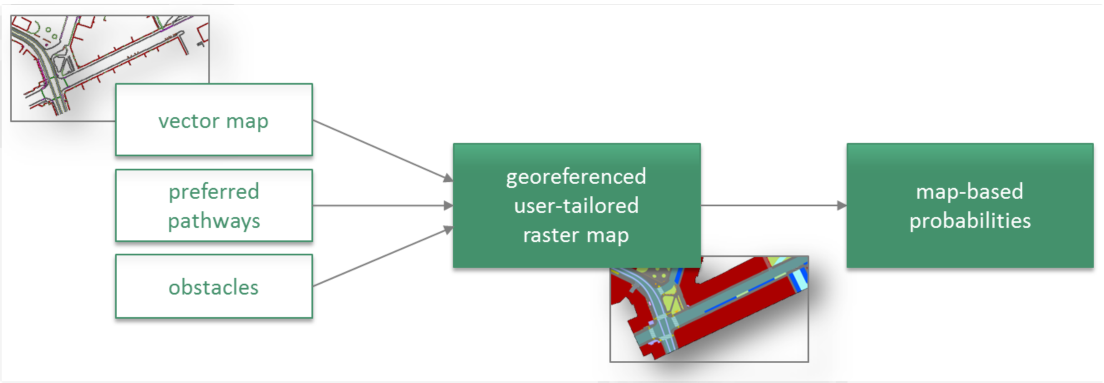


Figure 5.4: Concept for the generation of the user-tailored raster map.

As illustrated in Figure 5.4, the basis for the generation of the georeferenced user-tailored map was a vector map due to the absence of a suitable raster map. The vector map contains already most of the relevant information. However, the vector map describes the environment as a graph, which consists of nodes and arcs representing the environment's objects and features. Regardless which object is mapped, it is done by nodes and node connecting arcs. In case of a vector map, the verification of the current position relative to an object is done with the help of analytical geometry, coming along with high computational effort. On the contrary, mapping the environment as a raster and assigning an appropriate value, which represents the feature (e.g., building, street, pavement, zebra crossing, etc.) to each raster element, enables an easy and fast verification of the underlying object. Therefore, a conversion from the vector map to a raster representation is performed.

The original vector map is provided as CAD dataset by the surveying office of Graz. Within the CAD dataset, different objects are assigned to different layers. The information content of the map is based on geodetic surveys, which were performed in 2009. Consequently, the dataset had to be updated in areas, where construction works took place. The map coordinates correspond to the projected spatial reference system MGI Austria GK (east).

For the design of the user-tailored map, a consultant, who is blind by herself, was involved in the developing process. The map was generated within the project POScity in cooperation with the company Strauß & Hollinger : GeoIT OG and the consultant Alice Geiger. Details on the map generation can be found in Hafner et al. (2014a) and in the final report of the project POScity (Hafner et al., 2015). Alice Geiger gave an insight into the navigation manner of visually impaired persons, inspected the existing dataset of the surveying office, and supported the derivation of the map-based probabilities. The existing dataset was enriched by missing blind-relevant data, while irrelevant information was eliminated from the map. Based on Alice Geiger's expertise and recommendations, a concept for the integration of the user-tailored map into the positioning process has been worked out. Based on this concept, I defined the map requirements, whereby the user-tailored raster map itself was then generated by the company Strauß & Hollinger : GeoIT OG.

For the automatic generation of the raster map, the information content of the vector map had to be generalized. Figure 5.5 demonstrates the generation process of the user-tailored raster map from the vector map containing all relevant information via the generalization into the final product. The vector dataset illustrates the enriched and updated dataset of the surveying office. Within the first generalization, the goal is a consistent classification of the point data, while the second generalization focuses on the consistency with regard to line data. During this process, duplicates are eliminated and unambiguous classification keys are assigned to each feature. After the generalization, the dataset is topologically correct over all different layers.

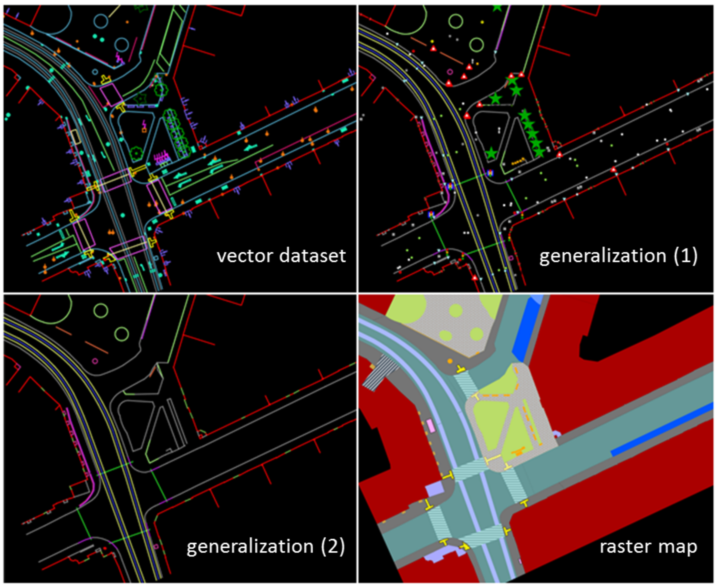


Figure 5.5: For the semi-automatic generation of the raster map, a generalization of the vector map had to be performed in advance. © Strauß & Hollinger : GeoIT OG

Due to the extensive work for the generation of the user-tailored map, the test area had to be restricted to a limited area of Graz. For the choice of the test area, special attention was paid on the presence of infrastructure relevant to blind people and the possibility to test various different pathway scenarios. The area needs to include tactile ground surface indicators, acoustic traffic lights, zebra crossings, etc. At the end, the surrounding of the campus of Graz University of Technology, comprising all kinds of features, had been chosen due to practical reasons having the test environment close to the Institute of Geodesy.

The final user-tailored raster map of the test environment is shown in Figure 5.6 and has a spatial resolution of 25 cm. Each color represents a special feature, for example, red symbolizes buildings, blue is a parking lot and green represents a grassed area. Since no information was available for the inner courtyards of the building blocks, the information was taken from a digital surface model or an aerial image.

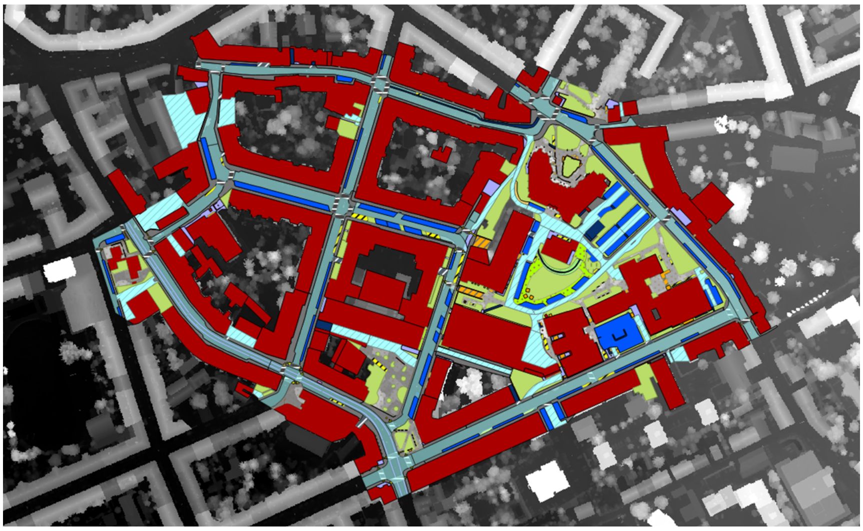
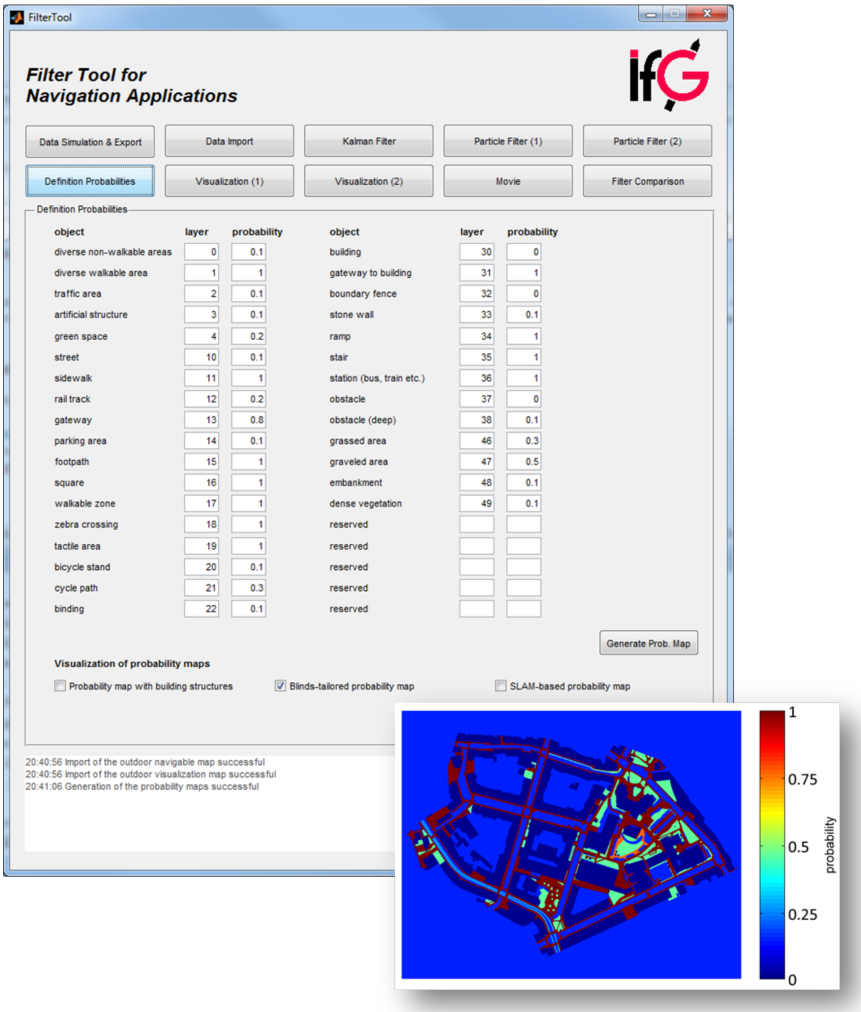


Figure 5.6: User-tailored raster map for the surrounding of the campus of Graz University of Technology. The raster map is visualized with a digital surface model in the background. © Strauß & Hollinger : GeoIT OG

The generated user-tailored raster map acts as basis for the derivation of map-based probabilities. To enable the modification of the probabilities at any time, the raster map had to be designed appropriately. Each raster element contains a feature value, which corresponds to a certain layer. As illustrated in Figure 5.7, within the GUI, the probability of each layer can be defined and adapted. Therefore, different probability maps can be generated to investigate the filter performance based on diverse spatial constraints. The map in Figure 5.7 represents an exemplary probability map of the test area. The particle filter uses such a probability map to enhance the estimation of the most likely position of the pedestrian.



Definition Probabilities

object	layer	probability	object	layer	probability
diverse non-walkable areas	0	0.1	building	30	0
diverse walkable area	1	1	gateway to building	31	1
traffic area	2	0.1	boundary fence	32	0
artificial structure	3	0.1	stone wall	33	0.1
green space	4	0.2	ramp	34	1
street	10	0.1	stair	35	1
sidewalk	11	1	station (bus, train etc.)	36	1
rail track	12	0.2	obstacle	37	0
gateway	13	0.8	obstacle (deep)	38	0.1
parking area	14	0.1	grassed area	46	0.3
footpath	15	1	graveled area	47	0.5
square	16	1	embankment	48	0.1
walkable zone	17	1	dense vegetation	49	0.1
zebra crossing	18	1	reserved		
tactile area	19	1	reserved		
bicycle stand	20	0.1	reserved		
cycle path	21	0.3	reserved		
binding	22	0.1	reserved		

Visualization of probability maps

□ Probability map with building structures ☑ Blinds-tailored probability map □ SLAM-based probability map

```
20:40:56 Import of the outdoor navigable map successful
20:40:56 Import of the outdoor visualization map successful
20:41:06 Generation of the probability maps successful
```

Figure 5.7: Definition of the probabilities for each layer, which is the basis for the generation of the probability map.

5.1.2 Height estimation within the Kalman filter

The algorithmic concept of the Kalman filter estimating the height is visualized in Figure 5.3. Since the probability map provides no height constraints, the particle filter is not necessarily required for the height determination. By outsourcing the height estimation into a Kalman filter, the computational efficiency of the filtering process could be improved. This is possible, since the Kalman filter assumptions regarding the linearity and the model's distribution suit for the application of estimating the current height.

The parameters which are estimated within the Kalman filter are the ellipsoidal height h referred to the reference frame WGS84, and the height component $\dot{h}$ of the velocity. Therefore, the state vector $\boldsymbol{x}_t$ has two components, which are:

$$\boldsymbol{x}_t = \begin{bmatrix} h \\ \dot{h} \end{bmatrix}_t . \tag{5.1}$$

The absolute height h_{GNSS}, based on the GNSS observations, and the relative height Δh_{baro}, gained from the barometer, serve as input for the Kalman filter, see Figure 5.3. h_{GNSS} is given as ellipsoidal height in the reference frame WGS84. Since the barometer delivers no absolute height information, no reference system has to be defined. Besides h_{GNSS} and Δh_{baro}, the pre-processing of the GNSS and the barometer sensor data provides the corresponding variances $\sigma_{h_{GNSS}}$ and $\sigma_{\Delta h_{baro}}$. Consequently, the input for the Kalman filter, the observation vector $\boldsymbol{z}_t$ and the covariance of the observations $\boldsymbol{R}_t$, can be defined as follows:

$$\boldsymbol{z}_t = \begin{bmatrix} h_{GNSS} \\ \Delta h_{baro} \end{bmatrix}_t , \qquad \boldsymbol{R}_t = \begin{bmatrix} \sigma^2_{h_{GNSS}} & 0 \\ 0 & \sigma^2_{\Delta h_{baro}} \end{bmatrix}_t . \tag{5.2}$$

As the observation and state vector have been identified, the time and measurement update of the Kalman filter, given by equation (3.15) to (3.19), can be derived. Therefore, the dynamic model and the relation between the observations and the unknown parameters of the state vector (measurement model) have to be defined.

Time update

The time update of the state $\tilde{\boldsymbol{x}}_t = \boldsymbol{\Phi}_t \hat{\boldsymbol{x}}_{t-1}$ within the Kalman filter is realized by

$$\begin{bmatrix} \tilde{h} \\ \tilde{\dot{h}} \end{bmatrix}_t = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}_t \begin{bmatrix} \hat{h} \\ \hat{\dot{h}} \end{bmatrix}_{t-1} , \tag{5.3}$$

where Δt is the shorthand notation for $\Delta t = t_{k+1} - t_k$. For the definition of the transition matrix $\boldsymbol{\Phi}_t$, a uniform velocity is assumed delivering the height h at epoch t by

$$\tilde{h}_t = \hat{h}_{t-1} + \Delta t \, \hat{\dot{h}}_{t-1}. \tag{5.4}$$

Besides the state, also the covariance has to be propagated from the previous to the current epoch within the prediction step. For the time update of the covariance given in equation (3.16) the system noise has to be defined to compensate for the idealized assumption of the dynamic model (Hofmann-Wellenhof et al., 2003). Referred to Legat (2002), the second-order time derivative $\ddot{h}$ of the height is suitable for the description of the system noise. Since the change of the height is defined by a uniform motion with constant velocity, the acceleration is qualified to account for any kind of disturbances with respect to the assumed motion. As described in Hofmann-Wellenhof et al. (2003), the impact of the acceleration $\ddot{h}$ on the state vector components h and $\dot{h}$ can be derived by principal laws of kinematics:

$$\begin{aligned}\begin{bmatrix}\Delta h\\ \Delta\dot{h}\end{bmatrix}_t &= \begin{bmatrix}\frac{1}{2}\Delta t^2\\ \Delta t\end{bmatrix}\hat{\ddot{h}}_{t-1},\\ &= \boldsymbol{N}_t\,\hat{\ddot{h}}_{t-1}.\end{aligned} \tag{5.5}$$

The matrix $\boldsymbol{N}_t$ is denoted as noise transition matrix. The covariance matrix $\boldsymbol{Q}_t$ of the system noise, which is required for the covariance update given in equation (3.16), can now be determined by applying the law of error propagation:

$$\boldsymbol{Q}_t = \boldsymbol{N}_t\boldsymbol{R}_n\boldsymbol{N}_t^T, \tag{5.6}$$

with

$$\boldsymbol{R}_n = \begin{bmatrix}\sigma^2_{\ddot{h}}\end{bmatrix}. \tag{5.7}$$

The variance $\sigma^2_{\ddot{h}}$ describes the uncertainty of the dynamic model. The covariance matrix $\boldsymbol{Q}_t$ enables the handling of the imperfect dynamic model by increasing the covariance $\tilde{\boldsymbol{\Sigma}}_t$ of the predicted state $\tilde{\boldsymbol{x}}_t$ corresponding to $\tilde{\boldsymbol{\Sigma}}_t = \boldsymbol{\Phi}_t\boldsymbol{\Sigma}_{t-1}\boldsymbol{\Phi}_t^T + \boldsymbol{Q}_t$. Consequently, the system noise influences the impact of the prediction to the final state estimate, whereby a low variance corresponds to a high impact of the prediction.

For investigation purposes, the system noise can be adapted within the user interface, since this parameter essentially influences the performance of the filter. In general, this parameter strongly depends on the input data, its covariance, and the assumed motion. Therefore, the adaption of the system noise has to be done with regard to the dynamic system and the corresponding observations.

Measurement update

If GNSS or barometer observations are available, the predicted state is updated according to equation (3.18) by incorporating the measurements. For the measurement update, the observation equations for both types of observations, h_{GNSS} and Δh_{baro}, have to be defined:

$$\begin{bmatrix}h_{GNSS}\end{bmatrix}_t = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}h\\ \dot{h}\end{bmatrix}_t, \tag{5.8}$$

$$\begin{bmatrix}\Delta h_{baro}\end{bmatrix}_t = \begin{bmatrix}0 & \Delta t\end{bmatrix}\begin{bmatrix}h\\ \dot{h}\end{bmatrix}_t. \tag{5.9}$$

The matrices linking the state vector $\boldsymbol{x}_t$ with the observations h_{GNSS} and Δh_{baro} are denoted by $\boldsymbol{H}_{GNSS}$ and $\boldsymbol{H}_{baro}$. As equations (5.8) and (5.9) show, the relation between the observations and the state vector components are given by linear functions of h and $\dot{h}$. Depending on the availability of observations, the state is either updated by one measurement group (GNSS or barometric data) or by both. The incorporation of both groups demands a combination of the design matrices $\boldsymbol{H}_{GNSS}$ and $\boldsymbol{H}_{baro}$:

$$\begin{aligned}\begin{bmatrix} h_{GNSS} \\ \Delta h_{baro} \end{bmatrix}_t &= \begin{bmatrix} \boldsymbol{H}_{GNSS} \\ \boldsymbol{H}_{baro} \end{bmatrix} \begin{bmatrix} h \\ \dot{h} \end{bmatrix}_t \\ &= \boldsymbol{H}_t \, \boldsymbol{x}_t,\end{aligned} \tag{5.10}$$

resulting in a design matrix $\boldsymbol{H}_t$, which is equal to an identity matrix of size $[2 \times 2]$ if a constant measurement frequency of 1 Hz is used.

Since h_{GNSS} and Δh_{baro} are assumed to be independent measurements, the covariance matrix $\boldsymbol{R}_t$ of both measurement groups depends on the covariance $\boldsymbol{R}_{GNSS}$ of the GNSS height h_{GNSS} and the covariance $\boldsymbol{R}_{baro}$ of the barometric height Δh_{baro} in the following form:

$$\boldsymbol{R}_t = \begin{bmatrix} \boldsymbol{R}_{GNSS} & 0 \\ 0 & \boldsymbol{R}_{baro} \end{bmatrix}_t, \tag{5.11}$$

with

$$\boldsymbol{R}_{GNSS} = \sigma^2_{h_{GNSS}} \tag{5.12}$$

$$\boldsymbol{R}_{baro} = \sigma^2_{\Delta h_{baro}}, \tag{5.13}$$

with $\sigma^2_{h_{GNSS}}$ and $\sigma^2_{\Delta h_{baro}}$ being the variances of the GNSS-based and barometer-based input data. The variance of h_{GNSS} is provided as NMEA output by the smartphone and is in the range of some meters. The actual accuracy strongly depends on the number of currently visible satellites and their constellation. The variance of Δh_{baro} is defined within the preprocessing of the barometer measurements. The accuracy is empirically determined based on several measurement datasets, whereby a standard deviation of $\sigma_{\Delta h_{baro}} = 0.7\,\mathrm{m}$ turned out to be the value which represents the barometer accuracy best.

The defined design matrices and the appropriate covariances of the observations are the basis for the measurement update, which is responsible for the update of the state and the covariance with regard to the determined Kalman weight $\boldsymbol{K}_t$ and the current observation vector $\boldsymbol{z}_t$ by applying equation (3.18) and (3.19).

5.1.3 2D position estimation within the particle filter

Outdoors, the particle filter combines the estimated GNSS position, the velocity and heading output of the PDR algorithm, and the information retrieved from the user-tailored raster map to reliably estimate the current position of the blind person (see Figure 5.3). The particle filter is chosen for the estimation of the geographic coordinates φ and λ due to its ability of integrating map information.

The principle framework of the particle filter is given by the Chapman-Kolmogorov equation defined in equation (3.5) and the Bayes' law given in equation (3.6). While the Chapman-Kolmogorov equation is the basis for the propagation of the particles, the Bayes' law incorporates the current measurements into the state estimation process. As the implementation of the Bayes filter is not computationally tractable, the particle filter approximates equation (3.5) and equation (3.6) by representing the state space by a set of particles. The importance sampling described in section 3.2.2 is the basis for the implementation of the particle filter. However, the practical implementation strongly depends on the application and is therefore described in the course of this section.

As shown in Figure 5.3, the developed particle filter incorporates the PDR data for the propagation of the particles, while the GNSS input and the map are used for the determination of the importance weights. Since information about the actual movement of the pedestrian is available, the PDR data is used instead of a common pedestrian motion model to propagate the particles from one epoch to the next. The importance weight is computed with respect to two uncorrelated data sources (GNSS and map) and provides the posterior PDF of the state.

Based on the Chapman-Kolmogorov equation in equation (3.5) and the particle representation of the state, the prediction within the particle filter can be defined by

$$\begin{aligned} \overline{bel}(\boldsymbol{x}_t) &= p(\boldsymbol{x}_t | \boldsymbol{z}_{t-1}, \boldsymbol{u}_t) \\ &= \sum_{N_s} p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}, \boldsymbol{u}_t)\, bel(\boldsymbol{x}_{t-1}), \end{aligned} \tag{5.14}$$

with $\boldsymbol{u}_t$ representing the transition data gained from the PDR algorithm. The measurement update incorporates the GNSS position solutions $\boldsymbol{z}_t$ and the map-based probabilities given by m in the following form:

$$\begin{aligned} bel(\boldsymbol{x}_t) &= p(\boldsymbol{x}_t | \boldsymbol{z}_t, m) \\ &= \eta_t\, p(\boldsymbol{z}_t, m | \boldsymbol{x}_t)\, \overline{bel}(\boldsymbol{x}_t) \\ &= \eta_t\, p(\boldsymbol{z}_t | \boldsymbol{x}_t)\, p(m | \boldsymbol{x}_t)\, \overline{bel}(\boldsymbol{x}_t). \end{aligned} \tag{5.15}$$

Since the GNSS positions and the map are independent data sources, the probability density function $p(\boldsymbol{z}_t, m|\boldsymbol{x}_t)$ can be split into two PDFs. One PDF represents the measurement likelihood $p(\boldsymbol{z}_t|\boldsymbol{x}_t)$, while the other one stands for the map-based probabilities $p(m|\boldsymbol{x}_t)$.

In the following, details on the initialization of the filter, the propagation model, the computation of the importance weights, and the resampling of the particles are given. The Dynamic Bayes Network of the designed particle filter is illustrated in Figure 5.8.

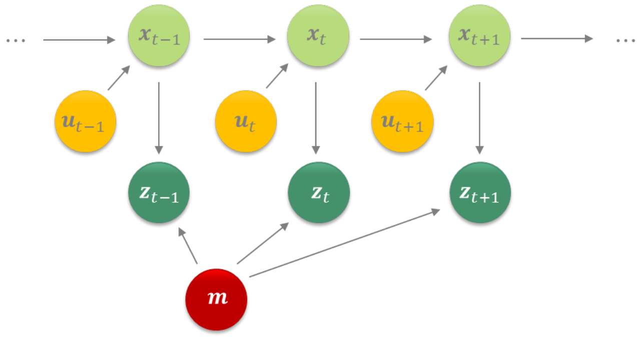


Figure 5.8: DBN of the designed particle filter.

Initialization of the particle filter

For the recursive state estimation within the particle filter, a once-only initialization of the particle space has to be performed. After the initialization, a particle set $\boldsymbol{\chi}_t$ represents the probability density of the state space. Within the positioning tool, a particle is defined by $\boldsymbol{x}_t^i = [\varphi, \lambda, \theta]^T$ comprising the geographic coordinates and the orientation of the pedestrian. The orientation corresponds to the heading (direction of the along axis) defined in section 4.3.1, in general, describing the walking direction of the person with respect to geographic north, i.e. the course angle.

Within the positioning tool, several methods for the initialization exist. The particles can either be randomly extracted from a uniform distribution over the whole state space, or can be drawn from a normal distribution with external information about the mean and the appropriate covariance. In case of the geographic coordinates φ and λ, the coordinates of the starting point of a trajectory (e.g. entrance of a building, crossing, etc.) or the first GNSS solution are qualified as mean for the Gaussian initialization. The corresponding covariance depends on the quality of the information. Regardless if the initialization is done by a random or a defined PDF, the user-tailored map is used as constraining factor to exclude improbable particles already during the initialization. The heading is initialized randomly within the interval $[0°, 360°]$ or by a Gaussian distribution $\mathcal{N}(\theta_{mag}, \sigma_{\theta_{mag}})$ with the magnetic heading θ_{mag} as mean and the heading's measurement noise $\sigma_{\theta_{mag}}$ as covariance.

Besides the particles' distribution in the state space, the number of particles N_s has to be defined within the initialization of the filter. Within the GUI, an arbitrary value for N_s can be chosen with regard to the application and the employed data. A number N_s between 500 and 1000 proved to be qualified for the outdoor positioning. An investigation on the actually necessary number of particles is given in chapter 6, which demonstrates that the choice for N_s depends on the input data and its quality. In a first instance, the N_s initialized particles have equal weights $\omega_{t_0}^i = 1/N_s$.

Particle propagation

Within the propagation step of the particle filter, a new set of particles is generated by sampling the state transition probability $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)$. This PDF is conditioned on the particle's state $\boldsymbol{x}_{t-1}$ of the previous epoch $t-1$ and the control or transition data $\boldsymbol{u}_t$. The control data $\boldsymbol{u}_t$ implies the actual velocity v_t and change in heading $\Delta\theta$ determined within the PDR algorithm. The PDR data is used for the propagation of the particles to enable a particle transition in the best possible way.

Another possibility for the particle propagation is the application of a generic motion model independently of any measurement data. In this case, the PDR is used for the computation of a PDR-based importance weight within the correction step. Particles which are propagated consistently to the actual velocity and heading would therefore get large weights, while particles whose propagation does not correspond to the determined motion are assigned with low weights. Investigations in the course of this thesis showed that within this approach, many propagated particles have a low PDR-based importance weight, and therefore, only a few particles contribute in the state estimation. The consequence is, that much more particles have to be generated. As a result, the particle propagation is done with PDR data to reduce the uncertainty of the propagation and thus, the number of necessary particles. Consequently, the vector $\boldsymbol{u}_t$, expressed by

$$\boldsymbol{u}_t = \begin{bmatrix} v \\ \Delta\theta \end{bmatrix}_t, \tag{5.16}$$

is used for the particle propagation. To preserve the diversity of the particles, the parameters v and $\Delta\theta$ are spread according to the measurement uncertainty of the IMU

$$v'_t = v_t + \mathcal{N}(0, \sigma_v) \tag{5.17}$$
$$\Delta\theta'_t = \Delta\theta_t + \mathcal{N}(0, \sigma_{\Delta\theta}), \tag{5.18}$$

by adding Gaussian noise corresponding to $\mathcal{N}(0, \sigma_v)$ and $\mathcal{N}(0, \sigma_{\Delta\theta})$ to the determined motion parameters. Based on v'_t and $\Delta\theta'_t$, the coordinates and the heading of the particles at time t are calculated by:

$$\theta^i_t = \theta^i_{t-1} + \Delta\theta'_t, \tag{5.19}$$
$$\varphi^i_t = \varphi^i_{t-1} + \frac{\Delta t\, v'_t\, \cos\theta_t}{R}, \tag{5.20}$$
$$\lambda^i_t = \lambda^i_{t-1} + \frac{\Delta t\, v'_t\, \sin\theta_t}{R\, \cos\varphi_0}, \tag{5.21}$$

where θ_{t-1}, φ_{t-1}, and λ_{t-1} correspond to the particles' parameters of the previous epoch $t-1$. The Earth radius R and a mean latitude φ_0 of the test ground are applied to transform the relative motion into the geographic reference system.

Within the defined filter, the change of the heading $\Delta\theta_{PDR}$ is used instead of the absolute heading θ_{PDR}. The current heading θ_t is estimated within the particle filter, as this reduces the PDR-typical drift due to the integration of additional constraints and observations within the estimation process.

The applied approach could be improved by defining a dynamic model based on PDR data which also accounts for accelerations. However, the accelerations which are typically caused by blind pedestrians are rather small, and therefore are in the range of the particles' diversity given by the measurement uncertainty of the IMU.

Computation of the importance weight

The sequential importance resampling, where the particles are drawn with regard to the importance weights, is the general framework for the implementation of the measurement update of the particle filter (Arulampalam et al., 2002; Ristic et al., 2004). The importance weight ω_t^i, which acts as basis for the resampling of the particles, is a combination of two independent groups of weights. While GNSS provides an observation-based importance weight ${\omega_{GNSS}}_t^i$, the user-tailored raster map delivers a map-based probability which corresponds to the importance weight ${\omega_{map}}_t^i$ (see Figure 5.3).

The importance weight ${\omega_{GNSS}}_t^i$, which represents the likelihood of the GNSS measurements ${\boldsymbol{z}_{GNSS}}_t = [\varphi,\, \lambda]_{GNSS}^T$ at a certain point in the state space, is determined by evaluating the measurement probability $p({\boldsymbol{z}_{GNSS}}_t|\boldsymbol{x}_t)$ based on ${\boldsymbol{z}_{GNSS}}_t$ and the appropriate covariance

$$\boldsymbol{\Sigma}_{GNSS} = \begin{bmatrix} \sigma^2_{\varphi_{GNSS}} & 0 \\ 0 & \sigma^2_{\lambda_{GNSS}} \end{bmatrix}. \tag{5.22}$$

In case of the GNSS observation ${\boldsymbol{z}_{GNSS}}_t$, the probability of the measurement is evaluated by determining whether ${\boldsymbol{z}_{GNSS}}_t$ corresponds to the normal distribution ${\boldsymbol{z}_{GNSS}}_t \sim \mathcal{N}(\boldsymbol{H}_t\boldsymbol{x}_t^i, {\boldsymbol{\Sigma}_{GNSS}}_t)$ with mean $\boldsymbol{H}_t\boldsymbol{x}_t^i$ and covariance ${\boldsymbol{\Sigma}_{GNSS}}_t$. Consequently, the GNSS-based importance weight ${\omega_{GNSS}}_t^i$ is gained by

$$\begin{aligned} {\omega_{GNSS}}_t^i &= p({\boldsymbol{z}_{GNSS}}_t|\boldsymbol{x}_t^i) \\ &= det(2\pi{\boldsymbol{\Sigma}_{GNSS}}_t)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}({\boldsymbol{z}_{GNSS}}_t - \boldsymbol{H}_t\boldsymbol{x}_t^i)^T {\boldsymbol{\Sigma}_{GNSS}}_t^{-1}({\boldsymbol{z}_{GNSS}}_t - \boldsymbol{H}_t\boldsymbol{x}_t^i)\right), \end{aligned} \tag{5.23}$$

with $\boldsymbol{x}_t^i$ being the particles which represent the state space. $\boldsymbol{H}_t$ corresponds to the design matrix describing the noise-free relationship ${\boldsymbol{z}_{GNSS}}_t = \boldsymbol{H}_t\boldsymbol{x}_t$ between the unknown state $\boldsymbol{x}_t$ and ${\boldsymbol{z}_{GNSS}}_t$. $\boldsymbol{H}_t$ is required to relate ${\boldsymbol{z}_{GNSS}}_t$ to the particle set $\boldsymbol{x}_t^i$ and has following structure:

$$\boldsymbol{H}_t = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{5.24}$$

Within the user interface of the positioning tool, the map-based probability can be defined for each layer of the user-tailored raster map, see section 5.1.1. To get the map-based probability, the particle's location has to be determined in relation to the map. In case that the particle $\boldsymbol{x}_t^i$ is located in an impossible particle space (obstacle, wall, indoor environment, etc.), a probability of zero is assigned to the appropriate map-based importance weight ${\omega_{map}}_t^i$. Otherwise, ${\omega_{map}}_t^i$ corresponds to the location probability which was defined in the GUI. Consequently, the map-based probability becomes

$$ {\omega_{map}}_t^i = \begin{cases} 0 & \ldots \text{impossible particle space,} \\ p_{map} & \ldots \text{otherwise.} \end{cases} \tag{5.25} $$

In comparison, the integration of the information content of a common map of the outdoor environment would provide the following map-based probabilities

$$ {\omega_{map}}_t^i = \begin{cases} 0 & \ldots \text{impossible particle space,} \\ 1 & \ldots \text{otherwise,} \end{cases} \tag{5.26} $$

differentiating only the impossible from the possible particle space.

The importance weight ω_t^i as input for the resampling is determined by combining ${\omega_{GNSS}}_t^i$ and ${\omega_{map}}_t^i$ by

$$ \omega_t^i = \left\langle {\omega_{GNSS}}_t^i, {\omega_{map}}_t^i \right\rangle . \tag{5.27} $$

Because of the fact that both groups of importance weights are uncorrelated, the final importance weight ω_t^i is given by the multiplication of ${\omega_{GNSS}}_t^i$ and ${\omega_{map}}_t^i$, since

$$ p(z|x) = p(z_1, z_2|x) = p(z_1|x)p(z_2|x) \tag{5.28} $$

for two independent observations z_1 and z_2. However, before combining ${\omega_{GNSS}}_t^i$ and ${\omega_{map}}_t^i$, the corresponding weights have to be normalized:

$$ {\omega_{GNSS}}_t^i = \frac{{\omega_{GNSS}}_t^i}{\sum_{N_s} {\omega_{GNSS}}_t^i}, \qquad {\omega_{map}}_t^i = \frac{{\omega_{map}}_t^i}{\sum_{N_s} {\omega_{map}}_t^i}. \tag{5.29} $$

After the computation of ω_t^i, the equal weights of the particle set $\boldsymbol{\chi}_t$ are replaced by the combined importance weights ω_t^i. Consequently, the particle set $\boldsymbol{\chi}_t = \left\langle \boldsymbol{x}_t^i, \omega_t^i \right\rangle$ represents the posterior probability of the state $\boldsymbol{x}_t$ based on all available input data. The corresponding posterior probability function $bel(\boldsymbol{x}_t)$ is gained by

$$ p(\boldsymbol{x}_t|\boldsymbol{z}_t, m) \approx \sum_{i=1}^{N_s} \omega_t^i \, \delta(\boldsymbol{x}_t - \boldsymbol{x}_t^i), \tag{5.30} $$

where $\delta(\cdot)$ is the Dirac delta measure (Ristic et al., 2004), and m represents the user-tailored raster map. $p(\boldsymbol{x}_t|\boldsymbol{z}_t, m)$ is the basis for the generation of a resampled set of particles for the consecutive epoch and for the determination of the walked trajectory. In section 6.1, the highest weighted particle, the sample mean, the weighted sample mean, and methods like k-means clustering are theoretically investigated for the derivation of the trajectory based on $p(\boldsymbol{x}_t|\boldsymbol{z}_t, m)$. In section 6.2, the optimal representative of the trajectory is identified with the help of realistic measurements.

Resampling

Within the resampling, a new set of N_s particles is generated by randomly sampling the states of the particles in proportion to their importance weights (Woodman, 2010). Due to the resampling of the particles, the degeneracy problem can be solved, since the particles are redistributed according to their actual probability. Therefore, as shown in Figure 5.9, by considering a resampling within the particle filter, the processing will be focused on regions of higher probability, and the number of particles with low weights can be reduced. After the resampling, the new set of N_s particles have equal weights $\omega_t^i = 1/N_s$.

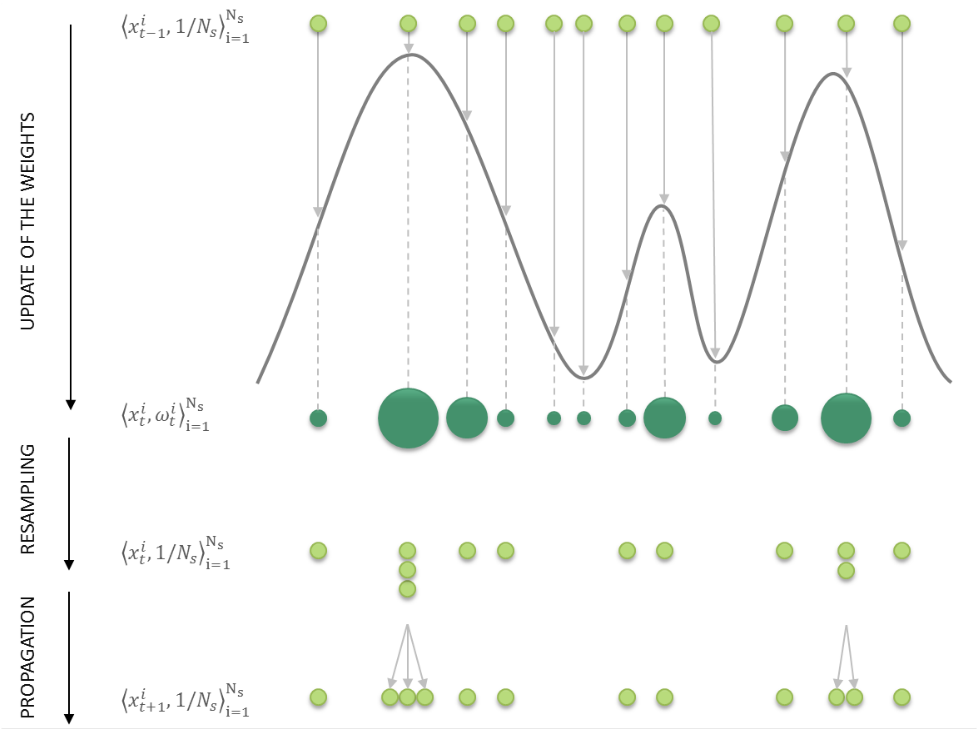

UPDATE OF THE WEIGHTS
RESAMPLING
PROPAGATION
$\langle x_{t-1}^i, 1/N_s\rangle_{i=1}^{N_s}$
$\langle x_t^i, \omega_t^i\rangle_{i=1}^{N_s}$
$\langle x_t^i, 1/N_s\rangle_{i=1}^{N_s}$
$\langle x_{t+1}^i, 1/N_s\rangle_{i=1}^{N_s}$

Figure 5.9: Resampling of the particles, in which particles with low weights are reduced.

