Technische Universität München
Department of Informatics

Master's Thesis in Informatics

A User-Assisted Creation of Semantic Indoor Models for Smarter Applications

Matthias Mögerle

TECHNISCHE UNIVERSITÄT MÜNCHEN
DEPARTMENT OF INFORMATICS

MASTER'S THESIS IN INFORMATICS

A User-Assisted Creation of Semantic Indoor Models for Smarter Applications

Eine Nutzer-Gestütze Erstellung von semantischen Modellen von Gebäuden für intelligente Geräte

Author	Matthias Mögerle
Supervisor	Prof. Dr.-Ing. Georg Carle
Advisor	Dr. Marc-Oliver Pahl, Stefan Liebald, M. Sc.
Date	May 15, 2018

Informatik VIII
Chair for Network Architectures and Services

I confirm that this thesis is my own work and I have documented all sources and material used.

Garching b. München, May 15, 2018

Signature

Abstract

This work presents a novel indoor positioning system for end users. The user experience of smart devices can drastically be improved by the knowledge of the user position. For example, the user tries to turn on the smart light per voice command. Since the environment usually can not easily determine the position of the user, the environment is not sure about which light to turn on. Most current indoor positioning systems require an explicit training phase. To increase the usability, this work uses continuous learning instead of an explicit training phase. We combine WiFi fingerprinting, clustering and labelling via requested user feedback. Thereby, semantic location information e.g. room names such as living-room or bathroom are provided to third party applications. Our indoor positioning system runs on an android device, using WiFi scans and accelerometer measurements. The positioning algorithm automatically determines whether the current WiFi fingerprint can be used as reference for the current location. The clustering algorithm requires user feedback, whenever a new and relevant user location is detected. Additionally, feedback is requested to increase the accuracy of an existing location. This work focuses on a minimal intrusiveness for the user. By this, the user can start the application without any prior knowledge and the application requests for feedback and improve the known locations over time. Thereby, the recorded time to be invested by the user are below ten seconds per day. In addition, an API provides the location and movement data to third parties which can be smart environments.

Zusammenfassung

Diese Arbeit präsentiert eine neue Möglichkeit der Positionierung innerhalb von Gebäuden. Die Nutzbarkeit von intelligenten Geräten kann durch Informationen zur Nutzer Position stark erhöht werden. So kann der Nutzer bereits per Spracheingabe eine Beleuchtung einschalten, jedoch weiß die Umgebung nicht welche Beleuchtung eingeschalten werden soll. Durch die Position des Nutzers kann dies stark verbessert werden. Diese Arbeit bietet existierendenden Anwendungen die Möglichkeit auf die semantische Position des Nutzers zur zugreifen. Dazu wurden WLAN Fingerabdrücke verwendet. Diese werden gruppiert und Benennung durch das Einbeziehen von Nutzer Rückmeldungen durchgeführt. Die Anwendung ist als Android Programm ausgeführt und verwendet den WLAN und Bewegungssensor des Mobiltelefons. Dadurch wird erkannt, wenn sich das Mobiltelefon bewegt und der Nutzer kann zu seiner zuletzt besuchten Position um Rückmeldung gebeten werden. Dadurch können neue Positionen hinzugefügt werden und die Genauigkeit für bestehende Positionen verbessert werden. Zudem kann dadurch auf eine anfängliche Trainingsphase verzichtet werden und der Nutzer das System direkt verwenden. Zusätzlich bietet eine Schnittstelle die Daten über die aktuelle und die vergangenen Postionen für weitere Anwendungen zur Nutzung an. Es konnte erreicht werden, dass der Nutzer für dieses System weniger als 10 Sekunden am Tag verwenden muss.

Contents

List of Figures

List of Tables

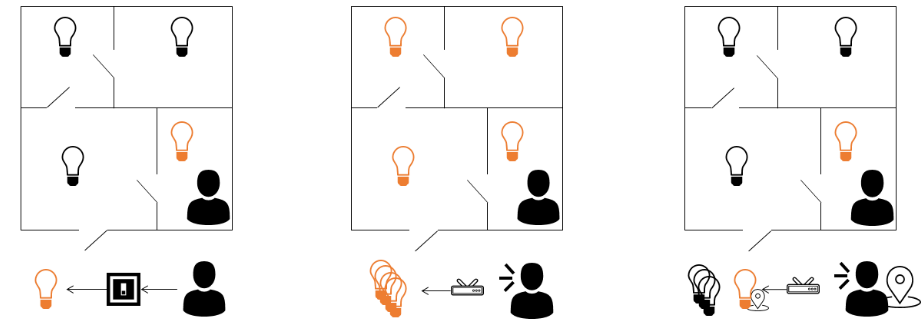
Chapter 1

Introduction

This master thesis is embedded in the context of smart environments where the user is surrounded with smart devices. One of such environments are smart home environments. These contain devices such as smart lights, smart plugs and smart locks. The connectivity between different smart devices can be achieved by existing platforms such as meSchup [1] or DS2OS [2]. In addition, smart environments can extremely benefit from contextual information about the user. This can be social context, surrounding conditions, location and infrastructure - in this thesis we focus on the user location.

1.1 Overview

The importance of the location context is addressed by the following example to control lights. The user is familiar using a switch to turn on the lights. In a smart environment, the lights can be switched on via voice control - "turn on the light". By this voice command the user intends to turn on the light at his current location. In case the user is using his smart phone to record the voice command, the issue is that the system is not aware of the user location. Therefore, it is not be able to only turn on the lights at the location of the user. This can lead to the behaviour that the environment can only turn on all the connected lights. A possible solution would be to match the location of the user with the location of smart devices. This enables a more natural behaviour by only turn on the lights at the current location of the user.



(a) Conventional Button Press (b) Speech to Smart Environment (Without location information) (c) Speech to Smart Environment (With location information)

Figure 1.1: Location context to turn on the light

In this thesis we want to present an indoor positioning system which is detecting locations and additionally provides semantic labelling of the detected locations. Current indoor positioning systems focus on detecting locations while their semantic labelling is less focused. According to Kim et al. [3] "people are going to want everyday applications to have location-awareness that goes beyond simple numerical latitude and longitude [..]" - places like 'my home' or 'Ed's office' which are within room-level granularity. This semantic labelling of locations requires the user input to label locations of interest.

1.2 Goals of This Thesis

The goal of this thesis is to develop a software for mobile phones to periodically monitor the users indoor location. The software should run on the mobile phone and not require specific hardware in its surrounding infrastructure. The mobile phone has various capabilities to sense its environment. These can for example be build-in GSM, GPS, WiFi and Acclerometer. Based on the recognized sensor data and without adjustments in the infrastructure, the application automatically learns about often visited places.

The goal is that this application is installed and kept at the users mobile phone. Unlike others, we aim to not ask for an extensive survey right after installing the application. Rather, the user should be asked during the localization process about his feedback. The goal of the user feedback is to give semantic meaning to detected locations. While asking the user for feedback, the requirement for feedback should be kept as small as possible and as non-intrusive as possible.

Thus, one result of this thesis should be a minimal intrusive approach for semantic labelling. To semantically label locations the user has to dedicate time and information. Therefore, the labelling must be done as minimal intrusive as possible. One use-case can be to provide location data and the history of visited locations to users - analogous to fitness trackers. According to Whitson et al. [4], 60% of U.S. adults are currently tracking their weight, diet or exercise routine. This does represent 'what' the user did. By using location information, additional information about 'where' the user has been are available. Whitson et al. [4] underlines the importance of location information for quantified-self approaches.

The overall objective of this thesis is to provide third party applications and smart environments with semantical location information about the users' location. The tracking can be done by the users' smart phone since it is shown that "smart phone proximity within the same room [...] as the user is true almost 90% of the time" [5]. This thesis further focus on three important aspects. The first focus lies on the clustering of indoor locations.We secondly focus on semantic labelling of locations in a minimal intrusive way. Last, we focus on providing the semantic locations through APIs to third party devices in the smart environment of the user. Therefore, environments can make more targeted decisions when controlling smart home devices (e.g. controlling light and music, showing notifications, providing adaptive UIs).

1.3 Outline

This thesis is structured in the following chapters: While this chapter gave an overall overview, the following chapter 2 analyses the important requirements of an indoor positioning system. Related work is analysed according to the requirements in the chapter 3.

Based on the requirements and related work, the design of this indoor positioning system is discussed in chapter 4. This contains discussion about the technology, clustering and labelling. Details of the implementation are provided in chapter 5.

The results of the user study conducted with this indoor positioning system are evaluated in chapter 6. The results of this work are concluded in chapter 7. This is based on the requirements and provides an outlook by its future work section.

Chapter 2

Analysis

This chapter analyses the most important requirements for the desired indoor positioning system. Thereby, various technologies, clustering algorithms and labelling methods are introduced. The result of this chapter are eight requirements which are used to evaluate the related work.

2.1 Problem Domain

In a world with increasing numbers of "smart" networked devices around us - such as lights, plugs, locks, TVs, phones, tablets - it is important to achieve targeted user experiences and avoid user frustration. One important means to achieve this is to include contextual information to the interpretation of explicit and implicit user interactions with smart devices and environments. One important component of this contextual information is the indoor location.

For instance, using a speech interface to control the lights at home. The voice command "turn the lights off" can only turn of all connected smart lights and not distinguished based on the user location. From the natural user behaviour, it is only intended to turn off the lights in the same room as the speaking user.

The accuracy of the smartphone location can be the key component to derive a user's indoor location. Related work [6] and [7] show how the indoor location of smartphones can reliably be determined on room-level and below using techniques such as movement detection and WiFi fingerprinting. Furthermore, its shown that "smartphone proximity within the same room [...] as the user is true almost 90% of the time" [5]. However, to make these research insights truly usable for context-sensitive applications in real world setups its necessary to a) link indoor location to semantic information and b) lower the burden for setting up indoor location sensitivity and semantic linking.

2.2 Problem Domain Solution Space with Identification of Requirements

This section describes all the possibilities to solve the individual requirements for an indoor positioning system installed on the smartphone of the user. A possible solution to detect the user's indoor location requires several functionalities. At first it must be clarified if the location of the user should be identified from the view of the user or from the view of the environment. In case the environment should sense the user, the environment can add new sensors to its infrastructure. In case the user wants to sense its location, infrastructure changes must be avoided. For this work the positioning should be user centric. This means the system should run on a mobile device to locate the user who is using this device.

The smartphone itself is equipped with various sensors like sender and receiver of GSM, Wifi and Bluetooth. Further smartphone sensors are for example accelerometer, gyroscope, GPS, compass and light sensors. Each sensor technology has its own sensor readings, sensor ranges and sensor accuracy. While GSM and GPS perform very well in outdoor environments, walls can cause significant errors in indoor environments. GPS and GSM work by the use of global infrastructure while WiFi and Bluetooth use local infrastructure to transmit data. The Internal Measuring Unit (IMU) contains sensors such as accelerometer, gyroscope, compass and completely focus on the data which were generated by the personal use of the smartphone. All of these sensor data change whenever the location of the smartphone changes. The accuracy of each sensor predicting the location is related to the type of sensor, its environment and surrounding infrastructure.

For each of the before mentioned sensor technologies, different sensor data are provided for changing environments. Different methods can be used in order to extract information about the changes of the location. For sending and receiving technologies - like GSM and WiFi - it is possible to detect the change of the environment by reading the incoming signals from the environment. Whenever a signal is received, the receiver must be in range of the sender. To further increase the accuracy, another approach is to use measurements of signal transmission delays called lateration. Time differences between multiple senders enable to calculate the position between these senders. Often, walls and other buildings influence the signals received for lateration. Therefore, a more robust approach is to detect the received signal strengths of multiple senders. These senders and their signal strengths create a measurement per location. This method is called fingerprinting.

The afore mentioned methods generate information about locations. To use the indoor positioning system, it requires the orchestrate single measurements into a whole environment context. Then the location of additional measurements can be evaluated.

Therefore, clustering is used to bring individual measurements of locations into context and provides the view on the whole scene. Probabilistic methods determine the classification of one measurement in respect to an available data set. k-Nearest-Neighbours is another clustering algorithm determining which k data are most related to the new measurement and locate the new measurement. Further approaches use Neural Networks and Support Vector Machine to cluster data and make correct predictions about the location of new measurements.

There are especially two types of positioning which can be done by indoor positioning systems. At first, the accurate position can be represented using Cartesian coordinates. Another approach is to use a topological model which does not focus on coordinates but on the user spatial context like "kitchen", "office", "bathroom" and others. A topological model can start on higher granularity and can get more precise like "home"->"living room"->"sofa". The selection of the model especially relies on the required resolution of the indoor positioning system and is defined on the supporting use-cases.

To let the indoor positioning system run on the user's device, the user must be motivated to install the application and to keep it running on his or her mobile device. Therefore, a well planned user-interface and user interactions are important requirements. Without the application being installed and running, the indoor positioning does not work. The indoor positioning system must be less intrusive and provide a real benefit for the user. At best, the benefit is existent right from the users initial use even without surrounding devices.

2.3 Useful Building Blocks - Technology Candidates

This section provides an overview of technologies as well as techniques and algorithms to be applied in the use-case of indoor positioning.

2.3.1 Sensor Technology to Sense the Environment

In this subsection, possible technologies for indoor positioning are presented. Their pros and cons are evaluated and a list at the end of this section summarizes their differences.

2.3.1.1 GPS - Global Positioning System

Global Positioning System (GPS) is used primarily for outdoor navigation. It is based on satellites which continuously transmit signals down to earth. Any GPS receiver can receive this GPS signals and calculate the current location of the user. Nowadays, smartphones contain GPS receivers which are used for outdoor positioning and naviagation. The user only consume the GPS singals but does not send any data. Because

GPS requires line-of-sight (LOS) transition between the satellites and the GPS receiver this provides an accuracy of 5 meters in outdoor locations. In comparision, the indoor positioning is more complex and according to Gu et al. [8] the accuracy ranges between 5 - 50 meters. This is due to much more additional influences and noise resulting from construction materials of the building, humans and additional devices inside the building.

2.3.1.2 GSM

Mobile cellular networks are available and provide wide coverage for mobile telephony and internet. Each receiver is logged into at one GSM cell which has a unique cell-ID. Different cells are overlapping each other to provide better coverage. In rural areas the number of cells is reduced while in cities more cells are available. Accoding to Liu et al. the localization of the cell and cell-ID provides an accuracy of 50 - 200 meters, depending on cell size [9].

Another approach is to use GSM fingerprints. Thereby, positioning can be done through checking all available GSM cells. A currently measured fingerprint can then be compared to prior recorded fingerprints. Otsason et al. [10] showed that this technique provides an median accuracy of 5 meters in large multi-floor buildings.

2.3.1.3 WiFi - IEEE 802.11

Wireless Local Area Networks (WLAN) are operating in the 2.4 GHz band. The scope of such networks ranges between 50 - 100 meters. IEEE Standard 802.11 is currently the most common used wireless networking standard [9]. Therefore, WiFi infrastructure is already included in most buildings today. Each network participant has its own MAC-Address and the signal strength of the network changes according to the location. Most indoor positioning systems using WLAN are based on these characteristics. Youssef et al. [11] is using a joint clustering technique for location estimation. The result is an accuracy of more than 90% within 2.1 meters. Another approach is to use neural-networks-based classifier [12] which resulted in an probability of 72% for an error smaller than 1 meter. According to Bolliger et al. [13], many of these systems have the initial problem that training measurements (offline phase) are required to setup the system. This means initially many measurements have to be taken before using the system in order to later localize the positions correctly. Most often the system gets more accurate, the more measurements were taken beforehand.
Ekahau[1] is a WiFi indoor positioning system which is used commercially. While it uses the existing WiFi infrastructure inside the building, each user has to wear an additional tag. The Ekahau system consists of three steps. At first a site survey is

[1] http://www.ekahau.com/

required to measure signal strengths at different locations. The second step is the mapping of measured data from the site survey to the real map. Finally, each user gets a tag emitting RF signals which are received by the access points and forwarded to a calculation engine identifying the user location.
Haeberlen et al. [14] presented a practical robust Bayesian method for topological localization. They focus on reduced cost in time and calculation power for the mobile device. More details about this approach are written in the related work section.

2.3.1.4 Bluetooth - IEEE 802.15 / Bluetooth Low Energy

Bluetooth (operating at 2.4-GHz) is a very light data transmission standard which ranges between 10 - 15 meters. Because Bluetooth transceivers are very small, Bluetooth is integrated in almost every smartphone.
Each bluetooth device has its unique ID to be identified. Bluetooth Low Energy (LE) is the latest standard. This even enables small devices like bluetooth beacons (e.g. iBeacons[2]). Beacons have especially been introduced by Apple to improve the accuracy of indoor locations. They simply transmit their unique ID and (battery-) status of the beacon. Those can be placed at nearly any location and can run on battery for several years. The signal intensity of beacons can be defined individually which allows to let them cover only a distinct area. This to use beacons in order to mark specific points of interest which then provides locational information.
Another local positioning system using Bluetooth is called Topaz[3]. It achieves 2 meter accuracy in 95% of the measurements. Kotanen et al. introduced another indoor positioning system called the Blutooth Local Positioning Application [15] which used Bluetooth fingerprints and an extended Kalman filter to compute 3D position with an accuracy of 3.76 meters.

2.3.1.5 RFID - Radio-frequency identification

Radio-frequency identification (RFID) using electromagnetic transmission to transfer data. RFID has been designed to identify individual products in a fast assembly process. A basic RFID system consists of RFID reader and RFID tags. Tags are storing data while receiver consume data which they can read from tags. RFID tags are RF compatible integrated circuits which are either passive or active.
Passive RFID tags operate without any battery. They are used like barcodes and provide information when they are read by the RFID reader. Passive RFID tags reflect the RF signal which is transmitted from the RFID reader and add information by modulating the reflected signal. In comparison to the active RFID tags the passive RFID tags are

[2] https://developer.apple.com/ibeacon/
[3] www.tadlys.com

lighter, smaller and have a more limited range which is between 1 - 2 meter [9].
Active RFID tags are small transceivers. They can actively transmit their ID plus additional data if required. Their range covers an area of up to 100 meters.
Hightower et al. [16] used active badges and developed a system localizing RFID position with an accuracy of 3 meter in a 3D room called SpotON. Their positioning system used additional hardware but some of nowadays smartphones contain RFID technology.

2.3.1.6 Vision Based

One very common method to detect positions - especially for mechanical parts - is to use cameras and evaluate their pictures and videos. The OPTPTRAK PROseries[4] uses three cameras to track 3-D positions of objects. These three cameras can cover a room of 20 cubic meters and a maximum distance of 6 meters.
Google has started to work on its own Indoor Positioning System called Visual Positioning Service[5]. It is using a camera on site of the user. The camera records the environment all the time and detects features. Feature recognition helps to localize the position of the user.
Another example is provided by Funk et al. [17] which used the camera at the user to provide navigation in indoor environments.

2.3.1.7 Optically

The use of visible and invisible light can be used to detect positions. Infrared Radiation (IR) can transmit data through infrared radiation and is primarily used for Wireless Personal Area Networks (WPAN). Infrared covers a short range and narrow-transmission-angle beam (line-of-sight) [18]. This provides the feature that IR signals do not transmit trough walls and enter other rooms.
One of the first indoor positioning systems is Active Badge [18] which was developed in the 1990s. Users had to wear a badge which is used to localize the persons within rooms. Unique IR signals are send per patch every 15 seconds. In each room sensors are sensing IR signals from the batch.
Another more modern approach is to use LED light bulbs. These flicker at a very high frequency. This frequency can be controlled and thereby be used to transmit data from the light bulb to the receiving device. Like a bluetooth beacon a light bulb can emit their ID and status which makes the mobile receiver aware of his context.

