Master Thesis Research

Study of Wi-Fi Fingerprint-Based Indoor Positioning on a smartphone

Author:	Thomas Vandenabeele
Supervisor:	Prof. Dr. Pan Yun
Ext. Supervisor:	Prof. Dr. Ir. Luc Claesen
Home University:	UHasselt & KULeuven, Belgium
Host University:	Zhejiang University, China
Major:	M.Sc. of Electronics and ICT Engineering Technology
Submitted Date:	June 2017

A thesis for the partial fulfillment of the requirements for the degree of

Master of Science in Electronics and ICT Engineering Technology

at Hasselt University and KULeuven, Belgium.

Acknowledgements (致　谢)

At the end of my engineering study I was presented with the exceptional opportunity to fulfil my master's thesis in China. I was fortunate to reside for four months at one of the country's best universities, Zhejiang University in Hangzhou. It is with great pleasure that I look back at the past four months. Not only in the research area I studied, but also on a personal level I have developed many new skills and interests, on top of the numerous important life experiences I acquired here. Because of this I will remember this study abroad for the rest of my life.

After this intensive period of four months, the time has come to write down this note of thanks as the finishing touch on my thesis. Writing this text has not always been easy. Therefore, I would like to reflect on the people that supported me throughout my period abroad.

First of all, I would like to express my gratitude to my supervisor at Zhejiang University, prof. dr. Yun Pan. I want to thank him for the excellent cooperation and the opportunities I was given to conduct my research at Zhejiang University. In particular, I would like to thank him for inviting me in his team of remarkable students and for helping me succeed my thesis. During the weekly meetings regarding my research he provided me with a lot of useful information and remarks, together with my fellow students Ling Zhang and Zhe Yang. Without their experience and knowledge, I would not have achieved the same results. I would also explicitly like to thank Ling very much for helping me out the first days after my arrival in China, for always providing me with useful information and to introduce me to the Chinese culture.

In addition, I would like to thank the other students in the lab for providing me with such a great time in China.

I would also particularly like to thank prof. dr. ir. Luc Claesen for encouraging me to study in China and for his valuable guidance during my stay. Without his experience and knowledge, it would not have been possible for me to study at Zhejiang University. Furthermore, I would also like to thank him for his advice and support relating to the structure and content of my thesis.

Finally, I would also like to thank my parents, family and friends for their wise advice, sympathetic ear and continuous support. With these people I could always discuss problems and findings regarding my thesis, as well as my experiences in China and many other things. I would like to thank everyone I explicitly or implicitly mentioned above collectively one last time for this unforgettable and inspirational experience of which I carry many pleasant memories for the rest of my life.

Thomas Vandenabeele
Hangzhou, June 2017

Abstract (摘　要)

Nowadays, there is a high demand of Location Based Services (LBS) for indoor environments. Because the widely used Global Positioning System (GPS) is unavailable indoors, many technologies and methods have been investigated to address this problem. Indoor positioning based on Wi-Fi fingerprinting has attracted significant interest due to its potential to obtain high accuracy at low costs. It can be applied to any indoor scenario where Wi-Fi networks are deployed without any additional hardware.

This thesis first examines the current developments in the field of indoor positioning and it investigates the problems with Wi-Fi fingerprinting in particular. In a second stage an Indoor Positioning System (IPS) is developed based on a novel implementation using a modified Weighted K-Nearest-Neighbors (WKNN) algorithm with prior Spearman's Rank Correlation Coefficient (SRCC) calculation. The proposed positioning algorithm also takes into account the number of signals being omitted during localization. Therefore, unreliable results have a smaller impact on the final result. The proposed system consists of two parts: an Android smartphone application and a webserver provided with the proposed algorithm written in Erlang. The proposed IPS achieves an exceptional accuracy with an average positioning error of approximately 80 cm using an up-to-date fingerprint database.

On the basis of the results of this research, it can be concluded that it is possible to use Wi-Fi fingerprinting for indoor positioning to obtain a state-of-the-art accuracy.

Keywords: Indoor Positioning, Wi-Fi, fingerprinting, WKNN, Spearman

Table of Contents (目　录)

List of Figures (图 目 录)

List of Tables (表 目 录)

List of Abbreviations (缩 略 列 表)

AOA Angle of Arrival.

AP Access Point.

BSSID Basic Service Set Identifier.

CDF cumulative distribution function.

CEPT European Conference of Postal and Telecommunications Administrations.

CSI channel state information.

ECO European Radiocommunications Office.

GPS Global Positioning System.

IMU Inertial Measurement Unit.

IPS Indoor Positioning System.

JSON JavaScript Object Notation.

KNN K-Nearest-Neighbors.

LBS Location Based Services.

LOS Line-Of-Sight.

MU Mobile Unit.

NICs network interface cards.

NLOS Non-Line-Of-Sight.

PDR Pedestrian Dead Reckoning.

RFID Radio-Frequency Identification.

RSS Received Signal Strength.

RSSI Received Signal Strength Indication.

SLAM Simultaneous Localization and Mapping.

SRCC Spearman's Rank Correlation Coefficient.

SVM Support Vector Machine.

TDOA Time Difference of Arrival.

TOA Time of Arrival.

UWB Ultra-Wideband.

vSLAM Visual Simultaneous Localization and Mapping.

WKNN Weighted K-Nearest-Neighbors.

1 Introduction

This chapter provides a short introduction of the purpose of this thesis. The objectives of this thesis are then formulated based on the given problem statement. To the end of this chapter the materials and methods to achieve these objectives and the research approach are introduced. The last section provides an outline of the chapters to come.

1.1 Problem statement and Motivation

Current developments in mobile device technology ensure that users can use these devices for navigation. In fact, for users it is a matter of course to use their mobile device for this application. However, indoor localization technology for indoor navigation is not obvious and it is a difficult task to improve the reliability and accuracy of the systems. The technology has already been examined for several years in research projects and many solutions have been proposed in literature.