Referring to Hol et al. (2006), the resampling modifies the weighted approximated density given in equation (5.30) to an unweighted density by eliminating particles with low importance weights and by duplicating particles with high importance weights. Therefore, the probability density in equation (5.30) is replaced by

$$p(\boldsymbol{x}_t|\boldsymbol{z}_t, m) \approx \sum_{i=1}^{N_s} \frac{1}{N_s}\,\delta(\boldsymbol{x}_t - \boldsymbol{x}_t^{i^*}), \tag{5.31}$$

with $\boldsymbol{x}_t^{i^*}$ representing the new set of particles. As stated in Hol et al. (2006) and Douc et al. (2005), several methods for the generation of a resampled set of particles exist. The investigations show, that all resampling algorithms lead to comparable results. Within the thesis, the *systematic resampling* has been chosen due to its simple implementation.

5.2 Indoor positioning

Referring to Alice Geiger, the blind consultant, it is impossible to generate a user-tailored map for indoor environments. The navigation behavior cannot be modeled in the same way as outdoors, because of the small-scaled structures and the very individual walking and navigation habits of blind persons in indoor environments. However, a SLAM approach, which maps the walking habits of only one person or if desired of several persons, could be qualified to incorporate navigation habits in the position estimation, see section 7.4.

Nevertheless, for nearly every public building appropriate building layouts exist, which can be integrated as spatial constraint within the particle filter. Hereby, the problem is the low public availability of these building maps. While maps of outdoor environments are widely available due to different online services, indoor maps are rarely accessible publicly and are not standardized yet. Though, the availability of indoor maps is increasing and the automatic providing of building layouts at the entrance of public buildings by any kind of data transfer is under discussion. As proposed in Woodman (2010), different companies like *Micello* (Micello, 2015), *HERE* (HERE, 2015), or *indoors* (indoors, 2015) are already providing the creation of detailed plans for navigation and location-based applications. At the moment, indoor maps are mainly available for public buildings like airports, shopping centers, or train stations. Within the thesis, an occupancy grid of the indoor environment has been established.

Besides the geometric constraints which are provided by the map, the state estimation in indoor environments is additionally supported by an algorithm which verifies the plausibility of pathways. This algorithm guarantees that rooms are only entered/left through doors, and floor changes are only possible at stairs or elevators. A so-called wall collision filter improves the 3D positioning by checking the plausibility of room and floor changes. Thereby, the reliability of the state estimation can be strongly improved.

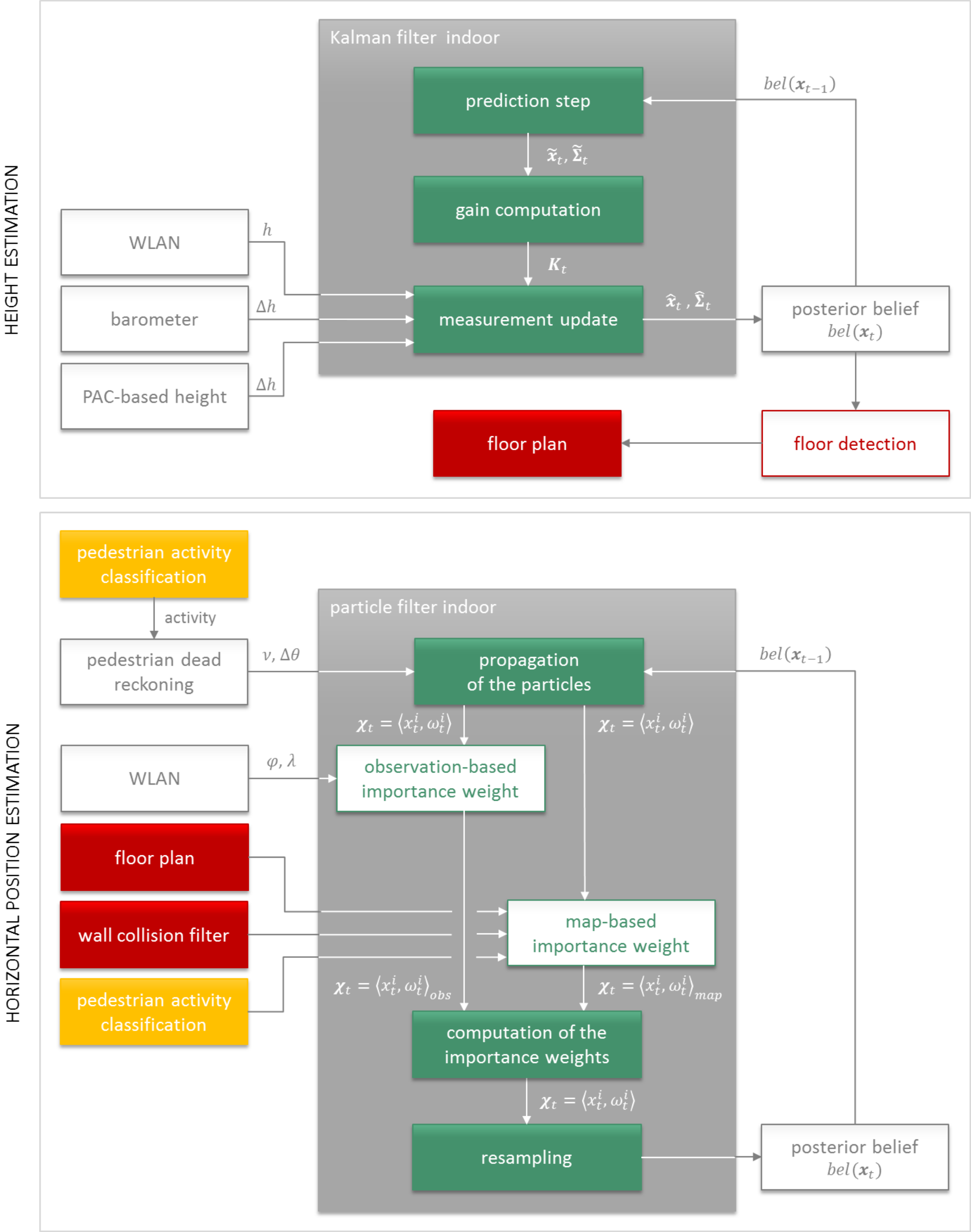

HEIGHT ESTIMATION
Kalman filter indoor
prediction step
$\tilde{x}_t, \tilde{\Sigma}_t$
gain computation
K_t
measurement update
$\hat{x}_t, \hat{\Sigma}_t$
$bel(x_{t-1})$
WLAN
h
barometer
Δh
PAC-based height
Δh
posterior belief $bel(x_t)$
floor plan
floor detection
HORIZONTAL POSITION ESTIMATION
pedestrian activity classification
activity
pedestrian dead reckoning
$v, \Delta\theta$
particle filter indoor
propagation of the particles
$bel(x_{t-1})$
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
WLAN
φ, λ
observation-based importance weight
floor plan
wall collision filter
pedestrian activity classification
map-based importance weight
$\chi_t = \langle x_t^i, \omega_t^i \rangle_{obs}$
$\chi_t = \langle x_t^i, \omega_t^i \rangle_{map}$
computation of the importance weights
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
resampling
posterior belief $bel(x_t)$

Figure 5.10: Rao-Blackwell estimation process for the positioning in indoor environments. The height, determined within a Kalman filter, is the basis for the floor detection which delivers the map of the correct floor. Besides the classified activity, the building layout acts as spatial constraint within the particle filter estimating the horizontal position.

As observations for the indoor RBPF, the PDR parameters v and $\Delta\theta$, the coordinates φ, λ, and h determined within the WLAN fingerprinting, and the barometric relative height Δh_{baro}, are used. In addition, the activity provided by the PAC and the corresponding relative height Δh_{PAC} are integrated in the state estimation.

Similar to the Rao-Blackwell filter for the outdoor environment, the filter approach for indoor environments consists of a central particle filter estimating the coordinates φ and λ (referred to WGS84), and of a Kalman filter, which provides the current ellipsoidal height h. Again, by outsourcing the height estimation, the computational efficiency of the state estimation could be increased. The difference to the outdoor approach is that the integration of map information requires the determination of the floor level. Therefore, the Kalman filter additionally identifies the actual floor. The algorithmic concept for the state estimation in indoor environments is demonstrated in Figure 5.10.

As illustrated in Figure 5.10, the height estimation integrates the absolute height h computed within the WLAN fingerprinting and information about the change in height Δh, which is either based on the barometer data (section 4.4) or results from a combined evaluation of the PDR and PAC output (see section 4.5.1). By estimating the height and the floor within the Kalman filter, much less particles are required within the particle filter, since the state estimation process can already be limited to a certain floor. Details on the Kalman filter and the appropriate floor detection are provided in section 5.2.4.

Within the particle filter, the propagation of the particles is controlled by the velocity v and the change in heading $\Delta\theta$ determined by the PDR algorithm, while the WLAN positions φ and λ are used for the importance sampling. The importance sampling is further supported by the building layout, the wall collision filter, and the spatial constraints derived from the actual activity. The particle filter concept is explained in section 5.2.5, while details for the map preparation and the design of the wall collision filter are given in section 5.2.1 and 5.2.2. The derivation of the activity-based constraints is discussed in section 5.2.3.

5.2.1 Occupancy grid map for indoor positioning

For the integration of the information content of a building layout within the particle filter, the spatial data has to be prepared in machine-readable form. A data structure which is suitable for representing buildings with multiple floors has to be defined for the processing of map-based information. Therefore, the three-dimensional building is modeled as 2.5-dimensional data. This means, each floor is represented by an appropriate floor map and a minimum and maximum height representing the vertical extension of the floor. Consequently, all features (obstructions and walls) of one floor are projected independently of their vertical extension onto the map of the corresponding floor. By providing the current height of the pedestrian, the correct floor can be identified by comparing the estimated height with the minimum and maximum height of the building's different floor levels.

Referred to Thrun et al. (2006), the indoor environment can either be represented by a feature-based or a location-based map. In general, a map is a list of objects in the environment and their locations in the form of $m = \{m_1, m_2, \ldots, m_N\}$, with N representing the total number of objects. In case of a feature-based map, the index $n = 1 \ldots N$ corresponds to the number of a feature and m_n contains the coordinates of that feature, while the index n with regard to a location-based map already represents a specific location. Considering a planar map, the map elements are given by $m_{x,y}$ instead of m_n corresponding to the dimension of the map. Consequently $m_{x,y}$ is the direct link to the feature's coordinates.

Within the thesis, a combination of the feature-based and the location-based representation is chosen. The building layout is mapped as location-based map representing occupied and non-occupied regions for the incorporation within the particle filter, while the room polygons and doors for the wall collision filter are stored as feature-based datasets. The location-based map, which corresponds to a raster map, needs a lot more storage capacity as the feature-based map. However, this fact can be neglected, because of the finite extension of the indoor environment. By providing the building layout as georeferenced raster map, the processing effort within the filter can be reduced due to the absence of geometric operations. A raster map, providing information about the occupancy of a pixel by a binary value, is called *occupancy grid map* (Thrun et al., 2006).

The feature-based representation is suitable for the wall collision filter, since doors and walls can be mapped easier by lines and polygons and the path verifications can be performed by simple geometric operations. Therefore, the spatial information has to be provided as dataset containing all relevant features and the corresponding positions. Details on the processing within the filter are given in section 5.2.2.

For the indoor positioning investigated within the thesis, a test area has been defined, on the one hand, for the construction of the occupancy grid maps and, on the other hand, for the installation of the WLAN fingerprinting database. The campus building Steyrergasse 30, in which the Institute of Geodesy is located, was chosen as test area. The building has five floors, with a planar extension of about $1300\,\mathrm{m}^2$ at each floor. Figure 5.11 exemplary illustrates the mapping of the five-story building as occupancy grid maps. Each floor of the building is represented by an occupancy grid map, which allows the verification of the occupancy of a grid cell, and an appropriate minimum and maximum height, which define the vertical extension of the floor.

For reducing the processing effort within the particle filter, a room index is assigned to every unoccupied pixel of the occupancy grid map prior to the state estimation. In this way, geo-

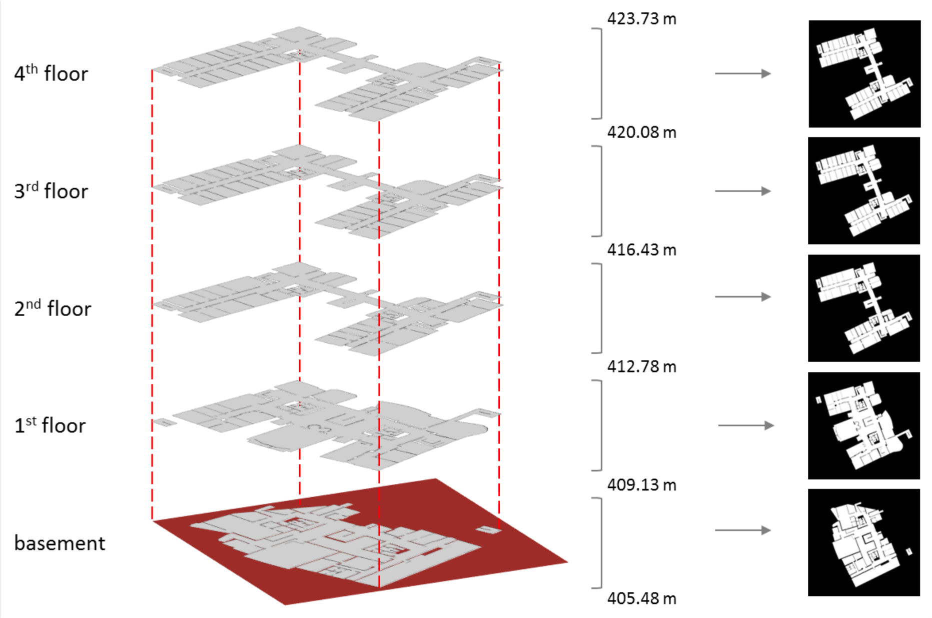


Figure 5.11: The occupancy grid maps and the height levels of each floor based on the original building layouts. The minimum and maximum height define the vertical extension of the floor.

metric operations for the determination of the current room can be circumvented. Therefore, each room had to be defined as a planar polygon and stored in a feature-based map. Every m_n element of the feature-based map contains the room index and the polygon coordinates. The room index is unambiguous and allows the identification of the room type, since the following classification key for the assignment of room indices has been defined:

$$\boldsymbol{m}_{\varphi,\lambda} = \begin{cases} 1 < m_{\varphi,\lambda} < 100 & \ldots \text{office room,} \\ 100 < m_{\varphi,\lambda} < 199 & \ldots \text{corridor,} \\ 1001 < m_{\varphi,\lambda} < 1999 & \ldots \text{stairs,} \\ 2001 < m_{\varphi,\lambda} < 2999 & \ldots \text{elevator.} \end{cases} \tag{5.32}$$

The classification into office rooms, corridors, stairs, and elevators enables a differentiation within the particle filter. Consequently, floor level transitions can be limited to stairways and elevators.

Based on the feature-based map containing the room polygons and the corresponding room indices, a developed room detection algorithm assigns a room index to every unoccupied pixel of the raster map. Therefore, geometric operations have to be performed only once to derive the properties of the location-based map $\boldsymbol{m}_{\varphi,\lambda}$. Consequently, each element of the map refers to a certain location and contains a room index. After this step, unoccupied pixels are represented by room indices, while occupied pixels are still represented by zero. Figure 5.12

shows the planar polygons defining the shape of the rooms (left) and the resulting occupancy grid map enriched by information on the room type (right).

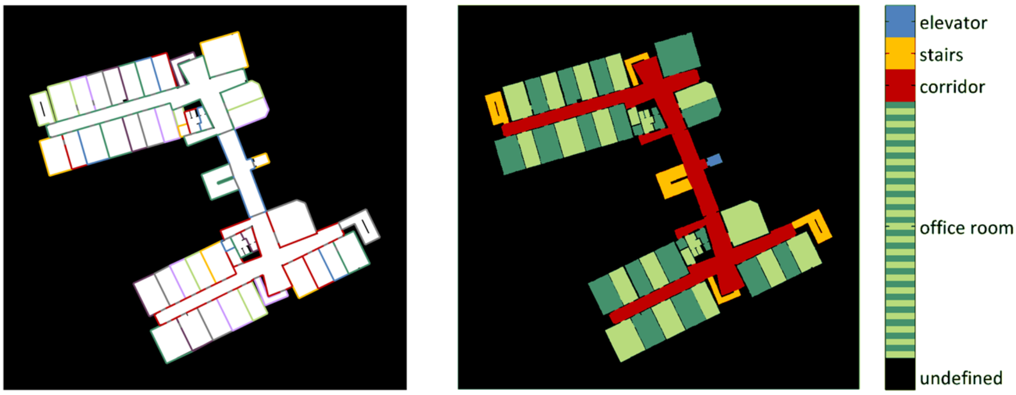


Figure 5.12: The defined room polygons (left) are the basis for the assignment of the room index to a certain location (right).

The room polygons in the left graph of Figure 5.12 are drawn in multiple colors, while the gray lines represent the doors within the building, which are required for the wall collision filter. The right hand graph of Figure 5.12 shows the result of the room classification. To illustrate the differentiation of the various office rooms, the rooms are colored in distinguishable greens. The resolution of the location-based map is 10 cm, which is sufficient for the positioning support within the filter.

The generated location-based map is used within the particle filter in several ways. On the one hand, to verify if a grid cell of the map is occupied or not, and on the other hand, to determine in which room the particle is located. The room index is used to avoid a floor level change in office rooms and corridors, but also for the verification of the plausibility of the particles' pathways by the wall collision filter.

For the wall collision filter, the geometry of the rooms and doors, and the connectivity of these elements has to be defined. Therefore, two feature-based datasets are required, which are connected by door indices. One dataset contains all room-relevant information, like the room index, the polygon defining the geometry of the room, the number of doors of a room, and corresponding door indices, which allow the unambiguous identification of a certain door. Thus, the feature-based map defining the room polygons for the room classification has to be extended by information about the doors. The second dataset comprises the door indices and the edge coordinates of the door. The door dataset is hereby directly linked to the room dataset by the unambiguous door index. The composition of the two datasets and the corresponding connection is shown in Figure 5.13. Based on these two datasets, the plausibility of the particles' pathways is proofed within the wall collision filter.

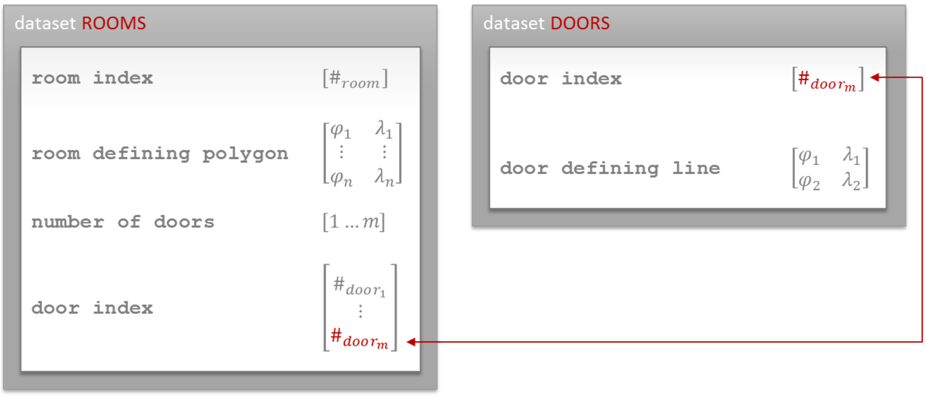

dataset ROOMS
room index
$[\#_{room}]$
room defining polygon
$\begin{bmatrix} \varphi_1 & \lambda_1 \\ \vdots & \vdots \\ \varphi_n & \lambda_n \end{bmatrix}$
number of doors
$[1 \dots m]$
door index
$\begin{bmatrix} \#_{door_1} \\ \vdots \\ \#_{door_m} \end{bmatrix}$
dataset DOORS
door index
$[\#_{door_m}]$
door defining line
$\begin{bmatrix} \varphi_1 & \lambda_1 \\ \varphi_2 & \lambda_2 \end{bmatrix}$

Figure 5.13: The feature-based datasets for defining the rooms and doors of the building.

5.2.2 Wall collision filter

The wall collision filter verifies the plausibility of the pathways after the particle propagation. In general, the wall collision filter checks the following cases:

- Did the particle take a valid path from one room to the next?
- Did the particle take a valid path from one floor level to the next?

After the application of the dynamic model based on the PDR motion parameters, the plausibility of the pathway of a particle has to be proofed. The propagated particle can be either in an occupied grid cell, or it was propagated along an impossible pathway (e.g., changing the room through a wall or the floor through the ceiling). Before the wall collision filter verifies the plausibility, the location of the particle has to be checked regarding the occupancy of the grid cell. A particle, which is assigned to a grid point representing a wall or an obstacle, has to be disregarded within the state estimation and should consequently not be treated within the wall collision filter. The left hand graph in Figure 5.14 exemplarily illustrates particles which are distributed without consideration of the environmental structures, while the graph on the right of Figure 5.14 demonstrates the particles' spatial distribution with regard to the building infrastructure.

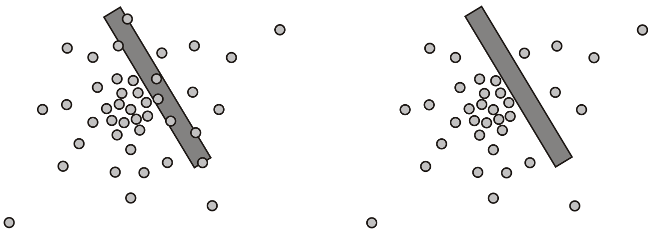


Figure 5.14: Distribution of the particles without/with consideration of the environmental structures.

The occupancy of the space is hereby verified by checking the grid cell's property, which is given in the location-based map $\boldsymbol{m}_{\varphi,\lambda}$. If the particle is located at an occupied map element $m_{\varphi,\lambda}$, the importance weight is set to zero, and therefore, the particle is not used for the later position estimation.

Figure 5.15 shows the transformation of the particles from one epoch to the next. Hereby, the wall collision algorithm has to check, whether the particle's pathway is valid. Particles will only be propagated from one room to the next if a door is between those rooms and the particle took exactly the way through.

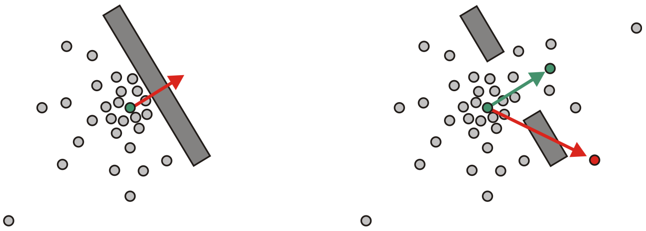


Figure 5.15: Verification of the plausibility of the propagated particles' path.

The left graph in Figure 5.15 shows the case, where two rooms are separated by a wall. If the PDR parameters indicate a motion to the next room, the wall collision filter will be responsible for assigning an importance weight of 0 to wrongly propagated particles. The right graph shows the case, where two rooms are connected by a door. Here, the wall collision filter has to verify if the particles took the way through the door. Particles which are propagated correctly are assigned with an importance weight of 1, the others with an importance weight of 0. The same plausibility check has to be done for the vertical pathway, guaranteeing floor transitions only at stairs and elevators.

The algorithmic concept of the wall collision filter is illustrated in Figure 5.16. The concept is tailored to the verification of room changes, however, the algorithm for the floor transition operates similarly. In indoor environments, besides the 2D coordinates and heading, the particle's state comprises the index of the room and the corresponding floor. This information enables a fast access to building relevant information within the wall collision filter. The floor is determined within the Kalman filter (5.2.4), while the current room is determined with the help of the location-based map $\boldsymbol{m}_{\varphi,\lambda}$ after the state transition within the particle filter (section 5.2.5). Based on the propagated particle's state $\boldsymbol{x}_t^i$ and its predecessor state $\boldsymbol{x}_{t-1}^i$, the goal of the wall collision filter is to derive an importance weight with regard to the taken pathway. Thus, the room indices of the predicted state $\boldsymbol{x}_t^i$ and its predecessor $\boldsymbol{x}_{t-1}^i$ are compared:

$$\#_{room}(\boldsymbol{x}_t^i) == \#_{room}(\boldsymbol{x}_{t-1}^i). \tag{5.33}$$

If the rooms are identical, an importance weight of 1 is assigned to the predicted particle. In case of different rooms indices, the validity of the pathway has to be verified. Therefore, the connectivity of these rooms has to be proofed. Since the indices of the doors $\#_{door_m}$ are given in the room dataset, the accordance of the doors can be figured out:

$$\#_{door}(\#_{room_1}) == \#_{door}(\#_{room_2}). \tag{5.34}$$

The next step checks whether the particle was propagated through the connecting door. Thus, the edge coordinates of the door are looked-up in the door dataset. Based on the door-defining line and the pathway, which is given by the particle's coordinates of the current and the previous epoch, the validity of the particle's propagation can be verified. If the motion corresponds to the possible way through a door, the particle is considered within the state estimation, otherwise an importance weight of 0 is assigned to the particle.

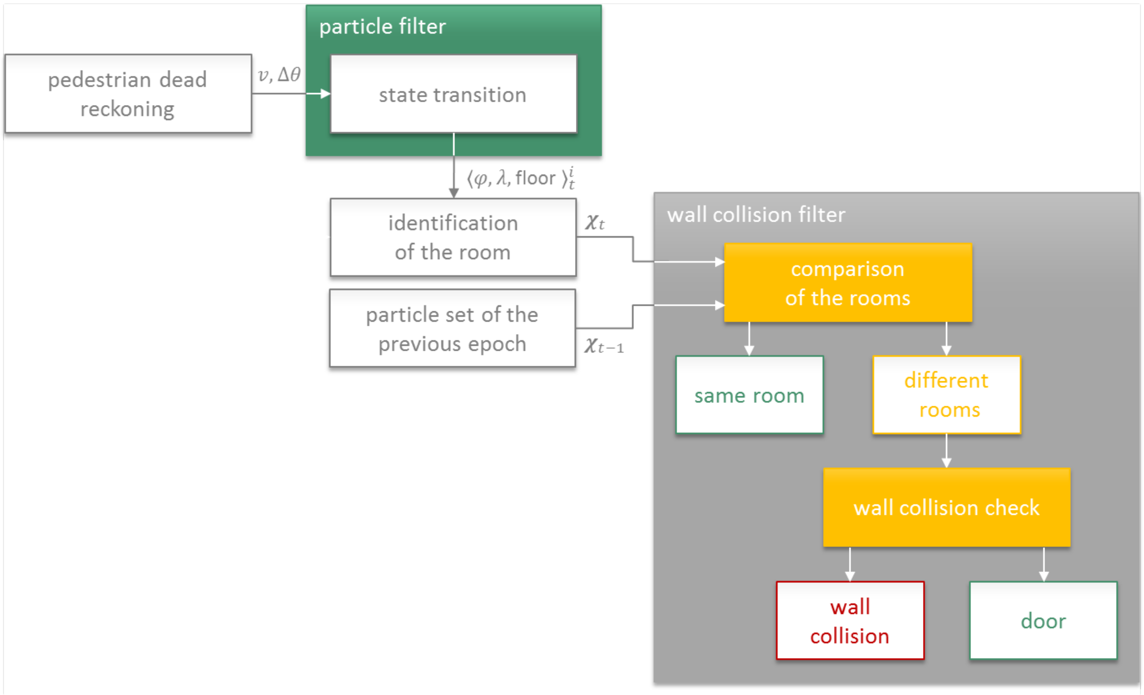


Figure 5.16: Algorithmic concept of the wall collision filter.

The plausibility verification for the floor change works similarly. Within this verification, the floor of the predicted and the predecessor state and the room type are additionally considered. Only if the particle's motion denotes a change in the floor level, the index of the room is checked concerning the possibility of a floor change. Red areas illustrated in Figure 5.17 symbolize, where floor transitions are possible according to the defined room type. The index must be in the intervals $1001 < \#_{room} < 1999$ and $2001 < \#_{room} < 2999$, which represent stairs and elevators (compare section 5.2.1). Therefore, only particles, whose vertical propagation was done at stairs or within an elevator, are assigned with an importance weight of 1.

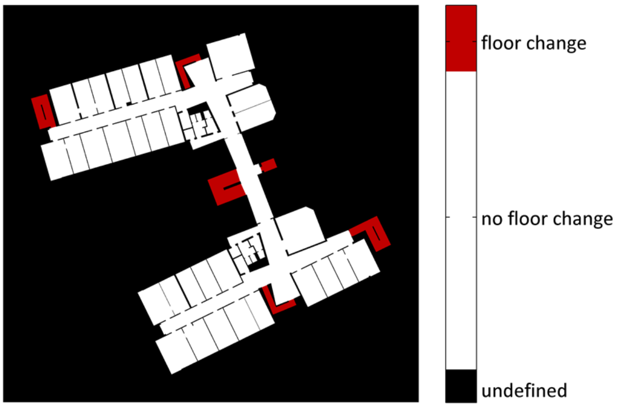


Figure 5.17: Floor changes are only possible at red-colored grid cells.

Consequently, following importance weights are assigned due to the path verification within the wall collision filter and the prior occupancy check:

$$\omega_{path}{}_t^i = \begin{cases} 0 & \ldots \text{ occupied grid} \\ 0 & \ldots \text{ unoccupied grid: invalid pathway} \\ 1 & \ldots \text{ unoccupied grid: valid pathway} \end{cases} \tag{5.35}$$

The outcome of the wall collision filter, the importance weight $\omega_{path}{}_t^i$, is considered for the determination of a map-based importance weight $\omega_{map}{}_t^i$, which is described in section 5.2.5.

5.2.3 Activity-based environmental constraints

In general, motion recognition is used as multiple support within positioning. There are various methods to improve the position estimation by motion related activities. An overview about the different domains is given in Frank et al. (2010). In Bayesian location estimation systems, the activity can be used for the selection of an appropriate movement model or to restrict the possible state space in combination with a floor plan. In contrast, Radu and Marina (2013) or Haibo et al. (2012) demonstrate the detection of floor changes based on the monitored activity, while Ibarra-Bonilla et al. (2014) adapt the step length within the PDR with regard to the current activity. Susi et al. (2013) show that an activity recognition is able to determine, how the person is carrying the smartphone (hand-held in walking direction, held in the swinging hand, etc.), which is necessary for the adaption of the PDR algorithm. On the contrary, Ling et al. (2010) use the activities *standing*, *walking*, and *walking fast* as assistance for the fingerprinting-based WLAN positioning with a smartphone. Since the smartphone used in Ling et al. (2010) updates the scanned WLAN information about every 8 to 10 seconds, the detected activity provides information for the interpolation of the motion in between.

Within this thesis, the activity, which is provided by the PAC algorithm, is used in multiple ways. As described in section 4.5, the employed PAC determines whether the person is *standing*, *walking*, *going upstairs*, or *going downstairs*. Combining the PAC output with the PDR data, changes in height can be derived if the step height is known, see section 4.5.1. Furthermore, the activity is used to adapt the step length estimation within the PDR due to the shorter step length at stairs. The third application is the retrieval of spatial constraints based on the activity in combination with the map.

Since some of the classified activities can only be performed at a limited number of places, the combined evaluation of the activity and the map delivers spatial constraints for the position estimation within the particle filter. Considering the activities defined in section 4.5, the activities *going upstairs* and *going downstairs* indicate that the pedestrian is located at the building stairways. In contrast, *walking* can be performed in office rooms and at the corridors. Based on the recall rates of the activity classification (see section 4.5), so-called PAC-based importance weights $\omega_{PAC}{}_t^i$ can be defined. If the recognized activity is *walking*, the probability that the pedestrian is in one of the office rooms or at the corridor is assumed to be higher. The magnitude of the probability is defined with respect to the recall rate of *walking*, which is 99%. Since stairways cannot be excluded due to the remaining probability of 1%, the following importance weight has been defined for *walking*:

$$\omega_{PAC}{}_t^i = \begin{cases} 0.9 & \ldots \text{office room, corridor,} \\ 0.1 & \ldots \text{stairs.} \end{cases} \tag{5.36}$$

As mentioned in section 4.5, the recognition of the activities *going upstairs* and *going downstairs* performs worse. The recall rate is about 95%. Consequently, the probability of being at stairs has to be defined less strictly, leading to a probability of

$$\omega_{PAC}{}_t^i = \begin{cases} 0.2 & \ldots \text{office room, corridor,} \\ 0.8 & \ldots \text{stairs.} \end{cases} \tag{5.37}$$

The above defined probabilities are visualized in Figure 5.18. The left hand graph illustrates the probabilities for the activities *going upstairs* and *going downstairs*, while the right hand graph symbolizes the $\omega_{PAC}{}_t^i$ for *walking*. As demonstrated, the pedestrian activity recognition allows the derivation of spatial constraints.

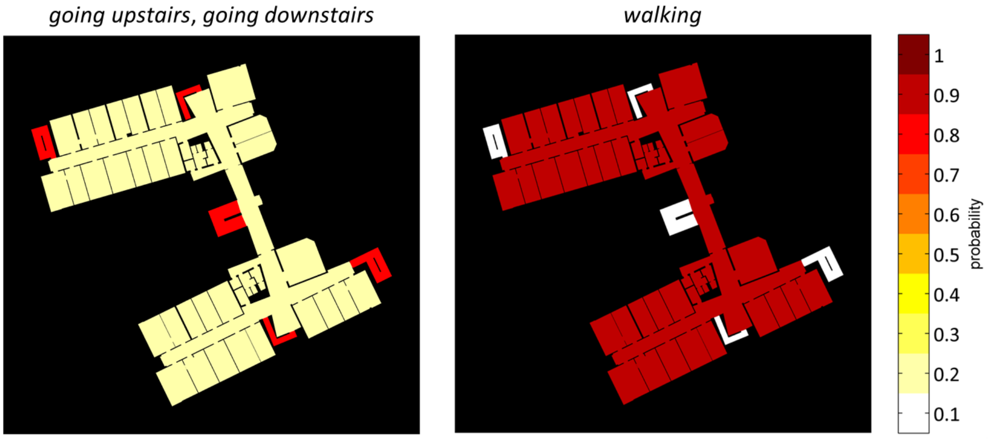


Figure 5.18: Defined probabilities with regard to the current activity. Left: probabilities for *going upstairs* and *going downstairs*. Right: probabilities for *walking*.

5.2.4 Height estimation within the Kalman filter

As illustrated in Figure 5.10, the estimation of the state parameters is done in a RBPF with the particle filter as core element determining the 2D coordinates, the heading and map-essential parameters like the current floor and room, while the Kalman filter is responsible for the height computation. Based on the height estimate, a floor detection is performed to provide the correct floor plan for the particle filter. The outsourcing of the height estimation is done due to the same reason as for the outdoor positioning. The reduction of the computational effort is even more important for the indoor positioning, since the wall collision filter causes an additional computational load.

For the indoor positioning, it is investigated, whether a relative height Δh_{PAC}, which is based on the PAC and PDR output as well as a standardized step height (see section 4.5.1), can be used instead of the barometric relative height Δh_{baro} in case that the smartphone has no internal barometer. Indoors, the absolute height information is provided by the WLAN fingerprinting and is a ellipsoidal height which is referred to WGS84. Inside buildings, a reliable height estimation is of great importance if a map is incorporated in the filter process. Therefore, Δh_{PAC} is investigated as alternative for Δh_{baro}.

In case of the availability of h_{WLAN}, Δh_{baro}, and Δh_{PAC}, the observation vector $\boldsymbol{z}_t$ and the corresponding covariance $\boldsymbol{R}_t$ are represented by

$$\boldsymbol{z}_t = \begin{bmatrix} h_{WLAN} \\ \Delta h_{baro} \\ \Delta h_{PAC} \end{bmatrix}_t , \qquad \boldsymbol{R}_t = \begin{bmatrix} \sigma^2_{h_{WLAN}} & 0 & 0 \\ 0 & \sigma^2_{\Delta h_{baro}} & 0 \\ 0 & 0 & \sigma^2_{\Delta h_{PAC}} \end{bmatrix}_t . \tag{5.38}$$

The covariance matrix $\boldsymbol{R}_t$ of the measurements contains the variances $\sigma^2_{h_{WLAN}}$, $\sigma^2_{\Delta h_{baro}}$, and $\sigma^2_{\Delta h_{PAC}}$ in the main diagonal, representing the uncertainties of the corresponding observations.

Since the observations are assumed to be uncorrelated, the off-diagonal elements are zero. However, within this thesis, Δh_{baro} and Δh_{PAC} are not evaluated in combination. Either h_{WLAN} is combined with Δh_{baro} or with Δh_{PAC}. The reason therefor is, that Δh_{PAC} is treated as an alternative for Δh_{baro} in case of the absence of a barometer. Consequently, the conformity of Δh_{baro} by Δh_{PAC} is the focus of research.

The state vector is the same as in the Kalman filter for the outdoor positioning, comprising the ellipsoidal height h and the velocity component $\dot{h}$ of the height, as given in equation (5.1). Based on the defined observation vector $\boldsymbol{z}_t$ and the state vector $\boldsymbol{x}_t$, the time and measurement update are defined in the following.

Time update

Since the state vector is equivalent for the height estimation in outdoor and indoor environments, the time update $\tilde{\boldsymbol{x}}_t = \boldsymbol{\Phi}_t \hat{\boldsymbol{x}}_{t-1}$ corresponds to the one defined in section 5.1.2, given by equation (5.3). Also the system noise covariance matrix $\boldsymbol{Q}_t$ is the same as presented in equation (5.6), whose derivation is based on equation (5.5). However, as mentioned in section 5.1.2, $\boldsymbol{Q}_t$ has to be defined with respect to the uncertainty of the assumed dynamic model and the incorporated measurements.

Measurement update

The measurement update is expanded by the PAC-based height component, while the GNSS input is replaced by height information of the WLAN fingerprinting. The measurement model for the WLAN input and therefore the design matrix H_{WLAN} corresponds to the one of GNSS given in equation (5.8), only the observation h_{GNSS} has to be exchanged by h_{WLAN}. The measurement model for the barometer data is exactly the same as in equation (5.9), while the measurement model for the incorporation of Δh_{PAC} is

$$\begin{aligned}\begin{bmatrix}\Delta h_{PAC}\end{bmatrix}_t &= \begin{bmatrix}0 & \Delta t\end{bmatrix}\begin{bmatrix}h\\ \dot{h}\end{bmatrix}_t \\ &= \boldsymbol{H}_{PAC}\,\boldsymbol{x}_t,\end{aligned} \tag{5.39}$$

with $\boldsymbol{H}_{PAC}$ representing the design matrix for the PAC data. Depending on the measurements availability, either one or two types of observations are integrated in the height estimation. If two observations are considered, the observation vector is either $\boldsymbol{z}_t = [h_{WLAN}, \Delta h_{baro}]^T$ or $\boldsymbol{z}_t = [h_{WLAN}, \Delta h_{PAC}]^T$. The appropriate measurement models and thus the design matrices of these two different approaches are given by

$$\begin{bmatrix}h_{WLAN}\\ \Delta h_{baro}\end{bmatrix}_t = \begin{bmatrix}\boldsymbol{H}_{WLAN}\\ \boldsymbol{H}_{baro}\end{bmatrix}\begin{bmatrix}h\\ \dot{h}\end{bmatrix}_t, \tag{5.40}$$

$$\begin{bmatrix} h_{WLAN} \\ \Delta h_{PAC} \end{bmatrix}_t = \begin{bmatrix} \boldsymbol{H}_{WLAN} \\ \boldsymbol{H}_{PAC} \end{bmatrix} \begin{bmatrix} h \\ \dot{h} \end{bmatrix}_t , \tag{5.41}$$

respectively, leading to the noiseless relation $\boldsymbol{z}_t = \boldsymbol{H}_t \boldsymbol{x}_t$ between the observations $\boldsymbol{z}_t$ and $\boldsymbol{x}_t$. Because of a measurement frequency of 1 Hz, the time interval Δt within the design matrices $\boldsymbol{H}_{baro}$ and $\boldsymbol{H}_{PAC}$, given in equations (5.9) and (5.39), can be replaced by a one.