[4] http://www.ndigital.com/
[5] https://www.androidpolice.com/2017/05/17/googles-new-visual-positioning-service-will-guide-indoor-locations/

2.3.1.8 Auditive

Ultrasound can not be heard by humans but is reflected well by objects. Evolution gave bats the possibility to emit ultrasound and navigate by receiving the reflections. Similar techniques can be used to provide indoor positioning. Information about the position and movement of humans and objects can be estimated by positioning ultrasound senders and receivers within rooms. The Active Bat positioning system[6] was designed at AT&T Cambridge and provides 3D position and orientation. A badge is carried by a human and sends regular ultrasound signals. These ultrasound waves are received by a matrix sensor system mounted on the ceiling. The room and thereby defines the position and orientation of the badge and thereby of the human. The achieved accuracy has been about 3 cm for about 95% of all measurements.
Other systems use audible sound which can be quite unique at different locations. For example some technical device emitting some vibrations and sound.

2.3.1.9 Inertial Measurement Unit (IMU)

In order to improve outdoor positioning and navigation additional sensors are used. These sensors are called Inertial Measurement Unit (IMU) and can consist of magnetic sensor, accelerometers and gyroscope. These sensor data allow to estimate the movement of the mobile device into which the sensors are build into. GPS works well for outdoor locations while it is hard to receive GPS signals in tunnels. Therefore, the IMU data are used to estimate the speed and direction of the user to provide him with navigational information even though there is no GPS signal available. This process is called dead reckoning. Ta et al. [19] used IMU data to reconstruct the path a user took inside a building. They comprehensively compared the different data and their possibilities to track the user.
Another approach using IMU data for indoor positioning is to use the sensor data to estimate the current movement state of the user. The state can be standing, walking, riding the bicycle. Bollinger et al. [6] used motion detection to trigger asynchronous interval labelling. Whenever the smartphone of the user was not moving the system processed measurements to improve the location accuracy.
While context awareness is important for many applications, Google published an Android Activity Recognition API[7] to receive the current movement state of the user. This covers the activities of being in a vehicle, on bicycle, on foot, running, still, tilting the phone and walking [8]. The API requires several seconds until a switch of action is detected.

[6] `http://www.cl.cam.ac.uk/research/dtg/attarchive/bat/`

[7] `https://developers.google.com/android/reference/com/google/android/gms/location/ActivityRecognitionApi`

[8] `https://developers.google.com/android/reference/com/google/android/gms/location/DetectedActivity`

2.3.1.10 Ultra Wide Band - UWB

Ultra wide band is using frequencies of 3.1 to 10.6 and 22 to 29 GHz for positioning. According to Mahfouz et al. [20], a precision of 1 cm in an environment of $15x25m^2$ can be achieved. Indoor positioning systems using UWB consists of a sender (the element which should be located) and receivers in the environment. Receivers are called anchors and receive the senders signal. Further, anchors communicate with each other to calculate the senders location.

Alarifi et al. [21] evaluated the use of UWB and concluded that UWB is a very promising technology - especially for applications requiring high precision. Through the use of high frequency, UWB does not require line of sight to precisely predict locations.

2.3.1.11 Summary

To conclude, all introduced technologies are summarized in the following table. Each column contains one specific feature. Each row represents one distinct technology. Thereby, the different technologies can be compared on different features.

Technology	Availability	Range	Outdoor Positioning Use	Indoor Positioning Use	Available Infrastructure in Buildings	Reliability	Accuracy
GPS	Globally	Globally	Hight	Low	Not Required	Relies on GPS satellites	5 meter (outdoor)
GSM	Mostly Globally	max. 35 km per cell	Medium	Medium	Not Required	Depending on GSM cells	5 meter (indoor)
WiFi	Requires WiFi infrastructure	50-100 meter	Low	High	High	Depends on WiFi Infrastructure	up to 2-3 meter (indoor)
Bluetooth (LE)	Requires Bluetooth infrastructure	10 - 15 meter	Low	High	Medium	Depends on Bluetooth Infrastructure	2 meter accuracy

Technology	Availability	Range	Outdoor Positioning Use	Indoor Positioning Use	Available Infrastructure in Buildings	Reliability	Accuracy
RFID	Requires RFID Tags	1 meter (passive) up to 100 meter (active)	Low	Medium	Low	Depends on RFID infrastructure	3 meter
Vision Based	Requires Cameras	20 cubic meters	Low	Medium	Low	Depending on cameras	na
Optically	Depending on light sending infrastructure	Requires LOS	Low	Medium	Low	Depends on light infrastructure	na
Auditive	Depending on audio infrastructure	Range of sound	Low	Medium	Low	Depends on audio infrastructure	na
IMU	Depending on sensors	Unlimited with increasing error	Low	Low (IMU only)	Not Required	High	Accumulative Error
UWB	Depending on UWB sensor and anchors	Range is defined through anchor locations	Low	High	Requires sender and anchors	High	High

Table 2.1: Summary - Technology Candidates

Each technology has its distinct focus. Compared by their capabilities, all technologies are different and have their distinct use case. This requires to select the technology dependent on the application.

2.3.2 Localization

The previous section described the possible technologies. This section focus on the localization techniques to process localization based on the input of the sensors. Positioning and localization are used in a very similar context. Within this thesis, positioning describes the whole process and result of an identified position, while localization only describes the sub-task of gathering sensor data and analyse them.

To model indoor RF signal propagation is not a simple task due to many possible interferences. Signals can be reflected, multipaths are recorded and line-of-sight (LOS) is not possible while changing to another room. Furthermore, there are building specific parameters like floors, architecture, surface and objects inside the building. Because of these reasons, no unique model for indoor positioning is available right now providing a solution to all of these topics. The next sections describe possibilities on how to estimate positions and locations using different algorithms and clustering methods.

2.3.2.1 Proximity (Cell ID)

For most RF technologies the sender has a unique ID. Therefore, the matching of this unique sender ID (cell-ID) provides a rough estimate about the position. If the measurement detects a sender, the measurement device and its user must be in the range of the sender which must be closer then the maximum range of the sender. This gives a very rough location estimate for receiver of this signal. Using GSM this methodology can identify in which cell a smartphone is currently located.
The signal strength can be taken into account to improve the accuracy. If the maximum and minimum signal strength are defined, the receiver can approximate if the sender is closer or more distant to the receiver. Indoor effects like reflections and attenuations have to be taken into account.

2.3.2.2 Triangulation

Triangulation uses geometric properties to calculate the position of a target device. This requires the use of the network infrastructure. Triangulation is either measuring lateration or angulation. Lateration calculates the position based on the knowledge of reference points and distances between these reference points. The angulation focuses on the angle of the object to multiple reference points. In indoor environments the lateration approach is used more common. In indoor environments triangulation is most often performed using the existing WiFi infrastructure.

Time of Arrival (TOA)

Time of Arrival is measuring the time which a signal takes to transmit from the sender to the receiver. This time is proportional to the distance between sender and receiver. To achieve 2D position accuracy at least three different reference points must be consolidated. It is important to synchronize all senders and transmit a timestamp per packet. The measurement can be processed using signals of different transmitted content.

Real Time of Flight (RTOF)

Real Time of Flight (RTOF) measures the time which the transmission takes from the sender to the receiver and back to the sender. This replaces the strict clock synchronization which is required for TOA. The complete round trip time is measured and provides data about the different delays. These are further used to evaluate the distance between the sender and receiver. This measurement still requires multiple reference points.

Time Difference of Arrival (TDOA)

While TOA focus on receiving signals from different reference points, Time Difference of Arrival (TDOA) sends the signals the opposite way - from the mobile devices to reference points. By sending one signal which is received by multiple reference points, the time difference of receiving the signals at different reference points is correlating the spatial distance between the sender and the reference point. Thereby, the location can be calculated by a central unit receiving all the time differences.

Signal Attenuation-Based (RSS-Based)

Because TOA, RTOF and TDOA imply the correlation of signal strength and distance between sender and receiver, this requires line-of-sight (LOS) transmissions which is in indoor environments only rarely the case. The RSS-Based approach uses the attenuation of the signal strength which lowers the signal strength between sender and receiver. The difference of signal strength can be used as an estimate for the distance. Wall Attenuation Factor (WAF) for example takes the number of walls and their influences like reflections on the RF signals into account. This model relies on its initial calibration effort of the real world environment. The positioning can be very inaccurate because of a large accumulated error according to Evnnou et al. [22].

2.3.2.3 Fingerprinting

Each network participant does have its own identifier (e.g. MAC-Address). This enables to identify and remember infrastructure devices like WiFi access points and repeaters. The Received Signal Strength (RSS) states the quality of a received RF signal like WiFi or Bluetooth. This provides an identifier and a RSS value per detected sender. Receiving signals of one or more sender and corresponding RSS values at the same position can be combined into one fingerprint. Measuring such fingerprints initially at distinct positions allows later localization of unlabelled measurements at these positions. The indoor positioning system RADAR introduced by Bahl et al. [23] is using k-Nearest-Neighbours to achieve an accuracy of 2-3 meters inside a building using WiFi scans. WiFi scans can be processed with an update frequency within seconds on regular devices. WiFi Fingerprints have been recorded by Google Street View Cars[9] for localization purpose. These WiFi data enabled users without GPS sensors to get locational information for the use of Google Maps[10].

2.3.3 Scene Analysis (Clustering)

One of the main goals of this thesis is to recognize locations which were visited more frequently. Entering these locations should be recognized and broadcasted to the users environment enabling context sensitive applications - this requires clustering. Clustering analyses the set of measured data from different locations. Therefore, predefined identifies are used e.g. the signal strength or time-of-flight. The clustering algorithm groups the measured data. Each clustered group has a common set of features which makes this group distinguishable from another group. Ideally, each identified group is identifying one specific (indoor) location. The clustering algorithm is identifying these groups using unique identifiers. To make the groups human readable, each group can be semantically labelled using location names as introduced in the labelling section.

Since indoor environments are changing over time, the indoor positioning system has to take care of the localization in this environment with varying measurements at the same location. As introduced before a fingerprint is one measurement of the environment, containing the characteristics the received senders and their signal strengths. The fingerprint is most likely to be unique for a specific location. At first, a data set has to be recorded and their locations must be labelled. After this data set is prepared, this allows to recognize locations by new measured fingerprints. The measured fingerprints do very often not exactly match the current measurement due to reflections and attenuation. Clustering is able to build clusters based on these varying data. The identification of new measurements is processed based on the defined clusters even though the new measurement does not exactly match one of the previous measurements at a distinct location. Different clustering approaches is presented below.

2.3.3.1 Probabilistic Method

Probabilistic methods consider the localization as a classification problem. As introduced by Yang et al. [24], in a given environment, there are n possible locations $L_1, \dots, L_n$ defined during an setup phase. During the working phase the user is using the system for localization. Therefore, one signal strength vector s containing the signal strengths

[9] https://googleblog.blogspot.de/2010/05/wifi-data-collection-update.html
[10] http://www.zdnet.com/article/google-explains-why-street-view-cars-record-wi-fi-data/

of the available networks is received which should be classified as one of those locations. This is done by the following function:

Choose L_i if

$$P(L_i|s) > P(L_j|s) \tag{2.1}$$

for $i, j = 1, .., n$ and $i \neq j$
The probability $P(L_i|s)$ defines L_i as most likely location.
While the probability P can be calculated using histograms it is possible to use a Gaussian distribution per network of a location in order to represent the received signal strength distribution. Instead of discrete points, this approach does only store the mean and standard deviation of the distribution which results in a reduction of storage. Furthermore, the setup measurements can be reduced because a small set of measurements already provide a comprehensive function of the distribution.

2.3.3.2 kNN (k-Nearest-Neighbours)

For k-Nearest-Neighbour clustering, the new measurement is compared to the existing setup measurements in the database. The k closest measurements are selected and their major locations decide about the location of the new measurement. Parameter k can be adapted to increase the performance of the location results.

2.3.3.3 Neural Networks

The use of a Neural Network requires an initial training phase. During the setup phase the RSS values and corresponding locations are adjusting the inputs of the neural network. The neural-network-based positioning system is using one hidden layer for its multilayer perception (MPL) network. To address the inputs the measurements during the working phase are represented as input vectors. This vector is multiplied by the trained input weight matrix [9]. The output vector has either two elements or three elements for a 2D or 3D estimated location.

2.3.3.4 SVM (Support Vector Machine)

A Support Vector Machine (SVM) is classifying measurements through the use of training data. These are used to construct an optimal separation between the different labels of the training data as surveyed by Liu et al. [9]. The measurements during the working phase are then separated into different labels according to the earlier defined separation of different locations.

2.3.3.5 DBSCAN

DBSCAN has been proposed by Daszykowski et al. and focuses on the clustering of not well formed clusters which do not have a typical circular shape but are for example a long line of highly related objects with high density [25]. Therefore, DBSCAN is first choosing one data point and checks if other data are close to this point. If there are enough data available this data point is a core point. If there would not be some but not enough core points close to it, DBSCAN marks them as reachable. If there would not be any other core point within reach, this point is marked as not reachable. The core points and related reachable points are clustered in the same cluster. This does guarantee to find cluster even though they are not well formed or can be separated by a straight line between the data points.

2.3.3.6 OPTICS

OPTICS [26] is related to DBSCAN but focuses on its major issue. The major issue of DBSCAN is the density of points for clusters. If the density varies between clusters DBSCAN struggles to identify the clusters appropriate. Therefore, Optics uses variations in the density do detect clusters with lower density and clusters with higher density correctly at the same time.

2.3.4 Environment Representation

To position the measurements in the real world, the real world needs to be abstracted for the representation in a computer. Two possibilities are introduced below. The environment can either be represented using a 2D or 3D map or by the use of labels per location which can be ordered topologically.

Whenever the application aims to represent the exact location in a room, a map of this room or building is required, Kartesian coordinates can match a drawn map. The benefits of a map are that further data of this map can additionally be used to train the algorithm such as coridors, doors or stairs. This has been used in the system of Ta et al. [19]. For example the user is not able to walk through walls and has to use the door which provides further estimates and information which can increase the accuracy of the user position.

Another approach are topological models which are used to avoid the requirement of real maps of the building. Those topological models start to label the positions and order these labels hierarchically. This provides first information about the building and gets more fine grain like providing the correct floor as another label. It continues getting to more fine-grained information like the exact room and even a specific position within the room. This model does not require an exact map but information about the relations between different label spaces. According to Haeberlen et al. [14], a topological model can dramatically reduce the time required to train the model.

2.3.5 User Information

Using specific features of the object which should be positioned can bring an additional benefit. If the object does not move, different assumptions can be made instead of a movable object. While it can move, the movements can only happen within the physical constraints. For a human it is not possible to change the room by walking through a wall. These information can be used to add additional accuracy [19] to a positioning system as introduced by Ta et al..
Another possibility is to recognize the movement possibilities of the user. This can be states like walking, standing, riding a bicycle, running and others. To recognize these patterns, the IMU of smartphones containing accelerometer and gyroscope can be used. These sensor data can be evaluated to abstract the current state of user movement. APIs offer the possibility to read this user state in order to provide contextual information to applications. Bolliger et al. used motion detection to improve the accuracy of indoor positioning [6].

2.3.6 Application User Interface

If the positioning application is installed on the users smartphone, different ways to interact with the user are possible. These define if the application is useful for the user or in contrast be removed from his smartphone within a short period of time.

2.3.6.1 No User Interface

The application can completely run as a background service on the smartphone. It completely works in the background, evaluates the user location and provides the location to further applications. This removes the possibility to put the user into the loop of labelling locations or to mark correct and false positions. This avoids giving the user the chance to correct or at least report any incorrect data.

2.3.6.2 Light User Interface

A light user interface can provide information to the user. This can contain information about the current location as done by Shin et al. [27]. By this, the user feels more informed by the application, because he has the chance to check which location is defined as the current location by the application. For users this gives the chance to simply consume the positioning service(e.g. using Google Maps[11]). An example can be to find the closest rest room. Thereby, the user does not get in touch with the underlying positioning measurements. The users can use the location service independent of the measurements in a non-cooperative way.

2.3.6.3 Interactive User Interface

An interactive user interface does not only represent data to the user, but in addition interact with the user. These interactions can be triggered by the application. The application can ask the user to label the current location or to vote if the current location matches the real location one example is RedPin [13]. By this the application can ask questions to reach higher accuracy and to label locations.

Another possibility is to provide the user the tools to improve the location data on his or her own. Thereby, the user can ask the application to start new measurements since a location is incorrect or the infrastructure changed. This can be the interface for an expert user improving the accuracy of the application. Instead of having one distinct expert-user, the applications can ask the user to cooperate. This then provides a collaboration between users and an increase of accuracy for the whole positioning of each individual user as surveyed by Yang et al. [24]. The contribution per single user remains very small. This approach to integrate the user has been used for the Redpin application introduced Bolliger et al. [13].

2.3.6.4 Involvement of User Feedback

Various applications of mobile devices already ask for feedback of the user. They use the feedback to improve their context sensitive behaviour as well as to improve the user experience within the application. While some user feel good to provide feedback to the application, other users are annoyed by frequent requests for feedback as Baumann et al. [28] introduce by different threshold levels. In addition to this, the way the application asks for feedback is an important criteria for the user whether he is willing to share his thoughts or if he does ignore the feedback request. An indoor positioning application has to deal with various locations which provides a high uncertainty. This can be reduced by asking the user for feedback. This can be to label the current location, or provide feedback about his environment and the correctness of calculated location of the application as implemented by RedPin [13].

[11] `http://www.maps.google.com/`

2.3.7 Localization Accuracy

To provide a certain level of service, the accuracy of the indoor positioning system must suit the application. Indoor accuracy can be separated into two main categories. Room-level accuracy and exact position accuracy. Room-level accuracy identifies the right room in which the mobile device and its user are positioned. Systems like Redpin introduced by Bolliger et al. [13] focus on room level accuracy to provide contextual information of the user. Other systems focus on the exact position to calculate the exact path of the user as introduced by Ta et al. [19].

2.3.8 Robustness on Changing Environments

Indoor environments are not completely static and can be changed by changing objects over time. Depending on the technology, the signals received from the indoor location are different. The aim is to provide an accurate localization while the environment might change. An approach to update the signal database is to make new reference measurements. This leads to re-calibration as introduced in the framework of Ta et al. [19]. In addition, Bolliger et al. [6] showed that the robustness of the measurements increase by either more measurements or more sender in reach of the localizing receiver.

2.4 Requirements for an Indoor Positioning System

To identify the user location within an indoor environment, several tasks needs to be considered. To sense the environment a sensor technology is required. The evaluation of these data is important to extract position information. An algorithm to process and label the measurements is mandatory. For the interaction with the user, the user interface needs to provide additional requirements for example to ask for user feedback. In the following list, all necessary requirements are explained in the following sections in more detail.

2.4.1 <R.1> Sensors

To do indoor positioning, one or more appropriate sensor(s) need to be chosen in order to read these sensor data and position the user. Some sensors need additional hardware within the environment to receive useful sensor data. As identified in figure 2.1, for most systems the user actively sends signals to the environment, which thereby detects the users' location. Another possibility is to employ the environment to send data to the user. Thereby, the user detects these signals and can position himself. In the 1990s, indoor positioning systems started used infrared and ultrasound. Today, GPS works well for outdoor environments providing line-of-sight (LOS) transition from the satellite to the GPS receiver. Because of walls and obstacles, this does not provide a sufficient reliability for indoor environments. Nowadays, indoor position technologies mainly use infrastructure components which are already deployed or can be deployed easily. WiFi infrastructures (Access Points) and Bluetooth infrastructures (Bluetooth Beacon) have to be mentioned. Another approach is to use led lights and transmit data via hight frequent flickering of the light. This flickering is detected and positioned. Movement sensors like accelerometer, gyroscope or compass can track the movement of the user and thereby increase accuracy of the indoor positioning system.