Up until this day, the need for high accuracy indoor smartphone localization is still big and booming. The main areas of application are to help users navigate inside large, open and complex indoor environments (e.g. airport, train station, shopping mall). The technology could also provide an understanding of the patterns and the customers visit for stores and businesses.

For indoor environments it is quite difficult to make use of the well-known GPS. Positioning with GPS can only be achieved by receiving signals from at least three to four GPS satellites at the same time. This is usually not available inside a building and makes the GPS useless and inefficient. To overcome these limitations, a Dead Reckoning solution is useful, in particular Pedestrian Dead Reckoning (PDR) algorithms [?] are very suitable for indoor positioning. PDR systems take advantage of the mobile measurement unit e.g. provided in mobile devices. These systems calculate the current location of a moving person on the basis of the starting point, the orientation, the number of steps and the step length of the pedestrian. Nevertheless, PDR systems often result in cumulative errors because the basic drift of the system will accumulate as time goes by. Handling different carrying scenarios and movements of the mobile device makes dead reckoning systems fairly complicated. Because of this, the goal of this thesis is to investigate the implementation of an IPS. This system could then be used as an additional technology to liquidate the basic drift of the PDR system [1, 2].

Therefore this research concerns the development of a Wi-Fi fingerprinting technique for indoor positioning. Although various studies have currently dealt with WiFi fingerprinting for indoor positioning, the novelty of this research is that it will endeavor to obtain better accuracy resulting from a novel implementation methodology. The widely used K-Nearest-Neighbors (KNN) algorithm will be adjusted with a prior SRCC calculation. The algorithm also takes the reliability of the measurements into account. The prototype could be used to navigate from one place in a building, to another place in the building in the future.

1.2 Objectives

In this thesis the focus lies on the development of a WiFi-based fingerprint indoor positioning system. To achieve this goal it is necessary to accomplish the following individual objectives:

1. Explain why WiFi is chosen for the indoor positioning system.
2. Obtain information (e.g. Basic Service Set Identifier (BSSID) and RSS) from nearby access points with mobile device via WiFi scan.
3. Create a radiomap of each access point in the building.
4. Search and implementation of the best location algorithm for fingerprinting.
5. The application will have to be tested and demonstrated on a mobile device, to check if the positioning accuracy is acceptable.

1.3 Summary of Contributions

The contributions of the work in this thesis are as follows:

- Development of a modified WKNN algorithm in Erlang
- Development of a Spearman's rank correlation coefficient algorithm in Erlang
- Development of a smartphone application capable for Indoor Positioning in C#
- Achieving reasonable accuracy with an average positioning error of approximately 0.8m with the overall positioning system

1.4 Material and methods

To achieve the objectives discussed in the previous section there are several approaches available. First the choice of the mobile device needs to be made. The project implements the positioning system on an Android smartphone. Development for iOS is currently not possible; Apple does not provide a public API to scan for nearby access points. Because of this, it is not possible to obtain the RSSI values of each access point.

The project will develop a Xamarin [3] application instead of a native Android application. The Xamarin platform allows the developer to create applications that are cross-platform. With the same code-base, written in C#, an application for iOS, Android and Windows can be developed. Xamarin Studio will be used as the IDE software. Because of this, one may decide later to build the Xamarin application for the iOS operating system from the moment Apple provides the public API for WiFI scanning.

The following hardware is used for the development and implementation of this project with the following specifications, as shown in Table 1.1.

1.5 Research Approach

The initial phase of this project mainly focuses on the research for the current state of the technology based on state-of-the-art. After this, the application is being developed, and successively optimized to a good working prototype. Ultimately, the results of the project are analyzed and reported.

The project is carried out in the Laoshengyi (老生仪) building at Zhejiang

Specification	Laptop	Smartphone
Name	MacBook Pro (Retina, 15-inch, Late 2013)	Nexus 5
Brand	Apple Inc.	LG
Operation System	macOS 10.12.3	Android 4.4.4
Processor	2 GHz Intel Core i7	Quad-core 2.3 GHz Krait 400 CPU
RAM-memory	8 GB 1600 MHz DDR3	2 GB

Table 1.1: Overview of used hardware.

University (Yuquan Campus) in Hangzhou, China. In particularly, the fifth floor will be used to carry out the experiments, to determine the position of the mobile device in two dimensions and to make sure the prototype will work in a relative small area. The system could also be used in different floors or public spaces if there are fixed access points available.

1.6 Outline

This thesis is divided in two main parts. Firstly the current techniques and technologies used for indoor positioning (chapter 2) are discussed. Then it focuses more on WiFi indoor positioning and the current developments in this area in chapter 3.

For the second part this thesis focuses on the implementation, development and the obtained results of the developed WiFi indoor positioning system. In the last chapter an overview of further improvements is proposed.

2 Indoor Positioning Systems

An IPS is a system to track an object or person situated within a building where the GPS is inadequate. An IPS is often implemented by the use of a portable, sometimes wearable, device.

2.1 Previous Developments

There are several non-radio [1, 2, 4–10] and wireless [11–18] technologies that have been studied in the last few years that could be used for positioning. Current and ongoing research focuses mostly on wireless indoor positioning techniques. Most of these approaches uses Wi-Fi or Bluetooth signals, taking advantage of existing wireless infrastructures already deployed inside the building.

2.1.1 Non-radio technologies

Infrared Signals One of the first indoor positioning systems is the active badge system introduced by R. Want et al. in 1992 [4]. In this study, they developed a wearable badge, similar to a name badge, that emits infrared signals. Receivers are placed at specific places in the building which make it possible to locate persons. There are obviously a lot of limitations. First of all, the receivers and the badge needs to be in the line-of-sight of one another. Another limitation is that infrared is a short-range transmission signal, therefore a lot of receivers are required, bringing the costs upward, even though the accuracy of the location depends on the number of receivers [4].