By assuming that the measurement groups are uncorrelated, the covariance matrix $\boldsymbol{R}_t$ of the observations corresponds to a matrix of dimension $[2 \times 2]$ with the covariance of each measurement group in the main diagonal. In case of the fusion of h_{WLAN} and Δh_{baro}, the covariance matrix $\boldsymbol{R}_t$ is given by

$$\boldsymbol{R}_t = \begin{bmatrix} \boldsymbol{R}_{WLAN} & 0 \\ 0 & \boldsymbol{R}_{baro} \end{bmatrix}_t , \tag{5.42}$$

while in case of the combination of h_{WLAN} and Δh_{PAC}, $\boldsymbol{R}_t$ corresponds to

$$\boldsymbol{R}_t = \begin{bmatrix} \boldsymbol{R}_{WLAN} & 0 \\ 0 & \boldsymbol{R}_{PAC} \end{bmatrix}_t , \tag{5.43}$$

with

$$\boldsymbol{R}_{WLAN} = \sigma^2_{h_{WLAN}}, \qquad \boldsymbol{R}_{baro} = \sigma^2_{\Delta h_{baro}}, \qquad \boldsymbol{R}_{PAC} = \sigma^2_{\Delta h_{PAC}}. \tag{5.44}$$

The variance $\sigma^2_{h_{WLAN}}$ is derived within the WLAN-fingerprinting algorithm, $\sigma^2_{\Delta h_{baro}}$ is fixed with an empirical value of $(0.7\,\mathrm{m})^2$ (see section 5.1.2), while the variance for Δh_{PAC} is assumed to be $\sigma^2_{\Delta h_{PAC}} = (0.1\,\mathrm{m})^2$.

Based on the defined matrices for the time and measurement update, the Kalman filter iteratively estimates the state $\hat{\boldsymbol{x}}_t$ and the corresponding covariance $\boldsymbol{\Sigma}_t$ by applying the formulas of the three Kalman filter steps, which are given by equation equation (3.15) to (3.19). Depending on the availability of the observations, the design matrix $\boldsymbol{H}_t$ and the covariance matrix $\boldsymbol{R}_t$ of the observations have to be adapted.

Floor detection

Based on the estimated height, the current floor is determined. The floor level is only modified, if the pedestrian is located at regions, where floor transitions are possible (compare section 5.2.2) and if an upward or downward motion is observed within the PAC.

5.2.5 2D position estimation within the particle filter

While in outdoor environments the derived probability map supports the particle filter in its positioning task, in indoor environments building layouts, a wall collision filter, and the recognized activity provide spatial constraints for the estimation of the 2D coordinates φ and λ. Similar to the outdoor approach, the PDR data is used for the propagation of the particles. The WLAN data, the building layout, the wall-collision filter, and the activity are integrated in the measurement update to derive a combined importance weight ω_t^i for the resampling. Based on the particle filter framework described in section 3.2.2, in the following, details on the particle filter initialization, the propagation of the particles, the computation of the importance weights, and the resampling are given.

Initialization of the particle filter

For an easy and fast verification of the particle's location with respect to the map, besides the position and orientation, the state $\boldsymbol{x}_t^i$ comprises the floor and the room indices, leading to a state vector of following structure $\boldsymbol{x}_t^i = [\varphi, \lambda, \theta, \#_{floor}, \#_{room}]^T$. For the initialization of the coordinates φ and λ, and the heading θ, the same possibilities exist as described in section 5.1.3. The only difference is that for the initialization of the coordinates φ and λ the floor has to be known to apply the spatial constraints of the correct floor plan. By incorporating the map already in the initialization process, the particles will be distributed on the floor with regard to the environmental structures. For the initialization of the floor index, either the floor is defined, or the first estimated height of the Kalman filter is used to determine the initial floor level. A floor level definition makes sense if the pedestrian enters the building from outdoors. Therefore, the building entrances and the corresponding doors have to be mapped, similar to the doors inside of the building, within feature maps containing the edge coordinates of the doors and the corresponding floor index $\#_{floor}$.

After the initialization of the particles, which is done randomly with respect to the defined distribution, the room index $\#_{room}$ of each particle has to be identified. Based on the georeferenced occupancy grid map $\boldsymbol{m}_{\varphi,\lambda}$, $\#_{room}$ can be figured out if the floor and the position is known.

The number of particles N_s can be arbitrarily chosen within the positioning tool corresponding to the quality of the input data, the real-time capability, and the accuracy, which is demanded for a certain application. Based on several test data sets, a number N_s of about 1000 turned out to be the best for the processing in indoor environments. Investigations concerning the best suitable N_s with respect to processing effort, accuracy, and system stability are given in section 6.3.

Particle propagation

Equal to the outdoor processing, the propagation of the particles is done by sampling from the transition probability $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)$, with $\boldsymbol{u}_t$ representing the control data which corresponds to the PDR parameters v and $\Delta\theta$. The particles are propagated according to the equations (5.17) to (5.21) of the outdoor particle filter but under consideration of the current floor. After the propagation of the particle set $\boldsymbol{\chi}_t$, the room, in which the particle is located, has to be determined. The propagation provides the prior belief $\overline{bel}(\boldsymbol{x}_t)$.

Computation of the importance weight

The importance weight ω_t^i for the resampling is influenced by the WLAN fingerprinting output and the map-based information. Both data sources provide uncorrelated importance weights, which are combined according to

$$\omega_t^i = \left\langle \omega_{WLAN}{}_t^i, \omega_{map}{}_t^i \right\rangle , \tag{5.45}$$

with $\omega_{WLAN}{}_t^i$ being the normalized WLAN-based importance weight, while $\omega_{map}{}_t^i$ is the normalized map-based importance weight. The distribution of the particles within the building in combination with the importance weights ω_t^i represents the posterior belief of the state

$$bel(\boldsymbol{x}_t) = p(\boldsymbol{x}_t|\boldsymbol{z}_{WLAN}{}_t, \boldsymbol{m}_{\varphi,\lambda}) = p(\boldsymbol{x}_t|\boldsymbol{z}_{WLAN}{}_t)p(\boldsymbol{x}_t|\boldsymbol{m}_{\varphi,\lambda}), \tag{5.46}$$

which is conditioned on the WLAN observation $\boldsymbol{z}_{WLAN}{}_t$ and the map $\boldsymbol{m}_{\varphi,\lambda}$.

For the evaluation of the measurement probability $p(\boldsymbol{z}_{WLAN}{}_t|\boldsymbol{x}_t)$, which is necessary to finally get $p(\boldsymbol{x}_t|\boldsymbol{z}_{WLAN}{}_t)$ (compare equation (3.6)), it is assumed that the WLAN observations $\boldsymbol{z}_{WLAN}{}_t = [\varphi,\ \lambda]_{WLAN}^T$ follow a normal distribution corresponding to $\boldsymbol{z}_{WLAN}{}_t \sim \mathcal{N}(\boldsymbol{H}_t\boldsymbol{x}_t^i, \boldsymbol{\Sigma}_{WLAN}{}_t)$ with mean $\boldsymbol{H}_t\boldsymbol{x}_t^i$ and covariance $\boldsymbol{\Sigma}_{WLAN}{}_t$. The evaluation of

$$\begin{aligned}\omega_{WLAN}{}_t^i &= p(\boldsymbol{z}_{WLAN}{}_t|\boldsymbol{x}_t^i)\\ &= det(2\pi\boldsymbol{\Sigma}_{WLAN}{}_t)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{z}_{WLAN}{}_t - \boldsymbol{H}_t\boldsymbol{x}_t^i)^T \boldsymbol{\Sigma}_{WLAN}{}_t^{-1}(\boldsymbol{z}_{WLAN}{}_t - \boldsymbol{H}_t\boldsymbol{x}_t^i)\right).\end{aligned} \tag{5.47}$$

delivers the WLAN-based importance weights $\omega_{WLAN}{}_t^i$ with regard to $\boldsymbol{z}_{WLAN}{}_t$ and the corresponding covariance

$$\boldsymbol{\Sigma}_{WLAN} = \begin{bmatrix} \sigma^2_{\varphi_{WLAN}} & 0 \\ 0 & \sigma^2_{\lambda_{WLAN}} \end{bmatrix} . \tag{5.48}$$

The importance weight $\omega_{WLAN}{}_t^i$ represents therefore the likelihood of the current WLAN positions based on the spatial distributed particles. The design matrix $\boldsymbol{H}_t$ is needed to relate the WLAN observation $\boldsymbol{z}_{WLAN\,t}$ to the set of particles $\boldsymbol{x}_t^i$ representing the unknown state $\boldsymbol{x}_t$, which is realized by $\boldsymbol{z}_{WLAN\,t} = \boldsymbol{H}_t \boldsymbol{x}_t$. The design matrix $\boldsymbol{H}_t$ for the incorporation of the WLAN observations has following structure

$$\boldsymbol{H}_t = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \tag{5.49}$$

if $\boldsymbol{x}_t = [\varphi,\, \lambda]$.

The map-based importance weight $\omega_{map}{}_t^i$ is a combination of the probabilities which are derived after the propagation of the particles with respect to following questions:

- Is the particle in an occupied or unoccupied grid cell?
- Did the particle take a valid pathway?
- Can the current activity be performed at the particle's location?

The probabilities with respect to the questions listed above are determined either by incorporating the occupancy grid map (section 5.2.1), by verifying the pathway within the wall collision filter (section 5.2.2), or by deriving activity based environmental constraints (section 5.2.3). The verification of the grid's occupancy and the pathway delivers a path-based weight $\omega_{path}{}_t^i$, while the weight with regard to the activity is given by $\omega_{PAC}{}_t^i$. The combined map-based importance weight $\omega_{map}{}_t^i$ is computed by

$$\omega_{map}{}_t^i = \left\langle \omega_{path}{}_t^i, \omega_{PAC}{}_t^i \right\rangle. \tag{5.50}$$

By combining all sources of input data, the equal weights of the particle set $\boldsymbol{\chi}_t = \left\langle \boldsymbol{x}_t^i, \omega_t^i \right\rangle$ can be replaced by the importance weights $\omega_t^i = \left\langle \omega_{WLAN}{}_t^i, \omega_{map}{}_t^i \right\rangle$ representing the probability conditioned on the observations and the map. The posterior probability $p(\boldsymbol{x}_t | \boldsymbol{z}_t, \boldsymbol{m}_{\varphi,\lambda})$ is then given by equation (5.30).

Based on the posterior probability $p(\boldsymbol{x}_t | \boldsymbol{z}_t, \boldsymbol{m}_{\varphi,\lambda})$, the trajectory of the pedestrian is determined. In section 6.1, five different approaches are discussed as representation for the estimated trajectory, while in section 6.3 the optimal method is identified based on actual measurements.

Resampling

As described in section 5.1.3, the new states of the N_s particles are created by randomly sampling the propagated states in proportion to ω_t^i according to the systematic resampling method. After the resampling, all N_s particles get the same weight $\omega_t^i = 1/N_s$.

6 Filter performance

Before analyzing the filter outcome, performance measures for the state estimate and the uncertainty in that estimate are defined in section 6.1. Based on these performance measures, the quality of the estimation is evaluated in terms of position discrepancies and error ellipses. The corresponding investigations for the outdoor as well as for the indoor environment are presented in section 6.2 and 6.3.

6.1 Performance measures

For the investigation of the estimation results, performance measures have to be defined. There are two different classes of performance measures. One class assumes the availability of a ground truth. The other class of performance measures is based on the estimated covariance matrix. For both classes of performance measures, the state estimate has to be identified. Since within the particle filter, the state estimate is given by particles, investigations regarding a suitable parameter representing the state have been performed. Parameters based on the following methods are analyzed:

- Identification of the particle with the highest importance weight ω_t^i.
- Computation of the sample mean.
- Computation of the weighted arithmetic mean.
- Finding a state representative based on k-means clustering.

Based on a suitable state representation, discrepancies between the estimated state and defined reference points can be computed. If no ground truth is available, the covariance is the basis for a quality statement. Therefore, an appropriate technique for the determination of the covariance has to be found, see section 6.1.2.

6.1.1 Representation of the state estimate

Within this section, several possibilities for the representation of the state estimate are analyzed and compared. The basis for the definition of potential representatives is the set of weighted particles $\boldsymbol{\chi}_t = \left\langle \boldsymbol{x}_t^i, \omega_t^i \right\rangle$ resulting from the measurement update, which is equal to the PDF defined in equation (5.30). This kind of representation is equal to the particle set after the resampling having the same weights ω_t^i, which is given by equation (5.31). However, the representation of the estimation by the set of weighted particles comes along with more possibilities for the representation of the state estimate as the spatial distribution of the particles and the importance weights can be treated separately.

Particle with highest probability: The simplest solution for the determination of the position estimate $\hat{\boldsymbol{x}}_t$ is to take the particle $\boldsymbol{x}_t^i$ with the highest probability

$$\hat{\boldsymbol{x}}_t = \underset{\boldsymbol{x}_t^i}{arg\,max}\, p(\boldsymbol{x}_t|\boldsymbol{z}_t). \tag{6.1}$$

As the particles representing the previous belief $\overline{bel}(x_t)$ in equation (3.58) have equal weights ω_t^i and η_t is only a normalization factor, equation (6.1) can be simplified to

$$\hat{\boldsymbol{x}}_t = \underset{\boldsymbol{x}_t^i}{arg\,max}\, \omega_t^i, \tag{6.2}$$

which means, that $\hat{\boldsymbol{x}}_t$ corresponds to the particle $\boldsymbol{x}_t^i$ providing the highest weight ω_t^i.

Outdoors, the state with the highest probability is a particle which is closest to the GNSS solution $\boldsymbol{z}_{GNSS\,t}$ and, corresponding to the user-tailored probability map, highly probable. Within buildings, only particles, whose propagation from the previous to the current epoch is valid according to the wall collision filter, are considered for the position estimate. Then, $\hat{\boldsymbol{x}}_t$ corresponds to the particle which is closest to the WLAN-fingerprinting solution in an unoccupied area, which is highly weighted due to the current activity.

The problem of taking the highest weighted particle as best estimate is, that if all particles have the same weight with respect to the map, the wall collision filter, and the current activity, the result is the particle closest to the absolute position solution provided by GNSS or WLAN. Consequently, the best estimate does not consider the collectivity of particles and strongly depends on the GNSS and WLAN position accuracy.

Sample mean: The sample mean of a set of particles $\boldsymbol{\chi}_t = \left\langle \boldsymbol{x}_t^i, \omega_t^i \right\rangle$ is given by the arithmetic mean of the N_s samples $\left\langle \boldsymbol{x}_t^i \right\rangle$

$$\hat{\boldsymbol{x}}_t = \frac{1}{N_s} \sum_{i=1}^{N_s} \boldsymbol{x}_t^i. \tag{6.3}$$

The sample mean considers only the distribution of the particles uncoupled from the importance weights ω_t^i and is therefore strongly influenced by the incorporated dynamic model and the transition data $\boldsymbol{u}_t$. The measurement update at time t has consequently no direct impact on the best estimate $\hat{\boldsymbol{x}}_t$ at the current epoch t. Only $\hat{\boldsymbol{x}}_{t+1}$ of the consecutive epoch benefits from ω_t^i through the particle resampling.

Weighted arithmetic mean: The weighted arithmetic mean considers the spatial distribution of the particles, which is given by $\boldsymbol{x}_t^i$, and the importance weights of the current epoch. The weighted arithmetic mean of $\boldsymbol{\chi}_t^i = \langle \boldsymbol{x}_t^i, \omega_t^i \rangle$ is given by

$$\hat{\boldsymbol{x}}_t = \frac{1}{\sum_{i=1}^{N_s} \omega_t^i} \sum_{i=1}^{N_s} \omega_t^i \boldsymbol{x}_t^i. \tag{6.4}$$

By incorporating both, the samples $\boldsymbol{x}_t^i$ and the appropriate importance weights ω_t^i, all kinds of input data are taken into account. Consequently, the weighted arithmetic mean is a more suitable representation of the location estimate compared to the sample mean.

k**-means algorithm:** The *k-means algorithm* is also known as *k-means clustering.* As stated in Hartigan (1975), the k-means algorithm partitions the set of N_s samples $\langle \boldsymbol{x}_t^i \rangle$ into a predefined number of groups/clusters, whereby the amount k of clusters has to be smaller than N_s (Figueiras and Frattasi, 2010). Each element of $\langle \boldsymbol{x}_t^i \rangle$ is assigned to the cluster, for which the Euclidean distance is a minimum:

$$\boldsymbol{x}_t^i \in C_j = \underset{C_j}{arg\,min} \left\| \boldsymbol{x}_t^i - \boldsymbol{\mu}_j \right\|, \tag{6.5}$$

where $\boldsymbol{\mu}_j$ is the mean value of the cluster C_j. The assignment of the samples to the clusters is done in an iterative fashion, where in a first step the center of the clusters are initialized randomly. Then, all samples are assigned to a cluster based on the Euclidean distance to the cluster centers. Then, the initialized center coordinates of the clusters are replaced by the mean values of the clusters

$$\boldsymbol{\mu}_j^{(l)} = \frac{1}{N_{C_j^{(l)}}} \sum_{\boldsymbol{x}_t^i \in C_j^{(l)}} \boldsymbol{x}_t^i, \tag{6.6}$$

whereby $N_{C_j^{(l)}}$ represents the number of samples in the cluster $C_j^{(l)}$ at iteration l. After that, the updated mean values $\boldsymbol{\mu}_j^{(l)}$ act as basis for a reassignment of the samples:

$$\boldsymbol{x}_t^i \in C_j^{(l)} = \underset{C_j^{(l)}}{arg\,min} \left\| \boldsymbol{x}_t^i - \boldsymbol{\mu}_j^{(l)} \right\|. \tag{6.7}$$

Equation (6.6) and (6.7) are repeated until the assignment of particles converges, which means, once the assignment of samples does not change in two consecutive steps, the algorithm terminates (Figueiras and Frattasi, 2010). After having executed the clustering, there are two different possibilities to estimate the pedestrian's current location based on the k-means algorithm. One possibility is to take the mean value of the cluster with the highest concentration of particles, which is defined by

$$\hat{\boldsymbol{x}}_t = \underset{\boldsymbol{\mu}_j}{arg\,max}\, N_{C_j}, \tag{6.8}$$

with N_{C_j} being the number of particles within the cluster C_j. The second method is based on the cluster with highest accumulated weight. The location estimate is therefore given by

$$\hat{\boldsymbol{x}}_t = \underset{\boldsymbol{\mu}_j}{arg\,max} \sum_{\boldsymbol{x}_t^i \in C_j} \omega_t^i, \tag{6.9}$$

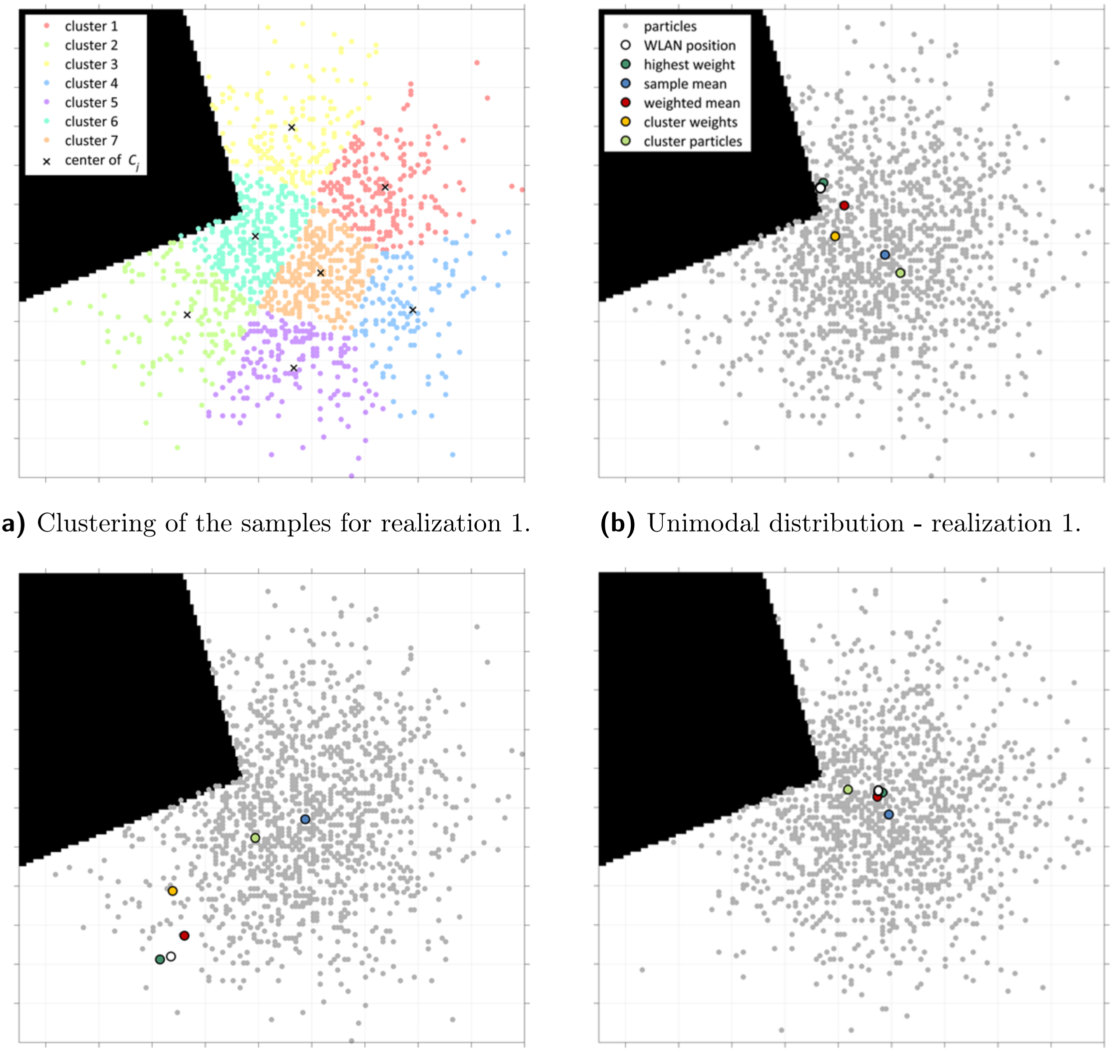
which takes the mean value of the cluster with highest accumulated weight. The sum of the importance weights ω_t^i of all particles belonging to the cluster C_j is consequently the decisive factor for choosing $\hat{\boldsymbol{x}}_t$.

Comparison of the different representations for the position estimation: In general, the choice, which method is most suitable for representing the state estimate, depends on the application. The dynamic and the measurement model, the involved sensor data and uncertainties, as well as the present environmental structures are all factors which have an impact on the goodness of fitting the true location. Therefore, two different scenarios, which represent diverse state transition probabilities, are investigated.

Figure 6.1 shows three realizations of a particle filter scenario in an indoor environment, in which WLAN delivers different 3D coordinates. The visualized particles are drawn after the prediction and are already corrected with respect to the building structures. After the propagation, each of the particles has the same weight. The importance weights ω_t^i are updated with according to the WLAN measurements. As in this scenario the building structures are already considered in the propagation, ω_t^i is only influenced by the WLAN solution. As a consequence, the resulting distribution is unimodal.

In Figure 6.1a, the clustering of the particles is exemplary shown for the first realization, whereby $k = 7$ is chosen for the amount of clusters within the k-means algorithm. The five different state representatives for the three realizations are illustrated in Figure 6.1b to Figure 6.1d. The particle with the highest importance weight is plotted in dark green, the sample mean in blue, and the weighted mean in red. The center of the cluster with the highest accumulated weight is drawn in yellow, while the center of the cluster with the highest particle concentration is light green. The distribution of the particles is the same in all three realizations, whereas the WLAN position varies in each of the figures and is displayed as black circle with a white filling.

The three different realizations in Figure 6.1 demonstrate that the suitability of the approaches strongly depends on the present scenario. If the particles' spatial distribution after the prediction coincides with the measurement likelihood (Figure 6.1d), which means that the WLAN measurement is located approximately at the center of the particles, the particle with the highest weight and the weighted mean are most qualified for the representation of the state $\boldsymbol{x}_t$. In contrast, in Figure 6.1b and 6.1c, where the PDFs of the dynamic and the measurement model are not conform, the WLAN measurement differs from the mean of the spatially distributed particles. Consequently, the choice of the state representation depends on the accuracies of the models.



(a) Clustering of the samples for realization 1.

(b) Unimodal distribution - realization 1.

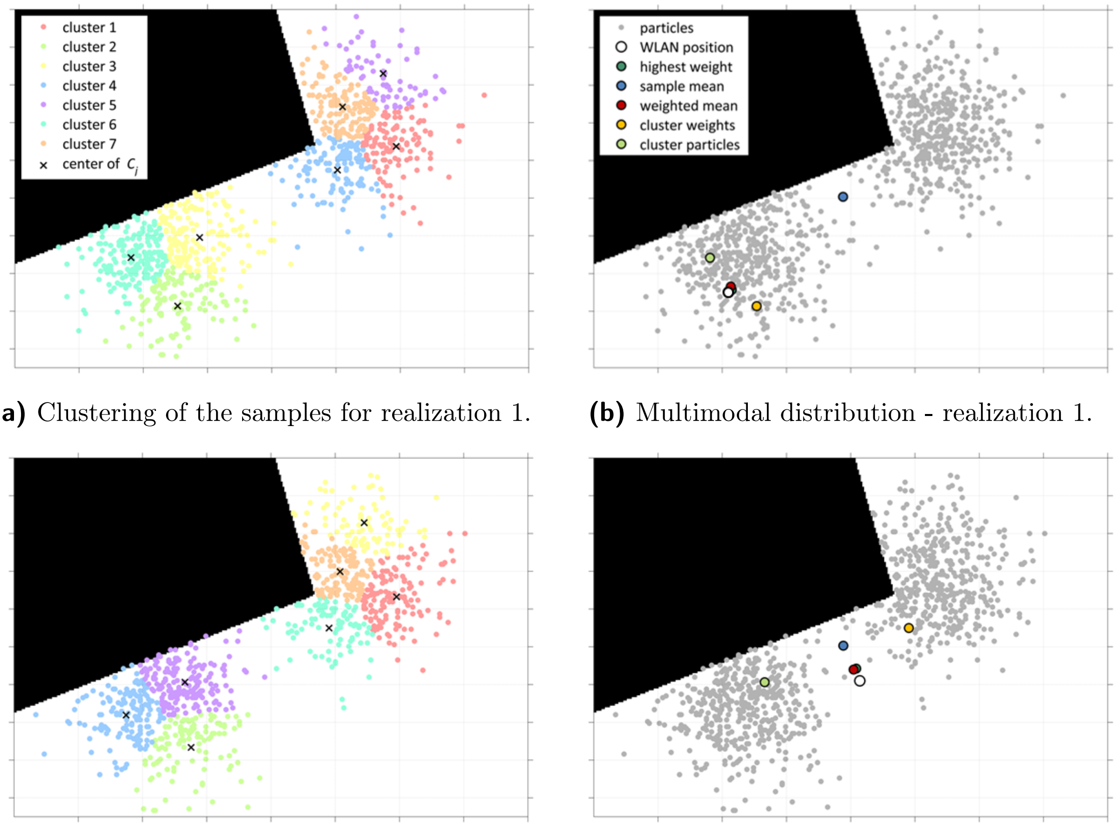
(c) Unimodal distribution - realization 2.

(d) Unimodal distribution - realization 3.

Figure 6.1: Comparison of the diverse state representations based on three different WLAN measurements, which are visualized in (b) to (d). The clustering of the samples is exemplarily shown for realization 1 in (a).

In all of the three realizations, the sample mean is the same, as it does not consider the current importance weights. Therefore, the sample mean is not qualified as state representative. None of the other approaches can be excluded, since the actual position is not defined. The three different realizations only demonstrate the principle concept of the diverse methods. As a consequence, the suitability of the particle with the highest weight, the arithmetic mean, and the two different k-means clustering methods as state representative are investigated with real measurement data in section 6.2 and 6.3.

Figure 6.2 shows another scenario based on a multimodal distribution. In statistics, a multimodal distribution corresponds to a probability distribution with two or more maxima (Ross, 2000). After the propagation of the particles, the particles are split into two separated areas exhibiting a multimodal distribution. Figure 6.2b illustrates a case, where the WLAN position solution coincides with one of the particle areas. In contrast, in Figure 6.2d the WLAN position solution lies in between the two separated particle areas. Now, the representatives suitability depends on the uncertainty of both involved models. Again, the sample mean



(a) Clustering of the samples for realization 1.

(b) Multimodal distribution - realization 1.

(c) Clustering of the samples for realization 2.

(d) Multimodal distribution - realization 2.

Figure 6.2: Comparison of the different representations for the position estimation in case of a multimodal distribution.

can be excluded, but, as Figure 6.2 demonstrates, in case of multimodal distributions, the k-means clustering methods have to be favored over the arithmetic mean and the highest weighted particle if the measurement does not provide information about which of the two areas is the right one.

6.1.2 Representation of the estimate uncertainty

If a ground truth of the trajectory is given, position discrepancies between the estimated state $\hat{\boldsymbol{x}}_t$ and the true position of the user can be determined. If no reference trajectory is available, the covariance $\Sigma_{\hat{\boldsymbol{x}}_t}$ of the state $\hat{\boldsymbol{x}}_t$ is the basis for analyzing the filter performance. While the position discrepancies indicate the absolute deviation of the estimated trajectory to a ground truth, the covariance only gives information about the precision of the state estimation based on the measurement uncertainties and the involved models.

Position residuals: As no reference system delivers the ground truth of the trajectory, reference points have been defined within the test environment. The coordinates of these reference points are known and therefore allow the determination of the differences between the estimated and the reference trajectory if the pedestrian passes one of the reference points. These differences are called position residuals within the thesis. For a fast identification of the reference points along the trajectory, e.g., indoors, markers were placed on the floor. Every time, the pedestrian crosses one of the markers, the ID of the reference point and the corresponding passing time are recorded. This is realized by a smartphone application which synchronizes the recorded time with the UTC time and exports the time stamps into a text file. This smartphone application is operated on a second smartphone by an observer recording the crossing of the markers.

The reference points are given by a set of horizontal geographic coordinates, which are defined by $\boldsymbol{x}^k_{ref} = \left[\varphi^k_{ref},\, \lambda^k_{ref}\right]^T$. Based on the reference points $\boldsymbol{x}^k_{ref}$ and the corresponding passing times t_{ref}, the horizontal position residuals between the states $\hat{\boldsymbol{x}}(t_{ref}) = \left[\hat{\varphi}(t_{ref}),\, \hat{\lambda}(t_{ref})\right]^T$ at t_{ref} and the reference points $\boldsymbol{x}^k_{ref}$ can be computed by

$$\Delta x^k_{horiz}(t_{ref}) = \sqrt{\Delta x^k_n(t_{ref})^2 + \Delta x^k_e(t_{ref})^2}, \tag{6.10}$$

with

$$\begin{aligned}\Delta x^k_n(t_{ref}) &= \left|\left(\hat{\varphi}(t_{ref}) - \varphi^k_{ref}(t_{ref})\right)\; R\right| \\ \Delta x^k_e(t_{ref}) &= \left|\left(\hat{\lambda}(t_{ref}) - \lambda^k_{ref}(t_{ref})\right)\; R\, \cos\varphi_0\right|,\end{aligned}$$

where $\Delta x^k_n(t_{ref})$ and $\Delta x^k_e(t_{ref})$ represent the position residuals in north and east direction, R is the radius of the Earth and φ_0 the mean latitude of the test ground. However, $\hat{\boldsymbol{x}}(t)$ is provided with an update rate of 1 Hz and has to be related to the time stamps t_{ref}. Therefore, an interpolation of the estimated states $\hat{\boldsymbol{x}}(t)$ with respect to t_{ref} is required to get $\hat{\varphi}(t_{ref})$ and $\hat{\lambda}(t_{ref})$. For the interpolation, the pedestrian's motion is assumed to be constant between two consecutive epochs.

Since the triggering of the time stamps is not done automatically, a bias can be observed in the passing times. The magnitude of the bias is strongly influenced by the reaction of the observer. Furthermore, the fact that the smartphone is carried handheld by the pedestrian, but the markers are placed on the floor, impedes a precise triggering of the time stamp. As most of the temporal bias is expected in the along track component leading to position residuals with a higher magnitude in the direction of the path, the position residuals are differentiated into along track and across track components, Δx^k_{along} and Δx^k_{across}. $\Delta x^k_{along}(t_{ref})$ and $\Delta x^k_{across}(t_{ref})$ can be determined through a rotation of $\Delta x^k_n(t_{ref})$ and $\Delta x^k_e(t_{ref})$ by the angle between the abscissa of the coordinate system and the along track direction of the reference path. Consequently, the text file containing the reference point IDs and the corresponding time stamps t_{ref} has to be expanded by a reference heading θ_{ref}. $\Delta x^k_{along}(t_{ref})$ and $\Delta x^k_{across}(t_{ref})$ can then be computed by the following rotation

$$\begin{bmatrix} \Delta x^k_{along}(t_{ref}) \\ \Delta x^k_{across}(t_{ref}) \end{bmatrix} = \begin{bmatrix} \cos\theta_{ref} & \sin\theta_{ref} \\ -\sin\theta_{ref} & \cos\theta_{ref} \end{bmatrix} \begin{bmatrix} \Delta x^k_n(t_{ref}) \\ \Delta x^k_e(t_{ref}) \end{bmatrix}. \tag{6.11}$$

Based on $\Delta x^k_{along}(t_{ref})$ and $\Delta x^k_{across}(t_{ref})$, the filter performance can be investigated in terms of along and across track errors.

Sample covariance: The covariance $\Sigma_{\hat{\boldsymbol{x}}_t}$ of the estimated state $\hat{\boldsymbol{x}}_t$ represents the estimate uncertainty, which is caused by the sensor- and model-related errors. Only the input data of the filter (see Figure 4.1 as well as section 5.1.3 and 5.2.5) influence $\Sigma_{\hat{\boldsymbol{x}}_t}$. Therefore, the covariance allows no interpretation about the absolute accuracy of the state estimation with respect to a ground truth. In general, the computation of the sample covariance is based on samples representing the probability density distribution of the estimate. In case of the particle filter estimation, the samples correspond to the particles.

Based on the estimated state $\hat{\boldsymbol{x}}_t$ and the particles $\boldsymbol{x}^i_t$, the sample covariance of the state $\hat{\boldsymbol{x}}_t$ can be determined by

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{x}}_t} = \frac{1}{N_s - 1} \sum_{i=1}^{N_s} (\boldsymbol{x}^i_t - \hat{\boldsymbol{x}}_t)(\boldsymbol{x}^i_t - \hat{\boldsymbol{x}}_t)^T, \tag{6.12}$$

with N_s representing the number of particles (Bronstein and Semendjajew, 1981; Ross, 2000). Since the covariance is computed before the resampling of the particles, the measurement probability $p(\boldsymbol{x}_t|\boldsymbol{z}_{t-1}, m)$ has no influence on $\boldsymbol{\Sigma}_{\hat{\boldsymbol{x}}_t}$. $\boldsymbol{\Sigma}_{\hat{\boldsymbol{x}}_t}$ is therefore based on the previous belief $\overline{bel}(\boldsymbol{x}_t)$ incorporating the state transition probability $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)$ and the posterior belief $bel(\boldsymbol{x}_{t-1})$ of the epoch $t-1$.

Sample covariance based on weighted samples: Besides the spatial distribution of the particles, this method also considers the importance weights ω_t^i. The sample covariance based on weighted samples represents the covariance of the posterior belief $bel(\boldsymbol{x}_t)$. Therefore, the state transition probability $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)$, the measurement probability $p(\boldsymbol{x}_t|\boldsymbol{z}_{t-1}, m)$, and the posterior belief $bel(\boldsymbol{x}_{t-1})$ influence the covariance estimation. The sample covariance incorporating the importance weights ω_t^i is defined by

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{x}}_t} = \frac{\sum_{i=1}^{N_s} \omega_t^i}{\left(\sum_{i=1}^{N_s} \omega_t^i\right)^2 - \sum_{i=1}^{N_s} \omega_t^{i^2}} \sum_{i=1}^{N_s} \omega_t^i (\boldsymbol{x}_t^i - \hat{\boldsymbol{x}}_t)(\boldsymbol{x}_t^i - \hat{\boldsymbol{x}}_t)^T, \tag{6.13}$$

whereby the Pearson weighted correlation matrix defined in Pozzi et al. (2012) is the basis of the sample covariance matrix in equation (6.13). Since the sample covariance based on weighted samples takes the prediction as well as the measurement model into account, this covariance representation is more suitable than the sample covariance.

Error Ellipse: Based on the covariance matrix $\Sigma_{\hat{\boldsymbol{x}}_t}$, the error ellipse at each epoch t can be computed. However, by representing the estimate uncertainty by an error ellipse, a normal distribution is assumed and therefore, the error ellipse represents a Gaussian approximation of the actual particle distribution. The size of an error ellipse is given by its semi-major axis a and semi-minor axis b, while α corresponds to the orientation of the ellipse with regard to the defined coordinate system. Details on the derivation of the parameters of the error ellipse can be found in Haug (2012). The semi-major and semi-minor axis a and b of the ellipse are defined by

$$a = \sqrt{\frac{\Sigma_{\hat{\boldsymbol{x}}_t(1,1)} + \Sigma_{\hat{\boldsymbol{x}}_t(2,2)} + k}{2}}, \tag{6.14}$$

$$b = \sqrt{\frac{\Sigma_{\hat{\boldsymbol{x}}_t(1,1)} + \Sigma_{\hat{\boldsymbol{x}}_t(2,2)} - k}{2}}, \tag{6.15}$$

with

$$k = \sqrt{(\Sigma_{\hat{\boldsymbol{x}}_t(1,1)} - \Sigma_{\hat{\boldsymbol{x}}_t(2,2)})^2 + 4\Sigma_{\hat{\boldsymbol{x}}_t(1,2)}^2}. \tag{6.16}$$

In equation (6.14), (6.15), and (6.16), $\Sigma_{\hat{\boldsymbol{x}}_t(1,1)}$, $\Sigma_{\hat{\boldsymbol{x}}_t(2,2)}$, and $\Sigma_{\hat{\boldsymbol{x}}_t(1,2)}$ are the elements of the covariance matrix $\Sigma_{\hat{\boldsymbol{x}}_t}$ of the estimated state $\hat{\boldsymbol{x}}_t$:

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{x}}_t} = \begin{bmatrix} \Sigma_{\hat{\boldsymbol{x}}_t(1,1)} & \Sigma_{\hat{\boldsymbol{x}}_t(1,2)} \\ \Sigma_{\hat{\boldsymbol{x}}_t(2,1)} & \Sigma_{\hat{\boldsymbol{x}}_t(2,2)} \end{bmatrix}. \tag{6.17}$$

The orientation of the error ellipse related to the defined coordinate system is given by

$$\alpha = \frac{1}{2} \arctan \frac{2\,\Sigma_{\hat{\boldsymbol{x}}_t\,(1,2)}}{\Sigma_{\hat{\boldsymbol{x}}_t\,(1,1)} - \Sigma_{\hat{\boldsymbol{x}}_t\,(2,2)}}, \tag{6.18}$$

with α being the angle between the abscissa of the coordinate system and the semi-major axis a of the error ellipse.

Referred to Hofmann-Wellenhof et al. (2003) and Hoepcke (1980), the probability, that the state $\boldsymbol{x}_t$ is located within the area covered by the error ellipse, is 39% if a normal distribution is assumed. As the error ellipse is based on a particle set $\boldsymbol{\chi}_t^i = \langle \boldsymbol{x}_t^i, \omega_t^i \rangle$ which does not correspond to a Gaussian distribution, it cannot be expected that the error ellipse covers 39% of the corresponding PDF. However, the error ellipse enables a visual interpretation of the size and orientation of the uncertainty of the state estimation.

6.2 Performance in outdoor environments

In the following, the performance of the outdoor positioning is analyzed. Since the Kalman filter for the height estimation is similar outdoors and indoors, the results are only given for the indoor environment in section 6.3.1. Indoors, the height estimation is responsible for providing the correct floor map and is therefore more important than outdoors. Thus, the main focus of the investigations is set on the particle filter performance and is reflected by the sub-categorization of this section:

- Performance investigations regarding the number of particles.
- Investigation regarding the state estimate representation.
- Performance gain due to the user-tailored probability map.
- Positioning performance of one selected trajectory.