2.4.2 <R.2> Localization

The signals of a sensors requires further processing in order to abstract any locational information. They are different on the computational effort, the reachable accuracy and the use of hardware. The most common techniques are described in the localization section 2.3.2. The most intuitive solution to localize a mobile device is to receive RF signals and identify their senders. Because each technology has only a limited transmission range, the receiver must be in the range of the sender. This gives a very rough estimate about the position while the accuracy depends on the technology used for the sender and its settings. For example GSM Cells (50 - 200 meters), WiFi networks (50 - 100 meters), Bluetooth senders (up to 15 meters) or RFID (several centimetres to 100 meters).

Another approach is to use triangulation. This requires multiple reference points to use geometric calculation in order to estimate the position of the user. In an ideal environment, the transition time between sender and receiver are correlating to the distance between sender and receiver. Therefore, the TDOA (Time Difference of Arrival) approach sends data to all receiving reference points at the same time. From the time distance between the arrivals at different reference points, the position of the sender within these reference points can be distinguished. A major issue is that there are reflections, attenuations and further influences in an indoor environment which reduce the accuracy of this approach. Measuring the attenuations is possible by reading the received signal strength (RSS) values.

2.4.3 <R.3> Scene Analysis (Clustering)

The sensor data of two measurements at the exact same location do most likely deviate from each other. To identify different locations while the measurements are not exactly identical, clustering is a common approach. Clustering techniques deviate on the methodology, on the required calculations and on the required data. The probabilistic method employs the Bayes theorem, to identify the location which is most likely to be the location of the new measurement based on a database with existing labelled measurements. The clustering of k- Nearest-Neighbours works by the evaluation of the location of the position of the k closest neighbours. Neural networks and support Vector Machines are deeper analysing the training data. The approaches of DBSCAN and Optics are developed to cluster datasets where the clusters have different distribution patterns and are not well surrounding a central reference point.

2.4.4 <R.4> Environment Representation

To make the data of locations feasible for the user, the environment needs to be abstracted. Thereby, the user can understand and interact with locations inside the application. This can be done using different approaches as introduced in the according analysis section 2.3.4. The real world environment can be represented by the use of a 2D or 3D map. Another possible solution is to use labels for different locations in the building. This allows to orchestrate them hierarchically which is called topological model.

2.4.5 <R.5> User Information

The user is defined though several physical properties. E.g. in case there are separated rooms, the user is not able to transit from one room into another room by going through walls. Contextual provided information about the user can help to make the indoor positioning system even more precise. IMU sensor data provide information about the current movement state of the user. The user can be in different states like to stand still, walk or run. These can all have influences on the position of the user and how fast he transited the location. In addition to the sensing of the environment, the sensing of the user provides valuable insights for indoor positioning systems.

2.4.6 <R.6> Application UI

The ultimate goal is to detect the location of the user. Data of sensors which are sensing the environment provide valuable insights. In addition to those, some information can only be provided by the user. In the work of this thesis such information are the semantic representation of the current location. This can be called room name or room identifier. Different approaches are presented in the Application UI section 2.3.6. An application which is not using any user input can record data of the user location simply as a background service. This does not give the user any chance to read information from the system. Another approach is the light user interface where the user can see the information about his current location, but can not interact with the system like asking to correct a wrong location. This can be done in the interactive user interface. Here the user is involved of the process of training the system to provide more accuracy.

Either the user can actively interact with the system or the system can ask the user for distinct feedback.

2.4.7 <R.7> Accuracy

The accuracy of the indoor positioning system can be defined on room-level or in more detail to the exact position with some error. The required accuracy depends on the field of application. Applications which need the user position for contextualization mainly work on room-level location. Applications which are interested in the path of the user are using exact positions. The true positive rate of predicted locations is important to provide benefits being used in real applications.

2.4.8 <R.8> Robustness

Indoor environments are constantly changing. This can be persons walking around, restructured rooms or changed weather conditions. To keep the trained positioning accuracy the system has to be updated. As introduced in the section robustness 2.3.8 robustness can be achieved by additional measurements updating the existing data and thus maintain the positioning accuracy.

2.4.9 Requirement Summary

Requirement	Name
<R.1>	Sensors
<R.2>	Localization
<R.3>	Scene Analysis (Clustering)
<R.4>	Environment Representation
<R.5>	User Information
<R.6>	User Application UI
<R.7>	Accuracy
<R.8>	Robustness

Table 2.2: Listing of Requirements

Chapter 3

Related Work

Location-aware computing as surveyed by Chen et al. [29] became important trough the existence of mobile computing devices. In early days location-awareness required additional hardware to identify the location of a device. Diffuese infrared pulese have been emitted by badges of the Active Badge System [16], which were detected by ceiling-mounted sensors. Another early system has been the Active Bat System[1] which uses ultrasound signals for localization. These systems have been introduced around 2000. Nowadays indoor positioning systems are are mostly using radio frequency signals such as WiFi or Bluetooth. Below, a listing of diverse indoor positioning systems is provided.

3.1 Gaussian Fit - Practical Robust Localization over Large-Scale 802.11 Wireless Networks [14]

Haeberlen et al. from Rice University published the paper Practical Robust Localization over Large-Scale 802.11 Wireless Networks [14]. The system is using WiFi Fingerprints in order to achieve room-level localization in an office building. Haeberlen et al. identified the following important factors for their system. At first, a very low training effort is crucial for their indoor location-sensing systems. Furthermore, the system must achieve high accuracy and must use available unmodified hardware. To make the system usable for a long period it must be robust for untrained variations of the sensor data.

[1] `http://www.cl.cam.ac.uk/research/dtg/attarchive/bat/`

3.1.1 Requirements Analysis

3.1.1.1 <R.1> Sensors

Haeberlen et al. used the WiFi signals for their positioning system. They used the existing 33 routers within their building and the buildings close to their building. A notebook with WiFi antenna has been used to receive the signals from these base stations.

3.1.1.2 <R.2> Localization

The localization has been performed by the recoding of WiFi fingerprints. Therefore, the indoor positioning system of Haeberlen et al. requires an initial training within the environment. They focused on keeping the training effort as small as possible. This resulted in the training time per room of one minute. A notebook is required to record around 25 fingerprints at each specific location. These fingerprints are used to train the system. During the use of the system these data are used to identify the current location of user.

3.1.1.3 <R.3> Scene Analysis (Clustering)

The indoor positioning system of Haeberlen et al. focuses on a room-level localization. During the measurement, each location (e.g. room or hallway) was treated as one single position. Large rooms have been split up into smaller locations with an average size of $24.6m^2$. The signal intensity distribution of each router has been approximated by a normal distribution. Haeberlen et al. showed an improvement over another approach using histograms. While the robustness increased, the number of required measurements decreased. The number of measurements especially depends on the size of the location area. In this paper Haeberlen et al. recommend to observe mean and standard deviations of the measured data in order to estimate if more samples are required. Below different algorithms for localization using fingerprints compared by Haeberlen et al. are introduced.

Bayesian Localization Framework

The Bayesian Localization Framework, solves the localization problem determining the agents position. It provides probabilities for different possible states $S = s_1, ..., s_n$. The probability distribution of the prediction completely relies on the current state observations $O = o_1, ..., o_m$. The estimated position is π_i, while the new estimate is π_i'.

$$\pi_i' = \frac{P(o_j|s_i)\pi_i}{\eta} \tag{3.1}$$

$$\eta = \sum_{i=1}^{n} P(o_j|s_i)pi_i' \tag{3.2}$$

Gaussian Fit Sensor Model

The Gaussian fit sensor model uses normal distribution functions for each base station (access point) to represent the singular measurements. Therefore, it only requires to store two variables per base station of a location. In addition, this reduces training effort compared to the use of histogram (introduced next) by half.

A fixed set of base stations $B = \{b_1, ...b_k\}$ matched with the received signal strength (RSS) values $V = \{0, ..., 255\}$ result in the observations $O = B \times V$. Each state s_i has its own probability for being the correct location. The signal intensity distribution $P((b_j, v)|s_i)$ per base station b_i determined by mean $\mu_{i,j}$ and standard deviation $\sigma_{i,j}$ of this specific base station and specific location. Probability to observe $b_j, v \in O$ at state s_i.

$$P((b_j, v)|s_i) = \frac{Gi, j(v) + \beta}{N_{i,j}} \tag{3.3}$$

$G_{i,j}(v)$ is a discretization of a Guassian probability distribution.

$$\sum_{v=0}^{255} P((b_j, v)|s_i) = 1 \tag{3.4}$$

Histogram sensor model

The histogram sensor model has been used in previous work of Haeberlen et al.. It stores each $P(o_j|s_i)$ in one table. Then for each state (location) s_i, the $P(o_j|s_i)$ is determined. Having enough measurements, the histogram sensor model does look like a Gaussian distribution curve. Because the Gaussian distribution curve only requires two parameters while the histogram has to store each measurement, the histogram sensor model requires way more storage.

Markov Chains

To model transitions between rooms, Haeberlen et al. introduce the use of Markov Chains. Markov Chains contain details about transitions between rooms and reduce the error impossible room transitions. The user is tracked while he or she is moving at 4 meters per second. This constraint is given because of the time to recalculate the next state.

<R.4> Environment Representation

The localization works through the use of a topological location based on Bayesian inference. Therefore, the environment is modelled topologically as graph of the environment. In addition, Markov localization is used to infer the next state from the current state.

<R.5> User Information

No additional information have been used by Haeberlen et al. gained from the user of the system.

<R.6> User Application UI

Because Haeberlen et al. focused on the location accuracy using WiFi, no user interface has been introduced.

3.1.1.4 <R.7> Accuracy

A correct identification of the room is achieved with an accuracy of over 95%. This includes static as well as dynamic localization which supports a walking speed of 3 meters per second. To test their system, Haeberlen et al. performed at least 100 base station scans in each of the 510 cells of their environment. To test the localization accuracy, they removed five scans random for each of the 510 cells and trained the system with the remaining data. Then they used the random chosen measurements to test their accuracy. This has been repeated for 100 times choosing different scans each time. The Gaussian method clearly exceeded accuracy of the approach using histograms. They additionally varied the training set size and measured the remaining accuracy. From the experience of the author, people close to the antenna influenced the measurement which for some measurements resulted in the identification of the localization in an adjacent office.

Calibration

Environmental changes like time-varying effects and different hardware can be approximated by a linear relationship. These two parameters can be adjusted for the user with little or no user-intervention.

$$c(i) = c_1 * i - c_2 \tag{3.5}$$

This linear relationship is used to calibrate specific measurements of a new device. Identifying c_1 and c_2 requires a test environment. The calibration parameters can be calculated through the least-squares method.

3.1.1.5 <R.8> Robustness

The system of Haeberlen et al. focused on robustness of various hardware devices and the time-varying phenomena of measurements. The different results of different WiFi adapters are considered in their implementation. Various hardware can be used and benefit from this localization system. Furthermore, variations of the measurements because of presence or absence of people in the office building like office traffic is considered. The system scales well for large buildings. They performed a test in an 12,000 square meter office which has been divided into 510 cells. To reduce the number of measurements, Haeberlen et al. evaluated how the number of measurements does influence the resulting accuracy. In addition, the impact of base station density on localization accuracy was evaluated. They show that the amount of measurements as well as of base stations the be reduced to a certain amount without drastic changes in accuracy.

3.2 Indoor Location Wi-Fi Fingerprinting using Invariant Received Signal Strength [30]

In this article, Husen et al. [30] propose an improvement on indoor positioning through focusing on the instability of random spatio-temporal disturbances of the received signal strength (RSS). They aimed to make approaches using RSS based location fingerprints more suitable for real world applications. Their method achieved a 17% increase in accuracy compared to regular indoor positioning which do not focus on RSS disturbances.

3.2.1 Requirements Analysis

3.2.1.1 <R.1> Sensors

Husen et al. used WiFi measurements to implement room level localization. They especially focused on getting measurements of the environment without any disturbances. Therefore, a smartphone has been attached to a remote controlled robot.

3.2.1.2 <R.2> Localization

During midnight, the smartphone recorded WiFi measurements during midnight which were then labelled and used as Reference Received Signal Strength (R-RSS). At each location (one position per room) 100 R-RSS measurements have been processed. In addition to the measurements processed during midnight, another measurement was performed during the day to collect the Invariant Received Signal Strength (I-RSS). These training data contained interferences with people moving through the building.

3.2.1.3 <R.3> Scene Analysis (Clustering)

These I-RSS measurements have been processed at various locations and were accessed in regard of variations compared to the Reference Received Signal Strength. To only work with data which were reliable, the following rules were implemented:

- **Observe RSS mean value:** When the mean value of an RSS for a network is less than -89 dBm this network is discarded, because for this specific location the RSS value is very low and thus unreliable.
- **RSS standard deviation:** When the standard deviation is greater than 5, these measurements are discarded.
- **Percentage of missing and undetected RSS values:** When more than 30% of the measurements at a location do not contain a certain network, the network is discarded from this location.

After accessing the difference between the pure measurement of R-RSS values and the measurements during the day, the system identified the Invariant Received Signal Strength (I-RSS) values. During a regular localization request, the system uses the fingerprint and checks the possible influences of environmental factors using the I-RSS values and estimates the user location.

3.2.1.4 <R.4> Environment Representation

The environment has been represented using room-level locations. Therefore, Husen et al. used room identifiers to label locations. This enabled them to process one set of training data within each room. Per room the measurement device stood still during the measurement.

3.2.1.5 <R.5> User Information

User information were not used to localize later measurements. These did only depend on the measured signal strength of the different networks and manual labelling during the measurements.

3.2.1.6 <R.6> User Application UI

A user interface has not been introduced since this system primary focused on the accuracy using invariant received signals compared to regular signals.

3.2.1.7 <R.7> Accuracy

Taking the Invariant Received Signal Strength (I-RSS) into account improved the performance of the indoor localization up to 93%. Using the same data and applying the conventional approach resulted in only 76% accuracy on room-level predictions.

Recalibration

To maintain accuracy, the initial measurements of the framework from Husen et al. has been recalibrated after several weeks to maintain its accuracy. To recalibrate, measurements at distinct locations must be processed.

3.2.1.8 <R.8> Robustness

A high robustness is achieved through the possibility to recalibrate the initial measurements after several weeks. This continuously ensures the quality of correct locational information. On the other hand, this requires time and effort to process new readings at different rooms.

3.3 IPIN Tracking Competition - Smartphone-based User Location Tracking in Indoor Environment [19]

Ta et al. submitted a paper for Track 3 of the 2016 IPIN competition. This competition provided a set of data which were measured by the organizers. The challenge was to construct the correct path the recording person took through the building. The building has been a multiple floor environment. Therefore, data from 12 different types of sensors were provided which were recorded. The goal has been to update a precise position of the user every 0.5 seconds.

3.3.1 Requirements Analysis

3.3.1.1 <R.1> Sensors

The data consists of 10 different sensor data recorded by three different smartphones. Ta et al. used data of GPS, WiFi, accelerometer, magnetic and gyroscope sensors. Each sensor data had a different sampling rate. In addition, camera data have been observed gain more insights on the recorded data of the competition.

3.3.1.2 <R.2> Localization

For localization purpose, the MAC-Addresses of available WiFi networks and WiFi fingerprints were abstracted from the data set. In addition, Ta et al. separated the user state into the walking and standing state. Thereby, they reconstructed the path which the user took through the building.

3.3.1.3 <R.3> Scene Analysis (Clustering)

In this paper, the tasks to reconstruct the path have been divided as follows. Since the dataset contained multiple buildings, the correct building has been identified at first. Ta et al. used the GPS signal as well as the WiFi MAC address data to uniquely identify buildings. The next task has been the floor identification where WiFi fingerprints were used to identify the floor ID. For clustering three algorithms have been tested. K-nearest neighbour (KNN), Random Forest (RF), Extreme Gradient Boost (XGB). While XGB provided the most accurate results. To detect the orientation and direction of the user, accelerometer, magnetic and gyroscope sensors were used. These allowed to separate between the state of walking and the state of standing. The following formula has been used to identify walking or standing of the user. $Gyro = (1 - \alpha) \times IntegratedGyro + \alpha \times AccMag$. The factor α is a threshold weighting the contribution of the integrated gyroscope as well as the accelerometer. At last, Ta et al. inferred the speed of the user by its step length.

3.3.1.4 <R.4> Environment Representation

The environment has been topologically segmented. Starting from the identification of the building, the floor-level, the room-level, to an accurate positioning within the room to reconstruct the path.

3.3.1.5 <R.5> User Information

Ta et al. used information about the user and the environment to improve the accuracy of their path predictions. Therefore, they avoid crossing of walls within their framework and use walls as boundaries. Rooms can only be changed through doors or corridors.

3.3.1.6 <R.6> User Application UI

Because the challenge only required to reconstruct the most accurate path the user took through the building, a user interface has not been implemented.

3.3.1.7 <R.7> Accuracy

This paper focused on tracking the user through the building and evaluated using different data. The positioning error after some time of use has been measured to compare different techniques. The error after 7 minutes only using WiFi resulted in 29.8 meters while the error declined to 24.5 meters after using the gyroscope and accelerometer additionally.

3.3.1.8 <R.8> Robustness

To increase robustness Ta et al. used several sensors which improved the accuracy even in difficult situations like small rooms.

3.4 Joint Clustering - WLAN Location Determination via Clustering and Probability Distributions [11]

In this paper Youssef et al. [11] from the University of Maryland propose a location system using WiFi clustering and probability distributions. They reached an accuracy of over 90% within 7 feet. Youssef et al. focused on two main features in their work. At first, they wanted to reduce the impact of the noisy nature of wireless measurements by using probability distributions to improve accuracy. Second, they reduced the computational requirements by the clustering of locations. They called this method Joint Clustering which is introduced in more detail below.

3.4.1 Requirements Analysis

3.4.1.1 <R.1> Sensors

Youssef et al. used WiFi for their indoor localization system. Received signal strengths of networks below -81 dB have been ignored because of the inaccuracy of these measurements.

3.4.1.2 <R.2> Localization

For each location all available networks and their received signal strength (RSS) were measured. By this, fingerprints have been generated. Per location, only the amount k visible networks were stored per fingerprint. k has been defined beforehand. For large k, the calculation became more complex and all locations must be covered by k networks. For location l the k networks with the strongest signal strength were selected. For each of these k networks per location, the RSS values are stored and their distributions were visualized as histograms.

3.4.1.3 <R.3> Scene Analysis (Clustering)

To locate the user, the RSS values of available networks at the users' location have been measured. Having k networks per location brings the risk of measuring less or more networks than the stored k networks. Therefore, Youssef et al. introduce parameter q which is smaller than k and is defining the size of a subset of networks. The q networks with the strongest RSS values were selected. These networks were then used to estimate the probability of each location using Baye's theorem $P(S)$. Bayes provides the chance to implement different likelihoods for different locations $P(l)$ while it is possible to give an equal likelihood to all locations.

$$P(l|S) = \frac{P(S|l) * P(l)}{P(S)} \tag{3.6}$$

While $P(S)$ is constant for all locations:

$$P(l|S) = P(S|l) * P(l) \tag{3.7}$$

The location having the highest probability is most likely the location of the user.

3.4.1.4 <R.4> Environment Representation

The environment has been modelled by dividing the 2D representation into equal cells of 5 feet which has been used due to the width of the corridors. The cells were identified by labels. This resulted in 100 cells for their experimental environment.

3.4.1.5 <R.5> User Information

The algorithm did not include any information about the user behaviour and boundaries like walls or other objects.