Magnetic Positioning Magnetic positioning is a commonly applied method. It is based on the iron structure of the building. Dedicated chips inside e.g.

a smartphone can sense these magnetic variations in the Earth's magnetic field. These variations are used to map the building and provide localization [5, 6].

Acoustic/Sound Positioning Another non-radio technology for indoor positioning is the use of acoustic or sound signals [19, 20]. Guoguo [20] is an indoor positioning system that achieves centimeter-level localization accuracy in several indoor environments. The system utilizes therefore an anchor network of low-cost devices spread over the space. Each node transmits modulated localization beacon signals using high-bandwidth acoustic signals. A smartphone application processes these signals and using a backend server the localization can be determined.

Inertial Measurements The tracked object or person carries a device with a Inertial Measurement Unit (IMU), this method is mostly applied in pedestrian dead reckoning systems [1, 2, 7]. These systems benefits from the availability of sensors in modern smartphones. With the proper algorithms on the sensor data, tracking the movement of an object or person is possible [1]. Recent research [**?**] focuses on a pedestrian tracking system using dead reckoning on a commercial off-the-shelf smartphone. The tracking system utilizes the smartphone's built-in Inertial Measurement Unit. The system even identifies three typical carrying modes of the smartphone during walking. This feature is used to optimise the tracking accuracy. The smartphone's user is able to localize itself in the environment assuming that the initial position is known based on step detection, step length and the holding mode of the smartphone. The PDR system is robust and accurate for people of different gender, height and walking speed. A sub-meter error accuracy is achieved, when walking a path with distance of 28m, for real-time tracking and localization of the smartphone user.

Visual Positioning There are two directions within visual positioning. First of all, it is possible to determine the position of an object or user by using a camera-enabled mobile device. Most methods implement visual markers in the environment. By locating and decoding these visual markers and measuring the angle from the device to the marker, it is possible to estimate the location of the device. Secondly, there are also visual positioning systems that use a database of images. The mobile device will estimate its location using interpolation on this image database.

A common implementation of visual positioning is Simultaneous Localization and Mapping (SLAM) [8, 9]. SLAM is concerned with the problem of building a map of an unknown environment by a mobile robot while at the same time navigating the environment using the map. SLAM approaches are employed in self-driving cars, unmanned aerial vehicles, autonomous underwater vehicles, planetary rovers, etc. SLAM will always use several different types of sensors, as well as optical sensors, named Visual Simultaneous Localization and Mapping (vSLAM) [10]. vSLAM algorithms are mostly vision- and odometry-based. These systems enable low-cost navigation in cluttered and populated environments. Because no initial map is required, a vSLAM robot possesses the ability to explore its environment without user-intervention. Because of this the robot is capable of building a reliable map and localize itself in the map [10].

2.1.2 Wireless technologies

One of the most common techniques for indoor localization is the use of radio-frequency signals. These systems most commonly use the concept of RSS, an indication of the received signal's power level measured by the receiver, for positioning. The inverse-square law applies to radio waves propagation, therefore distance determination is possible based on the relationship between transmitted and received signal strength. This makes radio-frequency signals particularly useful for localization, usually **Bluetooth** [11–13] or **Wi-Fi based positioning systems** [14, 15] are used.

Radio-Frequency Identification (RFID) tags are also used for implementing indoor positioning systems, e.g. the mTag project [16]. Fixed RFID readers are placed in the building and a passive RFID tag attached to e.g. a mobile phone or name badge. The location is determined by passing a RFID reader with this tag, resulting in an estimation of the user's location. The implementation of this system requires, of course, a close passage to prevent it from walking out of range of the reader.

The operation principle for positioning systems using Bluetooth or Wi-Fi techniques are similar. They are both a wireless technology that is already very well established in modern smartphones. Choi and Jang report up to 86% accuracy using fingerprinting in combination with Bluetooth technology [13]. The main advantage of Wi-Fi over Bluetooth systems is that the network infrastructure is already present (e.g. hotspots, etc.) and several access points are located in fixed positions. So it will not require an investment in specialized hardware. Wi-Fi based positioning systems are further discussed in the next chapter.

Another upcoming technology is **Ultra-Wideband (UWB)**. It is an RF technology that allows millimeter accurate positioning, consuming low power [17, 18]. But most current consumer type mobile devices are not yet equipped with this technology. Given the appropriate driving applications, it is to be expected that this technology will also be embedded in future smart phones and mobile devices.

2.2 Approach for Indoor Positioning

From the literature on indoor positioning a classification can be made: on the one hand, the quantities that are measured (the technology), and on the other hand, the manner in which these quantities are used to calculate a specific location (the techniques) [21]. Table 2.1 depicts an overview of this classification. The most important techniques are fingerprinting, signposting and trilateration. Received Signal Strength (RSS), Time of Arrival (TOA) and Angle of Arrival (AOA) are some technologies that can be used for indoor positioning. The possible positioning techniques can not only differ in the technology that is used, but also in the way in which the data are processed. Thus, another classification can be made between centralized and decentralized methods. Localisation is centrally calculated or distributed on each mobile device respectively [22].

Techniques	Technologies	Localization
Fingerprinting	RSS	Central
Signposting	RSS	Central or distributed
Trilateration	RSS, TOA, AOA	Central or distributed

Table 2.1: Overview of technologies and techniques.

The following sections explain the operation of the most important technologies and techniques for indoor positioning.

2.2.1 Technologies

RSS [23] defines received signal strength as the measurement of the power present in a received radio signal. This received power is mostly dependent on the distance between transmitter and receiver and is determined based on a large number of sample measurements of the received signal. Sampling will be taken over disjoint time intervals. RSS is most commonly shown as the power ratio in decibels (dB) of the measured power. The unit is therefore dBm.

Most hardware is equipped to measure the Received Signal Strength Indication (RSSI) of the received power. The advantage is that it consumes no extra bandwidth. With RSS, the distance is calculated either at the mobile station or the base station.

The drawback of the use of RSS signals is that it is heavily influenced by the structures in a building: walls, doors, floors, concrete, steel, furniture, machines, etc. as well as reflections, multi-path by these structures.