In section 6.2.1, investigations concerning the optimal number of particles N_s with respect to the required computation time and the resulting position residuals are presented. Since the particle filter estimation delivers a weighted set of particles, in section 6.2.2, the estimated state has to be identified. Therefore, the various possibilities for the state representation defined in section 6.1 are evaluated especially for the application of the outdoor RBPF. In the consecutive investigations, the identified parameter represents the estimated trajectory for the computation of the position residuals. In section 6.2.3, the performance gain due to the integration of the user-tailored probability map is demonstrated. Finally, in section 6.2.4 one measurement data set is chosen to exemplarily show one estimated trajectory and the corresponding evaluation results. The last section 6.2.5 treats the handling of unexpected walking behavior.

For the different investigations, the raw sensor data of several walked trajectories have been recorded and evaluated in a post-processing mode. The post-processing enables a statistical evaluation of the achieved position accuracy. Figure 6.3 shows besides the reference trajectory, along which the pedestrian walked within the test environment, the defined reference points representing the ground truth. The trajectory follows hereby the reference points with increasing numbers. For the investigations within the thesis, different trajectories have been analyzed to investigate the GNSS availability in the whole test environment. However, for the performance investigations presented in this document, the same trajectory was walked four times at different times of the day. Thus, a changing GNSS constellation is given for the evaluations. By processing four different walks based on different observation data, the reproducibility of the results can be investigated. Furthermore, statistics over different walks can be provided. The trajectory includes areas with low to high GNSS signal obstructions. The duration of each of the walks is about 13 minutes.

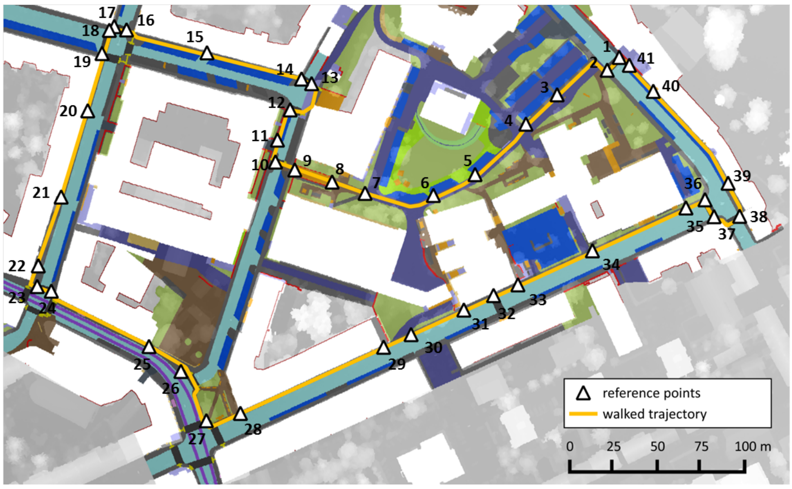
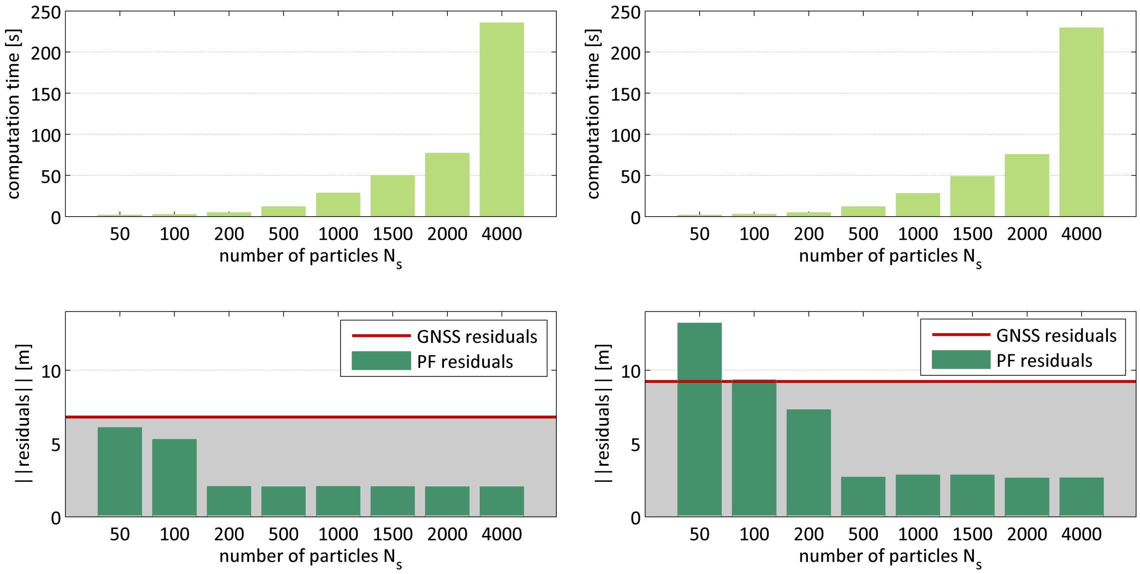


Figure 6.3: Walked trajectory within the test environment at the campus of TU Graz.

In general, the processing is done as described in section 5.1. However, within this section, besides the user-tailored probability map, also constraints derived from a common outdoor map are integrated within the particle filter to investigate the performance gain due to the user-tailored probability map.

6.2.1 Performance investigations regarding the number of particles

In Figure 6.4, the mean computation times and the mean horizontal position residuals $\mu_{\Delta x_{horiz}}$ are shown for two different walks. Each of the two walks is evaluated ten times with a varying number of particles $N_s = \{50, 100, 200, 500, 1000, 1500, 2000, 4000\}$. The particle filter processing is again repeated ten times, as the outcome of the particle filter is not reproducible due to the random initialization of the particles. The illustrated computation times and the residuals represent the mean of the ten evaluations for different N_s. In Figure 6.4, the residuals illustrated in the bottom row correspond to the horizontal position residuals defined in equation (6.10). The mean computation times given in the top row of Figure 6.4 are normalized to a trajectory length of 1000 s to make the results comparable.



(a) Walk 1.

(b) Walk 2.

Figure 6.4: The mean computation times and the mean horizontal position residuals vs. N_s for two different walks.

Figure 6.4 shows that the mean computation times are very similar for both walks. In both cases, the computation times are exponentially increasing with the growth of N_s. In Table 6.1, the mean computation times for the evaluation of 1000 epochs are also listed as numerical values together with the corresponding standard deviations σ_t. The little standard deviations indicate the similarity of the ten PF evaluations, while the mean computation times demonstrate that the processing can be done in real-time. The processing with 500 particles already comes along with position residuals which are comparable to the results with larger N_s and requires as little as 11.87 s and 11.78 s. Even if 1000 particles are chosen, the computation time for a walk of 1000 s is less than 29 s, which means that the processing of one epoch needs only 0.03 s.

As Figure 6.4 demonstrates, the mean values for $\mu_{\Delta x_{horiz}}$ are strongly decreasing with an increasing N_s from $N_s = 50$ to $N_s = 500$. After $N_s = 500$, no significant improvement can be observed. The mean of the horizontal position residuals converges to about 2.04 m and 2.64 m. In Table 6.2, the numerical values for $\mu_{\Delta x_{horiz}}$ and the corresponding standard

Table 6.1: Mean computation times and the corresponding standard deviations for two different walks with a length of 1000 s with varying N_s.

	walk 1		**walk 2**	
N_s	time [s]	σ_t [s]	time [s]	σ_t [s]
50	1.57	0.04	1.59	0.03
100	2.60	0.06	2.62	0.09
200	4.63	0.03	4.76	0.22
500	11.87	0.14	11.78	0.14
1000	28.35	0.10	27.92	0.08
1500	49.83	0.17	48.88	0.13
2000	76.97	0.30	75.40	0.13
4000	235.45	1.27	229.41	0.44

deviations $\sigma_{\Delta x_{horiz}}$ are given for different N_s. For reasons of comparability, the last row of Table 6.2 shows $\mu_{\Delta x_{horiz}}$ and $\sigma_{\Delta x_{horiz}}$ for the GNSS solution.

Figure 6.4 and Table 6.2 show that the increase of N_s from 100 to 500 particles comes along with a strong improvement of the position accuracy. Walk 1 already provides position residuals of 2.07 m with $N_s = 200$, which means that the doubling of N_s from 100 to 200 improves the mean horizontal position residual by about 60%. In case of walk 2, the gain from 100 to 200 particles leads to an improvement of only 22%. The reason therefor is the worse GNSS performance. However, the evaluation with 500 particles reduces the residuals significantly, providing an improvement of more than 70% compared to the scenario with

Table 6.2: Mean horizontal position residuals and the corresponding standard deviations for two different walks with varying N_s.

	walk 1		**walk 2**	
N_s	$\mu_{\Delta x_{horiz}}$ [m]	$\sigma_{\Delta x_{horiz}}$ [m]	$\mu_{\Delta x_{horiz}}$ [m]	$\sigma_{\Delta x_{horiz}}$ [m]
50	6.09	8.08	13.21	16.34
100	5.27	8.46	9.34	11.54
200	2.07	0.17	7.29	12.84
500	2.05	0.08	2.70	0.25
1000	2.06	0.09	2.84	0.31
1500	2.06	0.05	2.84	0.43
2000	2.04	0.08	2.64	0.26
4000	2.04	0.07	2.64	0.37
GNSS	6.80	3.79	9.23	4.91

$N_s = 100$. A further increase of N_s provides only very little or no quality improvements. Consequently, for the position estimation in outdoor environments a number of $N_s = 500$ is recommended.

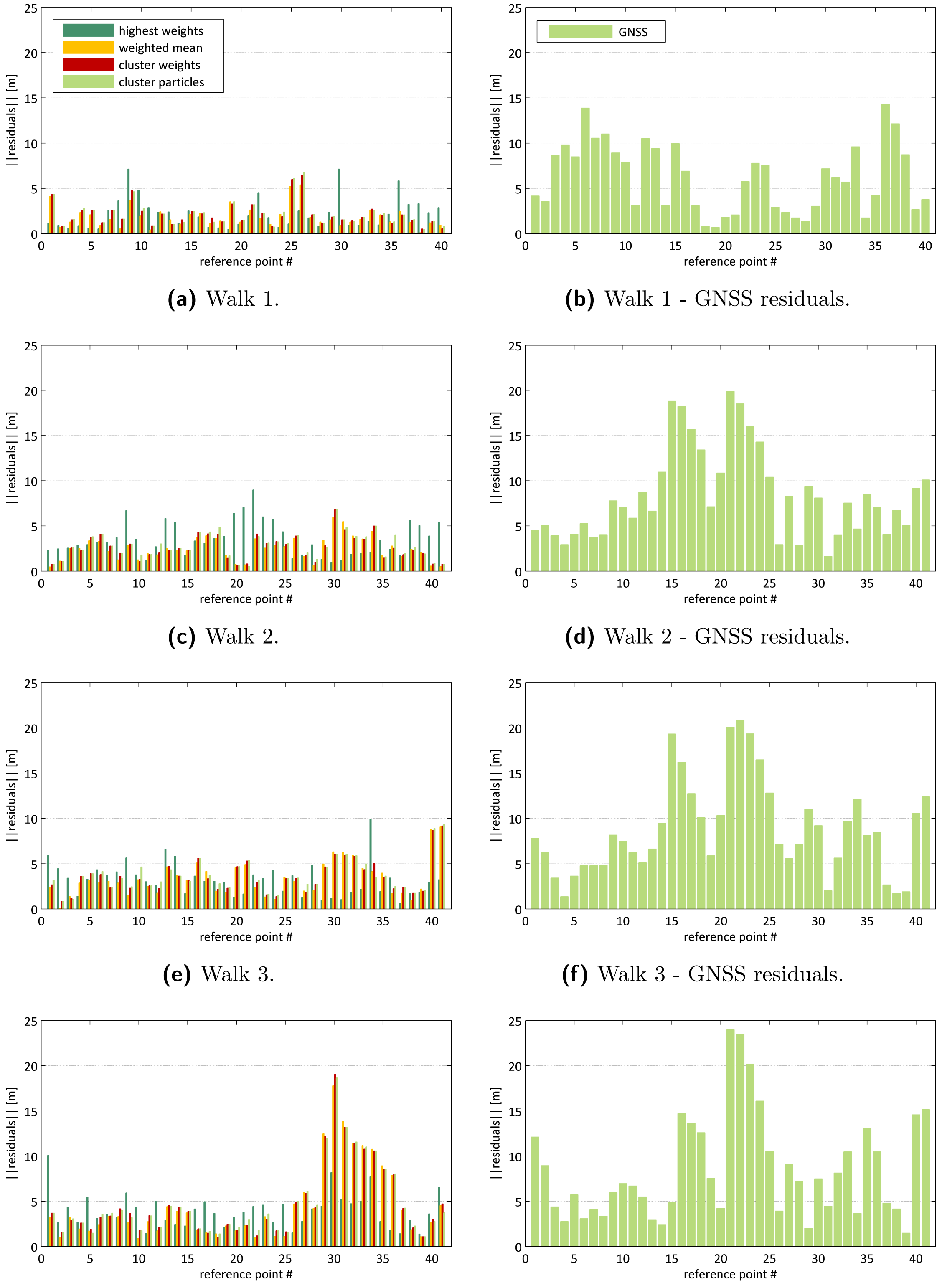
6.2.2 Investigation regarding the state estimate representation

Within the particle filter, the state estimation is represented by a particle set $\boldsymbol{\chi}_t = \langle \boldsymbol{x}_t^i \rangle$. The spatial distribution of the particles corresponds to the PDF of the estimated state. For the computation of position residuals and the visualization of error ellipses, a suitable state representative for the PF estimation has to be identified. Within this section, four different quantities are investigated with regard to the application of the RBPF in outdoor environments. As mentioned in section 6.1, the highest weighted particle, the weighted mean, and two different realizations of the k-means clustering (the mean of the cluster with the highest accumulated weight and the mean of the cluster with the highest particle concentration) are hereby analyzed in detail. The PF processing for this investigation is done with 2000 particles, as the focus is set on the identification of the best state representative and therefore, incalculable effects due to a too small N_s want to be avoided. As Table 6.2 shows, from a number of 2000 particles best results in terms of position residuals can be assumed.

For the identification of the optimal state representative, the position residuals are determined for each of the four different realizations. For this purpose, all walks are evaluated ten times, delivering representative mean values for the horizontal, along track, and across track position residuals. In Figure 6.5, the mean values of the horizontal position residuals averaged over ten evaluations are visualized for each reference point. For reasons of comparability, besides the different realizations of the state estimate, the GNSS residuals are visualized. While the GNSS position solutions of walk 1 show horizontal position residuals with magnitudes which are less than 14 m throughout the whole walk, the GNSS trajectories of walk 2, 3, and 4 have residuals higher than 20 m in the middle of the walks. In contrast, those three walks perform better than walk 1 in the first ten reference points.

For all of the four walks, an improvement due to the filtering can be observed. The horizontal position residuals Δx_{horiz} for the weighted mean, the center of the cluster with the highest accumulated weight, and the center of the cluster with most of the particles lead to comparable results. However, in most cases the weighted mean provides the residuals with lowest amplitudes. Furthermore, it can be observed, that in general a bad GNSS solution affects the result some epochs later.

The particle with the highest weight exhibits a behavior which strongly differs from the other three realizations, since it is most affected by the GNSS solution. If the propagated particles



(a) Walk 1.

(b) Walk 1 - GNSS residuals.

(c) Walk 2.

(d) Walk 2 - GNSS residuals.

(e) Walk 3.

(f) Walk 3 - GNSS residuals.

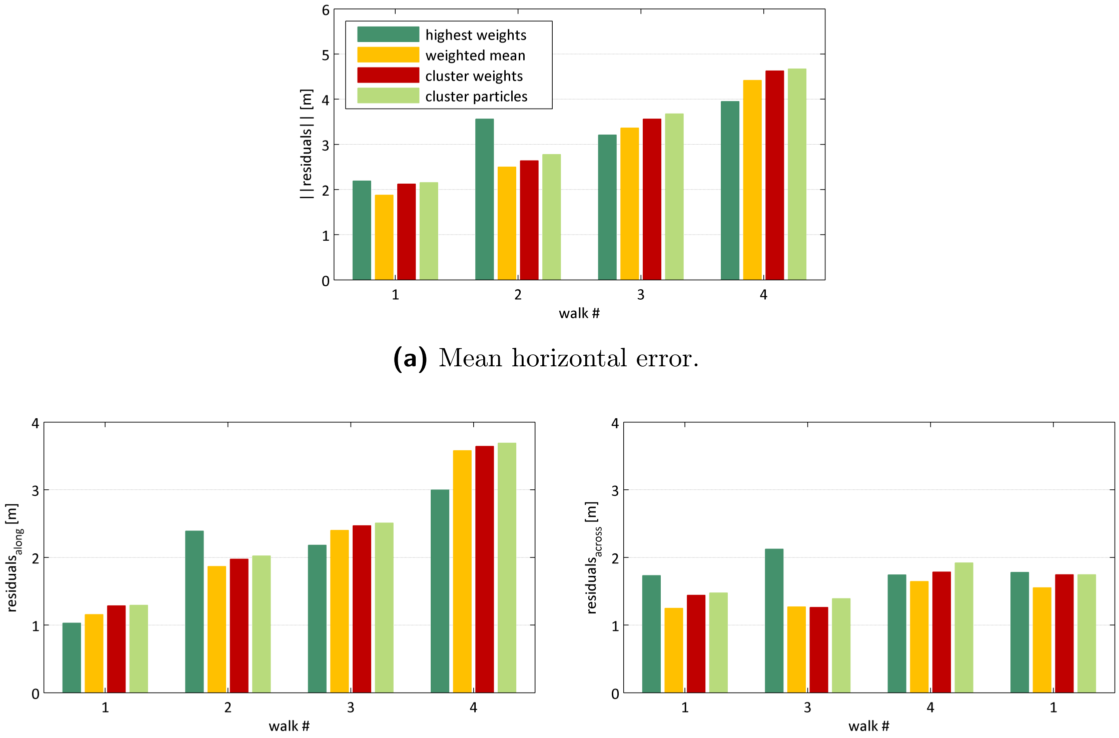
(g) Walk 4.

(h) Walk 4 - GNSS residuals.

Figure 6.5: Error in the horizontal position resulting from four processing strategies applied to four different walks.

are close to the GNSS solution, the particle with the highest weight corresponds to the GNSS position in an area which is highly probable due to $\omega_{map}{}_t^i$. This means, that in case of a bad GNSS position, the state estimate represented by the particle with the highest weighted is influenced most if the probability map does not constrain the solution.

Figure 6.6a shows the mean horizontal position residuals over all walks, while Figure 6.6b and 6.6c illustrate the corresponding mean values in along and across track direction. The mean values in Figure 6.6a indicate that the weighted mean provides residuals with the lowest amplitudes. The separation into along and across track components demonstrates, that in three of four cases the particle with the highest weight has the lowest along track residuals (Figure 6.6b), while the weighted mean works best for the across track direction (Figure 6.6c). The comparison of the different clustering methods reveals that the trajectories based on the cluster with the highest accumulated weight fits the ground truth better. Anyway, both clustering approaches provide residuals of higher amplitudes than the weighted-mean, although the time series of the residuals show comparable characteristics.



(a) Mean horizontal error.

(b) Mean along track error.

(c) Mean across track error.

Figure 6.6: Mean error resulting from four processing strategies applied to four different walks.

Finally, the weighted mean is chosen as representative parameter for the estimated trajectory due to following reasons:

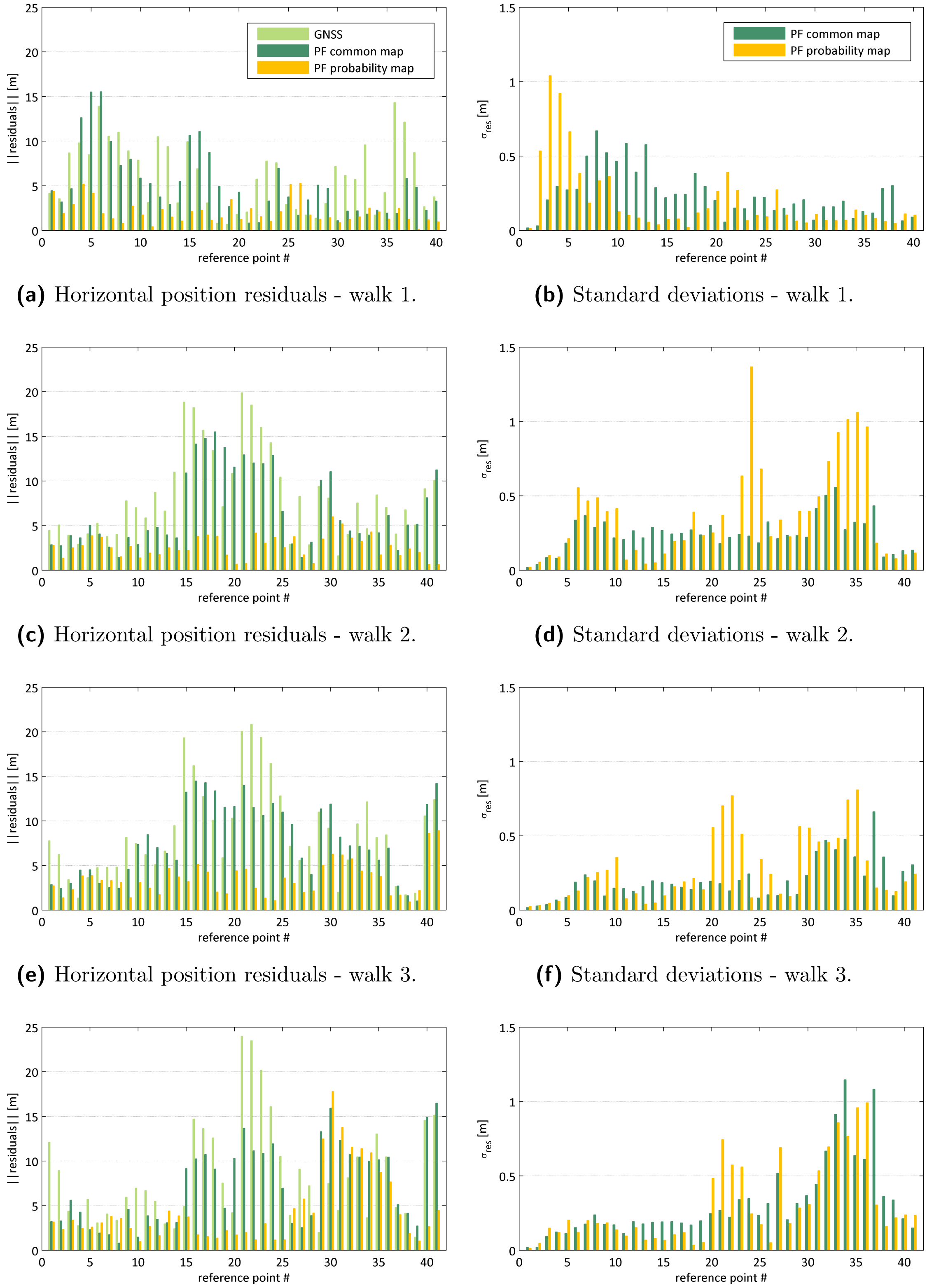
- The along track residuals are more affected by an incorrect time stamp recording, since the reference points are passed in the along track direction of the path.
- Low residuals in the across track component guarantee a positioning at the correct pavement, which is very important for a reliable obstacle warning and an effective navigation. The along track position accuracy mostly influences the timely announcement of guidance instructions. The maximum difference in the averaged along track position residuals can be observed for walk 4 and is about 70 cm (minimum is 3.00 m, maximum is 3.69 cm). Anyway, the guidance instructions have to be given in advance, and a difference of about 70 cm only induces a delay of about 0.6 s if the pedestrian walks with a speed of about 4 km/h.

6.2.3 Performance gain due to the user-tailored probability map

Within this section, the performance gain due to the integration of the user-tailored probability map is analyzed. Therefore, the particle filter is processed ten times with the user-tailored probability map, and another ten times, with a common outdoor map which represents the presence of buildings and obstacles. The estimated trajectory is represented by the weighted mean of the particle set $\boldsymbol{\chi}_t = \left\langle \boldsymbol{x}_t^i, \omega_t^i \right\rangle$, whereby N_s is chosen to be 2000.

In Figure 6.7, for each of the four different walks, the mean horizontal position residuals averaged over the ten evaluations (left hand side) and the standard deviations (right hand side), which represent the variability of the ten evaluations, are shown. Besides the residuals of the two different PF evaluations (user-tailored map vs. common map), the GNSS residuals are visualized to demonstrate the improvement due to the PF processing.

Figure 6.7a, 6.7c, 6.7e, and 6.7g show that the integration of the user-tailored probability map significantly improves the position estimation. Although the position residuals are not always the smallest over the whole trajectory in case of applying the PF integrating the user-tailored map, the mean of the residuals of all points are the lowest in each of the four walks, see Figure 6.8. It can be observed that walk 2, 3, and 4, visualized in Figure 6.7c, 6.7e, and 6.7g, have large position residuals between reference point 29 and 36. As illustrated in Figure 6.3, the reference points 29 to 36 are located along the longest straight path of the walked trajectory. As the position cannot be constrained in the along track component while the pedestrian is walking along a straight pavement, the position residuals are increasing with the length of the walk. The standard deviations visualized in Figure 6.7b, 6.7d, 6.7f, and 6.7h also show a higher spreading between the reference points 29 and 36.



(a) Horizontal position residuals - walk 1.

(b) Standard deviations - walk 1.

(c) Horizontal position residuals - walk 2.

(d) Standard deviations - walk 2.

(e) Horizontal position residuals - walk 3.

(f) Standard deviations - walk 3.

(g) Horizontal position residuals - walk 4.

(h) Standard deviations - walk 4.

Figure 6.7: Horizontal position residuals Δx_{horiz} averaged over ten evaluations and the corresponding standard deviations of two different particle filter evaluations.

The standard deviations of the mean horizontal position residuals, which are visualized in Figure 6.7b, 6.7d, 6.7f, and 6.7h, demonstrate that a worse GNSS performance partly causes higher residuals in the PF solutions. Moreover, in 6.7d, 6.7f, and 6.7h the decreasing accuracy of the PDR solution with increasing time can be observed in particular between reference points 27 and 36 due to the absence of drift reducing constraints.

The overall mean of the horizontal position residuals are visualized for each of the four walks in Figure 6.8. Figure 6.7 and Figure 6.8 reveal, that the filter performance strongly improves due to the integration of the user-tailored map. Although the evaluation with a common map already reduces the position residuals by about 11% to 20%, the integration of the user-tailored map comes along with an improvement of about 47% to 69%.

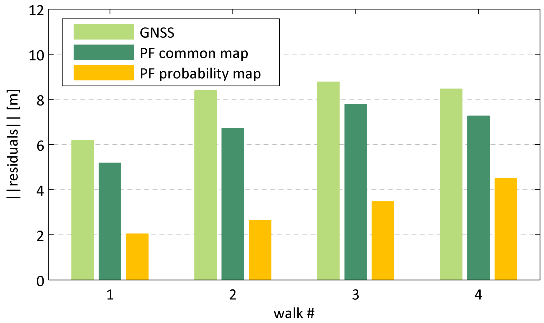


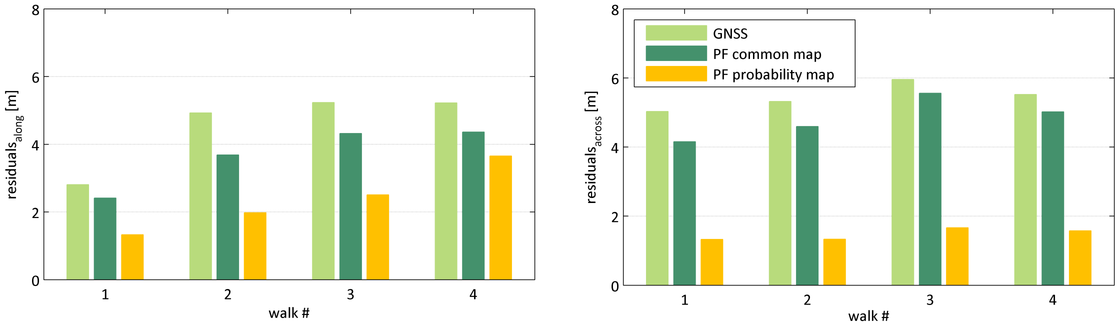
Figure 6.8: Mean of the horizontal position residuals for each of the four walks.

As Figure 6.8 shows, the mean of the residuals based on the PF integrating the user-tailored probability map of is lowest for walk 1 with 2.04 m, while walk 4 represents a less accurate result with a mean of 4.49 m. In general, all results are best for walk 1. The worst result in case of the user-tailored probability is obtained for walk 4, although the GNSS and the particle filter performance based on a common map are worst for walk 3.

The position residuals are further investigated in along and across track direction, because of the errors caused by the recording of the time stamps and due to the reason that the geometric constraints are assumed to have more influence on the across than on the along track component. Therefore, the along track and across track residuals are visualized separately in Figure 6.9. Although the evaluation is based on the same input data and processing parameters, the residuals shown in Figure 6.9 do not coincide with the residuals in Figure 6.9. This is due to the random initialization of the particles.

Figure 6.9 demonstrate that the across track residuals can be further reduced by the incorporation of the user-tailored probability map than the along track residuals. Although a diversity in the mean along track residuals can be observed for the PF based on the user-tailored map, the across track residuals nearly provide the same results for all walks.

Furthermore, the improvement due to the PF integrating the common map is more or less the same in each component.



(a) Along track residuals.

(b) Across track residuals.

Figure 6.9: Mean of the along and across track residuals for each of the four walks.

Summarizing, it can be stated that the user-tailored probability map strongly improves the position estimation. The horizontal position residuals can be reduced by 47% to 69%. Separating the along and across track components, a decrease of 30% to 60% can be established in the means of the along track residuals, while the mean values of the across track residuals are improved by 72% to 75%.

6.2.4 Positioning performance of one selected walk

Finally, one walk (walk 2) is chosen to exemplarily visualize a determined trajectory in the defined test environment. Figure 6.10 opposes the trajectories of the input data for the filter (GNSS and PDR) to the results of the PF integrating the user-tailored probability map and of the PF using the geometric constraints of a common map. In the background, the features of the user-tailored raster map are illustrated as a gray-scaled image. While the PDR trajectory is affected by a drift, which is typical for trajectories derived from inertial data, the GNSS trajectory spreads according to the uncertainties of a code-based SPP solution, which is in the range of about five to ten meters.

The particle filtered trajectory illustrated in yellow, which is based on the user-tailored map, coincides with the reference trajectory (see Figure 6.3) along the entire track of the pedestrian's walk. The trajectory follows the correct pavements, crosses the street at zebra crossings only, and ends in the point, in which the trajectory started. The horizontal position residuals, which are illustrated in Figure 6.11a, can be improved by 70%, while the gain yielded by the common map is only 23%. During the entire walk, the along track residuals of the PF integrating the user-tailored map are less than 5 m except in reference point 30

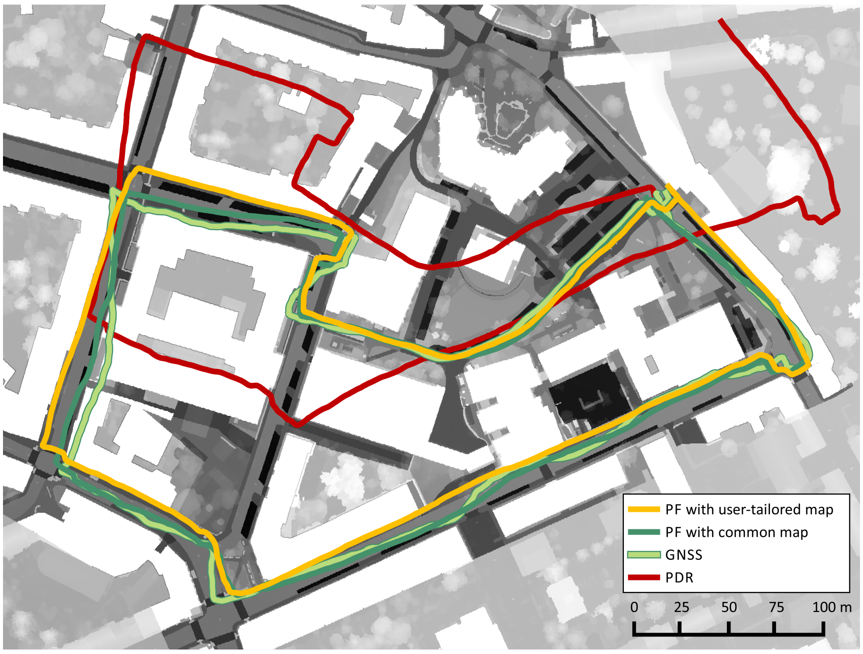
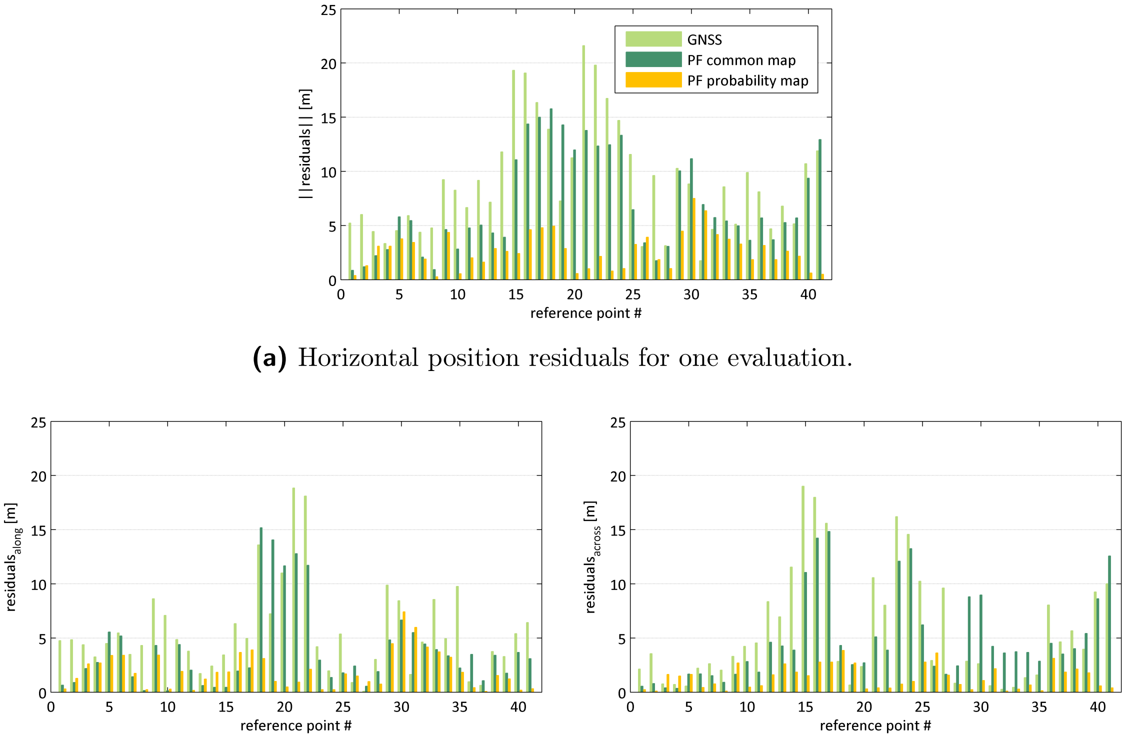


Figure 6.10: Comparison of the evaluation results of the particle filter integrating the user-tailored probability map and a common map.

and 31 (see Figure 6.11b), while the across track residuals have a maximum of 3.85 m (see Figure 6.11c). As the maximum across track residual show, the RBPF concept developed within this thesis provides a localization at the correct pavement for the entire duration of the walk. 68% of the along track residuals and 80% of the across track residuals are less than 2.2 m.

The comparison of both PF results reveals that the integration of the user-tailored probability map is absolutely necessary for an accurate positioning, which is emphasized by the horizontal, along track, and across track residuals visualized in Figure 6.11. It can be observed, that the user-tailored map has in general a higher impact on the across track component. In the along-track component the improvement is less, although the GNSS residuals are about the same in both directions (compare Table 6.3).

Table 6.3 summarizes the mean residuals of the GNSS and PDR trajectory, as well as the two different realizations of the particle filter. Furthermore, the results of a different evaluation approach, which does not incorporate PDR data within the propagation step, are listed. A common motion model assuming a uniform velocity was used for the particle propagation, whereas the measurement update is the same as described within section 5.1.3. The residuals of this PF approach are indicated with *wo PDR*, while the residuals marked with *w PDR* correspond to the results of the above described PF evaluations, which are based on a particle propagation with PDR data and a measurement update incorporating GNSS and map information. Again, the processing was done with the two different maps.



(a) Horizontal position residuals for one evaluation.

(b) Along track residuals for one evaluation.

(c) Across track residuals for one evaluation.

Figure 6.11: Horizontal, along and across track residuals of one evaluation for one selected trajectory (walk 2).

The residuals of the evaluations without PDR data are averaged over ten particle filter computations, because of the high diversity in the evaluation results. In general, the evaluation without PDR data lead to much worse results. In case of the incorporation of the user-tailored probability map even a degradation of the residuals compared to the GNSS trajectory can be observed. This is due to the insufficient propagation of the particles. If no information about the pedestrian's motion is used, and the particles are predicted with respect to an assumed motion model, the motion model's uncertainty must increase leading to a higher particle spreading. In case of a GNSS position solution at the false pavement, the PF consequently follows the wrong side of the street as no further information about the actual motion is available. In case of the incorporation of the common map, the GNSS performance can be slightly improved particularly in the across track component due to the general smoothing behavior of a navigation filter. The less strict constraints does not force the PF following the wrong pavement and therefore the common map leads to better results compared to the user-tailored map. The worse standard deviations of the results related to the user-tailored map (especially in the across-track component) demonstrate the effect of the strict constraints. Depending on the evaluation, the filter follows different pathways and therefore, the individual results do not coincide.

Table 6.3: Mean of the horizontal, along track, and across track position residuals.

		$\mu_{\Delta x_{horiz}}$ [m]	$\mu_{\Delta x_{along}}$ [m]	$\mu_{\Delta x_{across}}$ [m]
	GNSS	9.14	5.64	5.77
	PDR	69.23	38.16	50.56
w PDR	PF common map	6.99	3.87	4.84
	PF user-tailored map	2.66	1.99	1.37
wo PDR	PF common map	8.74 ± 0.35	5.72 ± 0.34	4.97 ± 0.22
	PF user-tailored map	10.24 ± 0.88	5.82 ± 0.49	6.74 ± 0.89

Finally, Figure 6.12 shows the error ellipses for every fifth second along the trajectory. Based on the weighted sample covariance given in equation (6.13), the ellipses are determined by applying equation (6.14) to (6.18). The error ellipses represent Gaussian approximations of the posterior PDFs, which are given by equation (5.30). The PF trajectory and thus the center of the ellipses are given by equation (6.4), which corresponds to the weighted sample mean. For a better interpretation of the resulting error ellipses, the GNSS trajectory is illustrated in black.

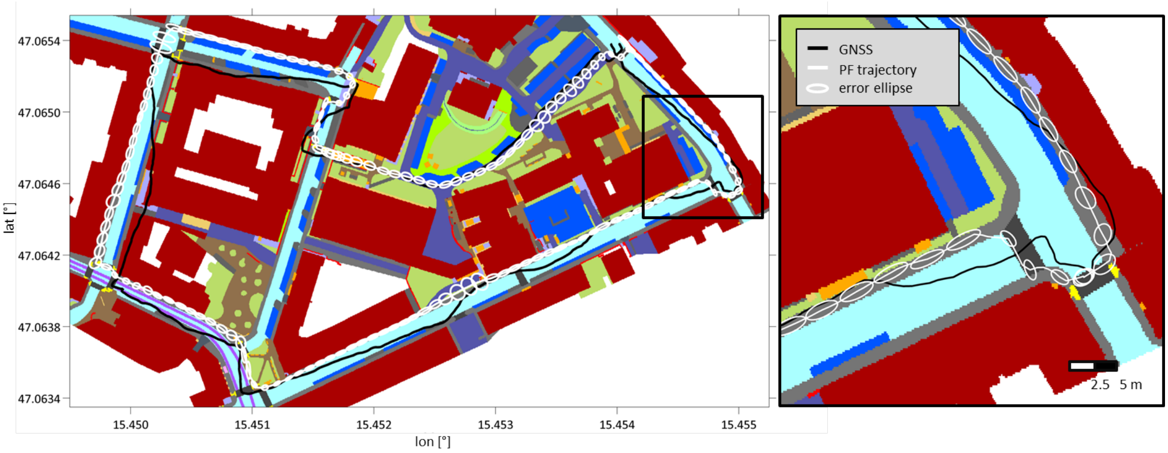


Figure 6.12: Error ellipses for every fifth second along the trajectory.