3.4.1.6 <R.6> User Application UI

The client side has been a simple application without focus on the user interface. The code to evaluate the position did run locally on each client. This kept the data about WiFi fingerprints and user positions local. This avoids privacy concerns of the user about external calculated positions. The privacy for the location of the user is guaranteed.

3.4.1.7 <R.7> Accuracy

The framework of Youssef et al. has been evaluated in a 20,000 square foot area. An accuracy of over 90% within 7 feet has been achieved while having very low computational requirements.

3.4.1.8 <R.8> Robustness

The computational efforts to locate the user are drastically reduced by the use of only the k strongest networks per location and the q strongest networks per measurement. If the positioning is running on mobile devices, energy consumption is a very important factor. The proposed solution focused on a small consumption of energy.

3.5 Redpin - Adaptive, Zero-Configuration Indoor Localization through User Collaboration [13]

Redpin[2] is a indoor positioning system developed by Bolliger et al. from the ETH Zurich. The system uses fingerprints of GSM, Bluetooth and WiFi. The framework is separated into a mobile client and a server. The main goals are to avoid the use of additional hardware. To provide a very easy setup and maintenance procedure and to provide at least room-level accuracy. It further empowers user collaboration to create the measurements in order to localize positions within buildings. Every user can create, modify and use location information which were created by other users. Redpin is open source and publicly available. The Indoor Positioning System of Bolliger et al. is separated into two main components. One is the server which is running as one instance. The other one is the mobile application which each user is running on his mobile device.

[2]`www.redpin.org`

3.5.1 Requirements Analysis

3.5.1.1 <R.1> Sensors

Smartphones are used as mobile clients. On these, data of GSM, WiFi and Bluetooth are used for the indoor location system. Each of these sensors is recording the available networks (senders) in its technology.

3.5.1.2 <R.2> Localization

The available networks are measured which Bolliger et al. call sniffing. The location identification takes place using an algorithm at the server which contains the so called locator component. All fingerprints are stored in the database on the server to provide locational information. The locator service on the server allows the users to receive a room-level location based on the measured fingerprint. The location which best matches the fingerprint is send back to the users mobile client. The location is only send, if it is below a defined threshold to reduce incorrect location information.

3.5.1.3 <R.3> Scene Analysis (Clustering)

At first, the mobile application is searching for available WiFi and non-portable Bluetooth networks. The measured fingerprint is then send to the central server which tries to locate the mobile device. If the user can be located the result is send back to the mobile application which shows the position of the user on the map. If the fingerprint location is unknown for this measurement, Redpin identifies the last known location of the user. In order to locate the user, Redpin continues to measure the available networks and compares them to the last three measurements of a known location. If the position still can not be located, the user is asked to label the name of his current location. Furthermore, he has to position himself on the floor-plan using a marker on the map in the application. In addition, Redpin offers the user the possibility to correct the identified location. Therefore, the database is continuously updated by changes and new entries of the users.

3.5.1.4 <R.4> Environment Representation

The environment is represented on a map using Cartesian coordinates in two dimensions. The floor plan is required in order to let the user process measurements and label these positions. In addition, this floor plan represents the position of the user.

3.5.1.5 <R.5> User Information

This paper focused on the collaboration of users. Additional information of the user possibilities are not used by this system.

3.5.1.6 <R.6> User Application UI

To avoid an extensive training phase, Redpin employs the users to train the system during their use. This not only avoids the initial training phase, it also accounts for a continuous training. On the mobile client the user can name and rename locations and place them on the map. The results and changes of the mobile application are send to the central server to either generate the map or localize the user. By empowering the user to add new, to create and to modify fingerprints and locations the system is more robust to changes in the environment. The visualization of the user location is shown via a mobile application to the user. The matching of a measured fingerprint is not directly linked to a position on the map. Instead it is linked to a symbolic location which for example can be a room number or room name.

3.5.1.7 <R.7> Accuracy

Bolliger et al. used the system in their office building to evaluate the accuracy of this application. The test environment consisted of 26 randomly chosen rooms where the smallest room has been 5 by 3 meters. The mobile application was installed on multiple mobile phones. The fingerprints of the 26 rooms have been added to the system and another mobile device was used to generate additional measurements and determine the correct location. These measurements took place during working hours and during the night on several days. The system identified the correct location in 90% of the time. A further test showed, that a collaborative approach can complete the whole map of their building within one day if only 20% of all working people contribute.

3.5.1.8 <R.8> Robustness

In the paper Bolliger et al. write that wrong predictions have mostly been linked to the settings of the threshold value. For this experiment no additional fingerprints were added to the system to improve accuracy or cope with changes in the WiFi infrastructure. In regards of the setup measurements, they noticed, that it is often sufficient to have only one fingerprint per room.

3.6 PILS - Improving Location Fingerprinting - Motion Detection and Asynchronous Interval Labeling [6]

Bolliger et al. from ETH Zurich published the framework PILS (adaPtive Indoor Localization System). The first key challenge has been how to make users contribute to label measurements without interrupting their work routine. The second challenge has been to let the system unobtrusively keep the radio map updated over days and weeks. PILS introduces a new concept of asynchronous interval labelling. This means the system detects the movement of the user and does not ask the user while he or she is walking but asks the user afterwards about feedback to label the prior location. PILS does address the challenge of end-user labelling for WiFi indoor positioning systems. To provide a high accuracy it is important to update positioning information frequently and to have a large dataset per location. This is performed while the system detected the device as stationary without requiring additional user input.

3.6.1 Requirements Analysis

3.6.1.1 <R.1> Sensors

The technical requirements for this system are an accelerometer as well as a WiFi receiver. For their tests they used Mac Books as their supporting hardware device.

3.6.1.2 <R.2> Localization

While the system is divided into the scanner module, the locator module and the motion detector module. The scanner module is listening for available networks and their received signal strength (RSS) in the format of fingerprints. The scans are processed every 5 seconds and forwarded to the locator module.

3.6.1.3 <R.3> Scene Analysis (Clustering)

To analyse the scene, the locator module compares the current measurement with the assembled radio map stored in the fingerprint database. To reduce the data size and provide faster calculations, normal distributions are used to store multiple RSS values of a location. To localize the new measurement, the probability of the new measurement in relation to the available locations is calculated and the result with the highest probability is reflected to the user.

3.6.1.4 <R.4> Environment Representation

The user had to label the locations via the use of their Mac Books. The locations have been labelled individually and are not taken into any relation. Therefore, each location was represented by the label of a unique identifier.

3.6.1.5 <R.5> User Information

One of the key developments of Bolliger et al. has been the motion detector module. This detects the current motion of the user by evaluating the accelerometer measurements. The accelerometer sampling is performed at 5 Hz while smoothing to the last 20 values is processed to avoid incorrect classification. The transitions between moving and stationary has been able to track the intervals with a delay of 2 - 4 seconds which was due to threshold values. The motion detector informed the locator about the start and the end of an interval like transiting from the stationary interval to the movement interval. This allows to label all measurements from the standing interval to the same location. Only significant motions are classified as moving to avoid wrong classifications. By this classification, the system is able to learn continuously and unobtrusively during each stationary period.

3.6.1.6 <R.6> User Application UI

The user interaction was able due to the installation on the participants Mac Books. In the task-bar the current estimated location has been shown. It has been possible for the user to add new locations and to state wrong locations. Because of the motion detection and interval labelling approach, the system was able to ask for time intervals. One example has been ["Where were you between 9:14am and 10:05am today?"] [6]. This technique has been called asynchronous interval labelling. The user therefore does not need to label the position instantly after he stood up and left the room but when he sits down next time and has the time to reply on this feedback request of the application.

3.6.1.7 <R.7> Accuracy

In a survey Bolliger et al. evaluated their indoor positioning system. Therefore, 14 participants have been selected all working in the same office space with an area of 1,000 square meters and about 70 rooms. The density of access points has been 0.23 access point per room while typically around 5 networks have been visible in each room. The application has been installed on the participants Mac Books for one month being visible in the task-bar. This always showed the estimated location, while the user has been able to add new locations. The users were never asked about checking or improving the measurements by the system. All interactions with the system has been for their own benefit of more accuracy. The evaluation after this study identified, that the accuracy got better over time. A major amount of the new room labellings has been done right at the first day.

3.6.1.8 <R.8> Robustness

Bolliger et al. enabled the user to provide feedback to the system about incorrect identified locations. By this, the system gets more precise over time and can adjust to changes in the infrastructure. To improve robustness, measurements were only taken when the client did not move.

3.7 Related Work Matrix

The previous described related work is summarized in the following table. The columns contain the related work while the rows contain the features, sorted by the requirements R.1 Sensor to R.8 Robustness.

Requirement	Gaussian Fit [14]	Invariant Received Signal Strength [30]	IPIN Tracking Competition [19]	Joint Clustering [11]	Redpin [13]	PILS [6]
<R.1> Sensor						
Sensor Technology	WiFi	WiFi	WiFi, GPS, IMU Sensors, Cameras	WiFi	WiFi	WiFi, Accelerometer
Adding additional Infrastructure	No	No	Cameras	No	No	No
<R.2> Localization						
Setup Phase	Used notebook to scan 25 initial fingerprints at each location	Smartphone on robot measures fingerprints during the night (100 scans per location)	Pre-provided competition data contains 12 different sensors of the user	Training phase: k-strongest fingerprints at each locations were used and stored	No training phase. Users are asked to label positions during use of the application	There is no training phase since the user trains the system during use

Requirement	Gaussian Fit [14]	Invariant Received Signal Strength [30]	IPIN Tracking Competition [19]	Joint Clustering [11]	Redpin [13]	PILS [6]
Localization	RF Fingerprints	RF Fingerprints	RF Fingerprints	RF Fingerprints	RF Fingerprints	RF Fingerprints
<R.3> Scene Analysis (Clustering)						
Scene Analysis (Clustering)	Bayesian Inference and Signal Intensity distribution instead of histograms	Use I-RSS values to identify location. Algorithm to calculate disturbances	WiFi Fingerprints and Extreme Gradient Boost. User Direction: IMU dead reckoning	Baye's theorem to extract the likelihood of different locations	Least Square Algorithm to calculate the error between locations and measurement	Probabilistic model storing RSS values in normal distributions
<R.4> Environment Representation						
Environment Representation	Topological Map	Topological Map	Map of the Building	Individual Labels	Individual Labels	Individual Labels
Requires Floor Plan	Only visualization	Only for visualization	Yes for path finding	No	Only for visualization	No
<R.5> User Information						
Uses User Information (State)	No	No	Yes	No	No	No
<R.6> User Application UI						
User Application UI	No	No	No	No	Yes	Yes
User Collaboration	No	No	No	No	Yes	No
<R.7> Accuracy						

Requirement	Gaussian Fit [14]	Invariant Received Signal Strength [30]	IPIN Tracking Competition [19]	Joint Clustering [11]	Redpin [13]	PILS [6]
Accuracy	Achieved a correctness for room-level accuracy in 95% of all measurements	93% room level accuracy compared to 76% using regular measurements without I-RSS	Path location error after 7 minutes has been reduced to 24.5 meters instead of 29 meters	Achieved an accuracy of more than 90% within a distance of 7 feet (2,13 meters)	90% correct labelling of rooms which have been as small as 5 by 3 meters	Achieved room-level accuracy with an improvement against data without interval labelling
Complexity and Scalability	Requires Expert for initial setup, around 100 initial scans per location	Requires Expert and robot for initial setup, 100 initial scans per location	Required deep inspection of environmental features and the recording of data	Initial Measurements are required at each location which should later be identified	Simple to deploy and fast to integrate since all users can contribute to the system	The system requires not initial setup while the user has to label locations
<R.8> Robustness						
Robustness	Does not update data	Does update data	Does not update data	Does not update data	Does update data by user	Does update data by user
Device Calibration	No (but proposed)	No (but proposed)	Yes	No	No	No

Table 3.1: Related Work Matrix

Chapter 4

Design

To target the established requirements of the requirement summary of the analysis chapter, this chapter discusses and evaluate the possibilities for each feature of the whole indoor positioning system. These insights on decisions should help further developers and architects of indoor positioning systems to make their decisions. For this purpose, the design of this indoor positioning system is discussed in the following sections. At first, an overview of all the required design components of an indoor positioning system is given. Followed by a detailed discussion on each necessary component - building blocks.

4.1 Design Components of an Indoor Positioning System

To gather indoor positioning data and use these, various elements are required. The process from sensor input to publishing the detected location through an API. The overview of required components is visualized in figure 4.1. Initially, the sensor technology needs to be defined. The selection of sensor(s) is related to environment and the required accuracy as well as the available hardware.

Figure 4.1: Overview of this Indoor Positioning System

The raw sensor input does not provide insights about locations. Therefore, clustering is applied, abstracting features and thereby clustering locations. This provides clusters of locations which do not have a semantic linking (e.g. name of location). The semantic linking is necessary to use the cluster information in the environment of the user. Therefore, labelling of these clusters is required. As a last element, the API is required.

By this the information about the location can be provided to external parties. This can contain the information about the current location, as well as historical data.

4.2 Block 1: Positioning Hardware and Devices

One of the most important decisions is the sensor(s) used for the indoor positioning system. There is no single best choice, but rather various sensors for various types of application. While some systems easily achieve an accuracy within several centimetres, they come with an additional effort and cost to install hardware in the environment. In this thesis we focus on the applicability and scalability and minimal intrusiveness within the end-consumer environment. The sensor to select must meet a certain accuracy and should have minimal costs.

4.2.1 Overview

As introduced in the sensor analysis 2.3.1, different sensor technologies have different level of accuracy. Therefore, graph 4.2 provides an overview of the accuracy per sensor technology. The gray color identifies that additional hardware must be installed. For example optical systems need additional installations, while in the environment of the user a wifi router is already available and is therefore marked black this graph.

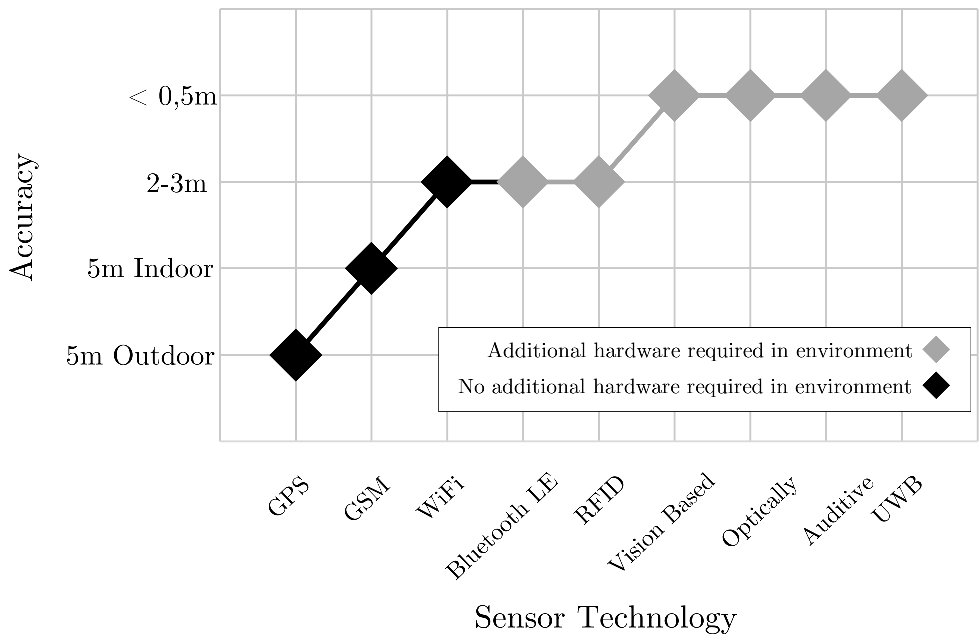


Figure 4.2: Accuracy of different sensor technologies

The goal of this thesis is room level granularity in the environment of the end-consumer. Therefore, GPS and GSM is not accurate enough for this application.

4.2.2 Installation Costs

The cost for the user increase by every hardware installation. Considering this, identifies that all of the very precise systems <0.5m come with additional hardware. This makes them less attractive to be installed in user environments. WiFi and Bluetooth LE are the remaining two candidates. They fulfil the required accuracy and do not need massive hardware installations.

4.2.3 Result: WiFi versus Bluetooth LE

Compared to Wifi, the Bluetooth LE standard is not very wide spread in home environments. This is due to the fact that Bluetooth LE beacons only have limited functionality and can thereby only signal their position and provide a limited amount of information. On the other hand, WiFi routers are wide spread since they are used for providing internet access in almost every building. Therefore, WiFi is the optimal candidate to do indoor positioning on room-level accuracy for consumers.

4.2.4 User Device - Smartphone

If possible, we want to avoid additional hardware in the environment of the user. Additionally, hardware which is already available at the user can be used since no additional costs arise. According to Dey et al. [5], the mobile phone is at the same location as the user during 90% of the day which makes it a highly personalized device. The sensors of a smartphone contain wifi, bluetooth, accelerometer and many more. Both features - technology and the personalization of the device - makes the smartphone a suitable device to be used for indoor positioning of the user. Additionally, an application can request the location of the user and represent the current user location.

4.3 Block 2: Localization using Radio Frequency Signals

Using Radio Frequency (RF) signals, different approaches can be used to make an estimate about the devices' location. All of them are explained in detail in the analysis section 2.3.2 of this thesis. Therefore, the following network diagram visualizes the strengths and weaknesses of each approach. The evaluation is processed on the following indicators:

- **Indoor Obstacles:** How do indoor obstacles influence the accuracy?

- **Multiple Sensors:** Do multiple senders increase the accuracy?
- **Possible Precision:** How precise can it possibly get?
- **Initial Setup:** How much effort is required for the initial setup?
- **Use in related work:** How often is this approach used in related work?

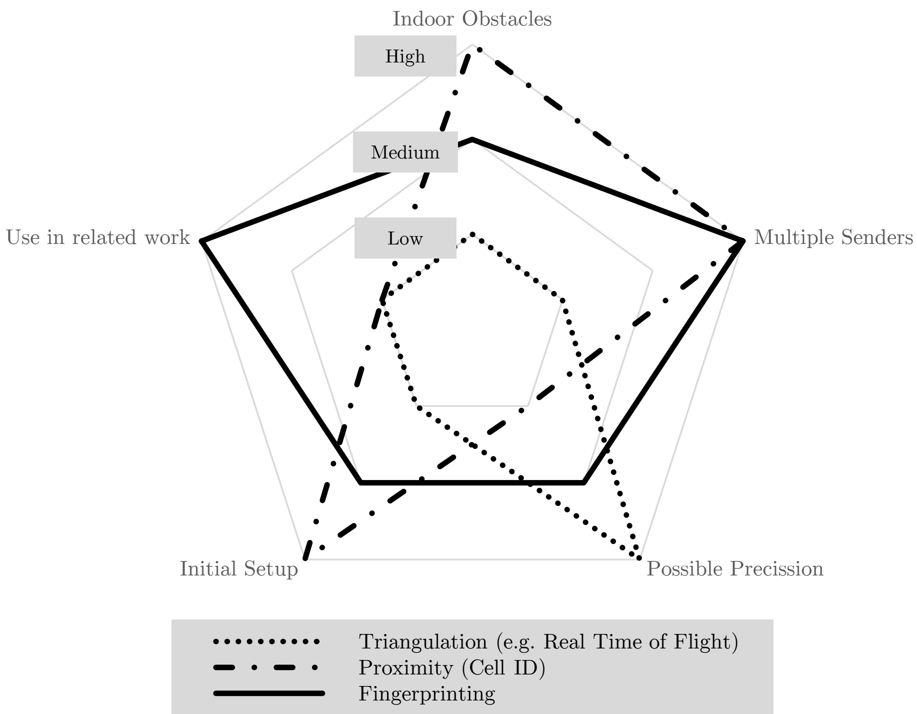


Figure 4.3: Accuracy influence factors

Each of the three techniques are discussed in the following sections.