TOA Position determination by TOA technology, is measuring the time that it takes for a signal to propagate from the mobile station to the base station, by using the finite velocity of propagation, approx. the speed of light (300 meters per microsecond)[1]. The time it takes for a signal to propagate from a transmitter to a receiver is in fact related to the distance traveled by the signal. A first drawback is the need of clocks with a very fine resolution. It also requires complete synchronization to obtain an accurate distance measurement, e.g. a time measurement error as small as 100 nanoseconds can result in a localization error of 30 meters [24].

UWB is a technology that implements time of arrival in a cost effective and accurate way.

A variant of TOA is Time Difference of Arrival (TDOA). TDOA bypasses the need for synchronization between transmitter and receiver. The localization is based on the difference in time of arrival at the receiving nodes. Because of this it requires a minimum of three nodes for basic operation (transmitter node and minimum two receiver nodes). The receiver nodes must no longer be synchronized with the transmitter. The difference in arrival time at the receiving nodes is in fact independent of the time base of the transmitter. Obviously, the receiving nodes should still be synchronized.

Figure 2.1 depicts a final disadvantage of TOA. For correct distance measurements a Line-Of-Sight (LOS) is needed, thus equivalent to the direct path (t1) between transmitter and receiver. If there is no LOS component received Non-Line-Of-Sight (NLOS), t2 will be chosen for the shortest arrival time. This will lead to an erroneous calculation of the distance. Such reflections of course also influence the accuracy of RSSI based methods.

AOA In a node network with nodes equipped with directive reception

[1] In vacuum, c = $299792458 m/s$. For simplicity, c = $3 * 10^8 m/s = 300 m/\mu s$ is used

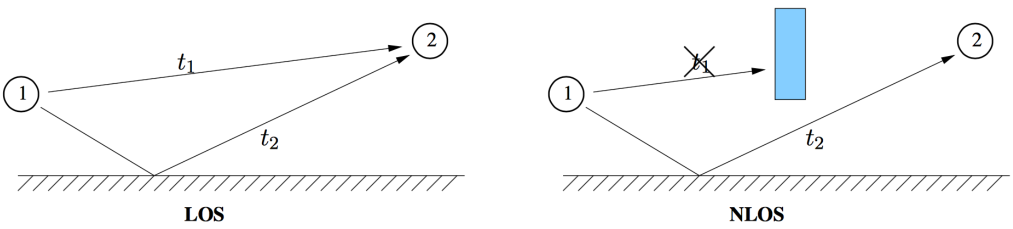


Figure 2.1: LOS/NLOS drawback of TOA [21]

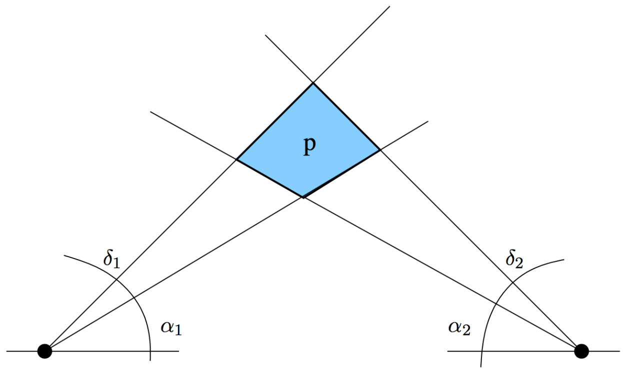


Figure 2.2: AOA visualization [21]

antennas, AOA technology could be used for localization, by making use of the reception angle of the signal.

As is shown in Figure 2.2, two reference nodes suffice to determine an unambiguous location area (p) at the receiving area. As the receiving antenna can distinguish smaller angles, the location area (p) becomes smaller.

It is possible to use AOA in combination with RSS or TOA to do fairly precise positioning. Keep in mind that this technology also gives incorrect results if there is no LOS between the nodes.

2.2.2 Techniques

Fingerprinting One of the most common indoor localization techniques is fingerprinting [6, 13, 25, 26]. This technique makes use of a database of signal values of each node in the network (mostly RSS values) measured on several specific locations. Therefore it is possible to build a radio map of the building for each node's measurements.

In order to determine the position of a blind node inside the building, one

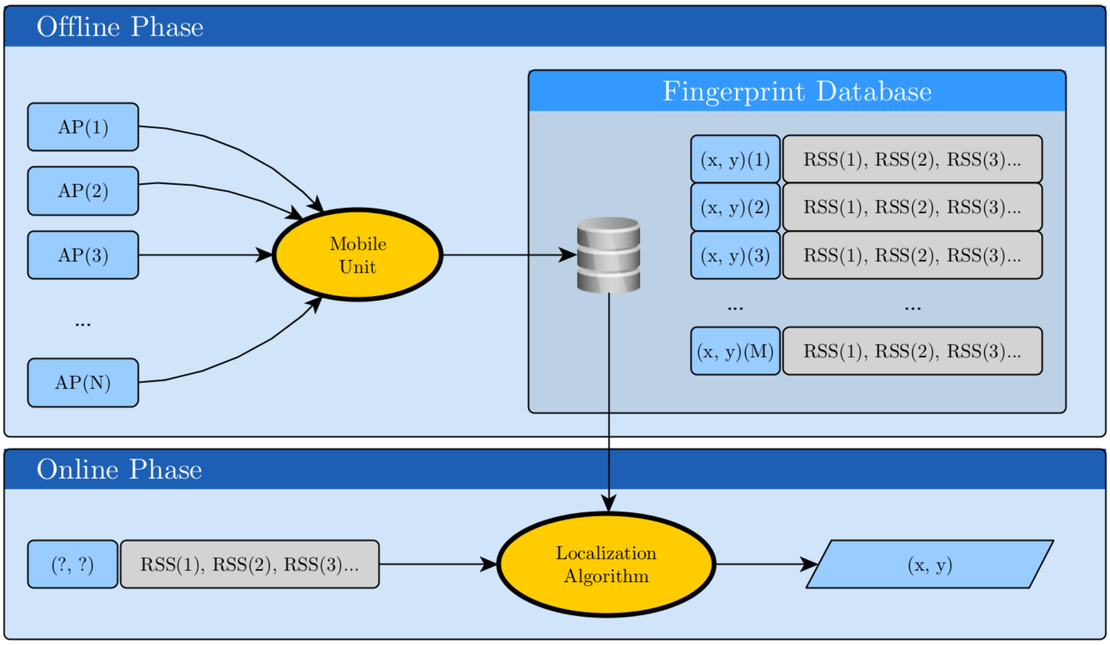


Figure 2.3: Fingerprinting workflow of training and online phase [30]