The left graph in Figure 6.12 gives an overview of the shape of the error ellipses along the whole trajectory, while the right graph shows a part of the trajectory increased in size. The size and orientation of the ellipses demonstrate the impact of the user-tailored map on the spatial distribution of the particles. In regions where several pathways are possible, the size of the error ellipse gets spread. Along pavements, the influence of the building borders can be observed, as the uncertainty in the along track component grows, but the across track error diminishes. By changing the walking direction, the shape of the error ellipse mutate due to the heading change and the various walking possibilities at crossings.

6.2.5 Investigation related to unexpected walking behavior

Due to the spatial constraints of the user-tailored map, an immense performance gain can be observed. However, the defined location probabilities are only valid as long as the blind pedestrian follows the assumed paths. Nevertheless, an approach has to be found, which also handles unexpected walking behavior and still benefits from the developed RBPF. Therefore, a test statistic has been introduced, which recognizes e.g., that a person crosses a street although there is no crosswalk. Figure 6.13 shows the case where the blind person assumes a crosswalk due to a beveled pavement and crosses the street.

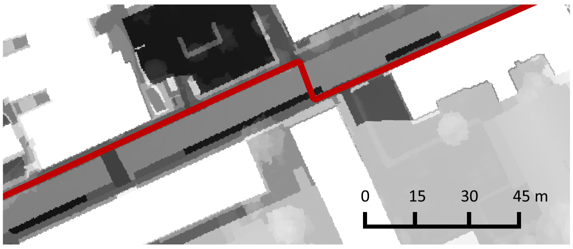


Figure 6.13: Unexpected crossing of the street.

Within the thesis, a concept for the handling of such unexpected movement has been developed and investigated. Since the method is not sufficiently mature for the integration within the RBPF, only a concept is presented at this place.

For the detection of an irregular movement an outlier detection algorithm for particle filter localization, originally developed by Maíz et al. (2009), has been adapted. Maíz et al. (2009) use a statistic, which is called *spatial depth*. With the proposed method, outliers among the measurements are detected and are then rejected for the measurement update within the particle filter. In general, spatial depth determines the probability of the measurement $\boldsymbol{z}_t$ based on a predicted measurement probability

$$p(\tilde{\boldsymbol{z}}_t|\boldsymbol{z}_{t-1}) = \frac{1}{N_s}\sum_{i=1}^{N_s} p(\tilde{\boldsymbol{z}}_t|\tilde{\boldsymbol{x}}_t^i), \tag{6.19}$$

with $\tilde{\boldsymbol{x}}_t$ representing the propagated particle set according to $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1}, \boldsymbol{u}_t)$. As stated in Maíz et al. (2009), the goodness-of-fit between the predicted probability $p(\tilde{\boldsymbol{z}}_t|\boldsymbol{z}_{t-1})$ and $\boldsymbol{z}_t$ can be determined by using the spatial depth statistic

$$D_{t|t-1}(\boldsymbol{z}_t) = 1 - \left\| \int S(\tilde{\boldsymbol{z}}_t - \boldsymbol{z}_t) p(\tilde{\boldsymbol{z}}_t|\boldsymbol{z}_{t-1}) d\tilde{\boldsymbol{z}}_t \right\| \tag{6.20}$$

where $S(\boldsymbol{x})$ is the spatial sign

$$S(\boldsymbol{x}) = \begin{cases} \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|}, & \text{if } \boldsymbol{x} \neq 0, \\ \boldsymbol{0}, & \text{if } \boldsymbol{x} = 0, \end{cases} \tag{6.21}$$

with $\boldsymbol{x} = \tilde{\boldsymbol{z}}_t - \boldsymbol{z}_t$. Given a set of particles $\{\tilde{\boldsymbol{z}}_t^i\}_{i=1}^{N_s}$, which are drawn from $p(\tilde{\boldsymbol{z}}_t|\boldsymbol{z}_{t-1})$, the spatial depth can be approximated as

$$D_{t|t-1}(\boldsymbol{z}_t) \approx 1 - \left\| \frac{1}{N_s} \sum_{i=1}^{N_s} S(\tilde{\boldsymbol{z}}_t^i - \boldsymbol{z}_t) \right\| . \tag{6.22}$$

Within the thesis, the same statistic is employed for the detection of an unexpected movement. Due to the spatial constraints, as soon as the pedestrian is unambiguously located at a pavement, the particles follow this pavement according to the determined motion parameters. If the person crosses the street in an area where no crosswalk is located, the spatial constraints would not allow the crossing. However, the processing would fail after some epochs since the heading θ does not coincide with the direction of the pavement. Therefore, the unexpected movement has to be detected before the processing terminates.

As within the thesis the relation between $\boldsymbol{z}_t$ and $\boldsymbol{x}_t$ is given by $\boldsymbol{z}_t = \boldsymbol{H}_t \tilde{\boldsymbol{x}}_t$ with $\boldsymbol{H}_t$ being an identity matrix of dimension 2×2, equation (6.22) can be rewritten as

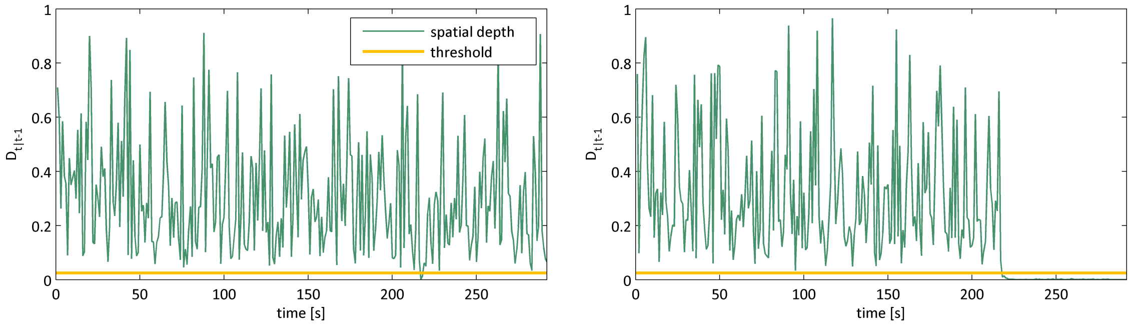
$$D_{t|t-1}(\boldsymbol{z}_t) \approx 1 - \left\| \frac{1}{N_s} \sum_{i=1}^{N_s} S(\tilde{\boldsymbol{x}}_t^i - \boldsymbol{z}_t) \right\| . \tag{6.23}$$

In case of an outlier, $D_{t|t-1}(\boldsymbol{z}_t)$ gets close to zero, while a conformity corresponding to $\boldsymbol{z_t} \sim p(\tilde{\boldsymbol{x}}_t)$ would result in a value close to one. For the detection of an outlier, a threshold has to be defined, which is 0.025 for the investigations within the thesis.

Based on equation (6.23), two different cases are investigated. One case treats the occurrence of an outlier in the GNSS data, while the second case covers the matter of unexpected movement. Since the RBPF handles outliers very well, detected outliers can be ignored. In contrast, unexpected movement has to be considered in the position estimation. Consequently, an appropriate algorithm has to distinguish these two cases.

Figure 6.14 shows a spatial depth investigation for two different trajectories - one with an outlier (Figure 6.14a) and one with untypical movement (Figure 6.14b). In both cases the spatial depth detects the periods in which the GNSS solution does not agree with the modeled path preferences. As Figure 6.14 demonstrates, an outlier clearly differs from an unexpected movement by the duration of $D_{t|t-1}(\boldsymbol{z}_t)$ being close to zero.

In the case, that the spatial depth detects that the pedestrian crossed the street without using the crosswalk, the RBPF may initialize the particles position according to the latest GNSS solution but keeps the heading.



(a) Spatial depth in case of an outlier.

(b) Spatial depth in case of unexpected movement.

Figure 6.14: Spatial depth investigation for the detection of unexpected movement behavior.

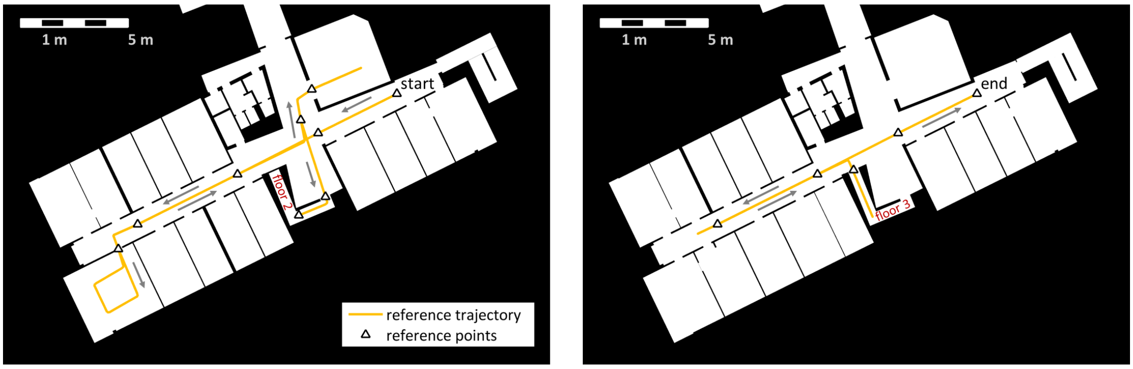
6.2.6 Conclusion - performance in outdoor environments

The investigations within this section revealed that the GNSS position solution can be improved by about 47% to 69% if the user-tailored probability map is integrated within the PF processing. For the processing, at least 500 particles are recommended, which come along with a mean computation time of 0.01 s for the evaluation of one epoch. However, the processing with 2000 particles guarantees the reduction of the mean position residuals to a minimum. The mean of the horizontal position residuals in the worst case is about 4.42 m, while the maximum mean residual in across track and along track direction are 1.65 m and 3.58 m respectively. The low residuals in the across track direction indicate that RBPF is able to localize of the pedestrian at the correct pavement. For each of the four walks, the along track residuals do not fulfill the accuracy requirements of blind pedestrians, which are in the range of about 2 m according to the sensing range of a blindman's stick. The worse along-track accuracy has to be taken into account for the temporal announcement of risks and obstacles and for giving guidance instructions. In contrast, the across track performance is very important for the provision of the correct warnings and guidance instructions and has to fulfill the accuracy requirements of 2 m.

6.3 Performance in indoor environments

As indoors, other spatial constraints are integrated within the RBPF compared to outdoors, the indoor positioning is analyzed separately. As described in section 5.2, an occupancy grid map in combination with the wall collision filter constrains the particle propagation within the building. Furthermore, motion-related activities provide location probabilities which are integrated within the RBPF.

For this performance investigations, raw sensor data have been recorded for post-processing, and several pre-defined reference points have been passed to have a ground truth. Three different trajectories have been chosen to analyze the performance of the filter. These trajectories are illustrated in Figure 6.15, 6.16, and 6.17. Besides the walked trajectories, the reference points are visualized. Figure 6.15 shows the first trajectory, where the pedestrian starts on floor level 4, follows the corridor, enters two different rooms before the person walks downstairs to floor level 3. On floor level 3, the pedestrian only walks along the corridor. The second trajectory follows the corridor on a single floor (floor level 4) forth and back (Figure 6.16). In contrast, the third trajectory focuses on walking along corridors on different floor levels (Figure 6.17).



(a) Floor level 4. **(b)** Floor level 3.

Figure 6.15: Trajectory 1 within the test ground of the building Steyrergasse 30.

These three trajectories have been selected due to the different impact of the spatial constraints given by the PAC, the map, and the wall collision filter. Trajectory 1 has the highest demands on the wall collision filter, while the determined activity is primarily needed for an effective floor level detection as it is performed within trajectory 1 and 3. At stairs, a correctly determined activity supports the positioning in reducing the step length and by providing

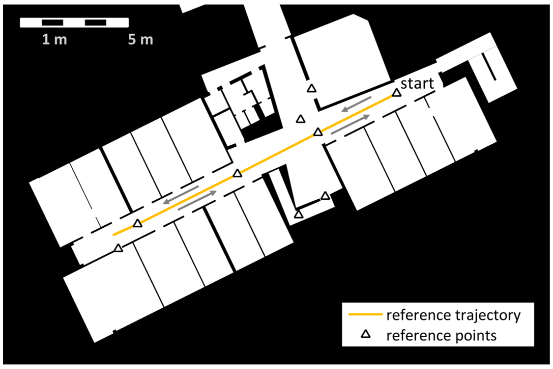
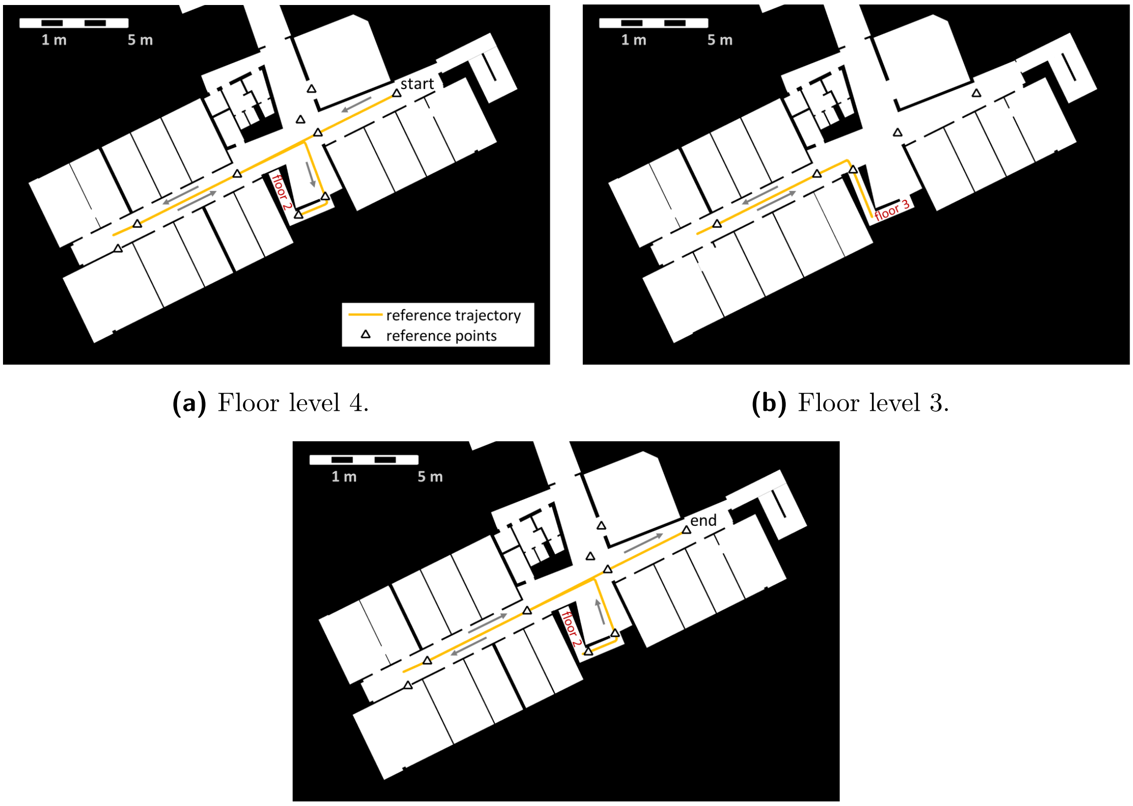


Figure 6.16: Trajectory 2 within the test ground of the building Steyrergasse 30.



(a) Floor level 4.

(b) Floor level 3.

(c) Floor level 4.

Figure 6.17: Trajectory 3 within the test ground of the building Steyrergasse 30.

appropriate probabilities for stairways. Trajectory 2 should demonstrate the quality of the position estimation for the case that no room is entered and the pedestrian stays at the same floor. Each of the three trajectories are walked several times, whereby the durations of the walks are about 200 s, 65 s, and 200 s for trajectory 1, 2, and 3. Based on these walks, similar investigations as shown in section 6.2 are conducted.

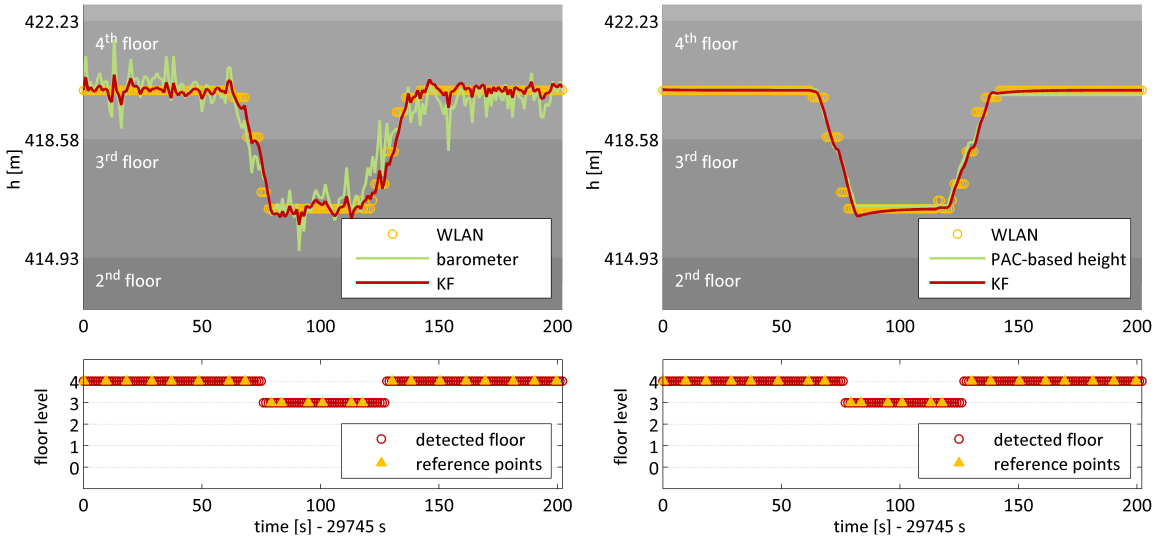
As the particle filter performance in indoor environments strongly depends on the provision of the correct floor plan, in a first instance, the efficiency of the floor detection is investigated (section 6.3.1). Then, investigations concerning the optimal number of particles N_s are done (section 6.3.2). Because of the small scaled indoor environment computational break-offs may occur. Therefore, besides the computation time and the position residuals, the break-off quota is analyzed. In section 6.3.3, the most suitable parameter for the representation of the state estimate is identified, while the performance gain due to the incorporation of the activity is demonstrated in section 6.3.4. Finally, one walk along trajectory 1 is investigated in detail (section 6.3.5). Hereby, the horizontal trajectory, the position residuals, as well as the error ellipses are analyzed. Consequently, the investigations for the indoor environment can be structured as follows:

- Performance of the height estimation.
- Performance investigations regarding the number of particles.
- Investigation regarding the state estimate representation.
- Performance gain related to the user-tailored probability map.
- Positioning performance of a selected trajectory.

For the different investigations, the measurement data are processed as outlined in the proposed filter design (section 5.2).

6.3.1 Performance of the height estimation

Since indoors, the position estimation needs the correct floor map, an appropriate height estimation based on redundant information and an efficient floor level detection has to be installed. Therefore, as described in section 5.2.4, on the one hand, a combination of h_{WLAN} and Δh_{baro} and, on the other hand, a fusion of h_{WLAN} and Δh_{PAC} are investigated. Trajectory 3 (Figure 6.17) is chosen for the investigation, since this trajectory provides a floor level change two times. In Figure 6.18, the evaluation results for one walk along trajectory 3 are shown. Figure 6.18a shows the results of the height estimation and floor detection for the combination of h_{WLAN} and Δh_{baro}, while Figure 6.18b visualizes the same for the fusion of h_{WLAN} and Δh_{PAC}. The results demonstrate that in both cases the floor transition is detected correctly.



(a) Height estimation based on WLAN and barometer data.

(b) Height estimation based on WLAN and PAC/PDR data.

Figure 6.18: Floor detection based on the estimated height delivered by the Kalman filter.

The amount of incorrectly detected floor levels is determined for three different Kalman filter evaluations. Following three scenarios are treated:

- Kalman filter based on WLAN height data only
- Kalman filter based on WLAN and barometer height data
- Kalman filter based on WLAN and PAC/PDR derived height data

For this investigation, five different walks along trajectory 3 are evaluated. The amount of incorrectly detected floor levels is listed in Table 6.4. The correspondence of the floor level is determined at the reference points, whereby one walk comprises 22 reference points. One incorrectly determined floor level implies that in one of the trajectory's reference points the floor level does not coincide with the actual floor level. For the floor level assignment within the floor detection, appropriate heights are defined. These are given by the medium heights of two consecutive floors.

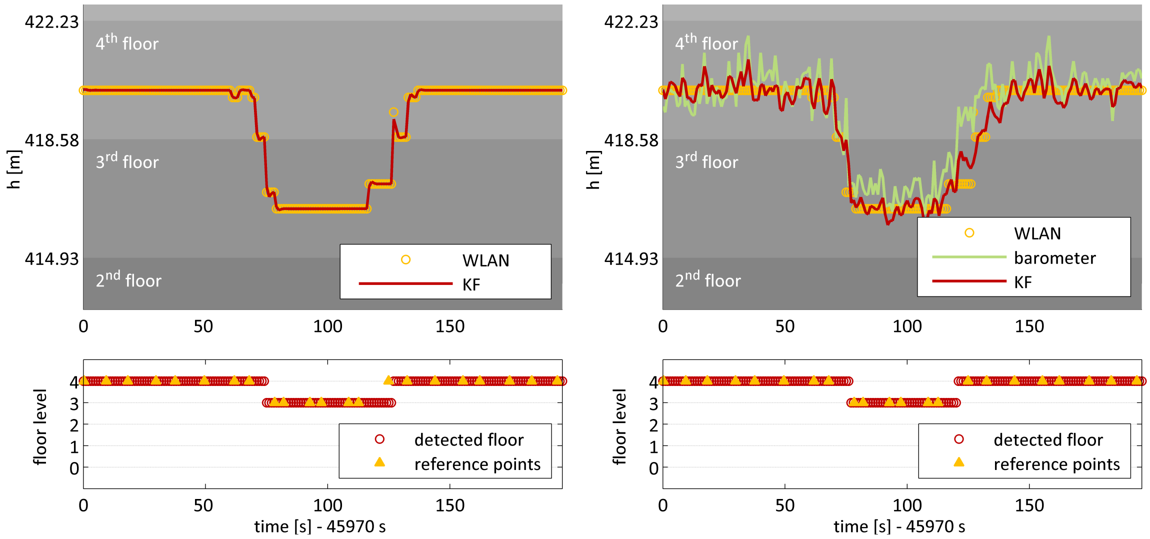
Table 6.4: Comparison of the Kalman filter evaluations in terms of incorrectly detected floor levels.

	h_{WLAN}	h_{WLAN} & Δh_{baro}	h_{WLAN} & Δh_{PAC}
incorrectly detected floors	4	2	0

Table 6.4 shows that the height estimation based on h_{WLAN} only comes along with four incorrectly detected floor levels out of 110. By integrating Δh_{baro}, the amount of incorrectly detected floor levels can be reduced to a number of 2. The best result with no incorrectly detected floor levels is achieved by a fusion of h_{WLAN} and Δh_{PAC}, but strongly depends on a correct detection of the current activity. However, if the pedestrian takes two stair steps by one step, the algorithm misinterprets the vertical change. This could be handled by defining an additional activity, which treats the motion over two stair steps. However, this scenario is quite unlikely in case of a blind person. A further possibility would be the integration of the barometer within the PAC.

Figure 6.19 exemplarily shows the performance gain which is caused by the fusion of h_{WLAN} and Δh_{baro}. Therefore, a walk is chosen, for which the KF evaluation delivers an incorrect floor detection in case of processing with WLAN data only. The support of the height estimation by Δh_{baro} provides the correct floor level throughout the complete walk (Figure 6.19b). In general, changes in height are recognized delayed by WLAN fingerprinting due to the frequency of the WLAN-scan update of 0.5 Hz.

Anyway, all of the reference points, for which the floor levels are incorrectly detected, are located in transition zones between two floors (stairs). At these zones, the spatial constraints



(a) Height estimation based on WLAN data.

(b) Height estimation based on WLAN and barometer data.

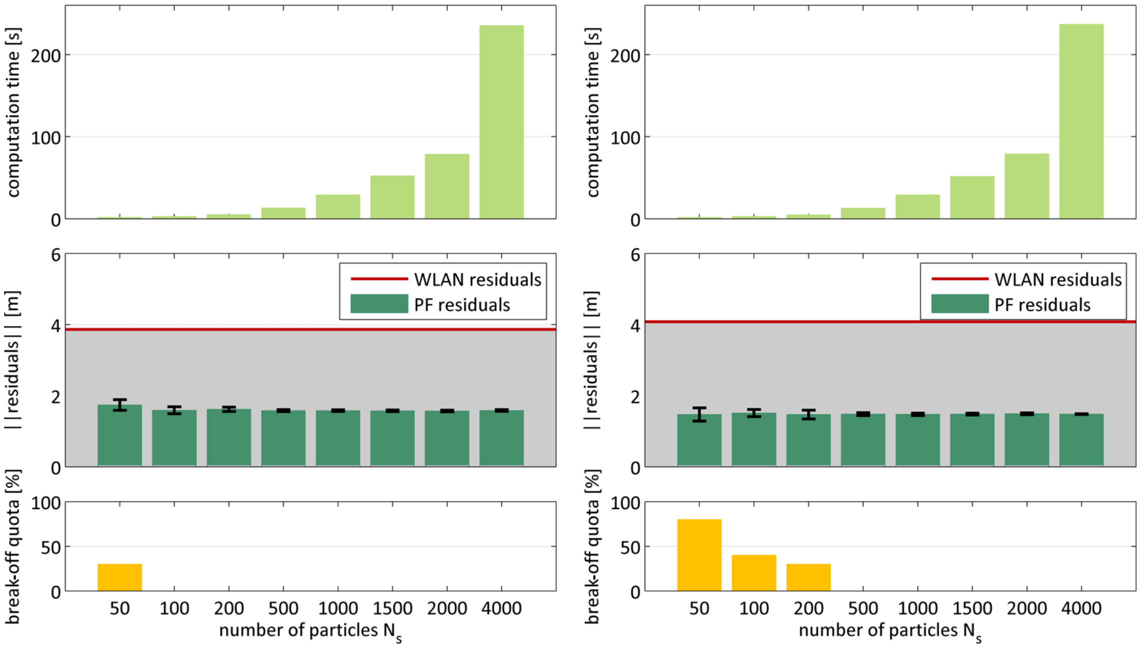
Figure 6.19: Contribution of the barometer to the correct floor detection.

due to the map are the same for both floor levels. However, as soon as the pedestrian leaves the stairs and reaches the corridor, the correct floor level has to be provided. Finally, it can be stated, that both KF approaches allow the detection of the correct floor level as all of the incorrectly detected floor levels are still located at stairs.

6.3.2 Performance investigations regarding the number of particles

Within this section, the mean computation time, the mean horizontal position residuals, as well as the computation break-off quotas are investigated for each of the three trajectories, by comparing two different walks. Hereby, the focus is set on a varying number of particles N_s, to find out the optimal N_s for the indoor processing. The particle filter is processed with $N_s = \{50, 100, 200, 500, 1000, 1500, 2000, 4000\}$. To obtain representative values, every walk is evaluated ten times with a varying number N_s. Consequently, in the following figures and tables, the mean values over all ten evaluations are illustrated.

Trajectory 1: Figure 6.20 shows the evaluation results of two walks along trajectory 1. The first row presents the mean computation time over ten evaluations normalized to a trajectory length of 1000 s, the second row shows the mean horizontal position residuals over all reference points, while the third row contains the quota of computation break-offs. The quota of computation break-offs represents the number of incompletely evaluated trajectories given in % of the overall number of evaluations (10). The computation times and the horizontal



(a) Walk 1.

(b) Walk 2.

Figure 6.20: Mean computation times, mean horizontal position residuals, and computation break-off quotas vs. N_s for two different walks along trajectory 1 (Figure 6.15).

position residuals are similar for both walks, while the quotas of the computation break-offs are different. In both cases, the increase of N_s comes along with higher computational stability. From $N_s = 500$ upward, no more break-off can be observed. In Table 6.5, the numerical values of the mean computation times and the corresponding standard deviations are given. As expected, the increase of N_s comes along with a longer computation time,

Table 6.5: Mean computation times and the corresponding standard deviations for two different walks along trajectory 1 normalized to a trajectory length of 1000 s.

	walk 1		**walk 2**	
N_s	time [s]	σ_t [s]	time [s]	σ_t [s]
50	1.70	0.14	1.62	0.04
100	2.72	0.09	2.64	0.11
200	4.83	0.05	4.70	0.09
500	12.95	0.21	12.79	0.12
1000	28.98	0.28	29.04	0.20
1500	52.25	0.71	51.43	0.32
2000	78.60	0.41	79.04	0.89
4000	235.38	0.86	236.87	1.94

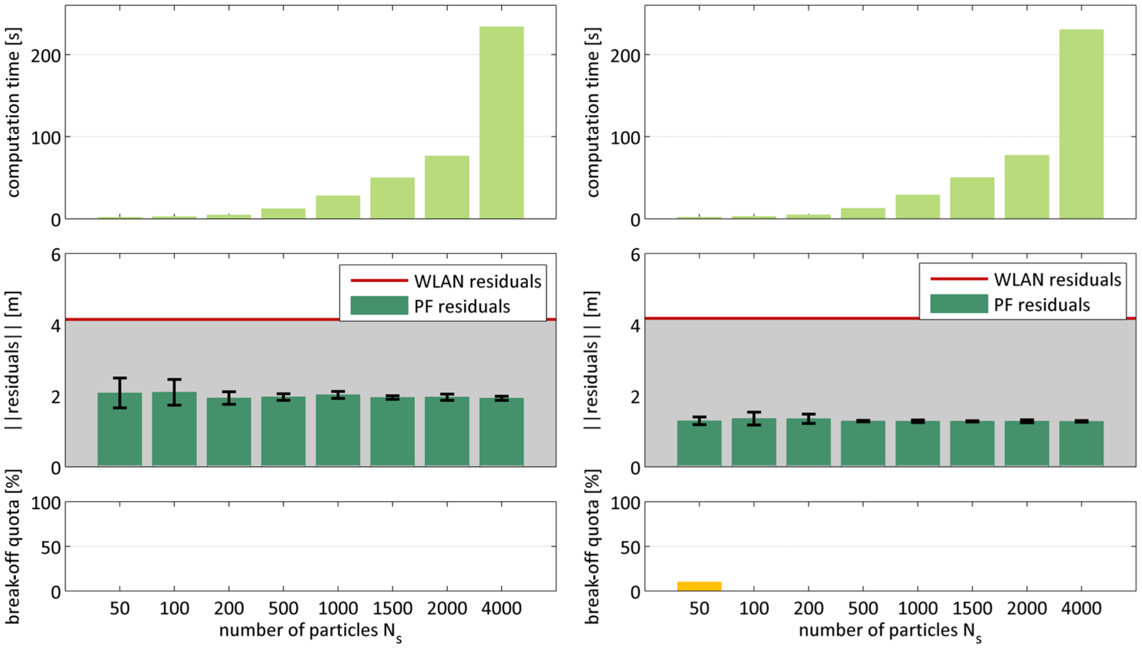
Table 6.6: Mean horizontal position residuals and the corresponding standard deviations for two different walks along trajectory 1 with varying N_s.

	walk 1			**walk 2**		
N_s	$\mu_{\Delta x_{horiz}}$ [m]	$\sigma_{\Delta x_{horiz}}$ [m]	break-offs	$\mu_{\Delta x_{horiz}}$ [m]	$\sigma_{\Delta x_{horiz}}$ [m]	break-offs
50	1.73	0.15	3	1.47	0.18	8
100	1.59	0.10	-	1.51	0.10	4
200	1.61	0.06	-	1.47	0.12	3
500	1.58	0.03	-	1.48	0.04	-
1000	1.58	0.02	-	1.47	0.03	-
1500	1.57	0.02	-	1.48	0.02	-
2000	1.57	0.02	-	1.48	0.02	-
4000	1.58	0.02	-	1.48	0.01	-
WLAN	3.86	1.86		4.08	1.98	
PDR	6.47	2.45		5.29	3.06	

which is characterized by an exponential growth. However, the convergence of the position solution can be improved, which can be verified by the decreasing standard deviation $\sigma_{\Delta x_{horiz}}$ of the mean position residuals $\mu_{\Delta x_{horiz}}$ in Table 6.6.

As Table 6.6 shows, the mean position residuals converge to about 1.58 m in case of the first walk and to about 1.48 m in case of the second walk. Furthermore, the standard deviations of the position residuals over ten evaluations are decreasing with increasing N_s. Due to reasons of comparability, the mean of the horizontal position residuals of the WLAN and PDR trajectories are listed. In case of WLAN and PDR, $\mu_{\Delta x_{horiz}}$ and $\sigma_{\Delta x_{horiz}}$ correspond to the mean value and the standard deviation over the horizontal position residuals in the reference points. All other values are based on ten PF evaluations. For the computation of the PDR residuals, the relative PDR trajectory is initialized at the walk's starting point.

Trajectory 2: The results for trajectory 2 are illustrated in Figure 6.21. The numerical values for the computation times are given in Table 6.7, while the position residuals and the number of break-offs are presented in Table 6.8. Again, the upper graphs in Figure 6.21a and 6.21b demonstrate that the computation time is exponentially increasing with a growing N_s. The small standard deviations σ_t of the computation times given in Table 6.7 indicate that the computation times are comparable for each of the ten particle filter evaluations. Furthermore, the comparison shows that the computation times are very similar for the two different measurement data sets. As Figure 6.21 and Table 6.8 show, although $\mu_{\Delta x_{horiz}}$ of the WLAN and PDR trajectories are in the same range for both walks (Table 6.8), $\mu_{\Delta x_{horiz}}$ of the



(a) Walk 1.

(b) Walk 2.

Figure 6.21: Mean computation times, mean horizontal position residuals, and computation break-off quotas vs. N_s for two different walks along trajectory 2 (Figure 6.16).

PF evaluations are different with about 1.93 m and 1.28 m. However, the higher $\sigma_{\Delta x_{horiz}}$ of the WLAN and the PDR trajectories of the first walk indicate a larger spreading in the horizontal position residuals which may cause the worse filter performance. In both walks, the standard deviations $\sigma_{\Delta x_{horiz}}$ are in general decreasing, indicating a convergence of the solutions with increasing N_s. In case of the second walk, one of the ten evaluations with 50 particles ended

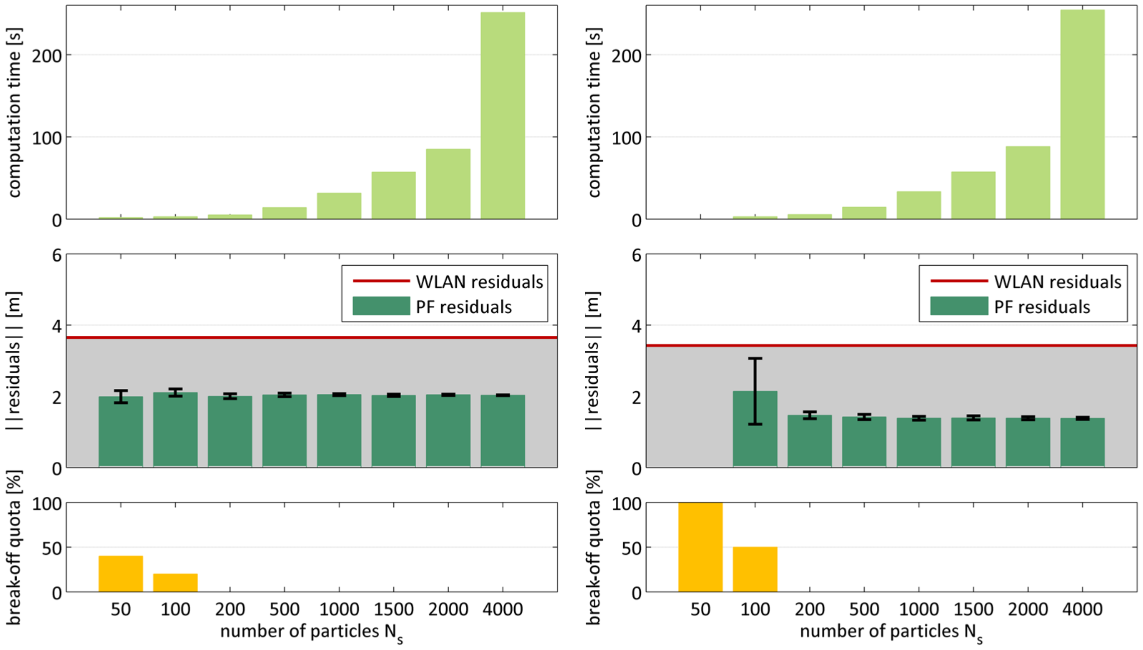
Table 6.7: Mean computation times and the corresponding standard deviations for two different walks along trajectory 2 normalized to a trajectory length of 1000 s.

	walk 1		**walk 2**	
N_s	time [s]	σ_t [s]	time [s]	σ_t [s]
50	1.73	0.44	1.80	0.33
100	2.50	0.12	2.66	0.11
200	4.43	0.17	4.68	0.15
500	12.00	0.19	12.54	0.16
1000	27.82	0.27	28.81	0.19
1500	49.69	0.36	49.91	0.51
2000	76.43	0.45	77.14	0.31
4000	233.66	0.75	230.23	2.29

Table 6.8: Mean horizontal position residuals and the corresponding standard deviations for two different walks along trajectory 2 with varying N_s.

	walk 1			**walk 2**		
N_s	$\mu_{\Delta x_{horiz}}$ [m]	$\sigma_{\Delta x_{horiz}}$ [m]	break-offs	$\mu_{\Delta x_{horiz}}$ [m]	$\sigma_{\Delta x_{horiz}}$ [m]	break-offs
50	2.05	0.41	-	1.29	0.11	1
100	2.09	0.36	-	1.36	0.18	-
200	1.93	0.17	-	1.35	0.13	-
500	1.96	0.09	-	1.28	0.02	-
1000	2.02	0.10	-	1.28	0.03	-
1500	1.95	0.05	-	1.28	0.02	-
2000	1.95	0.09	-	1.28	0.04	-
4000	1.93	0.06	-	1.28	0.02	-
WLAN	4.14	2.98		4.18	1.80	
PDR	4.73	2.32		4.22	2.04	

in a computation break-off. Due to the fact, that the pedestrian is walking at the corridor only, the trajectory is less restricted by the constraints given by the wall collision filter and



(a) Walk 1.

(b) Walk 2.

Figure 6.22: Mean computation times, mean horizontal position residuals, and computation break-off quotas vs. N_s for two different walks along trajectory 3 (Figure 6.17).

the activity. Consequently, less processing break-offs are caused.

Trajectory 3: Figure 6.22 shows the evaluation results for two walks along trajectory 3. The characteristics with respect to the computation times and the mean horizontal position residuals are similar to trajectory 1 and 2. However, the second walk cannot be processed with 50 particles. The break-off quota of the PF processing is 100%. Consequently, neither the computation time nor the mean horizontal position residual can be declared for the second walk for the PF processing with $N_s = 50$, see Table 6.9 and 6.10.