4.3.1 Triangulation

As triangulation (introduced in the analysis 2.3.2.2) can achieve a high accuracy, this accuracy is based on no signal reflections and line-of-sight. Having obstacles such as walls and humans in the environment of the user, makes a clear line-of-sight impossible and thereby reducing the indoor accuracy. To conclude the requirement to provide line-of-sight between sender and receiver makes triangulation not suitable for the users' environment.

4.3.2 Proximity (Cell ID)

Using cell ids (see 2.3.2.1) does not get influenced much by obstacles. This is because the detected location depends on the reach of a specific network and thereby being in the cell of a network. The initial setup is very fast, since it requires only a mapping of the cell id with its location. Each cell is as large as the range of the sender and thereby the accuracy is very inaccurate. For WiFi the range per router is usually much larger than one room, thus room level accuracy can not be achieved by proximity.

4.3.3 Fingerprinting

The mechanism of fingerprinting (introduced in analysis 2.3.2.3) detects all surrounding senders and the received signal strength for each of them. This recording at one point in time is called fingerprint. While obstacles might come into the way, the received signal strength for one or for multiple senders change. This combination of received signal strengths from various sensors do contain additional information: a) which networks are available b) how do the received signal strengths of networks relate to each other (RSS are equal, stronger or weaker). These information make the location predictions more robust against changes in the environment. The approach of fingerprinting is used in most of the related work research **Bolliger2009** and is for the above reasons for this and the other techniques the most promising for this indoor positioning system.

4.4 Block 3: Data Analysis to Determine Locations (Clustering)

Each measured raw data for itself does not provide an identification of its location. The data analysis of multiple measurements can determine a group of same measurements, which is called clustering.

4.4.1 Goal and Constraints

The goal is to use a clustering mechanism, which achieves the best possible room-level accuracy. There are various machine learning and clustering techniques available which is discussed below. In addition to the constraint of accuracy, the users' smartphone is an additional limitation. Smartphones are limited on energy, calculation power and storage. All of these three factors are all required for the computation of machine learning and clustering. Usually, these algorithms perform on well formed data. This means each dimension has a complete set of data and the dimensions stay the same. While for fingerprint scans, the detected networks are changing and therefore, the raw dataset is not well formed. For this reason, the data need to be prepared before being processed by clustering algorithms.

4.4.2 Discussion of Available Clustering Techniques

The considerable clustering techniques are mentioned in the analysis section of clustering techniques (2.3.3). To discuss which clustering technique is performing best, we want to mention two important requirements. At first, the clustering must be performed continuously. That's because the training phase should not be separated from the phase of use. Therefore, the clustering algorithm must be able to continuously add new data for its training set. Secondly, the dimensions of each data pint are not limited. Each fingerprint consists of n measurements. The number n can even be changing in the same environment since routers might be not received any more and networks can be removed/added in the environment. To perform machine learning or clustering, the set of input values is usually predefined before starting the algorithm - whereas it is not the case when using fingerprints. To conclude this paragraph, the clustering algorithm need to perform well on adding additional data and need to perform well on adding additional input values.

4.4.2.1 Neural Networks and Support Vector Machines

Neural Networks (2.3.3.3) and Support Vector Machines (2.3.3.4) are working on their initially trained data values. While, an initial calculation load is given the labellings can be processed without huge calculation effort. As one goal of this indoor positioning system is to continuously add new location and additional training data, the process of re-caluclating all values needs to be processed constantly and thus the load of calculations require power and cost energy. Therefore, this is hardly applicable to be used in real-time on a mobile device. Liu et al. [9] introduce its use on stationary devices.

4.4.2.2 DBSCAN and OPTICS

DBSCAN (2.3.3.5) and its adapted version OPTICS (2.3.3.6) are especially promising because of their capability to label not well formed clusters. While this sounds promising, the sparse data of many dimensions of fingerprints reduces the correctness of this method. Each fingerprint contains a subset of the whole recorded set of networks. For example at the one side of the floor there are networks from the right neighbours and on the other side of the floor, there are networks of the left neighbour. This makes the data very sparse in some dimensions. One solution is to add data to the sparse dimensions. This can either be done via the average value, zero or another number. Another solution is to remove this dimension. But as the example illustrates, these additional data of the left and right neighbour are extremely important since they make a significant difference of the room left on the floor and the room right on the same floor. Therefore, both options of add a value for sparse dimensions and to remove the dimension are both removing a lot of important data.

4.4.2.3 Probabilistic Method and k-Nearest-Neighbours

K-Nearest-Neighbours (2.3.3.2) is selecting the k-closest data points and uses them as a reference to label the new data point. This is done by calculating the error for each dimension of the new data point to the available data points. The k-closest are identified as the ones with the smallest error. The benefit is, that labelled data points can be added at all times and do not add an additional calculation effort. Still the regular calculation to identify a network is more expensive since it needs to iterate through all the data points and calculate the error.

The probabilistic method (2.3.3.1) is using a similar approach of comparison and identifying the smallest error. Initially, the new measurements are grouped. This is done by not storing the data pints as individual values, but to cluster them as a Gaussian distribution defined through mean and standard deviation. As identified by Yang et al. [24] this abstraction reduces the calculation effort to identify the location of a new data point.

4.4.3 Conclusion

To avoid sparse data, the number of networks which are very rarely available needs to be as minimal as possible. Therefore, a threshold of minimal accepted RSSI values needs to be set which has been introduced by [11] and [30]. The clustering algorithm must perform on a subset of dimensions. Because it can not be assumed that the the dimensions of the new data set and the training data set are completely matching. Therefore, we implemented k-nearest-neighbours with the data preparation using the probabilistic method surveyed by Yang et al. [24].

Benefits of the probabilistic method are that the cluster of labelled data is stored very compact. In addition calculations need to be done on less data, since the compact data of mean and standard deviation can be used. Each location can consist of several sub-clusters. This adds the benefit of accounting for different size of locations. For example a living room might be multiple times larger than another smaller room. Having only one cluster per location would make it more difficult to distinguish between such locations.

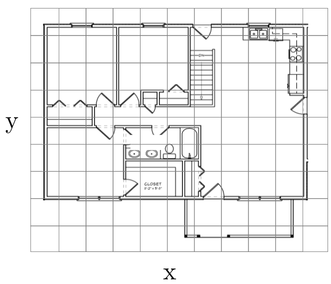
On this compact data set, an adapted algorithm of kNN calculates the error between the stored labelled data and the new measured fingerprint. The error are calculated per sub-cluster while all sub-clusters per location are averaged. This average error per location is combined to the other locations allows the ranking of all locations per measured fingerprint. A scheme of the exact flow is available in the implementation chapter 5.

4.5 Block 4: Digital Representation of User Environments

The users' location needs to be visualized in this indoor positioning system in order to enable the user labelling his current location. The Cartesian coordinates and room-level labels for location identifications have been introduced in the analysis section (2.3.4). Using GPS positions, the location is defined through longitude and latitude. In a building this could be applied by using x, y and z values in a Cartesian system. For indoor positioning the user needs to relate to a room, which makes it difficult for x, y and z coordinates. The z coordinate defines the height and thereby the floor level while x and y are the coordinates on the ground. It needs to be considered that this segmentation into coordinates as shown in figure 5.1a provides a segmentation sub room-level. For the user, individual Cartesian positions are hard to grasp. The user is more familiar with room level localization and names, while more detailed information can lead to confusion of the user.

4.5.1 Hierarchical Labelled Structure of Locations

Using only labels makes it difficult to get the rooms in a overall context. In a large building, rooms are typically enumerated which makes this more simple. In addition, a hierarchical tree of locations and sublocations can make clear references. This works, even though rooms might have the same name. An example can be a kitchen on the first and a kitchen on the second floor. Using this hierarchical approach as visible in figure 5.1b, the kitchens can still be separated using paths like ground floor / kitchen and upstairs / kitchen or 1st floor / kitchen. This approach is using labels and is ordering them hierarchically. Thereby, additional information are contained which can further be used for path finding through buildings.



(a) Location Coordinates

Home
- Lisa's Room
- Max's Room
- Kitchen
- Master Bedroom
- Bath
- Closet
- Living Room
- Deck

(b) Hierarchical Semantic Labelling

Figure 4.4: Possibilities to do Indoor Location Labelling

4.6 Block 5: User Involvement

The user involvement for our indoor positioning system should be as minimal intrusive as possible. For this reason, the user should not make an initial recording for all rooms but should from time to time provide location feedback to the indoor positioning system. Combining these location information with the automatically recorded fingerprints a representation of locations in fingerprints can be used. A very important question to be raised is how the user interacts with the system. The section of user involvement is separated into the following parts: 1. Motion detection using the users device. 2. Reduced User Interactions - Notifications. 3. User Interface and Data Visualization.

4.6.1 Motion detection to Indicate Movements

Whenever locations are changing, the user had to move to change the location. The possibility to identify movements can add additional information which are completely distinct from any fingerprinting methodology. These information can be used to identify which recordings clearly belongs together. As an example: if no movement has been detected for 10 minutes, all the WiFi recordings belong to the same location. This methodology has been published by Bolliger et al. [6] at the ETH Zurich.

All information which the user does not have to provide actively, are reducing the workload for him. Therefore, to identify the moments when the user is moving and standing provides very valuable information. Whenever the person is moving, it identifies, that the recordings do not need to belong together, while all recordings done in a non moving state of the device have been processed at the same location.

Fortunately, an accelerometer sensor is available in any modern smartphone. A sliding window algorithm to smooth the sensor data with a certain threshold to identify the movements precisely predicts when the device and thereby the user is moving which is demonstrated in the graph below. Test with the Android Activity Recognition API identified, that their activity recognition works well for longer durations of activities, but has been to slow to identify whether the user has changed the room which needs to be identified within 5 seconds.

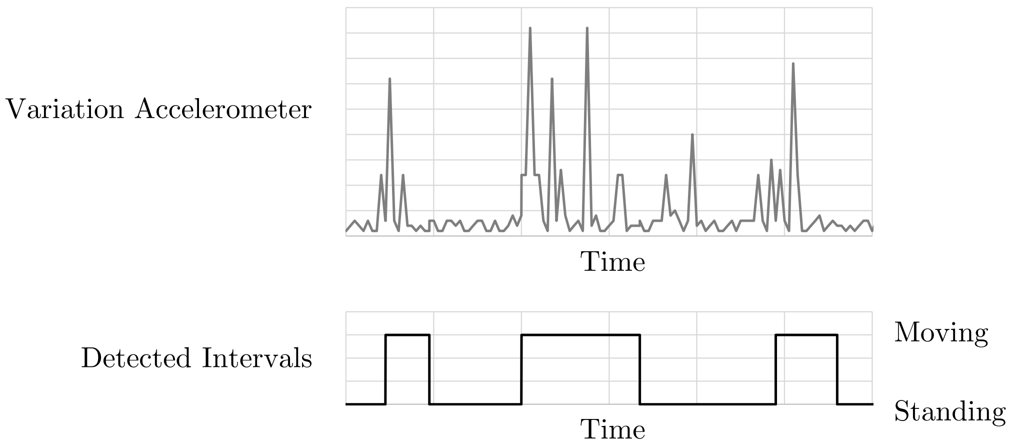


Figure 4.5: States of movement and standing through accelerometer values

4.6.2 Reduced User Interactions - Notifications

User interactions with smartphones are possible in two ways: a) internally motivated: the user opens an application, because he is motivated to send a message or open a website etc. The other possibility b) is that the user gets triggered externally, which could be an incoming call, an incoming message or any other notification of the phone. Google implemented an notification feature into Google Maps. It detects when the user has been at a distinct location for a certain time and asks him to provide feedback. This request gets pushed via notification on android devices. The text in the notification title says: "rate your visit".

Figure 4.6: Google Maps Notification - Rate your visit

For our indoor positioning system, notifications are a reduced intrusive method to get feedback from the user about his position. The feedback requests can be made whenever the algorithm needs to get more information to become more precise in the prediction of the specific location. This can be for the reason of adding a new location or the reason to improve the prediction of existing locations. These are two different types of notifications with different types of possible user actions.

4.6.2.1 Notification to Add a New Location

The notification to add a new location must be shown when the clustering algorithm detected a new location and is certain that this location is relevant to the user. This means the user should not be asked continuously for his location, but only at locations of his personal interest because it is a place which is often or regularly visited. The algorithm knows about the movements of the user and thereby about the moment, when a user is leaving the current location. This is the moment, where a notification can ask about the location the user just moved away. The notification to request this information does look as follows:

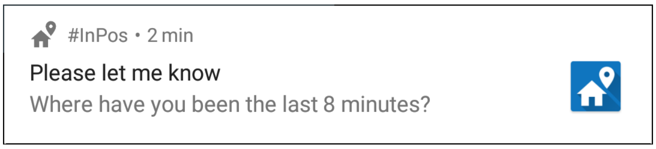


Figure 4.7: #InPos Notification to rate new location

If the user is selecting the notification in order to provide feedback for the current location, he is referred to a new page. This page contains a hierarchical list of all previously inserted locations. In addition, new locations can be added. To provide the rating of a location, this distinct location needs to be selected and saved.

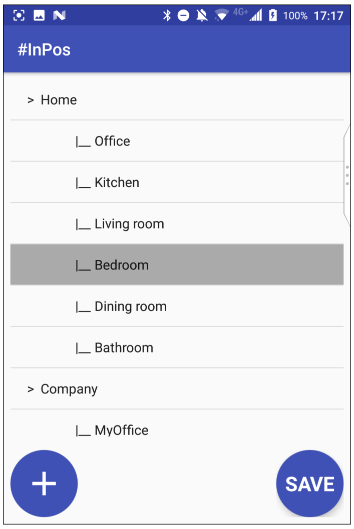


Figure 4.8: #InPos Activity to select the current location

4.6.2.2 Notification to Improve Existing Location Accuracy

Because of changes in the environment, the detected fingerprint is not constant at a specific location. Therefore, the algorithm needs to add additional labelled data to update a location and improve accuracy. For this reason, notifications can request these localization data from the user. To keep the user effort as minimal as possible, the notification already contains three possibilities to be selected by one click.

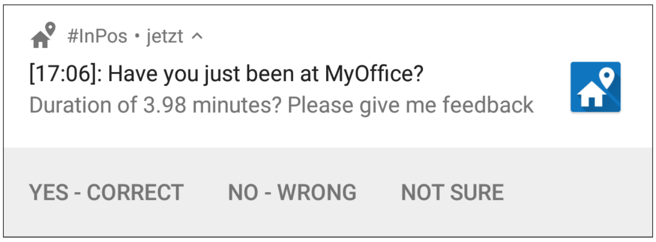


Figure 4.9: #InPos Notification to rate the detected location

- **Correct Location:** The user can confirm that the detected location is identified correctly.
- **Wrong Location:** The current identified location is not correct. In case the user is selecting this option, a page opens to select the current location. This is the same page as for adding a new location shown in figure 4.8.
- **Not Sure:** In case the user is not sure if the location has been correct, he or she can skip the notification by selecting this action.

4.6.3 User Interface and Data Visualization

Last section discussed the notifications which are triggering the user externally. In this section the internal motivations for the user are discussed. Two main internal motivations of the user can be identified: a) check the current detected location and history of locations b) take action to improve the correctness of detected locations.

4.6.3.1 Check Detected Location and History

To give the user a reason to interact with the application, there must be some kind of motivation for him. Nowadays, the user is quite eager about the self quantification such as wearable wrist bands providing fitness and health data. These applications provide an overview about the current status as well as a history about the past. These fitness tracking wearables can be connected to an indoor positioning system. Both should be as minimal intrusive as possible, provide as accurate data as possible and should keep the interest of the user to not remove this application from his smartphone. To address the user the same way as fitness tracker to, our indoor positioning has both views. The view of the current location can be seen in figure 4.10. The user does only see the surrounding locations, which are most probably at this moment - rated with a percentage.

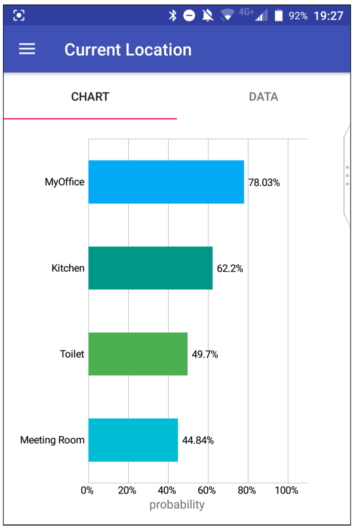
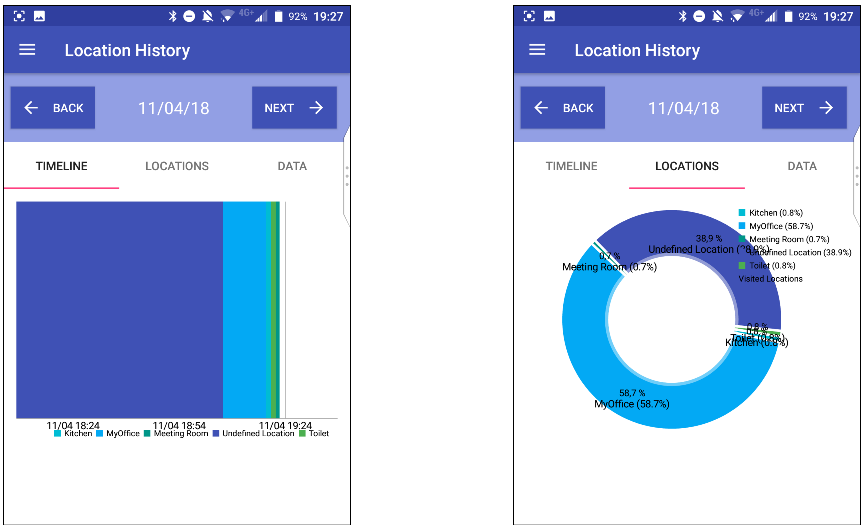


Figure 4.10: #InPosVis to inform the user about the current detected location

The history of visited locations can be provided in various ways. Showing the complex raw data to the user does not show him the information on a quick look. Therefore, the data need to be represented in some visualization. Various graphs are possible to do this visualization. Finally, the two charts which can be seen in figure 4.11 were chosen.



(a) Fragmentation History View

(b) Pie Chart History View

Figure 4.11: InPosVis History Visualizations

4.6.3.2 Take Action - Improve Correctness

After the user gets attracted by our indoor positioning system, it would be favourable to enable him to add additional locations whenever a new location is not detected or is not labelled correctly. This can be done by notifications as mentioned in the sections before, but in case the algorithm did not already identify a new location and trigger a notification, the user needs to be enabled doing labellings whenever he would like to add them. These labellings are called labelled recordings. Such labelled recordings can be processed via the application and be triggered by the user. Therefore, the user selects the specific room in the application and starts to add fingerprint recordings by a simple click on the recording button in the user interface.

4.7 Block 6: API to Distribute Location Data

While location data are nice feature for a user, the use-cases are still limited if this indoor positioning system would keep the information to itself. The real benefits arise, as soon as the information of the users' location are published to the users' environment. Then the environment can act on the basis of these location information. By environment two possibilities arise:

- **Locally** on the same mobile phone. This enables other applications to act in regard of the current location. An example would be that an application could send less notifications when the user is at work. Further, the UI of an application could look different. For some applications this is already done in the context of driving where the user interface has a reduced design and buttons get larger.
- **Publicly** the mobile phone can send the location to all services in the environment outside the own mobile phone. This enables the environment to take actions and avoid others. An example can be that the home environment knows where the user is located and can adjust the environment to a pre-set theme. Others can be the environment is switching on and off lights, depending on the user location.