compares the received signal with the values stored in the database. The best match (with all reference nodes) provides the position. From the best similarity (with all reference nodes), the position of the blind node can be calculated.

The method consists of two phases, a training (or set-up phase) and an online phase (or localization phase). Figure 2.3 shows these two phases for WiFi fingerprinting. In the training phase, the area is scanned for surrounding Access Point (AP) from the Mobile Unit (MU). After that, the fingerprint from each AP is stored together with its (x,y)-location in the database. This phase uses the so-called war-driving [27, 28] method. In the online phase a positioning algorithm is used that compares the current online measurements of the mobile unit to the fingerprints stored in the database. The best possible match is the mobile units (x,y)-location [29].

The size of the database used is directly related to the accuracy of the positioning and the number of reference nodes. The main disadvantage of the technique is the training phase. Building a radio map for each node in the building is very labor-intensive and time-cosuming for large spaces. Furthermore, these radio maps are a snapshot of the signal strengths of the moment of measurement causing into erroneous calculations at the time the signal strenghts change or one of the nodes become faulty.

With fingerprinting the localization is performed centrally, all data is processed at a central location (e.g. remote server with database), contrary to distributed positioning.

Signpost Positioning Signpost positioning is the simplest indoor positioning technique, moreover also the least accurate. This is, in other words, a highly simplified form of fingerprinting. One such application of the signpost algorithm is used by [31]. The exact location of all the access points needs to be known before using the signpost algorithm. It is also common to use a symbolic name for the acces point, e.g.: the name of a specific room or desk. The location of the blind node is linked with the symbolic name of the reference node that is received best. Therefore, the signpost algorithm makes it possible to predict in which room the blind node is situated.

The advantage of this technique is that there is no need for much calculations. The disadvantage is the low precision of the positioning (mostly room location estimation). The need to obtain a more precise location will result in an increase of the number of reference nodes. This added infrastructure will lead to additional costs to implement the signpost positioning technique.

By using signpost positioning the localization is performed centrally or distributed. Like fingerprinting, signpost positioning can also use a central database. It is also possible that the blind node determines its own location due to the limited computing power necessary.

Trilateration One of the most traditional ways of localization, is the use of trilateration. The measurement technologies for trilateration are: RSSI, TOA and AOA. Trilateration uses measurement of distances for determing locations using circles, spheres or triangles. In contrast to triangulation, trilateration does not use measurement of angles.

In a two-dimensional plane, the position of a blind node can be determined with the aid of the signal strengths of at least two reference nodes, converted to distances. It is known that the blind node then lies on two circles with their two radii equal to these distances. The position can be determined because the centers of these circles form a triangle together with the blind node.

In a three-dimensional plane, at least three reference nodes are needed. The position of the blind node can be determined via conventional geometric methods. The centers of the three circles or spheres together with their radii are sufficient information to determine a localization area (also shown in Figure 2.4). Since these

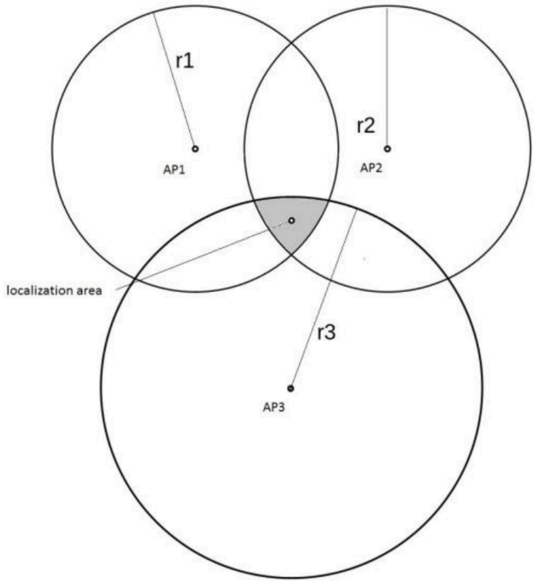


Figure 2.4: Geometric interpretation of TOA for indoor positioning [32]

calculations rarely provide an unambiguous location, the most probable position is often determined via interpolation techniques, such as the least squares method [32].

The Global Positioning System relies on this technique, except that GPS is in three dimensions. So spheres are needed instead of circles to calculate the distances from the location to the satellites. After the geometric triangulation with the three spheres two points of possible locations are determined. One of the points is not on the surface of the earth, so it can be eliminated, finding the current location. So if it is desired to use three-dimensional trilateration or multilateration for indoor positioning, one needs at least four reference nodes.

2.2.3 Comparison

Table 2.2 summarizes a comparison between these technologies and techniques by [33]. As can be seen from the table, it is very clear that fingerprinting is the most suitable method for the proposed indoor positioning system. Note that when using the other methods one needs to specify the location of the antennas. This is a major drawback in comparison to fingerprinting.

Method	Indoor accuracy	Coverage	LOS or NLOS	Affected by multipath	Cost
Signpost (RSS)	Low	Good	Both	No	Low
AOA	Medium	Good	LOS	Yes	High
TOA	High	Good	LOS	Yes	High
Fingerprinting	High	Good	Both	No	Low

Table 2.2: Comparison of indoor position methods. [33]

3
WiFi Fingerprinting

This chapter focuses on WiFi IPS with in particular the fingerprinting method. First there is a short introduction about why this method is implemented. After that some problems with WiFi signals are explained. Next there is a report of the explored related work that is most important for this thesis. At the end of this chapter a closer look will be given to the proposed system architecture for this thesis.