Table 6.9: Mean computation times and the corresponding standard deviations for two different walks along trajectory 3 normalized to a trajectory length of 1000 s.

	walk 1		walk 2	
N_s	time [s]	σ_t [s]	time [s]	σ_t [s]
50	1.91	0.17	-	-
100	2.87	0.14	3.02	0.08
200	5.21	0.09	5.55	0.27
500	14.05	0.17	14.57	0.26
1000	31.61	0.48	33.37	0.32
1500	57.21	1.31	57.45	0.70
2000	85.01	1.22	88.23	1.60
4000	251.31	1.70	254.48	1.60

Table 6.10: Mean horizontal position residuals and the corresponding standard deviations for two different walks along trajectory 3 with varying N_s.

	walk 1			walk 2		
N_s	$\mu_{\Delta x_{horiz}}$ [m]	$\sigma_{\Delta x_{horiz}}$ [m]	break-offs	$\mu_{\Delta x_{horiz}}$ [m]	$\sigma_{\Delta x_{horiz}}$ [m]	break-offs
50	1.99	0.17	4	-	-	10
100	2.10	0.10	2	2.14	0.92	5
200	2.00	0.07	-	1.46	0.10	-
500	2.04	0.05	-	1.42	0.07	-
1000	2.04	0.03	-	1.38	0.05	-
1500	2.02	0.03	-	1.39	0.06	-
2000	2.04	0.02	-	1.38	0.04	-
4000	2.03	0.01	-	1.38	0.03	-
WLAN	3.65	2.02		3.42	1.93	
PDR	3.69	2.39		6.06	3.38	

Although the evaluation is based on different measurement data sets, the computation times given in Table 6.9 are again very similar. As already discussed for trajectory 1 and 2, the small σ_t indicate the similarity of the processing times for all ten evaluations. The mean position residuals listed in Table 6.10 are converging to about 2.03 m and 1.38 m with decreasing $\sigma_{\Delta x_{horiz}}$. In comparison, the mean position residuals of the two WLAN trajectories are 3.65 m and 3.42 m, while the mean values for the PDR solutions are 3.69 m and 6.06 m. The higher break-off quota of the second walk is due to the worse PDR performance. However, if the number of particles is chosen correspondingly, the positioning performance of the second walk is very good with a mean $\mu_{\Delta x_{horiz}}$ of 1.38 m.

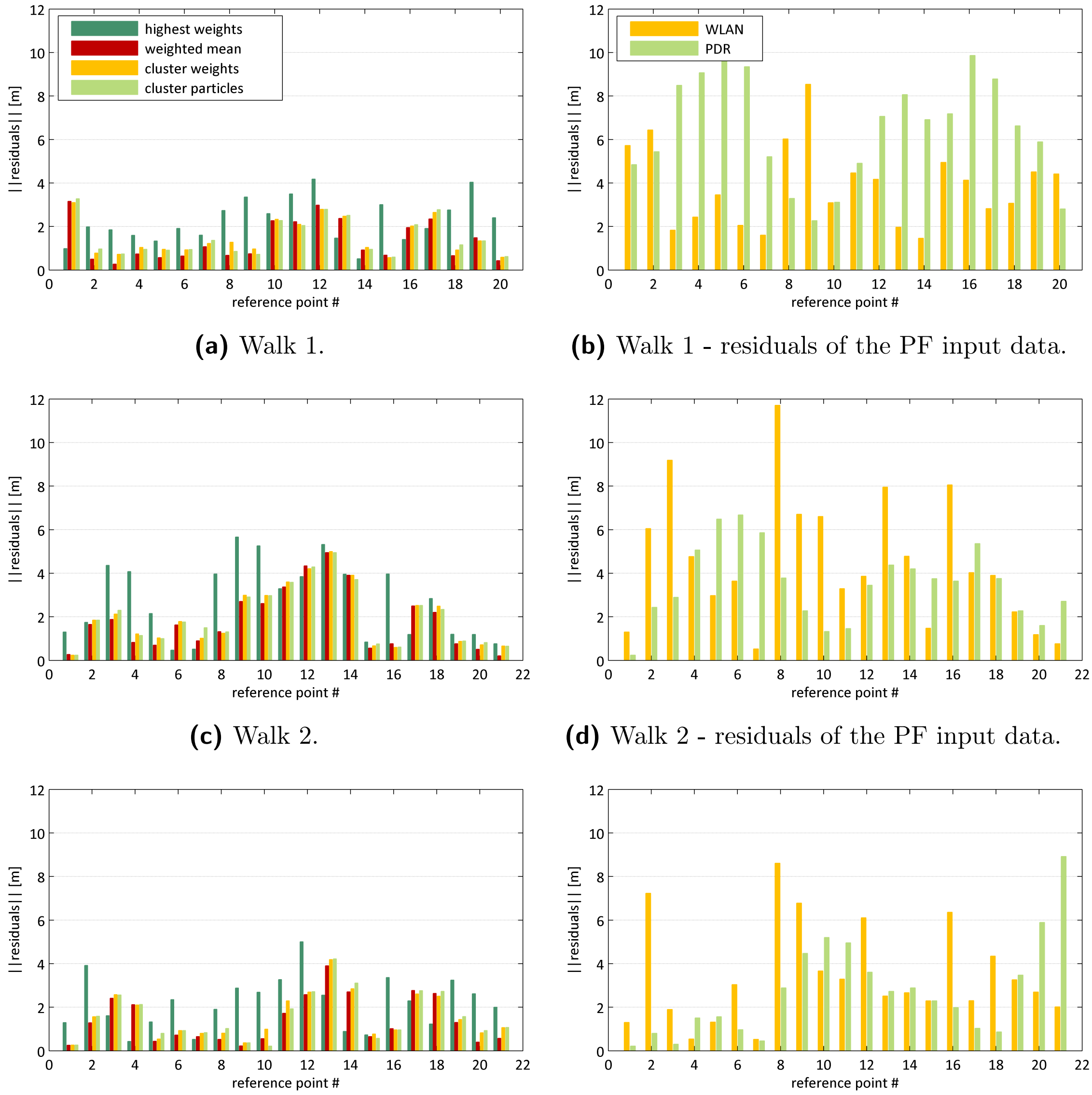
Comparison of the different trajectories: The evaluation results of the three different trajectories demonstrate that the position residuals are decreasing with increasing N_s. The computation times are comparable for all three trajectories. However, the computation times are highest for the two walks of the third trajectory due to occurrence of two floor-level changes. The evaluation of the second trajectory induces the least computational effort, since the pedestrian is walking at one floor only and does not enter any room.

The computation break-off quotas show, that trajectories, which have higher demands on the wall collision filter and the activity integration, come along with more break-offs. The second trajectory, which follows the corridor at one floor only, has nearly no break-offs, while trajectory 1 and 3 are hardly processable with 50 and 100 particles. However, a computational stability is achieved for each of the three trajectories if N_s is chosen to be higher than 500. The magnitudes of the converging position residuals are in the range of 1.28 m to 2.03 m and are independent of the trajectory characteristics. From a number N_s of 500, the residuals start converging to the final result except the first walk of trajectory 2. The mean of the residuals of this walk grows from 1.96 m to 2.02 m if the number of particles is increased from 500 to 1000. After that, the processing based on 1500 particles again shows a mean $\mu_{\Delta x_{horiz}}$ of 1.95 m. The good result with 500 particles is probably due to a better random initialization of the particles and a consequently good propagation of the particles.

In general, it can be stated, that at least 500 particles are required for the evaluation if computation break-offs should be circumvented. 500 particles come along with a computation time of 0.01 s for one epoch and the position solution starts converging. However, the computation with 1500 particles guarantees that the mean residual differs only at maximum 0.01 m from the minimum mean residual, while the computation still needs less than 0.06 s for one epoch.

6.3.3 Investigation regarding the state estimate representation

Indoors, the environmental structures and the involved algorithms are different compared to the outdoor environment. Therefore, the best parameter for the representation of the state estimate has been investigated separately. Again, the particle with the highest weight, the weighted mean, the center of the cluster with the highest accumulated weight, and the center of the cluster with the highest particle concentration are computed and analyzed under the aspect of finding the best representative of the estimated trajectory.



(a) Walk 1.

(b) Walk 1 - residuals of the PF input data.

(c) Walk 2.

(d) Walk 2 - residuals of the PF input data.

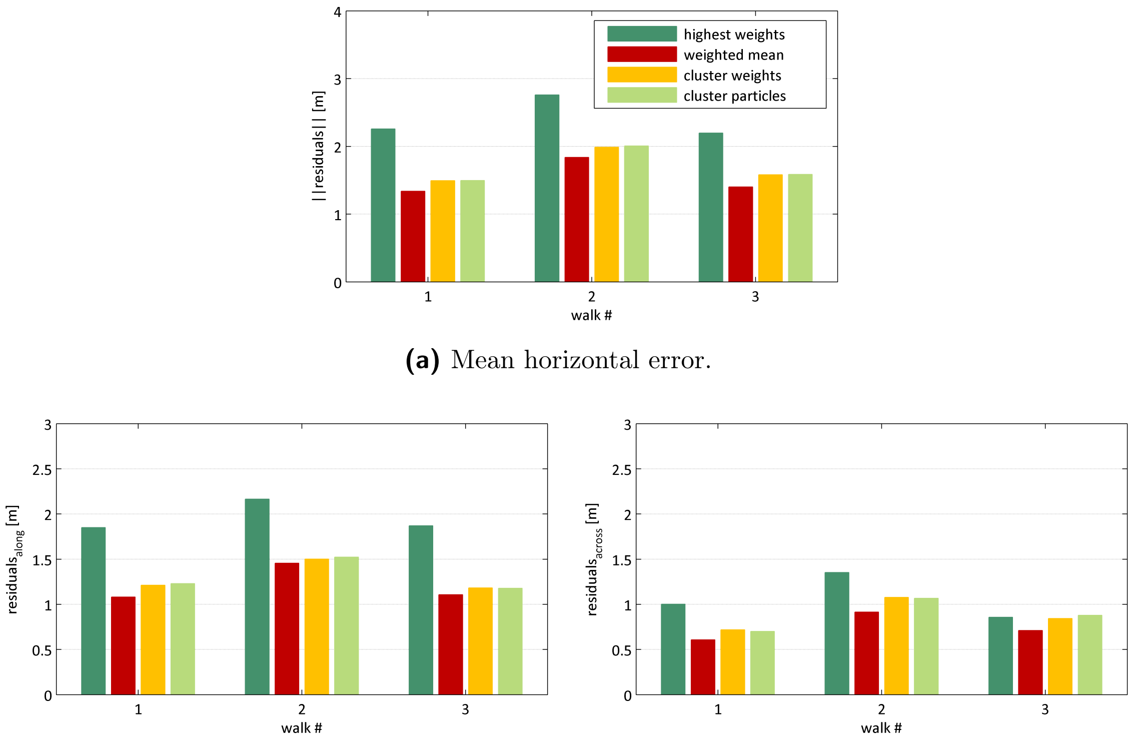
(e) Walk 3.

(f) Walk 3 - residuals of the PF input data.

Figure 6.23: Error in the horizontal position resulting from four processing strategies applied to three realizations of trajectory1.

As trajectory 1 includes walking on corridors and through rooms, but also a floor-level change (see Figure 6.15), three walks along trajectory 1 are taken as basis for this investigation. The particle filter processing is done with 2000 particles, as the focus is not the real-time processing, but the identification of the best representative.

The graphs on the left hand side of Figure 6.23 show the mean values of the horizontal position residuals over ten evaluations at each of the passed reference points, while the graphs on the right hand side illustrate the mean horizontal position residuals for the WLAN and the PDR trajectory. As the output of the PDR algorithm corresponds to a relative position solution, the PDR trajectory is initialized at the starting point of the walk for the computation of the residuals. Neither the WLAN trajectory nor the PDR trajectory is processed within a filter. In Figure 6.23a, 6.23c, and 6.23e, it can be observed that in indoor environments the trajectory is represented worst by the highest weighted particle. Concerning the other three strategies, no explicit statement can be given.



(a) Mean horizontal error.

(b) Mean along track error.

(c) Mean across track error.

Figure 6.24: Mean error resulting from four processing strategies applied to three realizations of trajectory 1.

However, by looking at the mean of the horizontal position residuals over all reference points for each of the three different walks in Figure 6.24a, the weighted mean can be identified as best representative. In addition, if the mean residuals are separated into an along track component (Figure 6.24b) and an across track component (Figure 6.24c), the weighted mean turns out to be best parameter for both components. The weighted mean leads to the minimum mean residual in each of the three walks. Furthermore, similar to the outdoor positioning, the investigations reveal that the mean across track residuals are smaller than the mean along track residuals.

6.3.4 Performance gain related to the integration of the activity

In this section, the performance gain related to the integration of the person's activity as spatial constraint within the position estimation is demonstrated. Anyway, only a small improvement is expected, since the spatial constraints provided by the defined activities *standing*, *walking*, *walking downstairs*, and *walking upstairs* mainly support the positioning filter at stairs (see section 5.2). However, as the stairs within the test ground are of small dimension, the importance weights ω_{PAC} strongly contribute to the positioning if the pedestrian follow the stairs.

Within this section, trajectory 3 (Figure 6.17) acts as a basis for the investigations, because of the fact that this trajectory provides two floor changes. Five different walks are evaluated, delivering the mean horizontal errors illustrated in Figure 6.25. These values are the mean over the horizontal residuals evaluated in the reference points, which are based on the mean of ten particle filter computations with 2000 particles. The trajectory is represented by the weighted mean μ_{wm}.

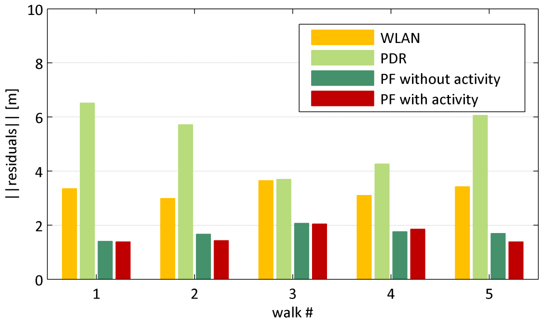
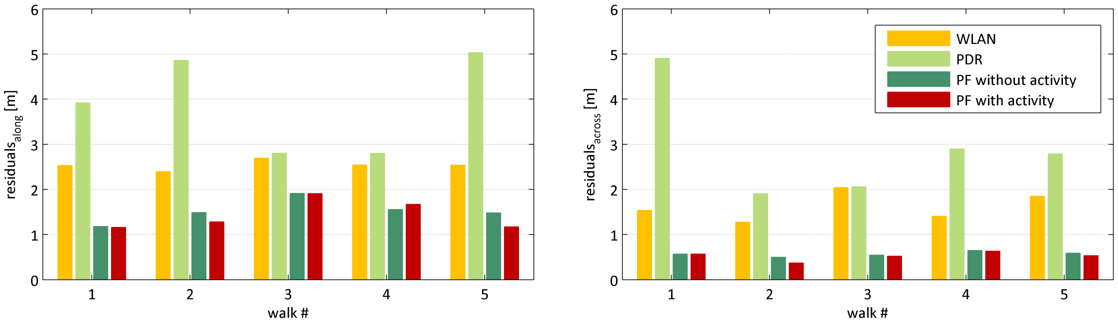


Figure 6.25: Mean horizontal errors for each of the five walks.

As Figure 6.25 shows, for four of the overall five walks, an improvement due to the activity integration can be observed. Only the position errors of the fourth walk increase slightly by about 9 cm due to the incorporation of activity-related spatial constraints. However, the along track and across track residuals of walk 4 illustrated in Figure 6.26 reveal that only the along track residuals are affected by the worse performance. In contrast, the across track component, which benefits more from the activity integration, is slightly improved by 2 cm.



(a) Mean along track error.

(b) Mean across track error.

Figure 6.26: Mean along and across track errors for each of the five walks.

The implemented filter approach for the indoor positioning improves the mean of the horizontal position residuals by about 40% to 60% if the WLAN trajectory is considered as reference. Hereby, the filter and the appropriate spatial constraints have a higher impact on the across track component, which comes along with an improvement between 55% (walk 4) and 75% (walk 3). However, the magnitude of the along track residuals can also be reduced significantly by about 29% to 54%.

Finally, one walk is chosen to exemplarily demonstrate the impact of the activity integration on the horizontal position residuals along the trajectory. These residuals are illustrated in Figure 6.27.

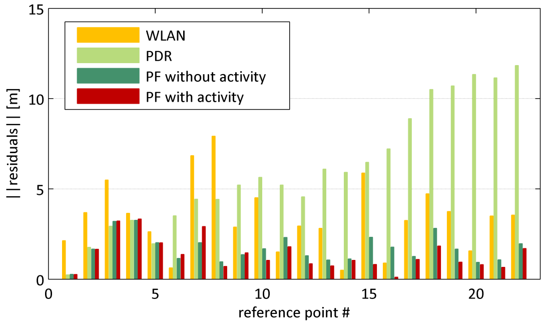


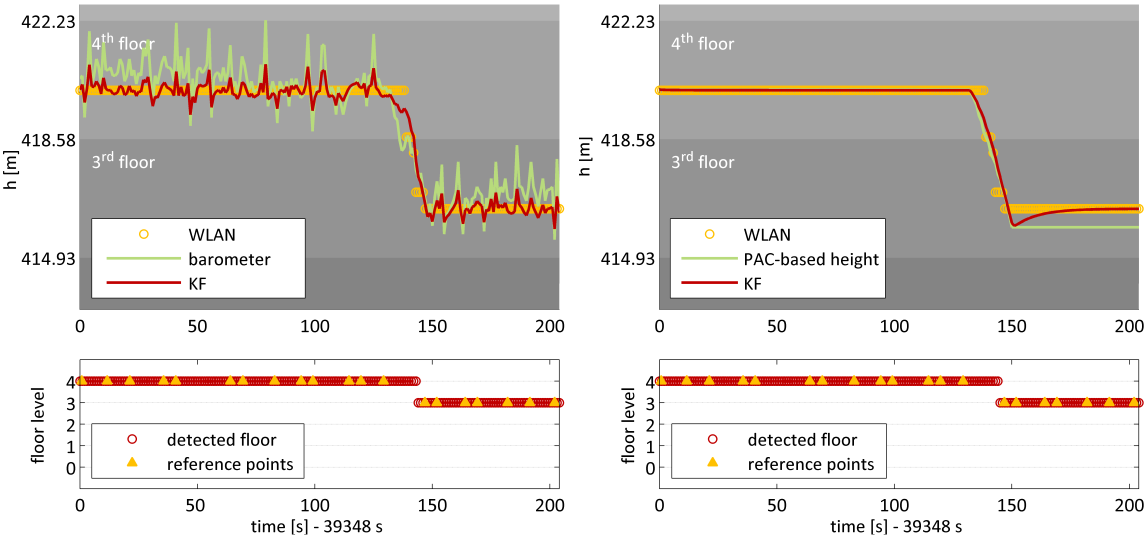
Figure 6.27: Mean horizontal error over ten evaluations of one selected walk.

The reference points 9 and 10 as well as 15 and 16 are located at the stairways. It can be observed, that the integration of the activity has a positive impact on the estimated states at the stairs, but also on the results of the immediately following reference points.

6.3.5 Positioning performance of one selected trajectory

Finally, one walk inside of the building is chosen to investigate the positioning performance of the trajectory in detail. Hereby, the focus was set on trajectory 1 due to its diversity concerning the walk characteristics (Figure 6.15). In a first instance, the outcome of the Kalman filter estimating the height and the consecutive floor detection is analyzed.

In Figure 6.28, two different Kalman filter evaluations are shown. Figure 6.28a illustrates the results of the KF integrating h_{WLAN} and Δh_{baro}, while Figure 6.28b presents the outcome of the KF integrating h_{WLAN} and Δh_{PAC}. In both cases, the floor transitions can be detected correctly. However, as Figure 6.28b shows, too many steps are classified as *walking downstairs* by the PAC algorithm and consequently, the accumulation of Δh_{PAC} leads to an overestimated downward motion. Anyway, it has to be mentioned, that the focus of the height estimation is the detection of the correct floor to provide, on the one hand, the floor map for the PF and, on the other hand, proper navigation instructions to guide the pedestrian through the building. Consequently, the accuracy of the absolute height information is not the central point of this research.

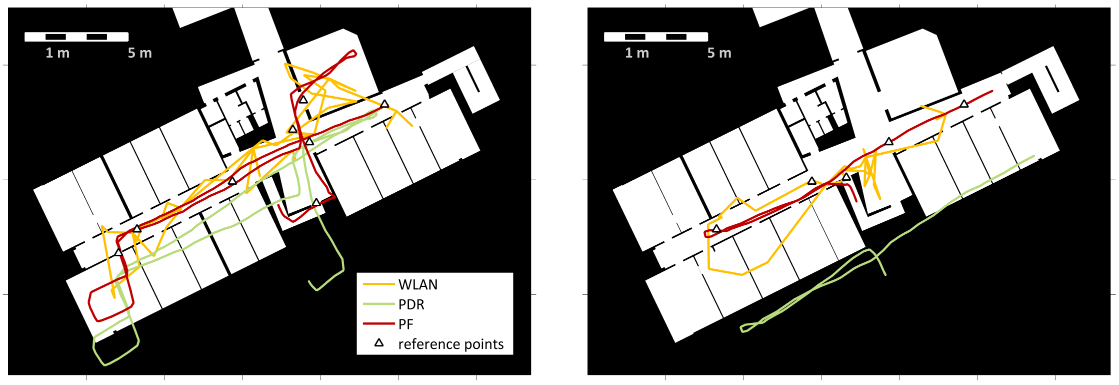


(a) Height estimation based on WLAN and barometer data.

(b) Height estimation based on WLAN and PAC/PDR data.

Figure 6.28: Results of the height estimation and the floor level detection.

Based on the provision of the correct floor map, the particle filter provides the estimated trajectory, which is visualized in Figure 6.29. The trajectory passes two floor levels, starting in floor level 4 and ending in floor level 3. As the position residuals in Figure 6.30 show, the maximum of the horizontal position residuals, which is 7.74 m for the WLAN trajec-

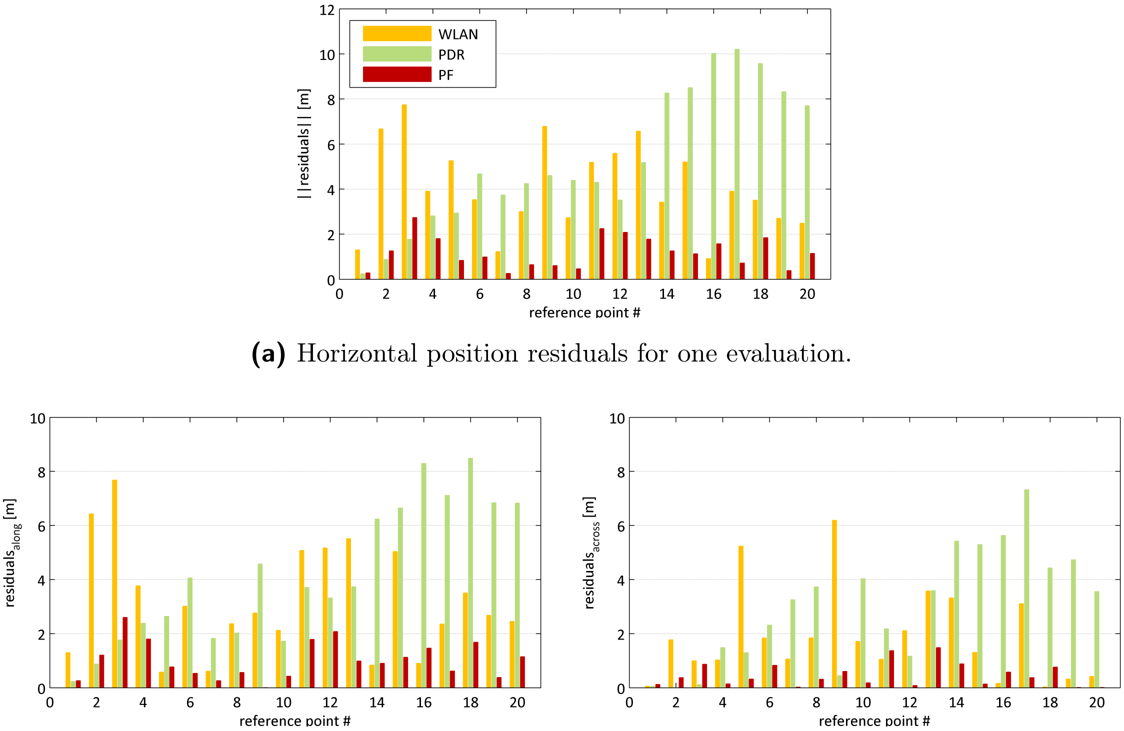


(a) Floor level 4.

(b) Floor level 3.

Figure 6.29: Evaluation result for one walk along trajectory 1.

tory and 10.21 m for the PDR trajectory, can be reduced to 2.74 m. The maximum of the along track residuals is 7.68 m, 8.49 m, and 2.60 m for the WLAN, PDR and PF trajectory, while the maximum of the across track residuals is 6.19 m, 7.32 m, and 1.48 m for the three trajectories. Therefore, the most improvement can again be gained in the across track component.



(a) Horizontal position residuals for one evaluation.

(b) Along track residuals for one evaluation.

(c) Across track residuals for one evaluation.

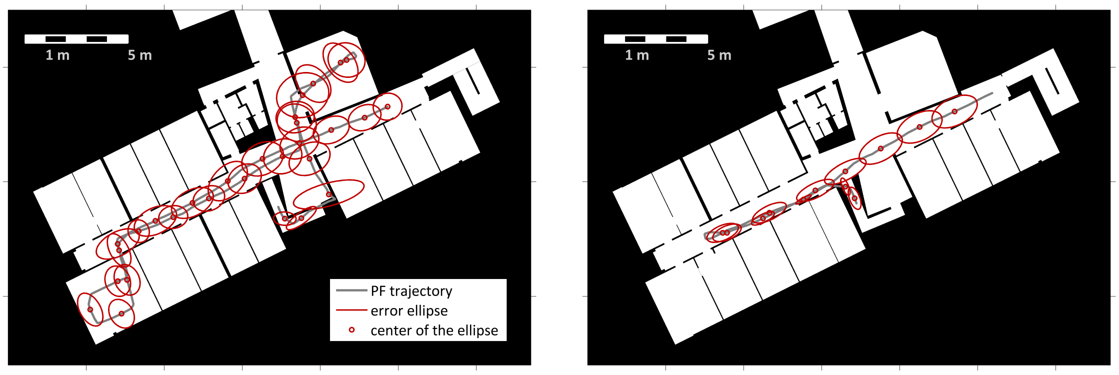
Figure 6.30: Mean errors for the selected trajectory 1.

Table 6.11 contains the mean values of the position residuals over the whole trajectory. The evaluation by the indoor positioning filter improves $\mu_{\Delta x_{horiz}}$ by about 70%, $\mu_{\Delta x_{along}}$ by about 68%, and $\mu_{\Delta x_{across}}$ by about 75%.

Table 6.11: Mean errors for one selected trajectory.

	$\mu_{\Delta x_{horiz}}$ [m]	$\mu_{\Delta x_{along}}$ [m]	$\mu_{\Delta x_{across}}$ [m]
WLAN	4.08	3.21	1.86
PDR	5.29	4.17	3.00
PF	1.20	1.03	0.47

In Figure 6.31, the error ellipses of every fifth epoch of the weighted mean trajectory are visualized. For the error ellipses, a Gaussian distribution is assumed (see section 6.1.2). Furthermore, the sample covariance based on weighted samples is adopted, as calculated in equation (6.4). The ellipse parameters are determined according to equations (6.14) to (6.18). Their scale corresponds to the scale of the map.



(a) Floor level 4.

(b) Floor level 3.

Figure 6.31: Visualization of the error ellipses for every fifth second along the PF trajectory.

Based on the size and orientation of the error ellipses, the impact of the wall collision filter and the activity-based probabilities can be revealed. The walls along the corridors act as spatial restriction while the pedestrian is walking straight. The orientation of the ellipse changes with the walking direction. If no constraints influence the position estimation, the ellipse starts increasing (see floor level 4, where the pedestrian enters the second room). While entering the room, the particle distribution gets restricted by the dimension of the door and therefore, the dimension of the ellipse gets smaller. Before leaving floor level 4 down the stairs, the ellipse has a large east/west extension. This is due to the untypical distribution of the particles. Some of the particles survived in the room next to the corridor in west direction and therefore influence the shape of the ellipse. However, by entering the stairs, the dimension of the ellipse is diminished drastically due to the restrictions provided by the detected activity of *walking downstairs* and the wall collision filter. In floor level 3 the influence of the WLAN and PDR data (see Figure 6.29) can be observed, which cause a drift to the south of the corridor.

6.3.6 Conclusion - performance in indoor environments

Within this section, the performance of the indoor positioning has been investigated in detail. The height estimation allows the detection of the correct floor independently of the input data for the Kalman filter. The fusion of h_{WLAN} and Δh_{baro} as well as the combination of h_{WLAN} and Δh_{PAC} are both possible solutions for the height estimation within a positioning tool being part of a smartphone-based navigation system. The investigations revealed that the PAC-based height can be taken as alternative for the barometric height if the smartphone has no internal barometer. However, the PAC-based height estimation is concentrated on stairs, which have a standardized step height, and only supports a vertical displacement of one stair per step.

The investigations concerning the particle filter performance revealed that the computation break-off quota can be reduced to zero if the particle filter processing is done with at least 500 particles. The processing with 1500 particles still provides a mean computation time of less than 0.05 s for 1 epoch, but guarantees that the position residuals converge to a minimum.

Furthermore, the positive impact of the activity integration on the position estimation could be shown and the weighted mean of the particles has been identified as best representation of the evaluated trajectory. The PF evaluations have demonstrated that the horizontal position residuals can be reduced by about 40% to 60% due to the processing with the designed RBPF for indoor environments. All of the presented evaluations provide a position accuracy which fulfills the requirements of a navigation system for blind pedestrians.

7 Simultaneous localization and mapping of the blind pedestrian's path

For the application of the designed Rao-Blackwell particle filters, maps of the outdoor as well as the indoor environment have to be available. In case of indoor environments, although architectural blue prints of most of the buildings exist, only a few buildings (mainly public buildings) are mapped with respect to the navigation and positioning requirements. On the contrary, the outdoor environment in developed countries is almost mapped completely. Common navigation systems are widely incorporating map information for map-matching the estimated state, calculating the optimal route to a destination, and to visually guide the user through the environment. However, user-tailored maps, as the one described in section 5.1.1, are not available and their development comes along with considerable work and consequently costs. As a consequence, the user-tailored raster map was only created for a limited test area, and therefore, the developed outdoor RPPF can only be applied within the test environment at the campus of TU Graz. However, the performance investigations in section 6.2 demonstrate that the integration of a probability map, representing the environment and the pathway preferences of visually impaired pedestrians, strongly improves the accuracy as well as the reliability of the state estimation. Therefore, within the thesis, the applicability of a Simultaneous Localization and Mapping (SLAM) approach has been investigated for the purpose of generating a probability map, which more or less corresponds to the probability map resulting from the user-tailored raster map.

As described in section 5.1.1, the probability map integrated within the outdoor RBPF represents, on the one hand, the occupancy of the grid cells and, on the other hand, the location probability with regard to a blind pedestrian. The goal of the SLAM is the direct derivation of the probabilities with respect to the walking and navigation habits of blind pedestrians instead of mapping the features of the environment (pavements, tactile ground surface indicators, etc.). Details on the designed SLAM and the resulting map are given in section 7.2, while the theoretical background and different existing SLAM approaches are presented in section 7.1. The incorporation of the SLAM-based probability map within the outdoor RBPF and the corresponding results are discussed in section 7.3. Finally, the applicability of the

SLAM for the derivation of a probability map, which represents the blind pedestrian's path preferences in indoor environments, is briefly investigated in section 7.4.

7.1 Simultaneous localization and mapping

Referring to Thrun et al. (2006), SLAM combines the probabilistic location estimation with the basic ideas of *occupancy grid mapping*, which refers to the generation of a consistent occupancy grid map from noisy and uncertain measurement data. Within the occupancy grid mapping, the evenly spaced grid, which is defined over the continuous space of locations, is assigned with occupancy values at locations which are covered by observations. Based on the defined grid, the posterior probability over the map $\boldsymbol{m}$ is computed conditioned on the current observations $\boldsymbol{z}_t$ and the current state $\boldsymbol{x}_t$, and is given by

$$p(\boldsymbol{m}|\boldsymbol{z}_t, \boldsymbol{x}_t). \tag{7.1}$$

The grid map $\boldsymbol{m}$ defines the space by finitely many grid cells $m_{\varphi,\lambda}$, leading to $\boldsymbol{m} = \{m_{\varphi,\lambda}\}$. Consequently, if a region is covered by observations, an appropriate occupancy value can be assigned to the corresponding grid cell $m_{\varphi,\lambda}$. The transition data $\boldsymbol{u}_t$ is neglected within the posterior over the map $\boldsymbol{m}$, since the state estimate $\boldsymbol{x}_t$ implicitly involves $\boldsymbol{u}_t$. Figure 7.1 shows the DBN model of the occupancy grid mapping, in which the map $\boldsymbol{m}$ is determined based on the observations $\boldsymbol{z}_{1:t}$ and the states $\boldsymbol{x}_{1:t}$. While $\boldsymbol{z}_t$ and $\boldsymbol{x}_t$ are time variant, the map $\boldsymbol{m}$ is time independent. In contrast to the DBN model of the designed particle filter illustrated in Figure 5.8, the map $\boldsymbol{m}$ is the object of interest instead of the state $\boldsymbol{x}_t$.

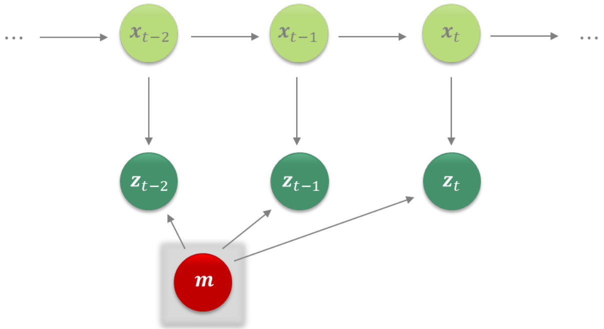


Figure 7.1: DBN of the occupancy grid mapping, where the map $\boldsymbol{m}$ is the object of interest.

SLAM can thus be formulated as extension of the occupancy grid mapping. As stated in Thrun et al. (2006), the goal of the SLAM is to estimate the map $\boldsymbol{m}$ and to localize the pedestrian relative to that map in parallel. Therefore, SLAM focuses on the simultaneous computation of the map $\boldsymbol{m}$ and the state $\boldsymbol{x}_t$. Referring to Thrun et al. (2006), the posterior with regard to the idea of SLAM is given by

$$p(\boldsymbol{x}_t, \boldsymbol{m}|\boldsymbol{z}_t, \boldsymbol{u}_t). \tag{7.2}$$

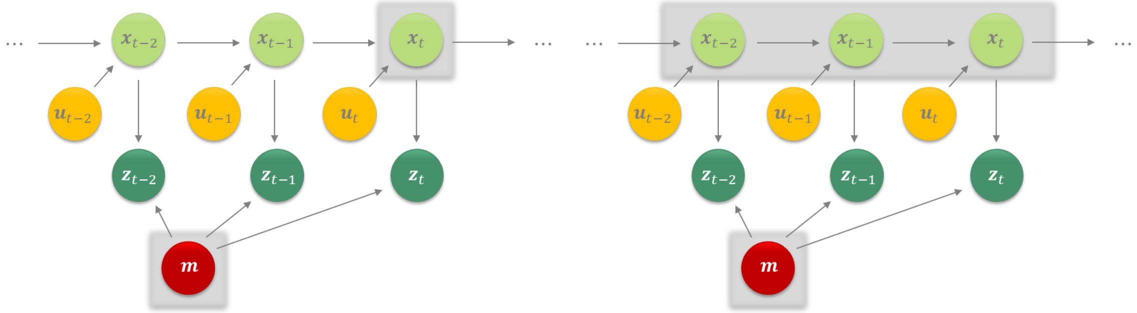
In contrast to the posterior of the occupancy grid mapping given in equation (7.1), the posterior in equation (7.2) is conditioned on the observations $\boldsymbol{z}_t$ as well as the control data $\boldsymbol{u}_t$, while $\boldsymbol{x}_t$ and $\boldsymbol{m}$ are the objects of interest. Equation (7.2) refers to the *online SLAM problem*, which only involves variables of the current epoch. The *full SLAM problem* determines a posterior over the entire measurement epochs up to time t

$$p(\boldsymbol{x}_{1:t}, \boldsymbol{m}|\boldsymbol{z}_{1:t}, \boldsymbol{u}_{1:t}). \tag{7.3}$$

The connection between the online and the full SLAM problem is given by

$$p(\boldsymbol{x}_t, \boldsymbol{m}|\boldsymbol{z}_t, \boldsymbol{u}_t) = \int\int \dots \int p(\boldsymbol{x}_{1:t}, \boldsymbol{m}|\boldsymbol{z}_{1:t}, \boldsymbol{u}_{1:t})\, d\boldsymbol{x}_1\, d\boldsymbol{x}_2\, \dots\, d\boldsymbol{x}_{t-1}, \tag{7.4}$$

in which the past poses are integrated out from the full SLAM to get the posterior of the online SLAM (Thrun et al., 2006). Within the thesis, the online SLAM problem is applied for the derivation of the SLAM-based probability map. Figure 7.2 illustrates the DBN of the online and the full SLAM problem, in which the elements to be estimated are marked in gray. The online SLAM focuses on the estimation of the current state $\boldsymbol{x}_t$ based on the observation and transition data of the actual epoch, while the full SLAM estimates the posterior over all states $\boldsymbol{x}_{1:t}$ based on all available input data at once.



(a) DBN model of the online SLAM, in which the map $\boldsymbol{m}$ and the current state $\boldsymbol{x}_t$ are the objects of interest.

(b) DBN model of the offline SLAM, in which the map $\boldsymbol{m}$ and the states $\boldsymbol{x}_{1:t}$ are the objects of interest.

Figure 7.2: DBN models for different SLAM implementations.

SLAM methods can be mainly classified into two categories depending on the underlying Bayes filter implementations. While the *EKF-SLAM* employs an extended Kalman filter for estimating the pose and building the map, the *FastSLAM* does the same on the basis of a Rao-Blackwell particle filter. The Rao-Blackwell method can be formulated by a factorization of the joint PDF $p(\boldsymbol{x}_t, \boldsymbol{m}|\boldsymbol{z}_t, \boldsymbol{u}_t)$. This joint PDF, which represents the posterior over the state $\boldsymbol{x}_t$ and the map $\boldsymbol{m}$, can be written as

$$p(\boldsymbol{x}_t, \boldsymbol{m}|\boldsymbol{z}_t, \boldsymbol{u}_t) = p(\boldsymbol{x}_t|\boldsymbol{z}_t, \boldsymbol{u}_t)\, p(\boldsymbol{m}|\boldsymbol{x}_t, \boldsymbol{z}_t), \tag{7.5}$$

leading to two separate problems, which can be solved within a sequential RBPF. The posterior $p(\boldsymbol{x}_t|\boldsymbol{z}_t, \boldsymbol{u}_t)$ is determined within the particle filter, while the map's posterior $p(\boldsymbol{m}|\boldsymbol{x}_t, \boldsymbol{z}_t)$ is estimated in parallel within an EKF, resulting in a joint PDF at the end.

Several SLAM particle approximations have been defined and implemented by names like Sebastian Thrun, Dieter Fox, Wolfram Burgard, Patrick Robertson, Michael Angermann, etc. In the following, the characteristics of these SLAM approaches are highlighted. The SLAM approaches of Thrun, Fox, and Burgard are mainly focused on the application within robotics, where robots are exploring the environment within buildings. The robot's odometry data is typically used for the particle propagation, while feature-detection sensors like cameras or laser scanners are involved for the retrieval of feature-based maps. An EKF-SLAM realization based on the robot's odometry and laser scan data is the *GraphSLAM* (Thrun et al., 2006). This EKF-SLAM was originally developed for robots moving in indoor environments. The approach has been expanded for the outdoor application, in which GPS information acts as additional observation (Thrun and Montemerlo, 2005). The *GraphSLAM* realized as a RBPF is described in Montemerlo et al. (2002).