Both of the above described applications should be able to request an API. The application could simply publish their data on request. This method can cause a lot of additional calculation since external services have to request the location using polling. Therefore, a subscription service on side of the indoor positioning can notify subscribers about updates whenever they happen. This reduces the amount of send messages and enables the environment to get notified. Because all is done centrally at the indoor positioning service, this service can decide which requesting service should be entitled to receive the indoor positioning data of the user. The implementation details are further described in the Implementation section at 5.7.

4.8 Resulting Solution - #InPos

Finally, the previous discussed building blocks can be combined into one indoor positioning system. This system starts with the input of the sensor using WiFi and the accelerometer.

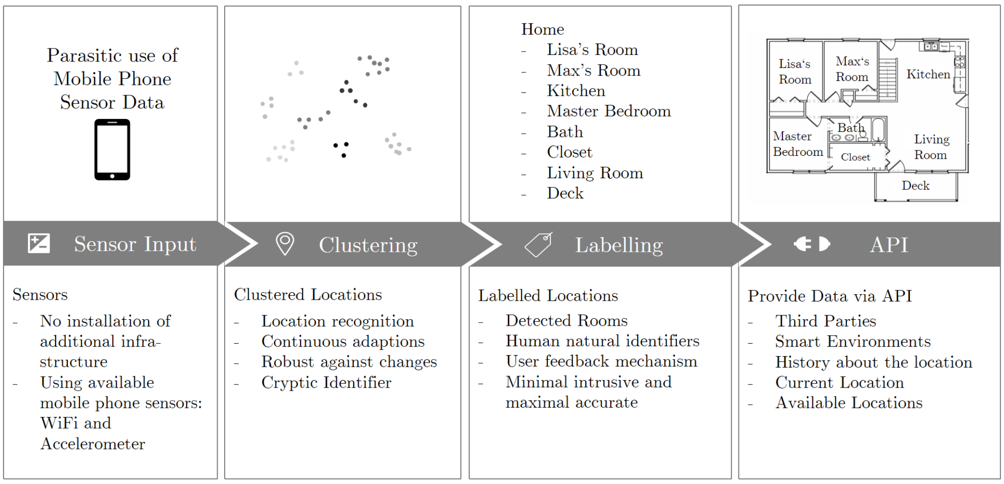


Figure 4.12: Complete Overview of this Indoor Positioning System

The accelerometer is identifying whether the user is still or moving to provide the clustering algorithm additional data. The clustering of the recorded WiFi data is done using k-Nearest-Neighbours. To reduce the calculation effort, the individual data points are grouped using a Gaussian distribution per data point group. The grouping is defined by the detected states (still/moving) of the user which are detected through the use of the accelerometer data. In addition, the clustering algorithm can request user feedback. Thereby detected clusters get labelled through the feedback of the user who can organize his environment via hierarchically structured semantic labels. After this labelling is done and locations are detected, these locations are published through a publishing service running as a subscription server on the users smartphone. By this, other services can subscribe themselves for position updates. A diagram of the complete overview can be seen in figure 4.12.

Chapter 5

Implementation

5.1 Introduction

This chapter provides an overview of the implementation and highlight the most important features. This thesis finally consists of two android applications. These can be installed individually, while the main application is #InPos which stands for Indoor Positioning. The second application is #InPosVis which is an application purely visualizing data from #InPos.

(a) #InPos

(b) #InPosVis

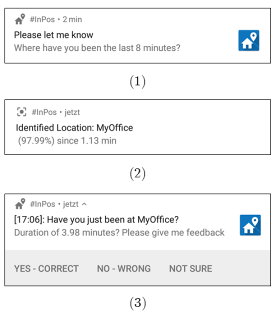
Figure 5.1: Application logos of this thesis

Furthermore, this indoor positioning system should be able to be reused and improved in the future. Therefore, it is planned to be published and open sourced.

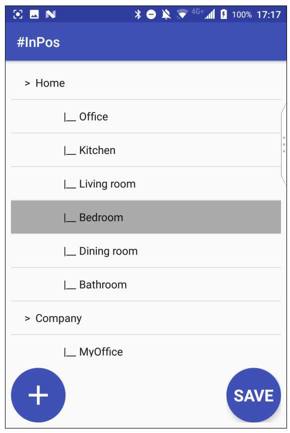
5.2 #InPos - Indoor Positioning

The #InPos application runs as a background service recording WiFi fingerprints and detecting movements of the user. The user is getting notified whenever the clustering algorithm wants to receive feedback in order to improve the accuracy or in order to add a new location. This application additionally offers the possibility for the user to add additional recordings for a selected location manually. Finally, this application contains much more features, which have been used to develop, test, iterate and improve the application.

In figure 5.2a screenshots of the implemented android notifications are shown. These are used as external triggers in combination with a vibration alert to catch the users attention. The select location activity (figure 5.2b) gets displayed as soon as the user clicks on the notification.



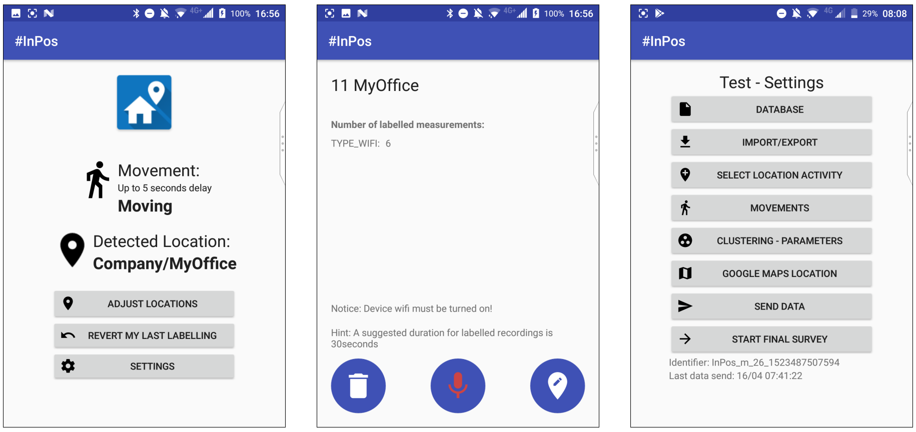
(a) #InPos - Notifications



(b) #InPos - Select Location Activity

Figure 5.2: #InPos Notifications and Location Selection

The following figure shows different android activities, which are used to interact with #InPos. The first activity (figure 5.3a) is the main activity, and gives the user an overview of current detections and the possible interactions. The centred activity (figure 5.3b) is used to make additional recordings per location, change the name or delete data. The recording button is switching to red, if the user is processing a labelled recording adding data. The right activity (figure 5.3c) shows further possible interactions, which are hidden for the user and only gets visible using the developer mode. This provides access to the database, movement detection algorithm and other parameters.



(a) #InPos - User Menu Activity (b) #InPos - Labelled Recording of Location (c) #InPosVis - Developer Menu

Figure 5.3: #InPos Activities for User and Developer

5.3 Reusable Framework

As discussed in the design chapter, an indoor positioning system can consist of different variants in each building block. For example various technology can be used for positioning and individual clustering mechanisms. Aiming to enable further developers in this area, this work is build as a framework which allows to exchange sensor technology and clustering algorithms easily. Therefore, each input sensor consists of the recording mechanism to generate the input data from this sensor. In addition, a specific clustering mechanism is added which operates on the data of the specific input of this sensor. Each of such **sensor - clustering** combination is called **positioning**. The class diagram (figure 5.4) does describe its design in more detail.

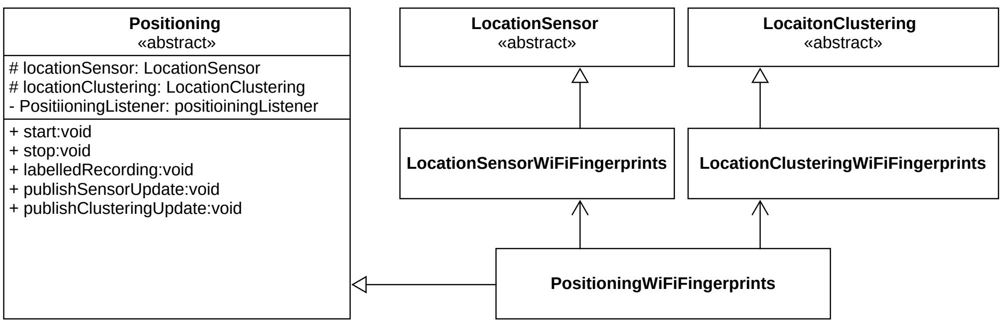


Figure 5.4: UML diagram of positioning handler and its sensor and clustering

Positioning enables to add various different sensor - clustering combinations (Location-

Sensor)

- WiFi sensor of smartphone (Analysis Wifi: 2.3.1.3)
- Bluetooth LE sensor of mobile phone (Analysis Bluetooth LE: 2.3.1.4)
- Android Location System build in mobile phone (Analysis Android Location System: 2.3.1)

Specific Clusterings (LocationClustering):

- For Bluetooth and WiFi Fingerprinting the adjusted kNN clustering has been implemented as discussed in the design chapter 4.4.3.
- Since the Android Location Service provides longitude and latitude measurements, fingerprinting is not applicable and therefore no clustering.

To manage all external commands such as start measurement, stop measurement, process a labelled recording - an additional controller is set on top. All **Position** objects are registered at this controller called **PositioningHandler**. The individual actions per command are executed within the **Position** object, which is delegating this to its sensor object and clustering object.

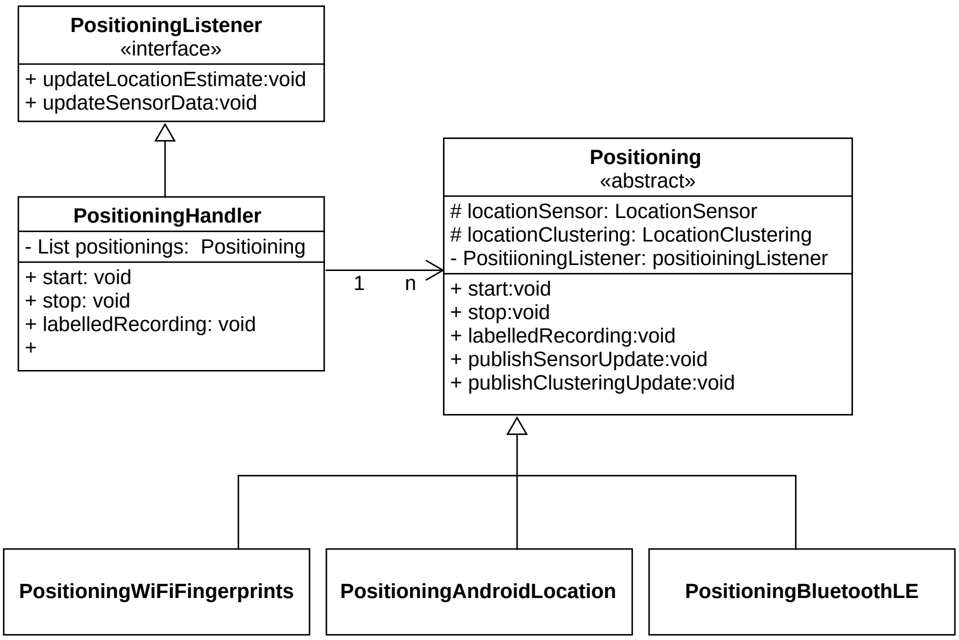


Figure 5.5: UML Diagram of PositioningHandler and various sensor positionings

Internally, the **Position Sensors** are publishing new recordings. These are published to the **Position** object which triggers the **PositioningHandler** that this sensor has an updated recording. Thereby, the **PositioiningHandler** is informing all **Positioining** objects and inform them that a specific sensor received updated data. With this information, the individual **Positioining** objects decide if they start to run further processing such as process a clustering or adapt their clusterings on the new data.

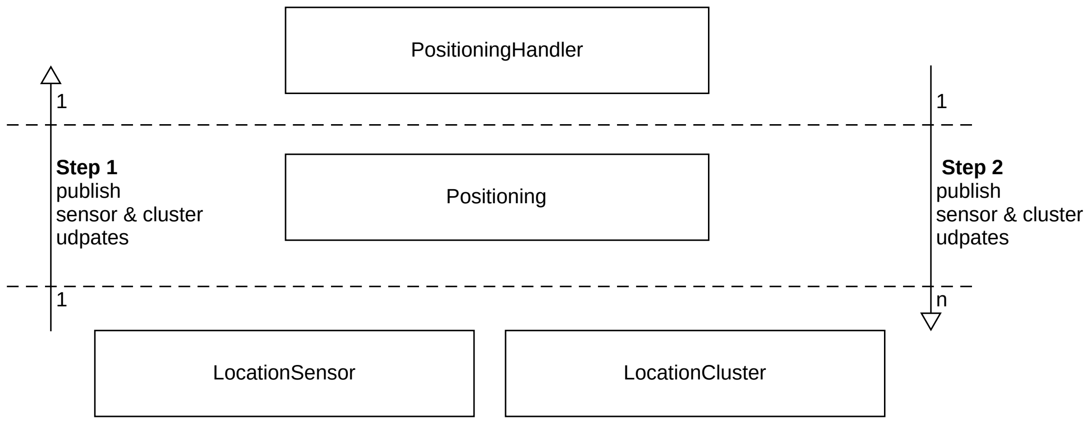


Figure 5.6: Layer Diagram of how information are propagated

This object and communication structure makes it simple to exchange individual components and replace them by another. This works for clusterings as well as for sensor-types.

5.4 Process of Clustering

This section provides a more detailed insights into the implementation of the clustering algorithm. There are two main task a) adding data: detect new locations based on a set of recorded fingerprints and b) localization: to identify to which location a recorded fingerprint belongs. These two processes are described in the following two sections.

5.4.1 Location Detection Process

Diagram 5.7 describes the process of adding new fingerprints and the evaluation if these belong to an existing location or might be the fingerprints of a new location. Since recordings of new locations are only done if the smartphone is at the same location, the movement detection is indicating if the device is still. If it is still, all the fingerprint recordings are added with this distinct interval_id. This interval_id enables to further select all fingerprints which have been recorded during that interval. Using the already introduced probabilistic method, the size of recordings per interval is reduced by gather multiple fingerprints and store them as one grouped fingerprint, containing mean and standard-deviation for each RSS measurement. This provides a set of grouped fingerprints for the same location.

The grouped fingerprint is then compared to the database of identified labelled locations. In case this gives no clear result, the algorithm can request user feedback to gain more certainty. Finally, the clustering generates an additional data point for that location which is added to the database and is used for the next labelling.

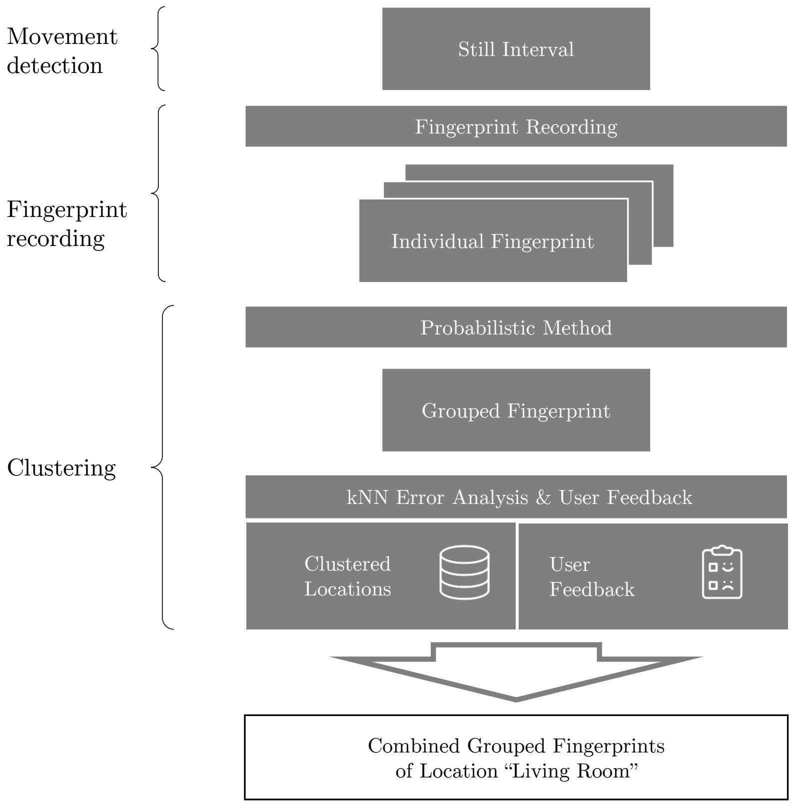


Figure 5.7: Diagram of grouping fingerprints and cluster locations

5.4.2 Fingerprint Labelling Process

While the previous described process creates and updates the database of locations and fingerprints of locations, the next process describes how the location labelling of a recorded fingerprint works.

The fingerprint is recorded independent of the interval. The condition if the user is moving or standing within this interval changes the scope of the input fingerprints. If the user is moving, the past fingerprints are only slightly considered, while if the interval is still, previous fingerprints of the same interval are considered, because the combination of them provides a more comprehensive dataset.

These fingerprint data are then evaluated by the adapted kNN algorithm, described in the design section 4.4.3. For each location, the error is evaluated and the current location gets predicted.

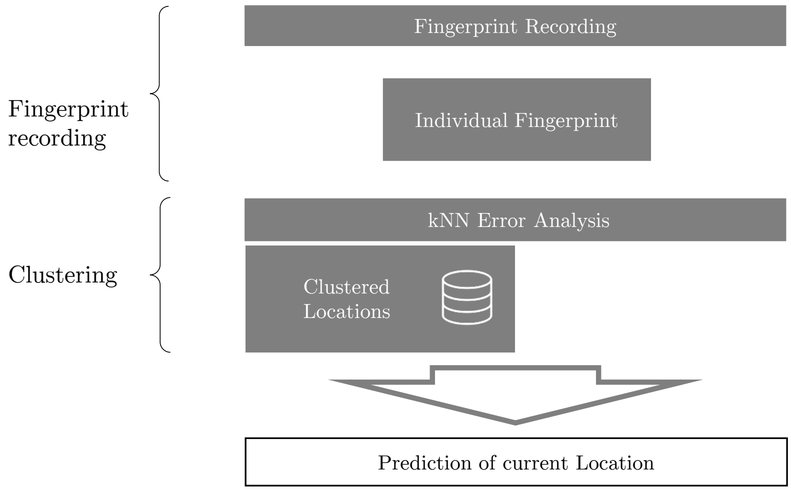


Figure 5.8: Diagram of fingerprint localization

5.4.3 History

The history of locations are additionally considered, when a new location estimate is calculated. For example, a person is getting into the way between the smartphone and the router. This immediately influence the RSS and the clustering algorithm might estimate another room. Smoothing on the level of results is used to correct such false predictions using the sliding window mechanism. This reduces the responsiveness because of the window size, but reduces the correct identified locations.

The complete history of locations is stored into the database, which enables API queries to read on previously visited locations.

5.4.4 Clustering Parameters

The complete clustering process for fingerprints is using parameters. Therefore, further adaptations, changes, and tests can made quickly. The below table gives an overview of the default values.