3.1 Introduction

In this part of the thesis, some recent research work on WiFi localization with a specific focus on fingerprinting-based localization techniques will be presented. Is is clear from the criterea discussed in the previous chapter, to chose fingerprinting as the technique for the proposed system.

3.2 Weaknesses of WiFi for indoor positioning

Using WiFi signals for indoor positioning is one of the most appropriate and favorable solutions because of the presence of IEEE 802.11 b/g/n access points in buildings. Besides this, WiFi also ensures some negative effects. Using WiFi for indoor positioning is therefore not without drawback.

3.2.1 Body Effect

European radio regulations are standardized by the European Conference of Postal and Telecommunications Administrations (CEPT). The European Radiocommunications Office (ECO) develops regulations for CEPT and obliges users of the 2.4 GHz frequency band to operate at low power. Therefore, devices

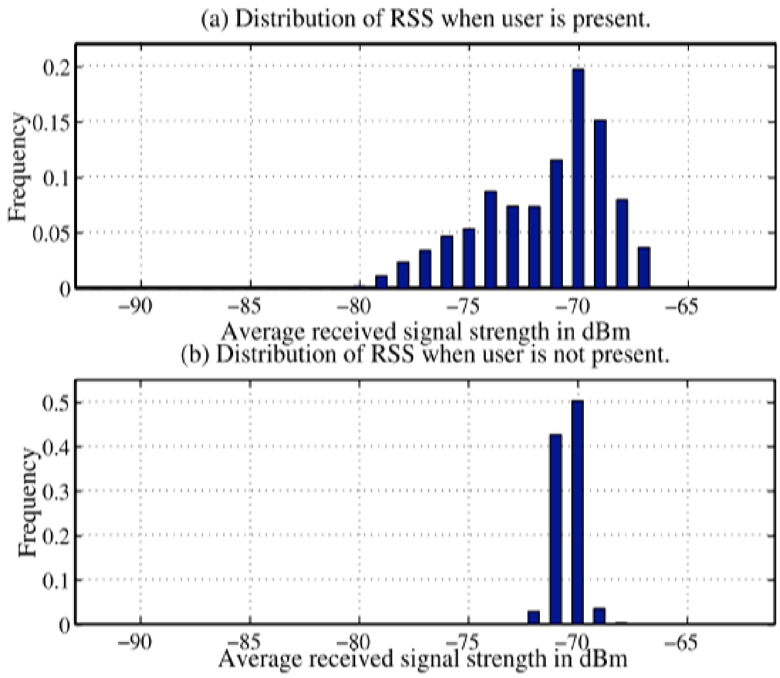


Figure 3.1: Results of body effect experiment by [26]

operate at a maximum power of 1 watt or 30 dBm [34]. According to [35], signal strength may drop 10-15 dBm due to the low penetration power. Note that this power los accounts for 14-21% of the total effective signal strength.

Because of this power los, the signal will highly influence the positioning system. Even when a user holds the mobile device in his hands, the path between the mobile device and the access point can be obstructed by the user, hence the name: body effect. The body of the user also acts as an additional antenna and disturbs the directional propagation/receiving pattern.

The body effect has been studied by K. Kaemarungsi and P. Krishnamurthy in [26]. They conducted an experiment to confirm this effect by measuring the signal strength at about 7 m removed from the access point with NLOS. The experiment took 2 hours, during the first hour, the user was present, while no user was present in the second hour. Figure 3.1 depicts the result of their experiment. Note that the presence of the user significantly changes the standard deviation of the signal strengths from 0.68 to 3.00 dBm and the mean from -70.4 dBm to -71.6 dBm.

By keeping the body effect in mind, it is clear that when the radio map is built for the IPS in the training phase, the user should be present, reflecting the online phase environment. However, there can also be other people and objects present and moving, thereby changing the power transmitted between sender and receiver. This is a major drawback for WiFi-based indoor positioning systems, especially in crowded environments.

3.2.2 RSS variations

Not only the body influences the Wi-Fi signal, but also differences in the environment e.g. physical objects contribute to the RSS variations in time. Due to walls and other structures, which are in the environment of the proposed positioning system, or by changes in the environment, multipath propagation usually prevail, causing significant fluctuations in the RSS. This effect, called fading, will be a disadvantage for the IPS when using a mobile device to observe the RSS [36]. This issue is most commonly remedied by calculating the average RSS value over a certain period. This should also be borne in mind in the proposed architecture for this thesis.

Another problem causing variations in RSS values is interference. The frequency band used by Wi-Fi radio signals is generally shared by other systems or devices e.g. microwave, Bluetooth devices. Interference may decrease RSS considerably when these devices are nearby and in operation.

RSS is thus greatly affected by the body effect, fading, and interference. Strong RSS values would be largely affected by fading, while weak RSS values may be effected by one or more of the three factors. Figure 3.2 depicts the results of an experiment that has been conducted in the lab to observe the changing RSS values in time. The Wi-Fi scanner from the Android smartphone collects the RSS values of a TP-Link WR340G access point for about 15 minutes with a sample rate of 1 second. Both the AP and the smartphone are situated in the lab environment with a distance of 1.5m from each other. The experiment has been conducted at a time when there was a normal activity in the lab. As can be seen in the figure, the lowest observed RSS value is -62 dBm and the maximum observed RSS value is -39 dBm: a difference of 23dBm. The average RSS value of the observations is -47.37 dBm and the median is -47 dBm. Note that there is a dip in the received signal strengths in the first minute of the experiment without an immediate explanation for this occurence.