In contrast, Robertson and Angermann process inertial measurements within a sequential RBPF based on the typical *FastSLAM* factorization to derive the structures within a building. *FootSLAM*, developed by Robertson, Angermann, and others (Robertson et al., 2009, 2010b), maps the unknown building environment by using the observations of the inertial sensors, which are attached to the shoes of the pedestrian walking around in the building of interest. The sensors measure the pedestrian's step while walking. An extended version of *FootSLAM* is *PlaceSLAM*, in which the humans are treated as sensors reporting the presence of recognizable features or places. As stated in Robertson et al. (2010a), the accuracy and stability of the *FootSLAM* can be significantly increased by integrating human-reported measurements. The combination of several cooperative *FootSLAM* is *FeetSLAM*, in which the data of various walks are combined (Robertson et al., 2011). In *WiSLAM*, the *FootSLAM* is extended for the generation of a map containing the position of WLAN access points (Bruno and Robertson, 2011). Therefore, WLAN RSS measurements from unknown access points are fused with odometric data derived from the foot mounted IMU. Similar localization and mapping approaches based on signal-strength measurements are realized by Huang et al. (2011) and Ferris et al. (2007).

Another SLAM implementation is the *SmartActionSLAM*, which is an Android smartphone application integrating the motion sensors of the smartphone and an optional foot-mounted

IMU to track a person inside of a building (Hardegger et al., 2013). Additional to the inertial data, which is preprocessed within a PDR approach, activity information like *standing* and *sitting* is incorporated within the SLAM. The SLAM focuses on the generation of a local landmark map for the monitoring of patients suffering from Parkinson.

7.2 Tailored design of the SLAM

The SLAM developed within the thesis derives a location-based map, which represents the occupancy of a grid cell $m_{\varphi,\lambda}$ and the location probability with respect to the path preferences of blind pedestrians. The SLAM is realized by a RBPF implementation. In contrast to the other mentioned SLAM methods, the implemented SLAM focuses on the derivation of location probabilities based on an already existing occupancy grid map. This means, the occupancy grid map is enriched by location probabilities, which are correlated with the walking and navigation habits of the blind pedestrians, instead of generating a map from scratch. Missing obstacles are implicitly included in the derived probabilities and therefore, the SLAM also improves the occupancy information. The idea is to take advantage of existing map information and to enrich this map by information which represents on the one hand, the spatial accessibility to blind pedestrians and, on the other hand, the path preferences of this special user group.

The path of the pedestrian is influenced by physical constraints like walls, obstacles, etc., and by the pedestrian's goal, such as reaching a destination. However, only noisy measurements are available to estimate the position and build the map. The areas accessible for blind pedestrians are implicitly derived by observing the motion and the current location of the person. The great advantage of the SLAM-based generation of the probability map is that individual habits and path preferences can be considered. The probability map can be either derived from measurement data of one individual person or can be realized as cooperative approach integrating measurement data of several blind pedestrians.

As discussed in Thrun et al. (2006), for the SLAM-based map generation different options may be considered.

- Which sensors are suitable for the generation of the SLAM-based map? Sensors or respectively sensor combinations, which deliver unambiguous position solutions, have to be favored over sensors leading to ambiguous solutions due to the faster convergence of the SLAM.

- Are the sensors accurate enough to map the real-world environment? The larger the noise of the sensors, the more difficult it is to generate a reliable map efficiently.

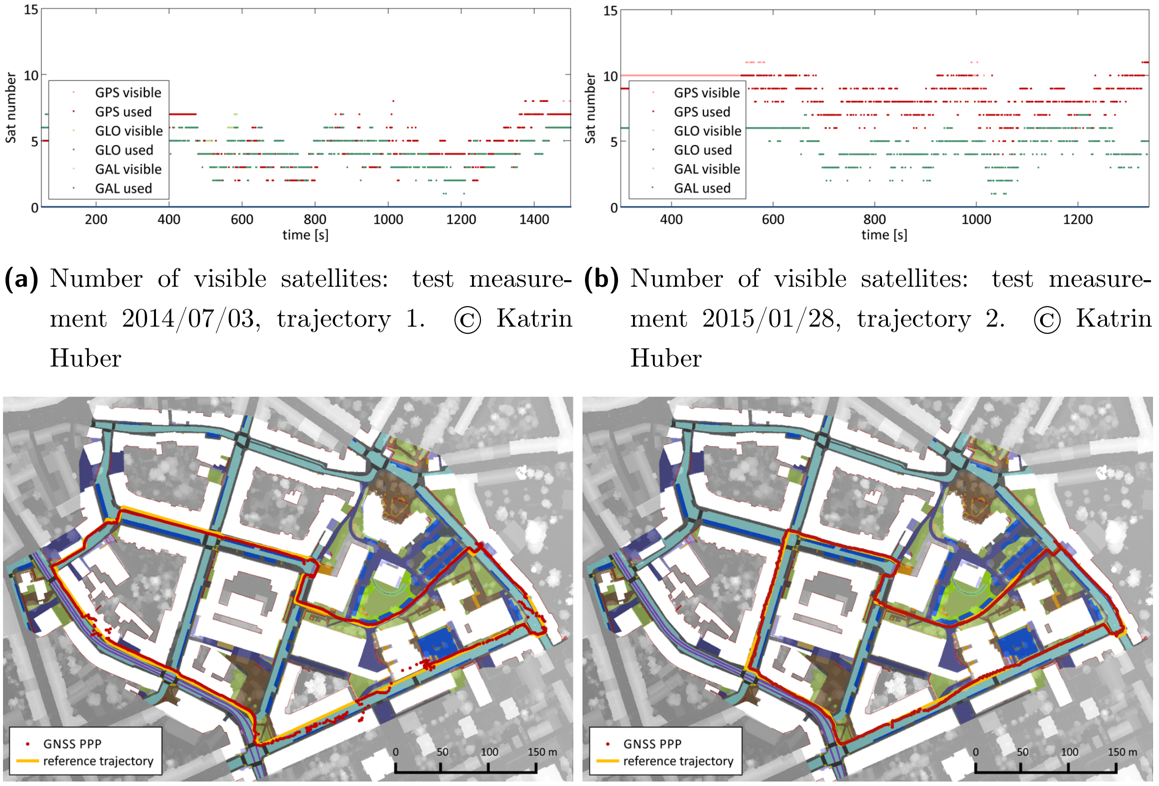
- Is the environment of interest sufficiently covered by measurements? The more often a certain region is passed, the more reliable the resulting SLAM-based map is.
- Which spatial extension should the grid covered within the SLAM have? Does the map have to be split into n separate mapping problems $\boldsymbol{m} = \{\boldsymbol{m}_{\text{part }1}, \boldsymbol{m}_{\text{part }2} \ldots \boldsymbol{m}_{\text{part }n}\}$?

These questions have to be kept in mind, while designing a SLAM approach for a specific application. In the following, the first three factors for the SLAM design are discussed with regard to the present case. The last of the above options is neglected within the investigations as for a first trial only the surrounding of the campus has been treated.

As mentioned in section 4.1.1, the accuracy of code-based GNSS solutions is in the range of some meters. For the mapping of the path preferences of blind pedestrians, which represent the presence of pavements, zebra crossings, tactile surface indicators, etc., the accuracy requirements are much higher. Therefore, the applicability of PPP for the measurement update within the SLAM is investigated. PPP is favored over differential and relative positioning, because of the fact that the PPP products are globally valid and easily accessible through online services (see section 4.1.4). The accuracy in case of kinematic applications is in the range of some decimeters if the ambiguities can be solved appropriately. Therefore, a continuous tracking of at least four satellites of the same satellite system is required. However, multipath and signal obstructions degrade the accuracy in urban environments.

Therefore, several test measurements have been performed in the surrounding of the campus of TU Graz to proof the applicability of multiple-GNSS PPP. The PPP evaluation revealed that in principle, PPP is suitable for the SLAM-based probability map generation, but is limited to restricted day times due to the extensive signal obstructions in urban environment. For the test measurements, the time span had to be planned in advance for an appropriate GNSS satellite constellation. The automatic generation of the probability map requires sufficient satellites in urban areas 24 hours a day, which cannot be guaranteed before Galileo and COMPASS reach their FOC status. However, several Galileo and COMPASS satellites will be launched in the near future, improving the situation drastically.

As common smartphones do not provide code and phase measurements, an external GNSS receiver and antenna had to be employed to record the appropriate GNSS observations. The Javad SIGMA, offering the possibility of recording GPS as well as GLONASS code and phase data of several frequencies, was chosen as receiver for the test measurements. The precise orbit and clock products are obtained from the IGS (International GNSS Service, 2014a). The evaluation of the GNSS data is done with the PPP software *PPPsoft* developed at the Institute of Geodesy, Working Group Navigation (Huber, 2015). Two evaluated PPP trajectories are exemplarily illustrated in Figure 7.3. Figure 7.3a and 7.3b show the number of theoretically visible satellites in bright colors compared to the actual measured satellites in darker colors. The lower two Figures 7.3c and 7.3d visualize the determined PPP trajectories against the background of a surface model and of information which is relevant for blind pedestrians (zebra crossings, pavements, etc.). In Figure 7.3, a PPP trajectory based on a good GNSS constellation (Figure 7.3d) is opposed to a PPP trajectory based on a bad GNSS constellation (Figure 7.3c).



(a) Number of visible satellites: test measurement 2014/07/03, trajectory 1. © Katrin Huber

(b) Number of visible satellites: test measurement 2015/01/28, trajectory 2. © Katrin Huber

(c) PPP evaluation of the test measurement: 2014/07/03, trajectory 1.

(d) PPP evaluation of the test measurement: 2015/01/28, trajectory 2.

Figure 7.3: Comparison of the PPP performance based on a bad (left) and a good GNSS constellation (right). The number of theoretically visible satellites compared to the number of actually observed satellites are given in (a) and (b), while the resulting PPP trajectories are illustrated in (c) and (d).

In Figure 7.3a, the number of visible GNSS satellites is between 2 and 8, resulting in a trajectory, which makes the mapping of the blind pedestrian's path preferences impossible in several areas (Figure 7.3c). Due to the environmental structures in urban areas, the number of visible satellites is strongly fluctuating and therefore, the ambiguities have to be resolved during the walk leading to positions with accuracies in the range of some meters in phases, where the ambiguities are unknown. In contrast, the GNSS constellation shown in Figure 7.3b leads to an estimated trajectory which nearly allows a mapping without any additional support from PDR data or an occupancy grid map (Figure 7.3d). Figure 7.3 demonstrates that PPP provides position accuracies, which enable the localization of the pedestrian at pavements and zebra crossings if the GNSS constellation is correspondingly good. In case of small GNSS outages and outliers, the PDR data in cooperation with the filter facilitates a reliable and continuous positioning.

However, due to absence of enough satellites during the whole day, the implemented SLAM is evaluated with simulated measurement data. GNSS PPP as well as PDR data are simulated with appropriate accuracies to enable a realistic performance evaluation of the SLAM. The existing framework for the outdoor positioning (section 5.1) was the basis for the development of the SLAM. The established positioning software has been expanded for the simultaneous mapping. The Kalman filter estimating the height is the same as presented in section 5.1.2, only the particle filter defined in section 5.1.3 had to be adapted. The adapted PF concept is visualized in Figure 7.4.

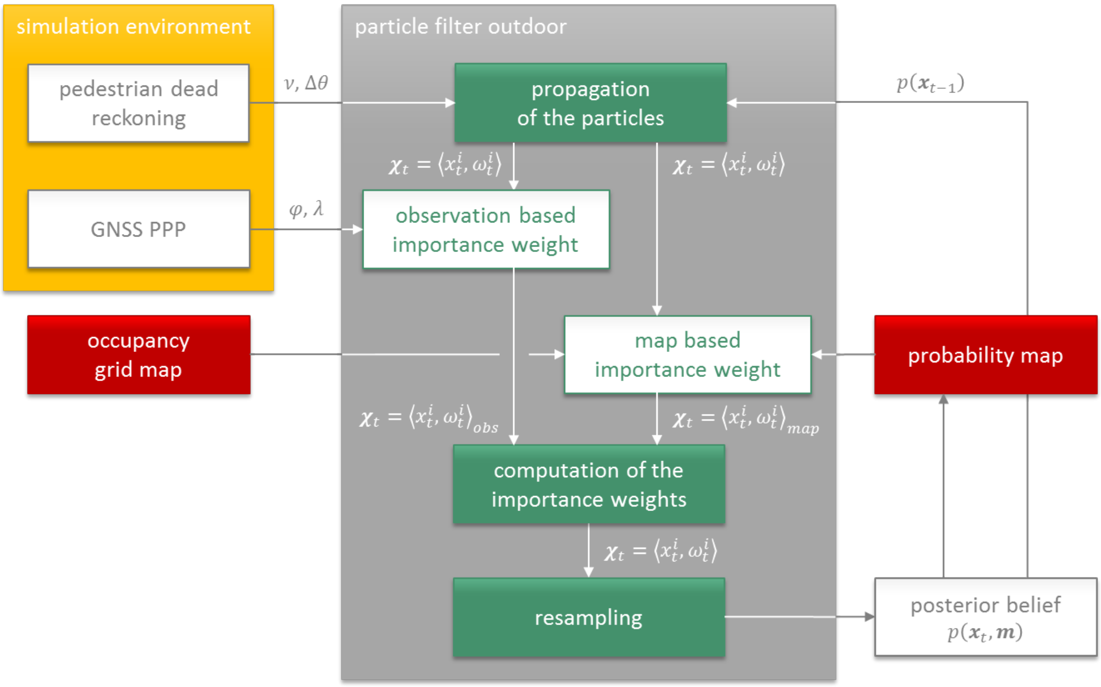

simulation environment
pedestrian dead reckoning
GNSS PPP
occupancy grid map
particle filter outdoor
v, $\Delta\theta$
φ, λ
propagation of the particles
$p(\boldsymbol{x}_{t-1})$
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
observation based importance weight
map based importance weight
probability map
$\chi_t = \langle x_t^i, \omega_t^i \rangle_{obs}$
$\chi_t = \langle x_t^i, \omega_t^i \rangle_{map}$
computation of the importance weights
$\chi_t = \langle x_t^i, \omega_t^i \rangle$
resampling
posterior belief
$p(\boldsymbol{x}_t, \boldsymbol{m})$

Figure 7.4: RBPF for the localization and simultaneous mapping of the pedestrian's path.

In contrast to the outdoor concept presented in section 5.1.3, the filter incorporates an occupancy grid map (Figure 7.5) which is iteratively updated by the posterior belief $p(\boldsymbol{x}_t, \boldsymbol{m})$ to finally provide the probability map representing the occupancy as well as the pedestrian's path preferences. The designed SLAM approach determines which areas are accessible to a blind pedestrian, whereby only areas which are spatially covered by measurements can be updated within the map. The more an area is explored, the more complete the resulting map is. The final map is a probabilistic representation of the blind pedestrian's motion derived from the posteriors over multiple epochs and trajectories. After some iterations a reliable probability map is provided.

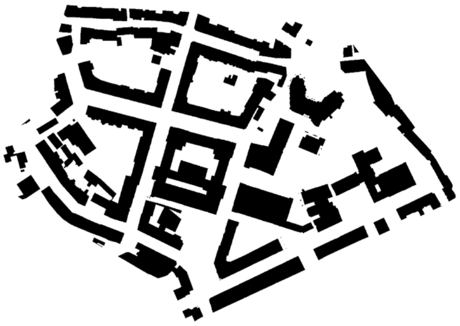


Figure 7.5: Occupancy grid map representing the buildings and obstacles of the environment.

As illustrated in Figure 7.4, the posterior belief $p\left(\boldsymbol{x}_t, \boldsymbol{m}\right)$ already acts as input for the consecutive epoch. The posterior of the state $\boldsymbol{x}_t$ is the basis for the propagation of the particles within the time update, while the derived probability map delivers an appropriate importance weight $\omega_{map}{}_t^i$ if the corresponding area is already covered by the pedestrian. Otherwise, $\omega_{map}{}_t^i$ is based on the occupancy grid map.

Based on the SLAM concept illustrated in Figure 7.4, the prediction and update step can be identified. The final goal is to determine the joint PDF of the state $\boldsymbol{x}_t$ and the map $\boldsymbol{m}$ given in equation (7.5). The estimation of a prior belief $\overline{bel}(\boldsymbol{x}_t, \boldsymbol{m}_t)$ is done within the prediction step, which is corresponding to equation (3.5) defined by

$$\begin{aligned}\overline{bel}(\boldsymbol{x}_t, \boldsymbol{m}_t) &= p(\boldsymbol{x}_t, \boldsymbol{m}_t | \boldsymbol{z}_{t-1}, \boldsymbol{u}_t) \\ &= \sum_{N_s} p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}, \boldsymbol{u}_t)\, bel(\boldsymbol{x}_{t-1}, \boldsymbol{m}_{t-1})\, d\boldsymbol{x}_{t-1}.\end{aligned} \tag{7.6}$$

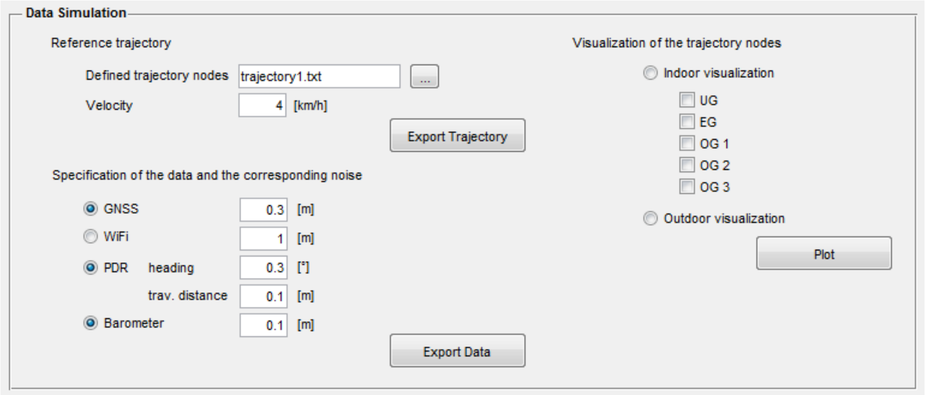
In contrast to equation (3.5), the belief is realized as discrete approximation and is used for the estimation of $\boldsymbol{x}_t$ and $\boldsymbol{m}_t$. The temporal update given in equation (7.6) incorporates the dynamic model $p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}, \boldsymbol{u}_t)$ based on the corresponding transition data $\boldsymbol{u}_t$.

The posterior belief is estimated within the measurement update according to equation (3.6). Besides the current measurements $\boldsymbol{z}_t$, the prior belief acts as input for the measurement update, which is given by

$$\begin{aligned}bel(\boldsymbol{x}_t, \boldsymbol{m}_t) &= p(\boldsymbol{x}_t, \boldsymbol{m}_t | \boldsymbol{z}_t) \\ &= \eta_t\, p(\boldsymbol{z}_t | \boldsymbol{x}_t, \boldsymbol{m}_t)\, \overline{bel}(\boldsymbol{x}_t, \boldsymbol{m}_t).\end{aligned} \tag{7.7}$$

The normalization factor η_t corresponds to $p(\boldsymbol{z}_t | \boldsymbol{z}_{t-1})$ and guarantees that the summation of the posterior belief over the whole state space results in 1. In contrast to equation (3.6), the measurement likelihood $p(\boldsymbol{z}_t | \boldsymbol{x}_t, \boldsymbol{m}_t)$ depends on $\boldsymbol{x}_t$ and $\boldsymbol{m}_t$.

For the derivation of the SLAM-based probability map, ten different trajectories have been defined by i coordinate triples $[\varphi,\, \lambda,\, h]_i^T$ along the trajectory. Based on these points, measurement data with different characteristics can be simulated (see Figure 7.6). The simulated trajectories can vary in the assumed velocity and the measurement uncertainties. Path variations due to the dynamic environment are covered by the measurement uncertainties of the simulated data. Problems caused by signal obstructions and multipath are not treated within the simulation, since the investigation focuses on the general ability of a SLAM approach for deriving location probabilities.


Data Simulation

Reference trajectory

Defined trajectory nodes: trajectory1.txt

Velocity: 4 [km/h]

Specification of the data and the corresponding noise

● GNSS 0.3 [m]

○ WiFi 1 [m]

● PDR heading 0.3 [°]

trav. distance 0.1 [m]

● Barometer 0.1 [m]

Visualization of the trajectory nodes

○ Indoor visualization

□ UG

□ EG

□ OG 1

□ OG 2

□ OG 3

○ Outdoor visualization

Figure 7.6: Environment for the simulation of GNSS PPP and PDR data.

For the simulation of the measurement data, a reference trajectory $\boldsymbol{x}_{ref} = \{\boldsymbol{x}_{ref_t}\}$ with $\boldsymbol{x}_{ref_t} = [\varphi,\, \lambda,\, h]_t^T$ is generated based on a defined velocity and a measurement frequency of 1 Hz. This reference trajectory acts as basis for the simulation of GNSS and PDR data. The GNSS PPP data $\boldsymbol{x}_{PPP}$ is simulated by

$$\boldsymbol{x}_{PPP} = \boldsymbol{x}_{ref} + \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}_{PPP}), \qquad \boldsymbol{\Sigma}_{PPP} = \begin{bmatrix} \sigma_{\varphi}{}_{PPP}^{2} & 0 & 0 \\ 0 & \sigma_{\lambda}{}_{PPP}^{2} & 0 \\ 0 & 0 & \sigma_{h}{}_{PPP}^{2} \end{bmatrix}, \tag{7.8}$$

whereby $\sigma_{\varphi PPP}$, $\sigma_{\lambda PPP}$, and $\sigma_{h PPP}$ correspond to the GNSS measurement uncertainty defined within the simulation environment (Figure 7.6). The investigations presented in the following are based on $\sigma_{PPP} = 0.3\,\text{m}$ for the GNSS PPP solution.

The PDR data comprising the heading θ_t and the velocity v_t have to be extracted from the coordinate differences in north and east direction, which are obtained by

$$\Delta n_{t-1,t} = (\varphi_t - \varphi_{t-1})\ R \tag{7.9}$$

$$\Delta e_{t-1,t} = (\lambda_t - \lambda_{t-1})\ R \cos \varphi_0 \tag{7.10}$$

with R representing the Earth radius and φ_0 the mean latitude of the test area. The heading θ is now given by

$$\theta_t = \arctan \frac{\Delta e_{t-1,t}}{\Delta n_{t-1,t}} + \mathcal{N}(0, \sigma_\theta), \tag{7.11}$$

with σ_θ being the heading's measurement uncertainty. The velocity v_t is computed by

$$v_t = \frac{\sqrt{\Delta n_{t-1,t}^2 + \Delta e_{t-1,t}^2}}{\Delta t} + \mathcal{N}(0, \sigma_v), \tag{7.12}$$

with Δt corresponding to the measurement update rate of 1 s and σ_v being the measurement uncertainty of the velocity. Within the simulation environment (Figure 7.6), σ_v as well as σ_θ can be defined, whereby following values $\sigma_\theta = 0.3°$ and $\sigma_v = 0.1\,m/s$ have been chosen due to the PDR evaluation results.

Based on the simulated PPP and PDR data, the SLAM is performed in an iterative manner as it is illustrated in Figure 7.4. The probability map is consecutively improved and updated within the SLAM. As soon as SLAM-based map probabilities are derived, the particle filter incorporates this information within the state estimation.

7.3 Outdoor positioning incorporating the SLAM-based probability map

Based on the simulated measurement data, the map is generated by the implemented SLAM algorithm, while the pedestrian is tracked simultaneously. For each of the ten reference trajectories (most left figure in Figure 7.7), five measurement data sets have been simulated, yielding fifty simulated trajectories. Figure 7.7 shows the iterative mapping process of the SLAM.

The SLAM-based probability map is visualized after every fifth trajectory, whereby regions which are covered by measurements are updated by appropriate probabilities. The more often a region is passed, the higher the probability is. The final SLAM-based probability map is illustrated in Figure 7.8, with the occupancy grid map (uniform light gray) and a digital surface model in the background. Besides the PPP and PDR data, the occupancy grid map has acted as basis for the SLAM-driven generation of the probability map. The SLAM has been performed with $N_s = 1500$ particles. In Figure 7.8, only areas which are covered by measurements have appropriate SLAM-based probabilities. The darker the red, the more often the grid cell $m_{\varphi,\lambda}$ is passed. After several SLAM iterations a reliable probability map of the environment is generated.

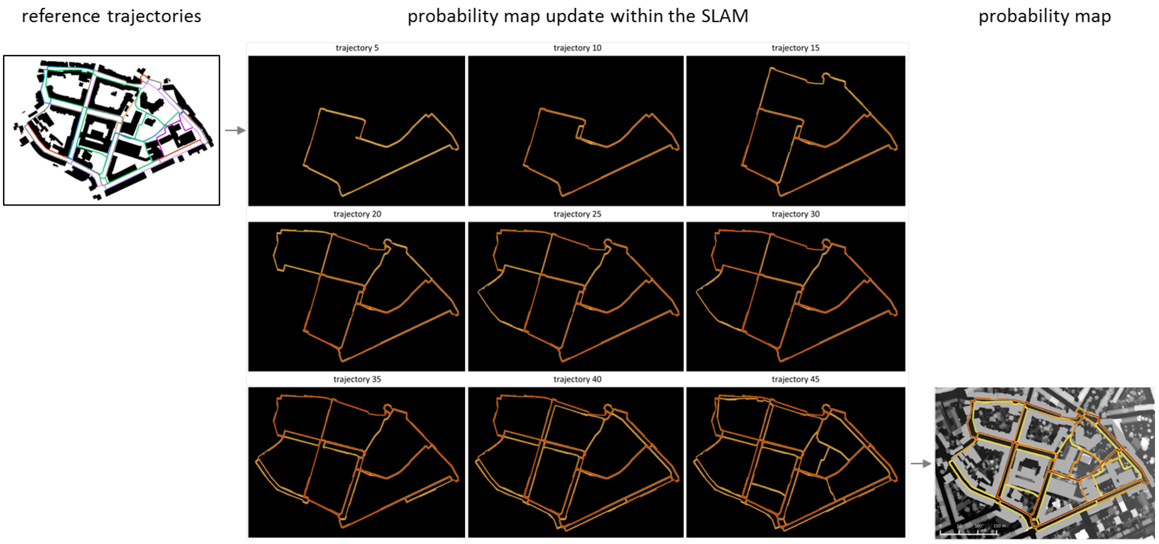


Figure 7.7: Iterative update of the probability map within the SLAM. The defined reference trajectories are the basis for the simulation of the measurement data.

The number of required particles have been investigated with regard to the computation time and the accuracy of the result. The computation time for each of the 50 trajectories has been determined for different numbers of particles $N_s = \{50, 100, 200, 500, 1000, 1500, 2000, 4000\}$. For an appropriate investigation of the computation time, the computation times of the trajectories are normalized to 1000 epochs. The mean computation time for 1000 epochs as a function of the number of particles N_s is shown in Figure 7.9a. The numerical values for the evaluation of one epoch are given in Table 7.1. As Figure 7.9a and Table 7.1 demonstrate, the mean computation time t is increasing with a polynomial of degree 2, which can be

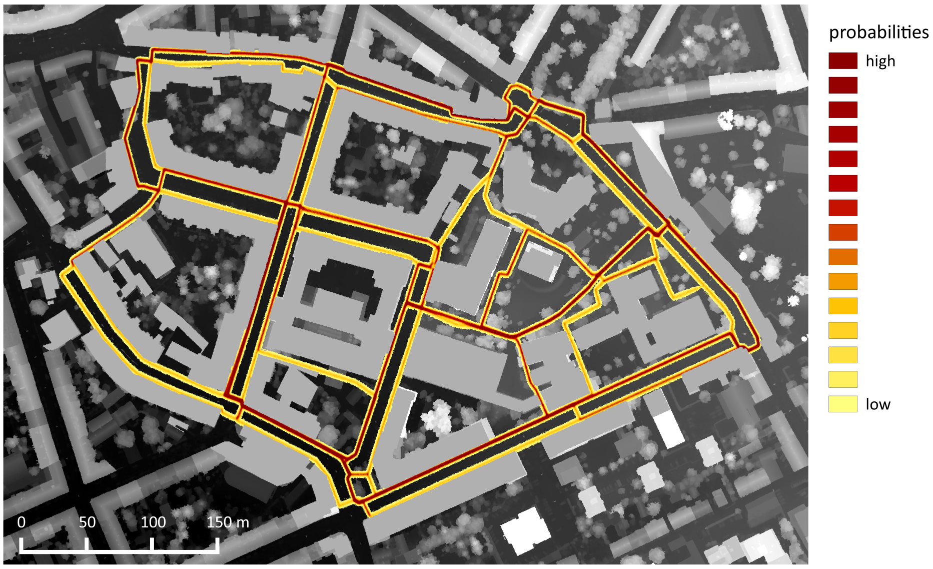


Figure 7.8: Probability map generated by the SLAM.

approximated by $t = a_0 + a_1 \cdot N_s + a_2 \cdot {N_s}^2$, whereby a_0, a_1, and a_2 are the coefficients of the polynomial.

Table 7.1: Mean computation time for one particle filter iteration (one epoch, $\Delta t = 1\,s$) for different number of particles.

N_s	50	100	200	500	1000	1500	2000	4000
time [s]	0.019	0.021	0.022	0.032	0.051	0.076	0.104	0.253

Provided that the input data of the filter is simulated, the horizontal position residuals of the SLAM trajectory can be determined by computing the differences between the SLAM position $\boldsymbol{x}_{SLAM} = [\varphi_{SLAM}, \lambda_{SLAM}]_t^T$ and the reference trajectories $\boldsymbol{x}_{ref_t} = [\varphi_{ref}, \lambda_{ref}]_t^T$. The horizontal position residuals are given by

$$\Delta x_{horiz} = \sqrt{\Delta x_n^2 + \Delta x_e^2}, \tag{7.13}$$

with

$$\begin{aligned} \Delta x_n &= (\varphi_{SLAM} - \varphi_{ref})\, R \\ \Delta x_e &= (\lambda_{SLAM} - \lambda_{ref})\, R \cos \varphi_0. \end{aligned}$$

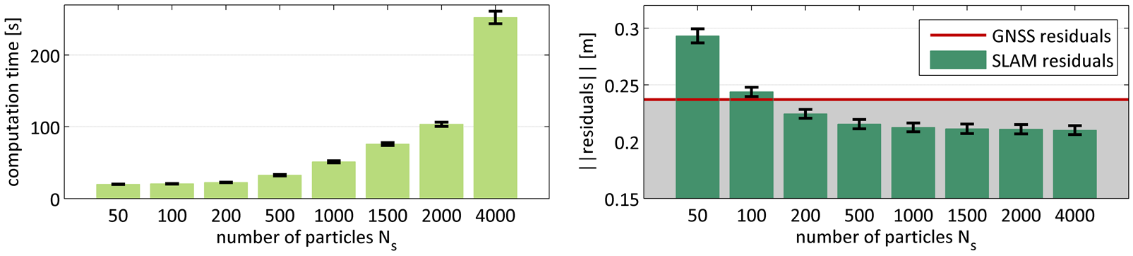
The terms Δx_n and Δx_e correspond to the position residuals in north and east direction, while $\boldsymbol{x}_{SLAM}$ is given by equation (6.4) representing the weighted mean of the particle set $\chi_t = \langle \boldsymbol{x}_t^i, \omega_t^i \rangle$. The numerical values for the mean horizontal position residuals $\mu_{\Delta x_{horiz}}$ over all epochs and trajectories are given in Table 7.2, together with the corresponding standard deviation $\sigma_{\Delta x_{horiz}}$.

Table 7.2: Mean horizontal position residuals and the corresponding standard deviations for different numbers of particles .

N_s	$\mu_{\Delta x_{horiz}}$ [cm]	$\sigma_{\Delta x_{horiz}}$ [cm]
50	29.3	0.62
100	24.4	0.41
200	22.4	0.39
500	21.5	0.41
1000	21.3	0.38
1500	21.1	0.42
2000	21.1	0.41
4000	21.0	0.40

The mean value $\mu_{\Delta x_{horiz}}$ is decreasing with an increasing number of particles N_s and converges to a number of 21.0 cm. The improvement of the position solution from $N_s = 1000$ to $N_s =$

4000 is marginal. The standard deviation $\sigma_{\Delta x_{horiz}}$ of the horizontal residuals is decreasing from $N_s = 50$ to $N_s = 100$, after that no significant differences can be observed. The convergence of the mean horizontal position residual $\mu_{\Delta x_{horiz}}$ can also be seen in Figure 7.9b, where the mean residuals are drawn as a function of N_s. Additionally to the SLAM residuals, the mean horizontal position residual of the GNSS PPP solution ($\mu_{\Delta x_{horiz}} = 0.237$ m) is indicated by the red line. If more than 200 particles are used, the SLAM result shows lower residuals than the GNSS PPP solution.



(a) Mean computation times of the SLAM normalized to a trajectory length of 1000 s as a function of N_s.

(b) Mean position residuals of the SLAM as a function of N_s compared to the mean position residual of the GNSS PPP solution.

Figure 7.9: Investigation of the SLAM performance with regard to the computation time and the position residuals.

As Figure 7.9 shows, the computation time increases with an increasing N_s, while the position residuals converge to a minimum of about 21.0 cm. Although the computation effort for the evaluation with 4000 particles is nearly five times higher as the evaluation with 1000 particles, $\mu_{\Delta x_{horiz}}$ improves only by 0.3 cm. In contrast, the increment of N_s from 50 to 1000 improves the positioning performance by 8 cm from $\mu_{\Delta x_{horiz}}(N_s = 50) = 29.3$ cm to $\mu_{\Delta x_{horiz}}(N_s = 1000) = 21.3$ cm.

As for each of the ten defined reference trajectories five different measurement data sets are simulated, the variations in the horizontal position residuals after each trajectory evaluation have been investigated. Within the SLAM evaluation, the probability map in each iteration is hereby taken from the previous epoch. The mean residuals have been computed for the 1^{st}, 2^{nd}, 3^{rd}, 4^{th}, and 5^{th} iteration using the residuals of all ten simulated trajectories with varying $N_s = \{50, 100, 200, 500, 1000, 1500, 2000\}$. The corresponding horizontal residuals are illustrated in Figure 7.10. Although an improvement of the residuals has been expected from one iteration to the next, the investigations have revealed that the variations are arbitrary and, consequently, are independent of the current iteration. The reason therefore is that an absolute offset in the GNSS PPP position solution cannot be reduced by the SLAM if the offset is located in an unoccupied region. The particle filter only enables a smoothing by the incorporation of the PDR data and the corresponding dynamic model.

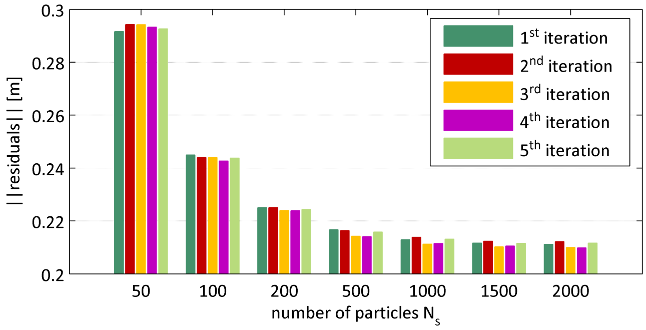


Figure 7.10: Mean position residuals of all trajectories after each iteration as a function of the number of particles N_s.

7.3.1 Application of the SLAM-based map within the RBPF

In a next step, the SLAM-based probability map is integrated instead of the user-tailored probability map within the particle filter processing based on the GNSS SPP solution. Thereby, the substitutability of the user-tailored probability map by the SLAM-based map is proofed. Figure 7.11 shows the results of one selected outdoor trajectory.

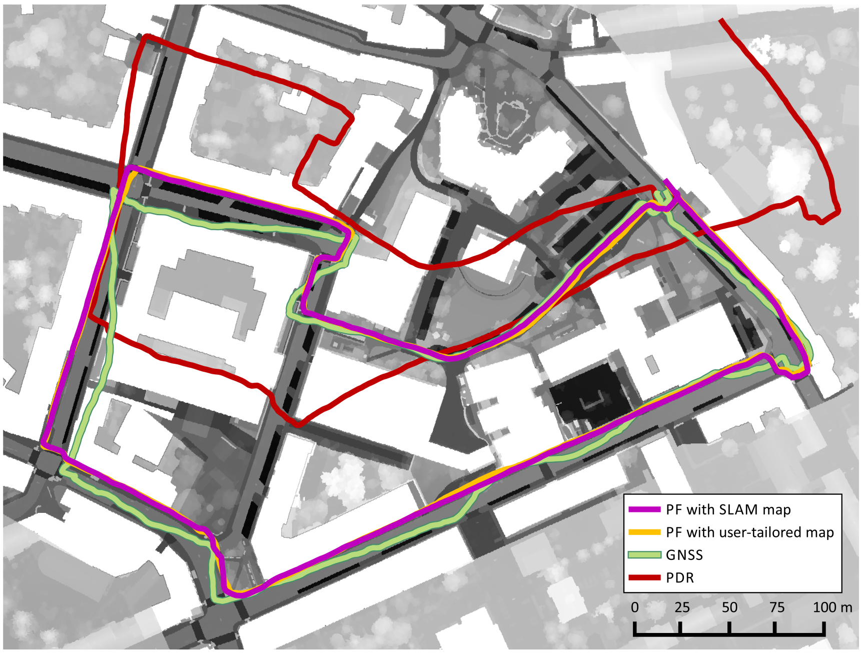
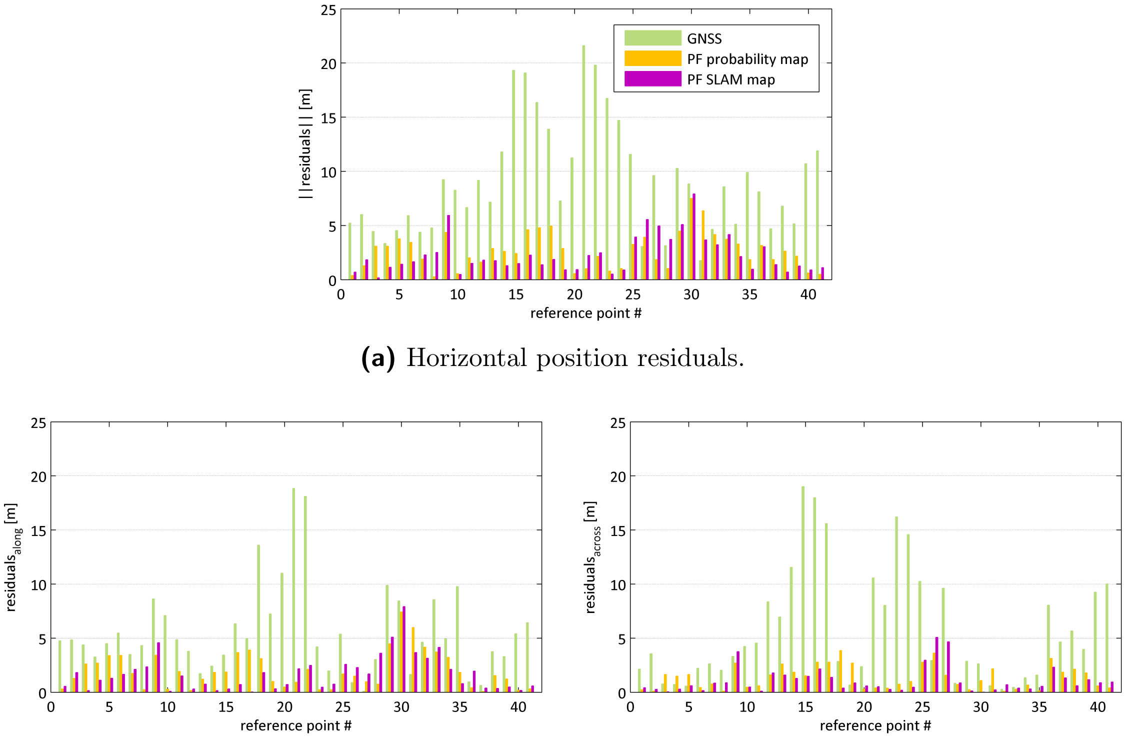


Figure 7.11: Comparison of the evaluation results of the particle filter integrating the user-tailored probability map and the SLAM-based probability map.