Type	Parameter	Default Value
int	AVOID_FIRST_N_ELEMENTS_TO_REDUCE_ERROR	1
int	NUMBER_OF_COMPARED_MEASUREMENTS_PER_LOCATION	20
int	DURATION_OFTHEN_LABELLING	5*60*1000
int	INTERVAL_SELDOM_LABELLING	150*1000
int	INTERVAL_OFTHEN_LABELLING	10*1000
int	ESTIMATE_HISTORY_CHECK_SECONDS_DELAY	45
int	maxNumberOfRecordedScans	25
int	maxNumberOfComparedScans	15
int	minNumberOfComparedScans	3
int	minSizeOfCluster	2
double	minAppearanceOfScans	0.0
int	minIntensityOfScans	-90
double	maxGroupError	1.5
int	fingerprintMaxCount	1
float	THRESHOLD_BELOW_LOCATION_IS_NOT_IDENTIFIED	0.1f
int	DURATION_BEFORE_REMOVE_NOTIFICATION_OF_RATING	15*60*1000
int	DURATION_BEFORE_REMOVE_NOTIFICATION_OF_REQUEST	30*60*1000
long	TIME_UNTIL_WHEN_RECORDING_CAN_NOT_BE_USED	16*60*60*1000
long	DURATION_OF_INTERVAL_BEFORE_CHECK_FOR_REQUEST	3*60*1000
long	TIME_AFTER_WHICH_UNLABELLED_DATA_CAN_BE_REMOVED	3*24*60*60*1000

Table 5.1: Fingerprint Clustering Parameters

We provide this table as an overview, while more detailed information about the impact of each parameter can be found in the implementation of this indoor positioning system.

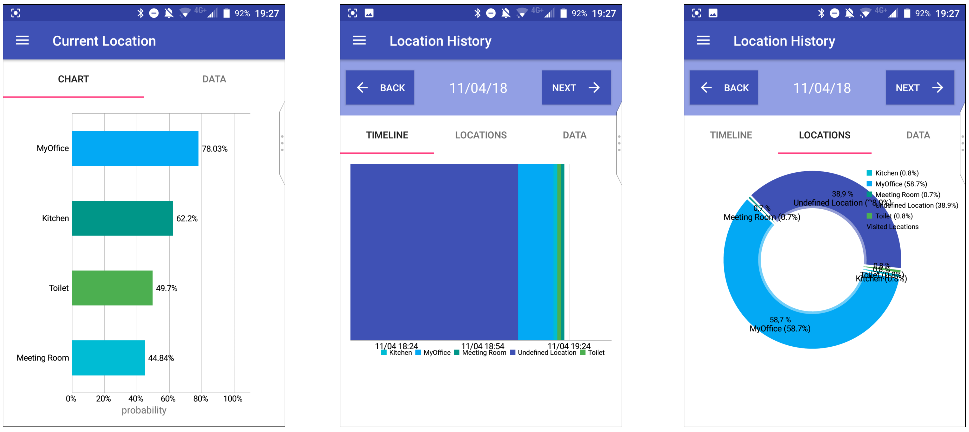
5.5 External Libraries

For both Android Applications, the Android SDK has been used. In addition, two libraries have been integrated.

- **org.apache.commons** has been used to run the server hosting the API.
- **org.nanohttpd:nanohttpd** has been used to process http requests to the API server.
- **com.obsez.android.lib.filechooser:filechooser** has been used to pick files and folders during the process of importing and exporting the database in #InPos
- **com.github.PhilJay:MPAndroidChart** has been used for the graphical elements in #InPosVis.

5.6 #InPosVis - Indoor Positioning Visualization

While #InPos is mandatory, #InPosVis adds additional information for the user but is not required for the clustering or positioning of the device. The first activity (figure 5.9a) shows the current location using percentage indicators of the certainty per location. The history data is visualized using a fragmented bar chart (figure 5.9b), to show the visited locations on the time axis. A pie chart (figure 5.9c) is displaying the ratio of visited location per day.



(a) #InPosVis - Current location

(b) #InPosVis - Fragmentation history view

(c) #InPosVis - Pie chart history view

Figure 5.9: #InPosVis Visualizations

#InPosVis is connected to the data of #InPos using the API which is introduced in the following section.

5.7 API Implementation

In order to provide other applications access to the location data of #InPos, a subscribe and publishing service for the current locations is implemented. This service is a webserver where external applications can request the current location. The response is a json string which can be parsed to either visualize these data or use these data for other location aware services. External webserver can subscribe themselves to receive messages on the change of the devices location. An example of the API functionality is the application #InPosVis.

The API is reachable at port 8776. Thereby, a request can look like: http://192.168.0.227:8776. Since this address is only reachable from the local network, a universal reachable server can be registered at #InPos. Thereby, #InPos publishes the location to a server which is not limited by a local network. The next table gives an overview about the reachable endpoints.

Path	Parameter
/	-
/CURRENT_LOCATION	-
/CURRENT_INTERVAL	-
/LOCATION_HISTORY	from, to, granularity
/INTERVAL_HISTORY	from, to, granularity
/REGISTER_UPDATE	address, services
/UNREGISTER_UPDATE	address, services

A complete request to subscribe a server (myservice:3030) at #InPos (192.168.0.227:8776) can look as follows:

http://192.168.0.227:8776/REGISTER_UPDATE
?address=http://myservice:3030
&services=[*interval_listener,location_listener*]

For example, this API can be used by frameworks for smart environments like meSchup [1] and DS2OS [2].

5.8 Testing of the Indoor Positioning System

In order to evaluate our indoor positioning system, a study where real users are using this indoor positioning system should be processed. To extract insights about the user behaviour, their activities and the sensor inputs have been recorded in a database. These data are then used for the evaluation of this thesis.

The study setup contained an initial request for the mail address and age of the user. The user data have then automatically be uploaded to a central server every 12 hours if the user has been connected to a WiFi network in order to save his mobile data. These provided regular insights into the current progress and indicates whether users are still active in the user study.

After the end of the user study, the data have been processed via sql queries to extract important measurements for the evaluation in a structured way.

Chapter 6

Evaluation

This indoor positioning system is intended to be a novel approach to gather indoor positioning data. Therefore, two factors are especially relevant: 1. the evaluation of the user intrusiveness and 2. the achieved location accuracy. The evaluation is structured in the following sections: At first, the setup of the evaluation containing a user study is described. Second, the interaction of the user is evaluated. The interactions are divided into externally motivated interactions (e.g. notifications) and internally motivated interactions (e.g. the interest to check the predicted location). The third section of this evaluation is about the labelling process. The labelling process is especially evaluated on the number of identified locations and on the accuracy of the predictions. The labelling process evaluation is followed by the usability evaluation of this indoor positioning system - including user feedback.

6.1 Evaluation Setup - User Study

The goal of the user study has been to evaluate the indoor positioning system in different environments while being used through real users. The study especially focused on the user interactions and detected locations. To gather insights into the initial setup and the use of the indoor positioning system thereafter. Because the user is visiting most of his relevant locations on a daily basis, the user study has been conducted throughout four days. During this time period, the indoor positioning system has been running as an application on the mobile phone of nine participants. Thereby, the system detected new locations and asked the participant to label them. In addition, the system predicted the position of the user. The user has been able to add additional measurements in order to increase the accuracy of the system. At the end of the study, the participants were asked to answer a System Usability Scale (SUS) test. The SUS test is a standardized test consisting of 10 predefined questions. Thereby, the usability of the system can be evaluated. One day after the SUS test, a survey of questions for qualitative answers has been conducted by the participants.

6.2 Evaluation User Interaction

Each system receiving or providing information does require some kind of interaction. Both, inputs and outputs can be part of a system. In the particular case of this work, the input (labelling of locations) heavily influences the output (accurate location) of the system. Therefore, the user interaction is crucial to enable the user to labelled locations and increase the number of detected locations and prediction accuracy.

6.2.1 Internal and External Motivation

Users can be motivated internally and externally to interact with the indoor positioning system:

- **Internal motivation:** This can be the personal interest of the current location detected by the system. Therefore, the time to check and read data is measured. In addition, the interest of the user to improve the location prediction has been recorded. Therefore, all manually added location recordings were measured by the evaluation layer.
- **External motivation:** These interactions are triggered by external factors. For this indoor positioning system, these have been notifications asking the user for feedback about his location. Thereby, the user had the chance to label the current location, while the user defined which notifications were answered and which were not.

6.2.2 User Interaction Measurements

The interaction of the user has been measured separately by using internal and external motivation as described in the previous section. The following graph shows the average time spend per day on the application. Users have been internally motivated to a) inform about the estimated location and b) to record additional labelled fingerprints. The externally motivated task is to answer the notifications. Both - internally and externally motivated actions - have been recorded.

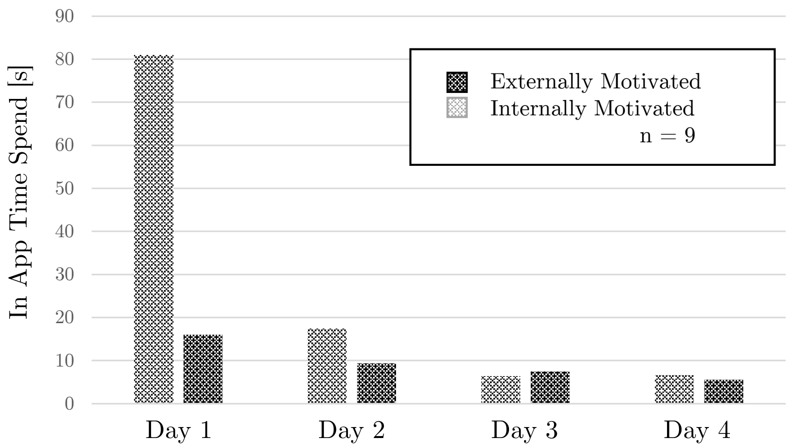


Figure 6.1: Average Time Spend in Application per Day

As can be seen in graph 6.1, the user does spend initially more time to interact with the system than at the following days. One reason for this is the initial setup. Here the user had the chance to add locations to the positioning system. These locations are then already available to quickly answer the request to label the current location. Over time most locations are known to the system. Therefore, the demand of asking to label new locations is reduced. Another reason might be that the interest of the participant for this study decreased over time.

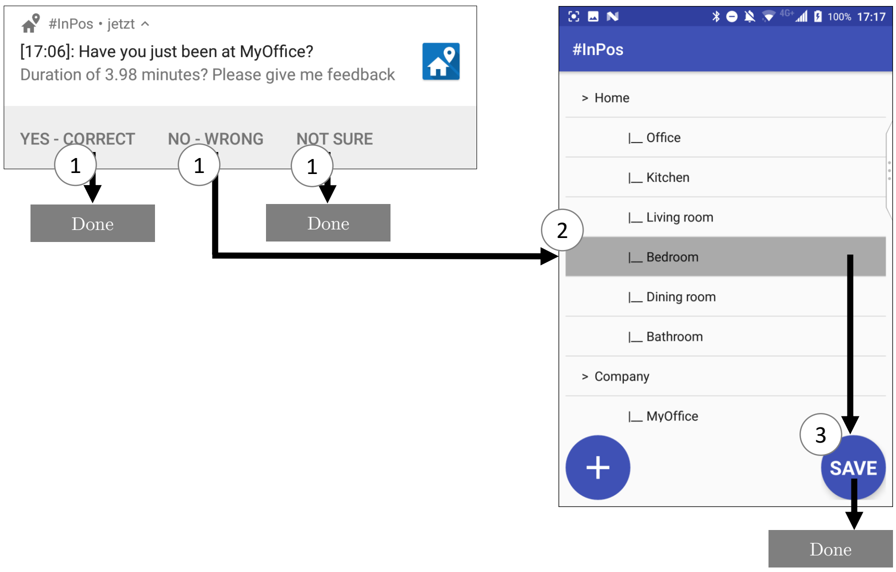


Figure 6.2: Feedback notification with possible number of clicks

Figure 6.2 is visualizing the notification and the possible user actions. Thereby the user can reply to a location request of the positioning system. If the identified location in the notification is correct, the participant can answer the feedback request by only 1 click. If the user does not want to answer the notification, it was possible to press "not sure". In case the detected location is wrong or not known, the click count increased to minimal 3 clicks. Additional clicks were required if a new location is added to the hierarchy of locations.

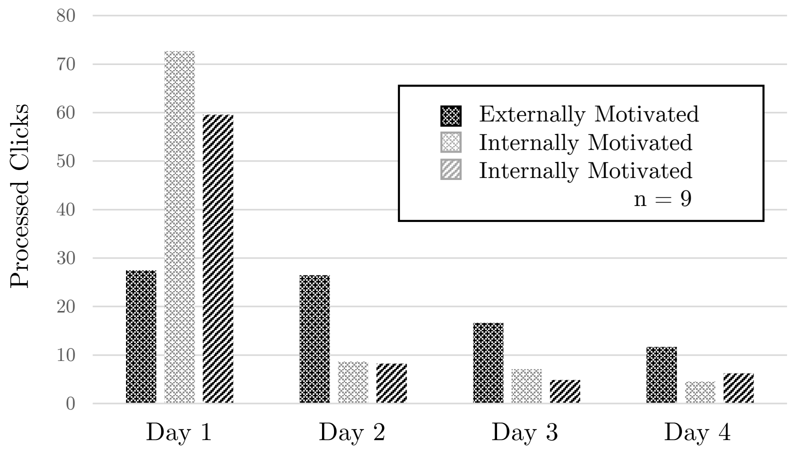


Figure 6.3: Average clicks in application per day

The average numbers of user interactions per day can be seen in the following table:

Day	Average Number of Notifications	Average Ratio Answered Notifications	Average Duration to Select a Location [s]
Day 1	13.6	75.67%	4.85 seconds
Day 2	19	67.27%	5.52 seconds
Day 3	10.9	66.22%	3.29 seconds
Day 4	9.3	72.88%	2.24 seconds

Table 6.1: Average User Interactions with Notifications

The table indicates, that the average ratio of answered notifications remained constant over time. The average ratio of answered notifications decreased over time as well as the average duration to select a location. Both columns have a peak on the second day of the user study. For the average number of notifications, this can be due to the increased need of the algorithm to gather labelling information in order to improve the precision. The average duration peak on the second day could be due to the additional effort of the user to add an additional location which must be added once. After the new location is added, it does appear in the list of locations and can be selected faster. This conclusion suits to the fact that the average duration per labelling is reduced over time. At the same time less locations are added as described in the section of detected locations 6.3.

6.2.3 Error Quantification

The users were asked to interact with the indoor positioning system answering notifications whenever they want. To see whether the user interaction continuously decreases the error, the following plot is drawn.

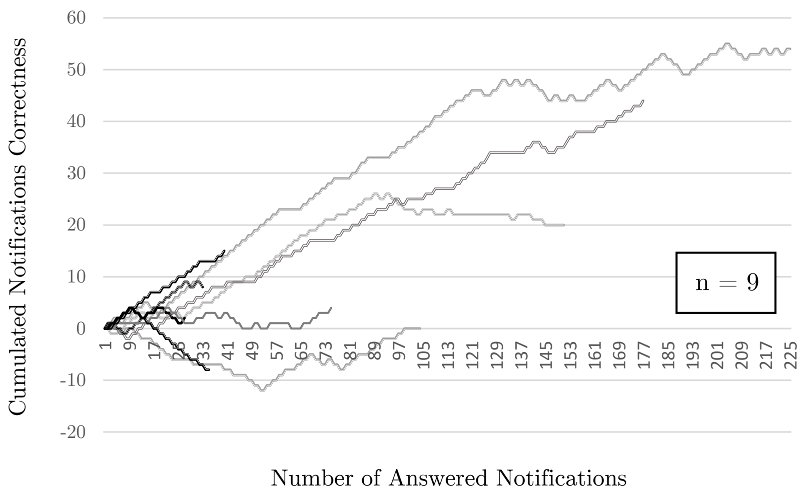


Figure 6.4: Correctness vs. user interactions (answered notifications)

On the x-axis, the number of answered notifications are shown and combined with the number of confirmed notifications on the y-axis. These are calculated as follows: If the notification asked for the correct location the counter by 1 increased while it has been decreased by 1 for the wrong predicted location. In case the user clicked on "not sure", the counter remained the same. Each line is one participant of the study and the increase or decrease of accuracy is visualized. Two signals turned to be negative while two others remained horizontal to the x-axis. These signals have less answered notifications than the signals which have a positive correctness. The signal with the negative initial correctness is turning positive again and reached the x-axis after 100 answered notifications. Therefore, it can be assumed that this signal turns even more positive and provides a more correct labelling in the future. For the under performing signals, a reason can be environmental influences for fluctuating WiFi signals. Evaluating these measurements lead to several conclusions. At first, for some environments the algorithm did indicate the correct locations faster then for others. Second, the overall correctness is increasing over time, even though the labellings might be wrong initially.

In the following section, the labelling is evaluated and therefore compared to the results of the user interaction within this section.

6.3 Evaluation Labelling

At first, the measured fingerprints can only provide a good prediction of the location, if the labelling works sufficient. Due to this, the accuracy of the labelling is especially important. Second, the amount of different locations is important to ensure them getting distinguished by the classification algorithm.

6.3.1 Time per Classification

The average duration to label the first 4 locations is 6.5 hours (starting from the initial labelling). While some users labelled all their locations within the first 10 minutes, others took around 10 hours. In addition, there can be a large delay until the first location is labelled. This is because the clustering algorithm has to evaluate whether a location is relevant (visited more frequent) or not (only once or seldom visited location). For this observation and decision, time need to pass by. The user can drastically reduce this initial duration by process additional labellings of locations manually.

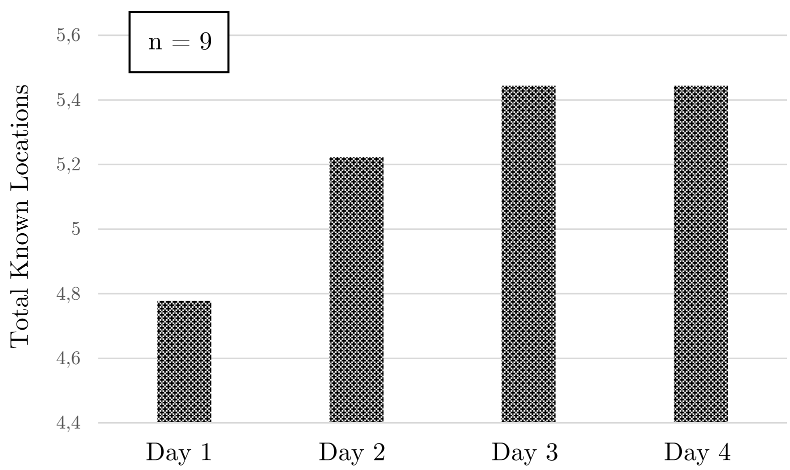


Figure 6.5: Average number of known locations

Graph 6.5 shows how the number of identified locations increase over time. While initially no locations are known, the number of known locations increases the most at the first and second day. Thereafter, the number of new locations decreases and is constant between day 3 and day 4. One reason for this limitation is that the environment of the participant is limited in rooms and the algorithm does filter locations which are not visited regularly by the user. This leads to this saturation of new locations which results in a decrease of new location labelling requests for the user and reduces the interaction with the indoor positioning system.

6.3.2 Error of Classification

The classifications need to meet a certain accuracy in order to provide a real benefit for the user. To not influence the evaluation of interaction and usability, the user has not been asked any additional information during his use during the study. To still get valid insights into the accuracy of the positioning system, the notifications answered by the participant are analysed. Notifications are only issued if the location system meets a certain level of confidence about a location and asks the user to confirm or deny its correctness. A confirmation by the participant identifies, the system has been correct about the location while denying identifies an error. The error can come from several factors such as no complete labelling of the users environment and external parameters such as weather, open/closed doors, changes in the environment. The rate of true positives and false positives can be seen in the histogram below.