3.2.3 Influence of the number of access points and reference points

The use of the number of access points is not directly related to the property of the Wi-Fi signal but it is certainly important to mention this as it can also be

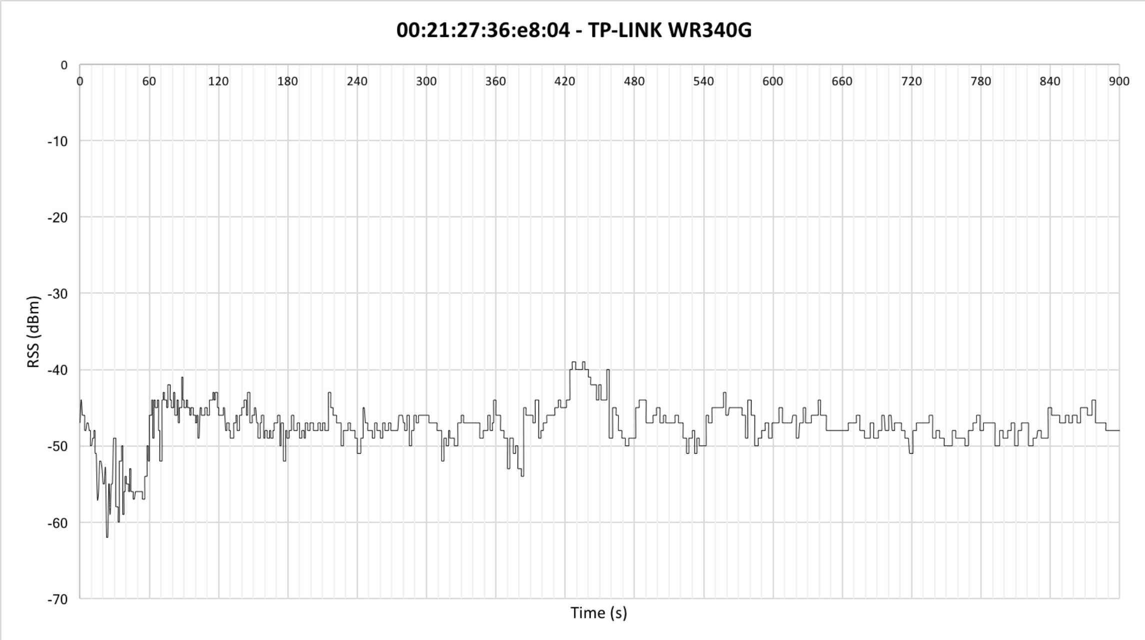


Figure 3.2: Experiment on RSS variations in lab

a disadvantage of using Wi-Fi. The number of access points in the environment, results in inaccurate fingerprint data, leading to poor performance. Placing additional APs to optimize the fingerprint data may incur additional costs and installing new APs can be time-consuming. [37] examines the relationship between the number of APs and reference points (RPs) in the fingerprint database that results into optimal localization results. An indoor space with dimensions of 11 x 23 m, should contain around 5 APs and around 66 RPs for optimal localization [37]. In fact, increasing the amount of APs or RPs barely influences the result [37].

3.2.4 Signal Aliasing

Another weakness that occurs with Wi-Fi signals is a phenomenon, called signal aliasing. According to [38], signal aliasing refers to:

> Two points that are far apart may be close together in signal space. Such aliasing can happen because of the complex indoor propagation environment. The signal strength at a point close to an AP may be similar to that at another point that is far away simply because of an obstruction (such as a wall) attenuating the signal received at the former point while the latter point receives an unobstructed signal.

To solve this problem, a first solution is the optimal placement of APs in the building.

Appropriate access point placement in the building layout is thus very essential in solving this problem.

This phenomenom should also be taken into account in the proposed architecture. The positioning algorithm should implement a solution for this. The RADAR system by [29] solves this by continuous user tracking. If the system is able to determine the location in an earlier position, it can chose between some calculated positions to determine the real position of the mobile device.

3.3 Related Work

There is a vast literature on indoor positioning systems with Wi-Fi fingerprinting [36, 39–45]. Most introduced systems achieve better accuracy by obtaining fingerprints from sources different from RSS [40, 46]. Alternatively, combining the Wi-Fi fingerprinting with IMU is also a common reasearch area [28]. This section elaborates on some of the most relevant systems useful for this thesis.

Xue W. et al. [36] present a positioning system with an improved Wi-Fi RSS measurement algorithm. After an analysis of the spatial resolution of the signal strength of Wi-Fi, they conclude that higher RSS values (good reception) produce smaller differential distance (better spatial resolution). So they propose a new RSS extraction algorithm where only high RSS values are employed for better positioning accuracy. They also prove that the mean of the RSS values is not an accurate reflection of the dynamic behavior of the RSS values. Instead, they first select M maximum RSS values and average these values for better positioning accuracy. The point of their system is to take a good value for M, this is determined by the curve smoothness index denoted by S and is defined as:

$$S = \sum_{i=2}^{N-1} \sqrt{(RSS_i - \frac{RSSI_{i-1} + RSSI_i + RSSI_{i+1}}{3})^2} \tag{3.1}$$

where N is the number of position points and $RSSI_i$ is the mean of the M selected maximum RSS values at the ith position point. The experiment concludes that there is not any consistent variation trend over the range of values of number M for every AP. Therefore one should make a sum of every smoothness index of every AP. Take M where the sum of every smoothness index is lowest, because the smoother the signal curve, the better quality of RSS value there is. The proposed algorithm is considerably better than the mean algorithm. It provides a positioning accuracy of