The trajectory is processed tow times, one time with the user-tailored probability map and another time with the derived SLAM-based probability map. The evaluation demonstrates that the SLAM-based probability map and user-tailored map lead to similar results. The corresponding position residuals of the illustrated trajectories are visualized in Figure 7.12.



(a) Horizontal position residuals.

(b) Along track residuals.

(c) Across track residuals.

Figure 7.12: Comparison of the horizontal, along track and across track residuals.

The amplitudes of the residuals of the PF integrating the SLAM-based probability map are a bit lower, but show that the accuracy is more or less the same as the one of the PF based on the user-tailored map, see also Table 7.3. In Table 7.3, the mean values of the horizontal, along track, and across track position residuals are given for the GNSS and PDR trajectory

Table 7.3: Mean of the horizontal, along track, and across track position residuals.

	$\mu_{\Delta x_{horiz}}$ [m]	$\mu_{\Delta x_{along}}$ [m]	$\mu_{\Delta x_{across}}$ [m]
GNSS	9.14	5.64	5.77
PDR	69.23	38.16	50.56
PF user-tailored map	2.66	1.99	1.37
PF SLAM-based map	2.28	1.69	1.07

as well as for the two different particle filter realizations.

Finally, it can be suspected, that the SLAM-based probability map could replace the user-tailored probability map in case that PPP would be applicable 24 hours a day. If GNSS phase observations would be available on smartphones, an individual SLAM-based probability map could be easily generated. The advantage of the SLAM-based probability map generation compared to the approach described in section 5.1.1, in which the raster map tailored to blind people acts as basis for the derivation of the probability map, is the possibility to adjust for environmental changes by rating current data higher than outdated data. Moreover, one may account for individual preferences by generating the map with data from a single person. Otherwise, the cooperative approach, incorporating data of various blind pedestrians, at least improves the spatial and temporal coverage of the map area.

7.4 Aspects of the application of SLAM in indoor environments

In indoor environments, it is impossible to generate a user-tailored raster map valid for each blind pedestrian due to the small-scaled environment and the very individual navigation habits of visually impaired people within buildings. The generation of the user-tailored raster map is very time- and cost-intensive and therefore cannot be tailored to one individual person. However, without any pre-knowledge, the SLAM offers a possibility to extract a probability map for indoor environments. As the probability map can be generated automatically, probability maps for individual blind persons could be realized.

In section 7.3, the applicability of the developed SLAM for the generation of a probability map representing the path preferences of blind pedestrians has been demonstrated for the outdoor environment. The combination of the GNSS PPP data and the PDR-based motion data is accurate enough to build a map representing relevant structures like pavements, zebra crossings, footpath, etc. For the realization of a SLAM within buildings, a position estimation with accuracies comparable to the PPP and PDR combination has to be provided for a reliable mapping of the areas accessible to blind pedestrians, since the environment is characterized by objects and areas of even smaller size.

Consequently, the combination of the fingerprinting-based WLAN positions with the PDR motion data does not suffice for the SLAM-driven mapping of the accessible and preferred areas of blind pedestrians. Other possibilities exist for positioning in indoor environments (Mautz, 2012), however, the use of a smartphone reduces the number of possible options. In general, fingerprinting-based Bluetooth positioning is an alternative to the WLAN positioning but requires, due to the lower signal range, a more dense network of access points.

As demonstrated in Cinefra (2013) and Faragher and Harle (2014), the accuracy of Bluetooth Low Energy is in the best case in the range of 1.5 to 2 meters. In comparison, the fingerprinting-based WLAN positioning provides accuracies in the range of 2.5 to 3 meters. The Bluetooth Low Energy signal range is about 10 m, which is two times lower than the WLAN signal range of about 30 m. Therefore, a denser network is needed for a continuous tracking. Nevertheless, the positioning accuracy of Bluetooth does not fulfill the requirements of the SLAM-driven probability map generation. Consequently, up to now, no smartphone-based positioning technology enables a SLAM of the probability map representing the path preferences in indoor environments.

8 Conclusion

8.1 Summary

The purpose of the thesis was the development of a positioning tool being the basis for a navigation system tailored to the needs of blind pedestrians. Therefore, in a first instance, the requirements related to the positioning of visually impaired people in terms of accuracy, reliability, and availability have been determined. A comprehensive market analysis with regard to navigation and assistance systems for blind persons was accomplished. Additionally, the contemporary research related to the navigation of visually impaired people has been discussed. Based on the market and research analysis, unsolved problems and possible improvements have been identified. In the design phase of the thesis, these aspects have been taken under consideration for the development of a positioning concept.

The investigations identified a smartphone as an affordable tool for visually impaired people. The comprehensive possibilities of a smartphone in terms of positioning, computing, and communication have been chosen to be the basis for the developments. Hereby, all possible smartphone sensors are used to have redundant information and to compensate the disadvantages of one sensor by the advantages of the others. The redundant and complementary data strongly contribute to the improvement of the availability, reliability, but also accuracy of the position estimation.

Then, the goal of the thesis was to exploit all possibilities of filtering to provide, outdoors as well as indoors, a position solution which fulfills the accuracy requirements of a navigation system for blind pedestrians. Therefore, the output of the smartphone-internal GNSS receiver and the WiFi scanner are used besides the sensor data of the accelerometer, gyroscope, magnetometer, and barometer to derive absolute and relative position information. As the available sensor data and the environmental structures are different for the outdoor and indoor environments, two filter approaches have been developed. While outside of the building, the the GNSS unit delivers absolute positions based on a single point positioning algorithm, a RSSI-based WLAN-fingerprinting technique supports the filter in indoor environments by providing 3D coordinates. Outdoors as well as indoors, a PDR algorithm determines the actual velocity and the change of the heading. Furthermore, the barometer gives information about the change in height. Details on the different preprocessing techniques have been given in the course of the thesis.

The particle filter is chosen as core element for the position estimation due to the fact, that this filter is optimal for the integration of multimodal distributions and non-linear motion and observation models under the aspect of estimating the state of a dynamic system in an infinite space. However, to reduce the computational load of the state estimation, the outdoor as well as the indoor filter is realized by a Rao-Blackwell particle filter concept. Thereby, parts of the unknown state are determined by a parametric Bayes filter. In case of the designed RBPF, the horizontal coordinates are estimated within the particle filter, while the actual height is the output of a Kalman filter.

Besides the preprocessed measurement data, the developed RBPFs integrate spatial information about the environment. The outdoor positioning is supported by a probability map which represents the path preferences of visually impaired pedestrians and the accessibility of the outdoor space. Due to the very individual navigation habits of blind persons in indoor environments and the small-scaled structures within a building, no equivalent probability map can be generated for the indoor use. Instead, an occupancy grid map and a wall collision filter facilitate the filter in its positioning task. For the provision of the correct occupancy grid map, a floor detection algorithm is installed within the filter. Furthermore, the activity gained from the PAC algorithm provides location probabilities. For example, in case of the detected activities *walking upstairs* or *walking downstairs*, the possible state space can be restricted to stairways.

The positioning performance of each of the two RBPF approaches has been investigated in detail on a PC with an Intel i7 desktop CPU with a processor speed of 3.2 GHz and 12 GB RAM using MATLAB. The raw sensor data of several walks along different trajectories in outdoor and indoor environments have been recorded and evaluated in a post-processing mode. As an external check for the ground truth, it has been defined that the pedestrian has to pass several reference points with known coordinates along the trajectory.

The outcome of the filters has been analyzed in terms of necessary number of particles, computation time and processing break-off quota. Furthermore, for the identification of a suitable state representation, four different processing strategies (particle with highest weight, weighted mean, center of the cluster with the highest accumulated weight, center of the cluster with the highest particle concentration) have been investigated. Based on the optimal number of particles and an appropriate representative of the state estimate, the impact of the map and activity integration on the position solution has been shown. Moreover, the position residuals in along track and across track direction, as well as the error ellipses of the position solutions are presented.

The investigations for the position estimation in outdoor environments revealed that at least 500 particles are necessary for the particle filter processing if the convergence of the position errors towards a minimum should be guaranteed. Less particles may come along with a much worse position accuracy. The processing of the particle filter based on 500 particles requires 0.01 s for one epoch. Furthermore, the weighted mean was identified as best state representative due to the minimum residuals gained by this approach.

The mean horizontal position residual of the whole trajectory is, in the best case, 1.87 m and, in the worst case, 4.42 m. However, the maximum of the mean across track residual is 1.65 m, which allows a continuous localization of the pedestrian at the correct pavement. The mean along track residual has a maximum of 3.58 m, which has to be taken into account for the announcement of guidance instructions. Finally, it can be stated, that the combined GNSS/PDR trajectory solution can be improved by about 47% to 69% if the user-tailored map is integrated within the PF processing. In comparison, a common outdoor map leads only to an improvement of only 11% to 20%.

To circumvent the labor-intensive generation of the user-tailored map, one of the research questions of the thesis is, whether the appropriate location probabilities can be automatically derived by a SLAM approach. However, due to the insufficient position accuracy of the GNSS code solution, GNSS PPP has to be employed for the SLAM-based map generation. As GNSS phase data, which is required for the PPP processing, is not accessible on smartphones, the SLAM was realized within a simulation environment. The investigations revealed, that SLAM is qualified to provide a probability map which represents the path preferences of blind pedestrians. For the evaluation of the applicability of the SLAM map, the user-tailored probability map was substituted by the SLAM-based map within the particle filter. The evaluation showed, that both maps lead to comparable results.

In indoor environments, the performance of the particle filter strongly depends on the use of the correct floor map. Therefore, the height estimation and the consecutive floor detection have been proofed with regard to their efficiency. The investigation revealed that the fusion of h_{WLAN} and Δh_{baro} as well as the combination of h_{WLAN} and Δh_{PAC} allow the identification of the correct floor level. Consequently, Δh_{PAC} can be taken as alternative for Δh_{baro} if the smartphone has no internal barometer. However, the PAC-based height estimation is limited to stairs which have a standardized step height and only supports a vertical displacement of one stair per walked step.

Since indoors the wall collision filter can cause processing break-offs, this effect has been inspected in detail. The computation break-off quota can be reduced to zero if the PF processing is done with at least 500 particles. 1500 particles result in a mean computation time of 0.05 s for 1 epoch, while the position residuals converge to a minimum. Furthermore, the weighted mean is again identified as best state representative. The integration of the activity, the map, and the wall collision filter has a positive impact on the position accuracy. The horizontal errors can be reduced by about 40% to 60% due to the processing with the designed RBPF. Especially at stairways and at room entrances, the position residuals can be strongly reduced. Summarizing the indoor results, all presented evaluations provide an accuracy which fulfills the requirements of a navigation system for blind pedestrians.

In general, the position residuals are smaller in indoor environments. This is due to the better accuracy of the WLAN fingerprinting and the small-scaled environmental structures within buildings. However, both positioning approaches strongly improve the original WLAN and GNSS accuracies. In summary, the two developed RBPFs estimate the 3D coordinates with an across track accuracy in the range of a blindman's stick. Indoors, also the accuracy of the along track component fulfills this requirement. The along track residuals of the outdoor solution may be higher than the sensing range of a blind pedestrian, but can be handled by an earlier announcement of guidance instructions and obstacle warnings.

8.2 Outlook

In a future step, the developed algorithms have to be adjusted for the application on a smartphone. The developed framework has served as tool for the investigation of the best performing procedures. During the development and implementation phase, the real-time character of the RBPFs has always been kept in mind. Nevertheless, the processing power of a smartphone is different from that of a desktop computer and therefore, the bottleneck for the application of the developed positioning filters on a smartphone is most likely its hardware performance.

Within this thesis, the potential of the SLAM-based map generation based on GNSS PPP has been demonstrated. However, up to now, the raw GNSS sensor data is not available on smartphones. If the GNSS phase measurements would be provided by smartphones, on the one hand, the SLAM-based map may be generated directly on the smartphone and, on the other hand, the performance of the outdoor positioning could be improved due to the better accuracies of the PPP solution.

As the indoor positioning depends on the availability of a WLAN network and an established radio map, the application of the indoor positioning approach is restricted to only a few buildings. However, the radio map may be also built on the basis of a SLAM approach. By collecting the data of several pedestrians, an appropriate radio map could be provided. Radu and Marina (2013) for example, use a crowdsourcing approach for the establishment of a WLAN fingerprinting data base. Thereby, motion and map information is used for the positioning and the update of the radio map.

List of Figures

List of Tables

Bibliography

Aichhorn K., Kuci D., Sommer C., Koglbauer I., Hafner P., Moder T., Valeskini S. S., Bernoulli T., Walder U., Witrisal K. and Heimo K. (2013). *LOBSTER Final Report*. Technical Report, Graz University of Technology.

American Foundation for the Blind (2009). Navigating by Phone: A Review of Wayfinder Access GPS and Mobile Geo. *AFB AccessWorld Magazine - Technology News for People Who Are Blind or Visually Impaired* 10(3).

Apostolopoulos I., Fallah N., Folmer E. and Bekris K. (2012). Integrated Online Localization and Navigation for People with Visual Impairments Using Smart Phones. In *Proceedings of IEEE International Conference on Robotics and Automation (ICRA), 2012* (pp. 1322 – 1329).

Apostolopoulos I., Fallah N., Folmer E. and Bekris K. E. (2014). Integrated Online Localization and Navigation for People with Visual Impairments Using Smart Phones. *ACM Transactions on Interactive Intelligent Systems* 3(4), 21:1 – 21:28.

Apple Inc. (2015). *VoiceOver für iOS. Unsere Stimme hilft dir weiter*. URL: `https://www.apple.com/at/accessibility/ios/voiceover/` accessed on 2015-01-20.

Arulampalam M., Maskell S., Gordon N. and Clapp T. (2002). A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing* 50(2), 174 – 188.

Bauer M. (2011). *Vermessung und Ortung mit Satelliten*. (6th ed.). Wichmann.

Bensky A. (2008). *Wireless Positioning - Technologies and Applications*. Artech House.

BEV - Bundesamt für Eich- und Vermessungswesen (2015). *APOS - Austrian Positioning Service*. URL: `http://www.bev.gv.at/portal/page?_pageid=713,1571538&_dad=portal&_schema=PORTAL` accessed on 2015-01-14.

Bevermeier M., Walter O., Peschke S. and Haeb-Umbach R. (2010). Barometric height estimation combined with map-matching in a loosely-coupled Kalman-filter. In *Proceedings of the 7th Workshop on Positioning Navigation and Communication (WPNC), 2010*.

Blinden- und Sehbehindertenverband Österreich (2007). *Wie viele Menschen mit Sehbehinderungen gibt es in Österreich?*. URL: `http://www.blindenverband.at/home/wissen/sehen/977?help=1` accessed on 2015-01-26.

BlindSquare (2012). *What is BlindSquare*. URL: `{http://blindsquare.com/}` accessed on 2015-01-20.

Bronstein I. N. and Semendjajew K. A. (1981). *Taschenbuch der Mathematik*. Teubner Verlagsgesellschaft.

Brown R. G. and Hwang P. Y. C. (1997). *Introduction to Random Signals and Applied Kalman Filtering*. John Wiley & Sons, Inc., New York.

Bruno L. and Robertson P. (2011). WiSLAM: Improving FootSLAM with WiFi. In *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2011* (pp. 1 – 10).

Burgard W., Fox D., Hennig D. and Schmidt T. (1996). Position tracking with position probability grids. In *Proceedings of the First Euromicro Workshop on Advanced Mobile Robot*.

Chen R. (2012a). Introduction to Smart Phone Positioning. In *Ubiquitous Positioning and Mobile Location-Based Services in Smart Phones*. IGI Global.

Chen R. (2012b). *Ubiquitous Positioning and Mobile Location-Based Services in Smart Phones*. IGI Global.

Chen R., Pei L. and Chen Y. (2011). A Smart Phone Based PDR Solution for Indoor Navigation. In *Proceedings of the 24th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS 2011)*.

Chen R., Pei L., Liu J. and Leppäkoski H. (2012). WLAN and Bluetooth Positioning in Smart Phones. In R. Chen (Ed.), *Ubiquitous Positioning and Mobile Location-Based Services in Smart Phones*. IGI Global.

China Satellite Navigation Office (2013). *BeiDou Navigation Satellite System Signal In Space Interface Control Document*. URL: http://www.beidou.gov.cn/attach/2013/12/26/20131226b8a6182fa73a4ab3a5f107f762283712.pdf accessed on 2014-12-25.

Chumkamon S., Tuvaphanthaphiphat P. and Keeratiwintakorn P. (2008). A blind navigation system using RFID for indoor environments. In *Proceedings of the 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology, 2008 (ECTI-CON 2008)* (pp. 765 – 768).

Cinefra N. (2013). *An Adaptive IIndoor Positioning System based on Bluetooth Low Energy RSSI*. Master's thesis, Politecnico di Milano.

Ding B., Yuan H., Jiang L. and Zang X. (2007). The Research on Blind Navigation System Based on RFID. In *Proceedings of the International Conference on Wireless Communications, Networking and Mobile Computing, 2007 (WiCom 2007)* (pp. 2058 – 2061).

Douc R., Cappé O. and Moulines E. (2005). Comparison of Resampling Schemes for Particle Filtering. In *Proceedings of the 4th International Symposium on Image and Signal Processing and Analysis (ISPA), ISPA 2005* (pp. 64 – 69).

Doucet A., de Freitas N. and Gordon N. (2010). *Sequential Monte Carlo Methods in Practice*. Springer-Verlag New York.

Ekahau (2015). *Real-Time Location System*. URL: http://www.ekahau.com/ accessed on 2015-01-20.

Eltis - the urban mobility observatory (2008). *NOPPA - Navigation and Guidance System for the Blind in Finland*. URL: http://www.eltis.org/discover/case-studies/noppa-navigation-and-guidance-system-blind-finland accessed on 2015-01-19.

EPOSA (2015). *EPOSA - Echtzeit Positionierung Austria*. URL: http://www.eposa.at/ accessed on 2015-01-14.

ESA - Technology Transfer Programme (2003). *More Autonomy for Blind People thanks to Satellite Navigation.* URL: `http://www.esa.int/Our_Activities/Space_Engineering_Technology/TTP2/More_autonomy_for_blind_people_thanks_to_satellite_navigation` accessed on 2015-01-19.

European GNSS Agency (2014). *EGNOS portal.* URL: `http://egnos-portal.gsa.europa.eu` accessed on 2014-12-25.

European Space Agency (2012). *Galileo fact sheet.* URL: `http://download.esa.int/docs/Galileo_IOV_Launch/Galileo_factsheet_2012.pdf` accessed on 2014-12-25.

European Space Agency (2014a). *ESA navipedia.* URL: `http://www.navipedia.net` accessed on 2014-12-25.

European Space Agency (2014b). *Galileo Satellites - Status Update.* URL: `http://www.esa.int/Our_Activities/Navigation/Galileo_satellites_status_update` accessed on 2014-12-25.

Fallah N., Apostolopoulos I., Bekris K. and Folmer E. (2013). Indoor Human Navigation Systems: A Survey. *Interacting with Computers* 25(1), 21 – 33.

Faragher R. and Harle R. (2014). An Analysis of the Accuracy of Bluetooth Low Energy for Indoor Positioning Applications. In *Proceedings of the 27th International Technical Meeting of The Satellite Division of the Institute of Navigation (ION GNSS+ 2014).*

Ferris B., Fox D. and Lawrence N. (2007). WiFi-SLAM Using Gaussian Process Latent Variable Model. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2007.*

Figueiras J. and Frattasi S. (2010). *Mobile Positioning and Tracking - From Conventional to Cooperative Techniques.* John Wiley & Sons.

Fischer C., Muthukrishnan K., Hazas M. and Gellersen H. (2008). Ultrasound-aided Pedestrian Dead Reckoning for Indoor Navigation. In *Proceedings of the first ACM international workshop on mobile entity localization and tracking in GPS-less environments, Co-located MOBICOM 2008* (pp. 31 – 36).

Fox D., Hightower J., Liao L., Schulz D. and Borriello G. (2003). Bayesian Filtering for Location Estimation. *Pervasive Computing, IEEE* 2(3), 24 – 33. doi:`10.1109/MPRV.2003.1228524`.

Frank K., Nadales M. J. V., Robertson P. and Angermann M. (2010). Reliable Real-Time Recognition of Motion Related Human Activities using MEMS Inertial Sensors. In *Proceedings of the 23rd International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2010)* (pp. 2919 – 2932).

Ganz A., Schafer J., Gandhi S. and Puleo E. (2012). PERCEPT Indoor Navigation System for the Blind and Visually Impaired: Architecture and Experimentation. *International Journal of Telemedicine and Applications* 2012, 12 pages.

GPS World (2015). *Upcoming GNSS Satellite Launches.* URL: `http://gpsworld.com/resources/upcoming-gnss-satellite-launches/` accessed on 2015-01-14.

Grewal M. S. and Andrews A. P. (2008). *Kalman Filtering - Theory and Practice Using MATLAB.* (3rd ed.). John Wiley & Sons, Inc., Hoboken, New Jersey.

Groves P. D. (2013). *Principles of GNSS, Inertial, and Multisensor integrated Navigation Systems.*

(2nd ed.). Artech House.

Hafner P., Huber K., Moder T., Wieser M., Hollinger G. and Strauß C. (2014a). Development of a Positioning Tool for the Navigation of Visually Impaired People. In *Proceedings of the 27th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2014)* (pp. 1867 – 1878).

Hafner P., Huber K., Wieser M., Hinterberger F., Weber R., Hollinger G., Strauß C. and Geiger A. (2015). *POScity Final Report: New Positioning Approach for Visually Impaired People in Urban Areas*. Technical Report, POScity consortium.

Hafner P., Moder T., Bernoulli T. and Fösleitner C. (2014b). Projekt Lobster - Entwicklung von Positionierungstools zur Unterstützung von Ersthelfern. *AVN - Allgemeine Vermessungsnachrichten* 2014(6), 387 – 398.

Haibo Y., Tao G., Xiaorui Z., Jinwei X., Xianping T., Jian L. and Ning J. (2012). FTrack: Infrastructure-free floor localization via mobile phone sensing. In *Proceedings of IEEE International Conference on Pervasive Computing and Communications (PerCom) 2012* (pp. 2 – 10).

Hall M., Frank E., Holmes G., Pfahringer B., Reutemann P. and Witten I. H. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations* 11(1), 10 – 18.

Hardegger M., Mazilu S., Caraci D., Hess F., Roggen D. and Tröster G. (2013). ActionSLAM on a Smartphone: At-Home Tracking with a Fully Wearable System. In *Prooceedings of the 4th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2013)* (p. pages 8).

Hartigan J. A. (1975). *Clustering Algorithms*. John Wiley & Sons.

Haug A. J. (2012). *Bayesian Estimation and Tracking*. John Wiley & Sons, Inc., Hoboken, New Jersey.

HERE (2015). *HERE. Maps for Life.*. URL: `https://www.here.com/business/?lang=en-GB` accessed on 2015-02-17.

Hoepcke W. (1980). *Fehlerlehre und Ausgleichsrechnung*. de Gruyter, Berlin, New York.

Hofmann-Wellenhof B., Legat K. and Wieser M. (2003). *Navigation - principles of positioning and guidance*. Springer-Verlag Wien New York.

Hofmann-Wellenhof B., Lichtenegger H. and Wasle E. (2008). *GNSS - Global Navigation Satellite Systems; GPS, GLONASS, Galileo & more*. Springer-Verlag New York.

Hol J. D., Schön T. B. and Gustafsson F. (2006). On Resampling Algorithms for Particle Filters. In *Nonlinear Statistical Signal Processing Workshop*.

Huang J., Millman D., Quigley M., Stavens D., Thrun S. and Aggarwal A. (2011). Efficient, generalized indoor WiFi GraphSLAM. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), 2011* (pp. 1038 – 1043).

Huber K. (2015). *Precise Point Positioning with Ambiguity Resolution for real-time applications*. Ph.D. thesis, Graz University of Technology.

Huber K., Berglez P., Hofmann-Wellenhof B., Weber R. and Troger M. (2011). The development of enhanced algorithms for rapid precise point positioning. *Oesterreichische Zeitschrift für Vermessung und Geoinformation* 2, 114 – 121.

Humanware (2015). *Trekker Breeze handheld talking GPS*. URL: `http://store.humanware.com/hus/trekker-breeze-handheld-talking-gps.html` accessed on 2015-01-19.

Ibarra-Bonilla M. N., Escamilla-Ambrosio P. J., Ramirez-Cortes J. M., Rangel-Magdaleno J. and Gomez-Gil P. (2014). Step Length Estimation and Activity Detection in a PDR System Based on a Fuzzy Model with Inertial Sensors. In O. Castillo, P. Melin, W. Pedrycz, and J. Kacprzyk (Eds.), *Recent Advances on Hybrid Approaches for Designing Intelligent Systems* (pp. 631 – 645). Studies in Computational Intelligence, Vol. 547. Springer-Verlag New York.

indoors (2013). *Bluetooth Low Energy solution released*. product homepage. URL: `http://indoo.rs/bluetooth-low-energy-solution-released/` accessed on 2015-01-22.

indoors (2015). *Company website*. URL: `http://indoo.rs/features/#offlinemap` accessed on 2015-02-17.

Information and Analysis Center for Positioning, Navigation and Timing (2014). *GLONASS constellation status*. URL: `https://glonass-iac.ru/en/GLONASS` accessed on 2014-12-23.

International GNSS Service (2014a). *International GNSS Service*. URL: `http://www.igs.org` accessed on 2014-12-29.

International GNSS Service (2014b). *MGEX - The Multi-GNSS Experiment*. URL: `http://www.igs.org/mgex` accessed on 2014-12-25.

Japanese Aerospace Exploration Agency (2014). *Quasi-Zenith Satellite-1 "MICHIBIKI"*. URL: `http://global.jaxa.jp/projects/sat/qzss/index.html` accessed on 2014-12-25.

Khider M. (2005). *Implementation of a Simulator/Demonstrator for the SoftLocation Concept using Bayesian Filters*. Master's thesis, Universität Ulm.

Kremer E. (2005). *Einführung in die Mathematik der BAYES STATISTIK für Aktuare und Andere*. Logos Verlag Berling.

Kushki A., Plataniotis K. and Venetsanopoulos A. (2012). *WLAN PosPosition Systems - Principles and Applications in Location-Based Services*. Cambridge University Press.

Kuusniemi H., Chen Y. and Chen L. (2012). Multi-Sensor Multi-Network Positioning. In R. Chen (Ed.), *Ubiquitous Positioning and Mobile Location-Based Services in Smart Phones*. IGI Global.

Legat K. (2002). *Pedestrian Navigation*. Ph.D. thesis, Graz University of Technology.

Li B., Harvey B. and Gallagher T. (2013). Using Barometers to Determine the Height for Indoor Positioning. In *Proceedings of the 4th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2013)*.

Ling P., Chen R., Liu J., Chen W., Kuusniemi H., Tenhunen T., Kröger T., Chen Y., Leppäkoski H. and Takala J. (2010). Motion Recognition Assisted Indoor Wireless Navigation on a Mobile Phone. In *Proceedings of the 23th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS+ 2010)* (p. 10 pages).

Loomis J., Golledge R. and Klatzky R. (2008). *UCSB Personal Guidance System (PGS)*. project homepage. URL: `http://www.geog.ucsb.edu/pgs/main.htm` accessed on 2015-01-20.

Lópes-Salcedo J. A., Capelle Y., Toledo M., Seco G., López Vicario J., Kubrak D., Monnerat M., Mark A. and Jiménez D. (2008). DINGPOS: A Hybrid Indoor Navigation Platform for GPS and

GALILEO. In *Proceedings of the 21st International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2008)* (pp. 1780 – 1791).

Majithiya P., Khatri K. and Hota J. K. (2011). Indian Regional Navigation Satellite System. *Inside GNSS* 6, 52 – 56.

Mautz R. (2012). *Indoor Positioning Technologies*. Habilitation Thesis.

Maíz C. S., Míguez J. and Djuríc P. M. (2009). Particle Filtering in the Presence of Outliers. In *Proceedings of the 15th Workshop on Statistical Signal Processing (SPP 2009)* (pp. 33 – 36).

Micello (2015). *Micello - mapping the great indoors. Company website*. URL: `http://www.micello.com/` accessed on 2015-02-17.

Moder T., Hafner P., Wisiol K. and Wieser M. (2014a). 3D Indoor Positioning with Pedestrian Dead Reckoning and Activity Recognition Based on Bayes Filtering. In *Proceedings of the 5th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2014)*.

Moder T., Hafner P., Wisiol K. and Wieser M. (2014b). *3D Indoor Positioning with Pedestrian Dead Reckoning and Activity Recognition Based on Bayes Filtering*. Poster.

Moder T., Wisiol K., Hafner P. and Wieser M. (2015). Smartphone-based Indoor Positioning Utilizing Motion Recognition. In *Proceedings of the 6th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2015)*. In review.

Montemerlo M., Thrun S., Koller D. and Wegbreit B. (2002). FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem. In *Proceedings of the AAAI National Conference on Artificial Intelligence* (pp. 593 – 598).

Nilsson J.-O., Gupta A. K. and Händel P. (2014). Foot-mounted inertial navigation made easy. In *Proceedings of the 5th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2014)*.

Pahlavan K. and Krishnamurthy P. (2013). *Principles of Wireless Access and Localization*. John Wiley & Sons, Ltd.

Park S. K. and Suh Y. S. (2010). A Zero Velocity Detection Algorithm Using Inertial Sensors for Pedestrian Navigation Systems. *Sensors* 2010(10), 9163 – 9178.

Pozzi F., Matteo T. D. and Aste T. (2012). Exponential smoothing weighted correlations. *The European Phycisal Journal B* 85(6), 21.

Radu V. and Marina M. K. (2013). HimLoc: Indoor Smartphone Localization via Activity Aware Pedestrian Dead Reckoning with Selective Crowdsourced WiFi Fingerprinting. In *Proceedings of the 4th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2013)* (p. 10 pages).

Rajamäki J., Viinikainen P., Tuomisto J., Sederholm T. and Säämänen M. (2007). LaureaPOP Indoor NavigationService for the Visually Impaired in a WLAN Environment. In *Proceedings of the 6th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications (EHAC' 07)* (pp. 96 – 101).

Ran L., Helal S. and Moore S. (2004). Drishti: An Integrated Indoor/Outdoor Blind Navigation System and Service. In *Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications (PerCom'04)* (pp. 23 – 30).

Rantakokko J., Rydell J., Stromback P., Handel P., Callmer J., Tornqvist D., Gustafsson F., Jobs M. and Gruden M. (2011). Accurate and Reliable Soldier and First Responder Indoor Positioning: Multisensor Systems and Cooperative Localization. *Wireless Communications, IEEE* 18(2), 10 – 18.

Renaudin V. (2014). *The Fundamentals of Inertial Navigation Systems*. Tutorial.

Renaudin V. and Combettes C. (2014). Magnetic, Acceleration Fields and Gyroscope Quaternion (MAGYQ)-Based Attitude Estimation with Smartphone Sensors for Indoor Pedestrian Navigation. *Sensors* 14(12), 22864 – 22890.

Ristic B., Arulampalam S. and Gordon N. (2004). *Beyond the Kalman Filter - particle filters for tracking applications*. Artech House.

Robertson P., Angermann M. and Khider M. (2010a). Improving Simultaneous Localization and Mapping for pedestrian navigation and automatic mapping of buildings by using online human-based feature labeling. In *Proceedings of the Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION* (pp. 365 – 374).

Robertson P., Angermann M. and Krach B. (2009). Simultaneous Localization and Mapping for Pedestrians using only FootMounted Inertial Sensors. In *Proceedings of the UbiComp 2009, ACM* (pp. 93 – 96).

Robertson P., Angermann M., Krach B. and Khider M. (2010b). SLAM Dance - Inertial-Based Joint Mapping and Positioning for Pedestrian Navigation. *Inside GNSS* 2010(5), 48 – 59.

Robertson P., Garcia Puyol M. and Angermann M. (2011). Collaborative Pedestrian Mapping of Buildings Using Inertial Sensors and FootSLAM. In *Proceedings of the 24th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS 2011)* (pp. 1366 – 1377).

Ross S. M. (2000). *Introduction to Probability and Statistics for Engineers and Scientists*. (2nd ed.). Harcourt - Academic Press.

Sanz J., Juan J. and Hernández-Pajares M. (2013). *GNSS Data Processing - Volume I: Fundamentals and Algorithms*. ESA.

Sendero Group (2014). *BrailleNote GPS version 2014*. company homepage. URL: `http://www.sendergroup.com/products/shopgps.htm#info` accessed on 2015-01-20.

Susi M., Renaudin V. and Lachapelle G. (2013). Motion Mode Recognition and Step Detection Algorithms for Mobile Phone Users. *Sensors* 13, 1539 – 1562.

Thrun S., Burgard W. and Fox D. (2006). *Probabilistic Robotics*. MIT Press.

Thrun S. and Montemerlo M. (2005). The GraphSLAM Algorithm With Applications to Large-Scale Mapping of Urban Structures. *International Journal on Robotics Research* 25(5//6), 403 – 430.

Titterton D. H. and Weston J. L. (2004). *Strapdown Inertial Navigation Technology*. (2nd ed.). MIT Lincoln Laboratory.

Virtanen A. and Koskinen S. (2004). *NOPPA Navigation and guidance System for the Visually Impaired*. URL: `http://virtual.vtt.fi/virtual/noppa/noppa%20eng_long.pdf` accessed on 2015-01-19.

Wieser M., Hofmann-Wellenhof B., Mayerhofer B. and Pressl B. (2007). A Navigation Concept for Visually Impaired Pedestrians in an Urban Environment. *Österreichische Zeitschrift für Vermessung & Geoinformation* 2007(2), 159 – 165.

Wired (2007). *Design Student Creates Innovative Guiding Cane For Blind*. Magazine. URL: `http://www.wired.com/2007/09/design-student/` accessed on 2015-01-20.

Wisiol K. (2014). *Human-Activity Recognition*. Master's thesis, Graz University of Technology.

Woodman O. J. (2010). *Pedestrian localisation for indoor environments*. Ph.D. thesis, University of Cambridge.

World Health Organization (2014). *Visual impairment and blindness*. URL: `http://www.who.int/mediacentre/factsheets/fs282/en/` accessed on 2015-01-23.

Zarchan P. and Musoff H. (2005). *Fundamentals of Kalman Filtering: A Practical Approach* , Vol. 208. (2nd ed.). American Institute of Aeronautics, Inc.

Appendix A

Basic concepts in probability

In the following, the basic concepts in probability are given. The formulation is based on the notation of Haug (2012) and Thrun et al. (2006).

A.1 Multivariate statistics

A random/stochastic variable varies with regard to a probability distribution and cannot be defined by a deterministic value. Within the thesis, the dynamic system is defined by multi-dimensional stochastic variables, therefore multivariate probability distributions are required to describe the system's stochastic behavior. The sensor measurements, the transition data, the state, as well as the environment are all modeled as random variables within a Bayes filter. All these variables can have any arbitrary values according to specific probabilistic laws. Furthermore, the stochastic system can change with time and consequently, the underlying probability distribution is changing. To describe such a stochastic variable, a dynamic or transition equation is needed. The process of calculating these probabilistic laws is called probabilistic interference (Thrun et al., 2006).

The *Probability Density Function* (PDF) represents the probability that a stochastic variable X has a specific value x and is defined by

$$Prob\,\{X = x\} = p(x). \tag{A.1}$$

In general, probabilities are always positive $p(x) \geq 0$. The probability that the stochastic variable has a value within the limits η and ξ is determined by

$$\begin{aligned} Prob\,\{\eta \leq X \leq \xi\} &= P\,\{\eta \leq X \leq \xi\} \\ &= \int_{\eta}^{\xi} p(x)dx. \end{aligned} \tag{A.2}$$

Considering a multidimensional variable $\boldsymbol{X}$, the probability that this variable has a value within the interval $\boldsymbol{\eta} \leq \boldsymbol{X} \leq \boldsymbol{\xi}$, one has to take the dimension of the variable into account:

$$\begin{aligned} Prob\,\{\boldsymbol{\eta} \leq \boldsymbol{X} \leq \boldsymbol{\xi}\} &= P\,\{\boldsymbol{\eta} \leq \boldsymbol{X} \leq \boldsymbol{\xi}\} \\ &= \underset{\mathbb{R}^{n_X}}{\int_{\boldsymbol{\eta}}^{\boldsymbol{\xi}}} p(\boldsymbol{x})d\boldsymbol{x} = \prod_{i=1}^{n_x} \int_{\eta_i}^{\xi_i} p(x_1, \ldots, x_{n_x})dx_i, \end{aligned} \tag{A.3}$$

with n_X being the dimension of $\boldsymbol{X}$.

The *Cumulative Distribution Function* (CDF) represents the probability $Prob\,\{X \leq \xi\}$ and is for the one-dimensional case

$$P(\xi) = \int_{-\infty}^{\xi} p(x)dx. \tag{A.4}$$

If ξ goes to infinity, the cumulative probability $P(\xi)$ becomes one. The Cumulative Distribution Function (CDF) for a multidimensional random variable is given by

$$P(\boldsymbol{\xi}) = \underset{\mathbb{R}^{n_X}}{\int_{-\infty}^{\boldsymbol{\xi}}} p(\boldsymbol{x})d\boldsymbol{x}. \tag{A.5}$$

If the random variable X is discrete instead of continuous, the integral of the PDF and the CDF in equation (A.2) and (A.4) is substituted by a summation representing the discrete probability distributions:

$$P\,\{\eta \leq X \leq \xi\} = \sum_{\eta}^{\xi} p(x), \tag{A.6}$$

$$P(\xi) = \sum_{-\infty}^{\xi} p(x), \tag{A.7}$$

with the property that the sum of all probabilities is equal to one

$$\sum_{x} p(x) = 1. \tag{A.8}$$

The joint distribution $p(x, y)$ of two stochastic variables X and Y is called *joint PDF* and can be determined by

$$p(x, y) = p(x)p(y) \tag{A.9}$$

if the random variables X and Y are independent. The *marginal PDF* of X is

$$p(x) = \int_{-\infty}^{\infty} p(x, y)dy. \tag{A.10}$$

The *conditional probability* $p(x|y)$ represents the probability of a random variable X under the assumption that the value of Y is known and corresponds to y:

$$p(x|y) = p(X = x|Y = y). \tag{A.11}$$

The conditional probability can be determined by

$$\begin{aligned} p(x|y) &= \frac{p(x,y)}{p(y)} \\ &= \frac{p(x)p(y)}{p(y)} = p(x) \end{aligned} \tag{A.12}$$

if the random processes X and Y are independent and $p(y) > 0$. Combining the conditional probability in equation (A.12) with the marginal PDF given in equation (A.10) yields to the *theorem of total probability*

$$p(x) = \int p(x|y)p(y)dy. \tag{A.13}$$

For the discrete case the integral within the theorem is replaced by a sum:

$$p(x) = \sum_y p(x|y)p(y). \tag{A.14}$$

The *Bayes rule* follows from the relation between a conditional PDF $p(x|y)$ and its inverse $p(y|x)$. The Bayes rule for the continuous case is defined by

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{p(y|x)p(x)}{\int p(y|x)p(x)}, \tag{A.15}$$

while the definition for the discrete case is

$$p(x|y) = \frac{p(y|x)p(x)}{\sum p(y|x)p(x)}. \tag{A.16}$$

The Bayes rule is the basis for probabilistic interference. The conditioned probability $p(x|y)$ corresponds to the *posterior probability distribution* of X after incorporating the measurements y, which is done by $p(y|x)$. $p(y|x)$ is the inverse conditional probability specifying the probability of Y assuming that x was the case (Thrun et al., 2006). The probability distribution $p(x)$ is denoted as *prior probability distribution*, which is the distribution of X before having any knowledge about y. Since the denominator of the Bayes rule does not depend on x, it is the same for any value x and can therefore be treated as normalization constant η for the posterior probability $p(x|y)$:

$$p(x|y) = \eta\, p(y|x)\, p(x). \tag{A.17}$$