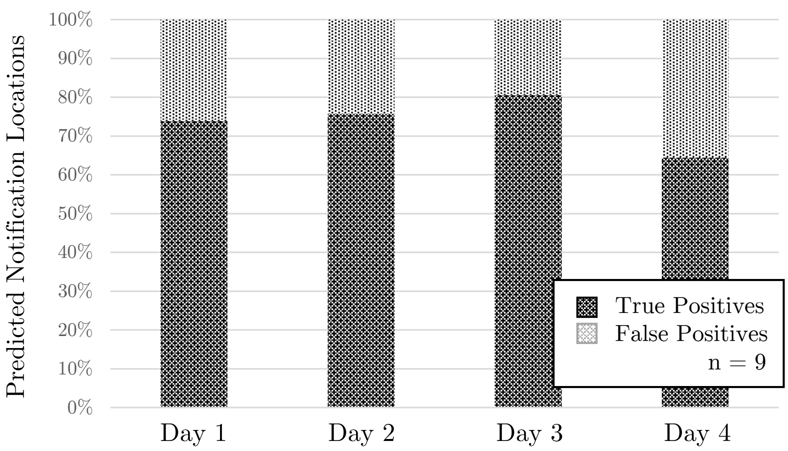


Figure 6.6: Correct Predicted Locations in Notifications

Graph 6.6, is based on the replies for feedback notifications. Whenever the user has been asked for the current position, the clustering algorithm combined this request with a predicted location. In case the location is correct, the user confirmed the location. In case the location is wrong predicted, the user can submit the correct location and the clustering uses this additional data. The graph shows the ratio of correct predictions and wrong predictions. The true positives are starting at the first day above 70% and increase the following two days. Not in line with this trend is day 4 where the rate of true positives is below the first day of the survey.

In general the goal is to have the best possible ratio in favour for true positives. The correctness of the predictions used for notifications are not representing the overall correctness of this indoor positioning system. Further, the notifications are only used, when the clustering requests additional data. To test the overall correctness, a parallel measure to record the exact location of the system is required. Bolliger et al. [6] and others already showed the possibility of room-level accuracy using WiFi fingerprints. Therefore, the goal of this thesis is to combine fingerprinting with clustering and user feedback to develop and evaluate an indoor positioning system for the real user. Therefore, the focus of its evaluation is on the user interactions with this system. Focus on additional accuracy data could be part of future work.

The overall accuracy has been requested by interviews with users. Thereby, statements such as "worked very well" or "worked in around 90% of the time" were received. The next section provides more details about the usability.

6.4 Evaluation Usability

One of the novelties for this indoor positioning system is to do indoor localization for the user in a minimal intrusive way. Therefore, the usability evaluation focus on the perception of the user during the use of the positioning system.

6.4.1 System Usability Scale (SUS)

One parameter which has been measured during the use of the system is the interaction, as described in the section of the user interaction evaluation. A first indicator for the use and usability can be the number of notifications which the user did or did not answer. To additionally gather more data about the intrusiveness and the users perception, the System Usability Scale (SUS) test has been processed. The SUS test is a standard tool to evaluate the usability of a system and consists of the following ten questions:

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.

4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

Each question is answered by the user of the system who can rate between strongly disagree (value 1) and strongly agree (value 5).

The participants of the survey have been asked after completing 4 days of the user study. The results of this questionnaire can be found in graph 6.7.

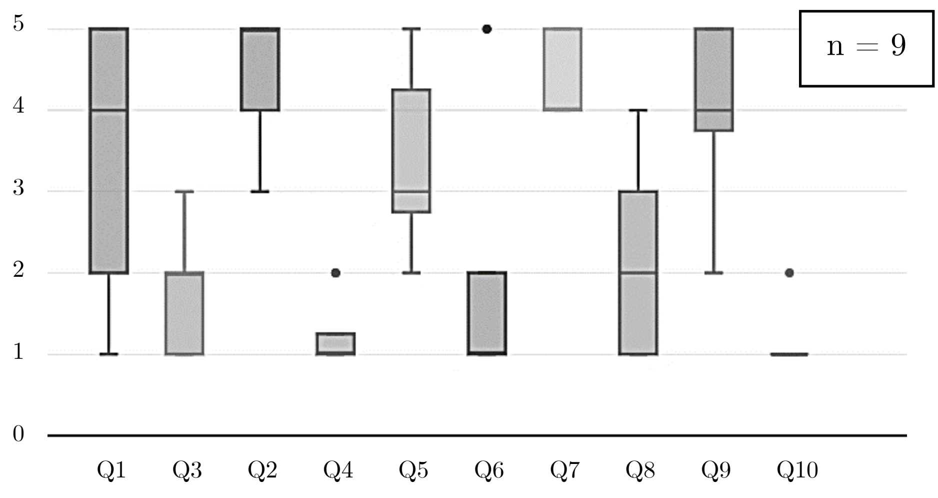


Figure 6.7: Results of System Usability Scale Questions

The average SUS index is 79.72. According to Bangor et al. [31] this is in the range between the adjectives good and excellent. This means the users felt confident using the system. The next section provides more qualitative insights of participants.

6.4.2 Qualitative Survey

In addition to the results of the SUS test, a survey of qualitative questions has been processed by the participants one day after the final date of the user study. The goal has been to receive qualitative feedback and opinions asking open questions about the use of the indoor positioning system.

6.4.2.1 Distraction

Some of the questions focused on the distraction of the user. Asking for the distraction caused by the application, the participants answered clearly that they want to avoid as much notifications as possible. Overall the the notifications of this application did not distract them. The answer of one participant summarizes is describing his experience with notification request of #InPos as follows: "Feedback notifications are kept short and simple. This needs less time and avoids distraction".

6.4.2.2 Use of Indoor Positioning System

The participants have used the indoor positioning system for four days. The system has not been integrated in other applications in their home. Therefore, the benefit for them has been quite limited. One benefit has been the #InPosVis - visualization of the current and past location data. The evaluation of the application use, identified already that the user did not use the application very frequently internally motivated. The same result has been provided through the additional questionnaire. Half of the participants were interested to have a look at the history of their locations. These are the same which answered they would install the application in the future.

6.4.2.3 Use Cases

The participants mentioned that this indoor positioning system could be of use for their home as well as external environments such as office and public buildings. For the home environment they already have very imaginations about use cases which can be beneficial for them. These can be all sections of home automation such as controlling the lights, audio system, and even doors. Another suggestion is to link reminders of todos to locations.

For large buildings such as shopping malls and offices, some participants suggested to extend it providing indoor navigation and share the position. The great benefit is that no additional hardware would be required and no data of the user is published.

6.5 Evaluation Summary

The goal of the evaluation has been to show the usability of this indoor positioning system while being used by real users in their real environment. It has been shown the participants felt comfortable using this indoor positioning system. The process of providing feedback has been a non intrusive way to get information from the user about the current location to label detected locations.

The survey about use cases for such indoor positioning system showed that users already have several use cases in mind (see use cases 6.4.2.3). These especially focus around smart home environments. Further, half of all participants can imagine using indoor positioning in their environment.

Even though the participants have not been asked explicit to label rooms by their own, they have been very motivated to do so. This has lead to an exploration of the first 4 rooms within 6.5 hours (starting from the labelling of the first location)(identified in evaluation section 6.3.1 time per classification).

From the results of this user study it can be seen, that the introduced clustering algorithm of this work depends on many factors. One of the most important is the number of labelled data. Another is the environment which contain the number of available networks, changing network signals and room size.

Analysing the user behaviour the user study identified (table 6.1), that at the fourth day, the users spend already less than ten seconds to answer feedback requests.

The SUS score reached 79.72 which states a user experience which is ranked between good and excellent (SUS results in figure 6.7). The participants of the study have been very positive about installing an application on their mobile phone and get the service of indoor positioning without any additional hardware. Using notifications to receive feedback is accepted by the user, even though it must be focused to not ask the user too many times in a short period.

Finally, the evaluation proved, that the user do not need to process a distinct training phase. This is a huge differentiation to most of the mentioned indoor positioning systems in the related work section. The user can start to use the system right from the the installation on his smartphone.

Chapter 7

Conclusion

This section concludes this work about indoor positioning. Therefore, the key aspects are summarized. Afterwards, additional future work is proposed.

7.1 Conclusion - Requirements

The indoor positioning system of this thesis focused on the use for users in the real word. In the analysis section, requirements have been defined. Based on them this thesis is summarized. The analysis resulted in the following eight requirements.

7.1.1 <R.1> Sensors and <R.2> Localization

Indoor positioning can be achieved in different levels of accuracy. While additional hardware can improve the accuracy, WiFi fingerprinting can provide a room-level accuracy. We have proven that WiFi fingerprinting has an accuracy which is satisfying the users needs. In addition the implementation has shown that WiFi fingerprinting can be conducted using a smartphone which does not require the user to add any additional hardware. Further, the user already carries the smartphone by his side most of the time.

7.1.2 <R.3> Scene Analysis (Clustering)

For analysing the WiFi fingerprints and predict the location data clustering is required. In this work a novel process from individual fingerprints to room-level clustering has been introduced. Compared to other approaches referred in the section of related work, our indoor positioning system focus on the integration of user information. These can be the measured information about when the user is moving and when the user is still. To match identified location clusters with human readable labels, the user is required to provide feedback. During the user study, the indoor positioning system has been running as a background service on android devices. The results showed that after two days, the average user had to spend less than 10 seconds per day to provide feedback to the location requests.

7.1.3 <R.4> Environment Representation

The environment representation of this indoor positioning system is linked to the user interaction. This thesis showed that the user can immediately understand the hierarchical labelled structure of rooms in his environment.

7.1.4 <R.5> User Information

This work used three different types of user information.

- **User Movement:** The internal accelerometer detects if the mobile device is moved and thereby detects movements of the user.
- **WiFi Fingerprints:** The WiFi networks surrounding the user and their received signal strengths are recorded to generate WiFi fingerprints and recognize previously visited locations.
- **User Feedback:** The labelling of locations requires the feedback of the user. This adds the context information to a recognized cluster of fingerprints.

We showed that an indoor positioning system can provide contextual data of locations.

7.1.5 <R.6> User Application UI

While in this work the user interaction is important to label locations, the interactions of the user have been reduced as much as possible. Therefore, the user does not need to do an initial training phase and thereby interacting with the UI. In this approach, the user only has to answer notifications and the indoor positioning system is clustering locations and increase accuracy without explicit labellings.

7.1.6 <R.7> Accuracy

The accuracy of the indoor position depends on various inputs. At first, the available networks and their signal strengths vary. Second, environmental constraints such as walls, windows, and moving humans change the detected fingerprints. Third, the clustering depends on the given data which can lead to incorrect clusterings. Finally, it depends on the feedback of the user and the visited locations. This makes it hard to provide one number for the accuracy. The recorded user study data show that for some environments the clustering did not perform very well, while for others the correct location has been detected very accurate in the impression of the user.

7.1.7 <R.8> Robustness

The mechanism of requesting user feedback is a strong tool for robustness against changes. Old fingerprints of an location are prioritised lower than the latest fingerprints. Thereby, updates - which are recorded in the latest fingerprints - improve the accuracy over the old measurements. Further, the algorithm requests user feedback in case its certainty decreases.

7.1.8 Interface to other Applications

This indoor positioning system provides data to third party applications through a subscription service. Thereby, third party applications can register their server at the indoor positioning system and whenever the location changes, the third party gets notified.

Internally the system is set up as a framework enabling further technologies to be used for positioning. Currently, Wifi, Bluetooth LE and Android Location Service are implemented.

7.2 Future work

This indoor positioning system is currently at a state where it can be used by real users and where the individual components like the background service, notifications, clustering and sensor recordings work together. This work can be used as a basis for further implementations and extensions.

7.2.1 Robustness and Clustering

As visible in the evaluation section, the robustness is not given in all environments. Since this depends on various parameters, these parameters can be further analysed and their impact on the overall result measured. Elements can be the user interaction, WiFi signals, clustering algorithm.

The clustering mechanism can then be further improved on the results of these detections. Another approach is to add another clustering algorithm and measure the difference on the predicted location.

7.2.2 Integration to Interact with Devices

The goal is to make the indoor positioning system be used by real users outside the user study. This requires use cases and third parties which can interact based on location input. Such third parties can be smart environment orchestrators like DS2OS and meSchup, and other applications on the smartphone.

7.2.3 Activities

Users complained that they are in the car or public transportation quite often during the day. This thesis did not make use of GPS data so far. Mixing indoor and outdoor positioning data can track the position of the user continuously. In addition, the activities of the user can be added as location. One example can be to use the android activity service. This service can detect when the user is driving in a car. This information can be used to label the position of the user as "in a car" instead of the currently shown "undefined location".

7.2.4 Share Location Learnings

It is often important to know on the first visit in large buildings where the is the current location and where do you have to go. To get information on the initial visit, the database must already provide a clustering of the fingerprints for the locations in this environment. This can be achieved using a central database for fingerprints and techniques.

Bibliography

[1] Thomas Kubitza, Alexandra Voit, Dominik Weber, and Albrecht Schmidt. An IoT infrastructure for ubiquitous notifications in intelligent living environments. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct - UbiComp '16*, pages 1536–1541, New York, New York, USA, 2016. ACM Press.

[2] Marc-Oliver Pahl, Georg Carle, and Gudrun Klinker. Distributed smart space orchestration. In *NOMS 2016 - 2016 IEEE/IFIP Network Operations and Management Symposium*, pages 979–984. IEEE, apr 2016.

[3] Donnie H. Kim, Kyungsik Han, and Deborah Estrin. Employing user feedback for semantic location services. In *Proceedings of the 13th international conference on Ubiquitous computing - UbiComp '11*, page 217, New York, New York, USA, 2011. ACM Press.

[4] Jennifer R. Whitson. Gaming the quantified self. *Surveillance and Society*, 11(1-2):163–176, 2013.

[5] Anind K Dey, Katarzyna Wac, Denzil Ferreira, Kevin Tassini, Jin H Hong, and Julian Ramos. Getting closer: an empirical investigation of the proximity of user to their smart phones. *Proceedings of the 13th international conference on Ubiquitous computing*, pages 163–172, 2011.

[6] Philipp Bolliger, Kurt Partridge, Maurice Chu, and Marc Langheinrich. Improving location fingerprinting through motion detection and asynchronous interval labeling. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5561 LNCS:37–51, 2009.

[7] Wilhelm Kleiminger, Christian Beckel, Anind Dey, and Silvia Santini. Using unlabeled Wi-Fi scan data to discover occupancy patterns of private households. *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems - SenSys '13*, pages 1–2, 2013.

[8] Yanying Gu, Anthony Lo, Senior Member, and Ignas Niemegeers. Wireless Personal Networks. *Communications*, 11(1):13–32, 2009.

[9] Hui Liu. Survey of Wireless Indoor Positioning\nTechniques and Systems. 37(6):1067–1080, 2007.

[10] Veljo Otsason, Alex Varshavsky, Anthony Lamarca, and Eyal De Lara. Accurate GSM Indoor Localization. *Pervasive and Mobile Computing*, 3(6):698–720, 2007.

[11] Moustafa A Youssef, Ashok Agrawala, A Udaya Shankar, A Udaya Shankar, and A Udaya Shankar. WLAN location determination via clustering and probability distributions. *Pervasive Computing and Communications, 2003.(PerCom 2003). Proceedings of the First IEEE International Conference on*, (August 2017):143–150, 2003.

[12] Siddhartha Saha, Kamalika Chaudhuri, Dheeraj Sanghi, and Pravin Bhagwat. Location determination of a mobile device using IEEE 802.11 b access point signals. *Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE*, 3:1987–1992, 2003.

[13] Philipp Bolliger. Redpin - Adaptive, Zero-Con guration Indoor Localization through User Collaboration. *Melt'08*, pages 55–60, 2008.

[14] Andreas Haeberlen, Eliot Flannery, Andrew M. Ladd, Algis Rudys, Dan S. Wallach, and Lydia E. Kavraki. Practical robust localization over large-scale 802.11 wireless networks. *Proceedings of the 10th annual international conference on Mobile computing and networking - MobiCom '04*, page 70, 2004.

[15] A. Kotanen, M. Hnnikinen, H. Leppkoski, and T. D. Hmlinen. Experiments on local positioning with Bluetooth. *Proceedings ITCC 2003, International Conference on Information Technology: Computers and Communications*, pages 297–303, 2003.

[16] Jeffrey Hightower, Gaetano Borriello, and Roy Want. SpotON: An indoor 3D location sensing technology based on RF signal strength. *Uw Cse*, (March 2000):16, 2000.

[17] Markus Funk, Robin Boldt, Bastian Pfleging, Max Pfeiffer, Niels Henze, and Albrecht Schmidt. Representing indoor location of objects on wearable computers with head-mounted displays. In *Proceedings of the 5th Augmented Human International Conference on - AH '14*, pages 1–4, New York, New York, USA, 2014. ACM Press.

[18] Roy Want, Andy Hopper, Veronica Falcao, Jonathan Gibbons, Veronica Falcão, Jonathan Gibbons, Veronica Falcao, and Jonathan Gibbons. The Active Badge Location System. *ACM Transactions on Information Systems (TOIS)*, 10(1):91—-102, 1992.

[19] Viet Cuong Ta, Dominique Vaufreydaz, Trung Kien Dao, and Eric Castelli. Smartphone-based user location tracking in indoor environment. *2016 International*

Conference on Indoor Positioning and Indoor Navigation, IPIN 2016, (October):4–7, 2016.

[20] M.R. Mahfouz, Cemin Zhang, B.C. Merkl, M.J. Kuhn, and A.E. Fathy. Investigation of High-Accuracy Indoor 3-D Positioning Using UWB Technology. *IEEE Transactions on Microwave Theory and Techniques*, 56(6):1316–1330, jun 2008.

[21] Abdulrahman Alarifi, AbdulMalik Al-Salman, Mansour Alsaleh, Ahmad Alnafessah, Suheer Al-Hadhrami, Mai Al-Ammar, and Hend Al-Khalifa. Ultra Wideband Indoor Positioning Technologies: Analysis and Recent Advances. *Sensors*, 16(5):707, may 2016.

[22] Frédéric Evennou and François Marx. Advanced integration of WiFi and inertial navigation systems for indoor mobile positioning. *Eurasip Journal on Applied Signal Processing*, 2006:1–11, 2006.

[23] Paramvir Bahl Padmanabhan and Venkata N. RADAR: An in-building RF based user location and tracking system. *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, 2(c):775–784, 2000.

[24] Sungwon Yang, Pralav Dessai, Mansi Verma, and Mario Gerla. FreeLoc: Calibration-Free Crowdsourced Indoor Localization.

[25] M. Daszykowski and B. Walczak. Density-Based Clustering Methods. *Comprehensive Chemometrics*, 2:635–654, 2010.

[26] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, Jörg Sander, Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander. OPTICS. In *Proceedings of the 1999 ACM SIGMOD international conference on Management of data - SIGMOD '99*, volume 28, pages 49–60, New York, New York, USA, 1999. ACM Press.

[27] Beom Ju Shin, Kwang Won Lee, Sun Ho Choi, Joo Yeon Kim, Woo Jin Lee, and Hyung Seok Kim. Indoor WiFi positioning system for Android-based smartphone. *2010 International Conference on Information and Communication Technology Convergence, ICTC 2010*, pages 319–320, 2010.

[28] Paul Baumann, Marc Langheinrich, Anind K Dey, and Silvia Santini. Quantifying the Uncertainty of Next-Place Predictions.

[29] Guanling Chen and David Kotz. A Survey of Context-Aware Mobile Computing Research. *Dartmouth Computer Science Technical Report*, 3755:1–16, 2000.

[30] M.N. Husen and S. Lee. Indoor Location Wi-Fi Fingerprinting using Invariant Received Signal Strength.

[31] Aaron Bangor, Phil Kortum, and James Miller. Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale. *J. Usability Stud.*, 4:114–123, 2009.

Icons - Flaticon

The icons used in the application and in this thesis are provided by Freepic from http://www.flaticon.com.