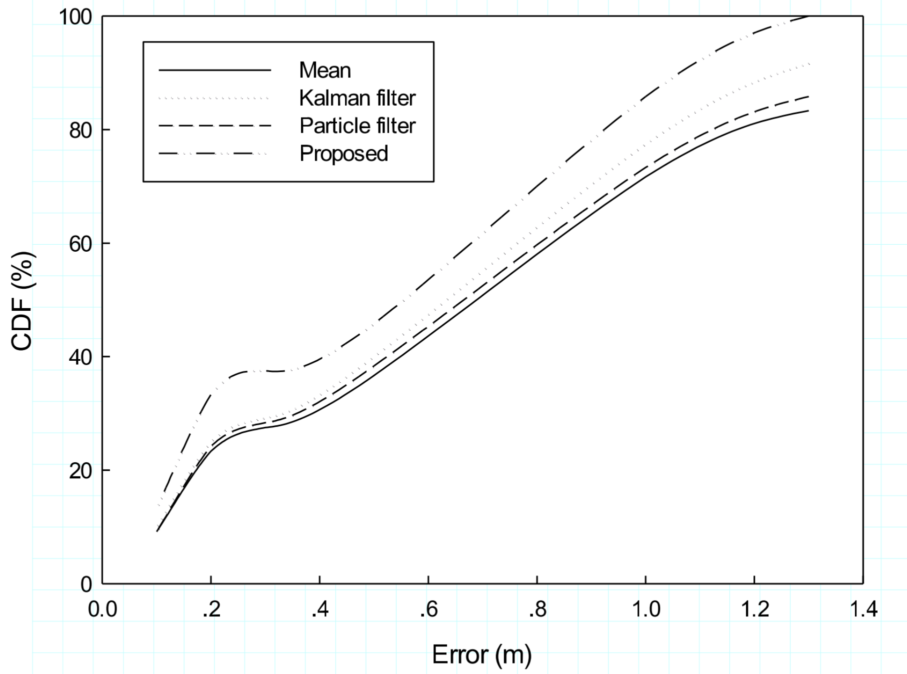


Figure 3.3: Comparison of location accuracy of proposed algorithm in [36] with three other algorithms in terms of cumulative distribution function (CDF)

84% at 1.0m error threshold, an improvement of 14% with respect to a classic mean-RSS algorithm, see also Figure 3.3. This picture shows the cumulative distribution function (CDF) for the positioning error. The horizontal axis is the positioning error in meters for the given probability functions. The vertical axis is the probability in percent. So the CDF shows the probability for the positioning errors less than or equal to a specific error. The algorithm has a lower computational complexity than the Kalman filtering and the particle filter algorithm. Because of the stronger ability of interference tolerance, the proposed algorithm has a better robustness than the other algorithms. It is a good idea to implement this method for the proposed IPS.

Another interesting indoor positioning system is the DeepFi system [46]. Many existing IPS using fingerprinting obtain the fingerprints from RSS values due to the simplicity and low hardware requirements. Because of the high variability over time and the coarse information, RSS does not exploit the many subcarriers in an orthogonal frequency-division multiplexing (OFDM) system for richer multipath information. DeepFi therefore implements the use of channel state information (CSI) from network interface cards (NICs). Compared with RSS, CSI is able to provide more information of the channel. Thus, replacing the use of RSS with CSI for indoor positioning systems results in a better ability to distinghuis locations. Therefore, time-varying effects are able to be overcome, improving the performance of the IPS. DeepFi only considers the amplitude of CSI, and the CSI phase information is ignored, which is largely due to the randomness and unavailability of the raw phase information [47]. CSI information for all the subcarriers and all the antennas is collected through the device driver and analyzed with a deep network with four hidden layers. About 60% of the used test points result in an error under 1m. Two experimental environments have been used, which result in a mean error of 1.2m in a living room scenario and 2.3m in a laboratory scenario. There exists abundant multipath and shadowing effects in the laboratory scenario, resulting in a larger mean error. Although DeepFi achieves good localization performances, the system still requires to create a fingerprint database via war-driving [27, 28] in the offline phase. An effective approach to reduce the burden of war-driving is crowd-sourcing, where the load of fingerprinting is shared by multiple users [48]. This research direction is very impressive but inaccessible for this thesis. CSI is in fact unavailable on most commercial WiFi devices and currently not available on the Android platform. DeepFi obtains the CSI from the Intel WiFi Wireless Link 5300 NIC. This is possible after modification of the firmware and the wireless driver [49].

As mentioned above, crowd-sourcing is a good way to reduce the cost of the fingerprint calibration process. UnLoc [28] extracts identifiable indoor places as landmarks of a building (e.g. an elevator imposes a distinct pattern on a smartphone's accelerometer, a corridor-corner may overhear a unique set of WiFi access points, a specific spot can have an unusual magnetic fluctuation [28]). UnLoc then applies dead-reckoning schemes to track users between these landmarks. New landmarks in the building are discovered from analyzing sensor data from different users in the building. This autonomous discovering can further help localization. Some WiFi fingerprinting systems can even automate this training via crowd-sourcing using mechanisms of increasing sophistication (e.g. Redpin [50], OIL [51], WiFi-SLAM [52], Zee [53], RCILS [40], etc.).

Another crowd-sourcing system is the EZ [39] Localization algorithm. The system's users carry Wi-Fi-enabled devices traversing a space in normal course. Even if the physical layout of the building or the placement of the APs is not known a priori, EZ can calculate an estimation of the user's location. This is achieved by mapping the relationship between RSS values and locations, with some occasional location report or fix from GPS of the device at the entrance or near a window. Other users are then located based on the RSS mapping scheme based on the relative signal measurements. EZ defines the received signal strength s_{ij} at a location with vector $\mathbf{x_j}$ of a user given distance d_{ij} from the i^{th} access point using traditional path loss model:

$$s_{ij} = s_i^0 - 10\gamma_i \log d_{ij} + \mathbf{R} \tag{3.2}$$

$$d_{ij} = \sqrt{(\mathbf{x_j} - \mathbf{c_i})^T(\mathbf{x_j} - \mathbf{c_i})} \tag{3.3}$$

where s_i^0 is the transmit power, the RSS from this i^{th} access point at a distance of one meter, γ_i, the path loss exponent, captures the rate of fall of RSS in the vicinity of the i^{th} AP and $\mathbf{c_i}$ is the estimated location vector of AP i. $\mathbf{R}$ is a random variable that hopes to capture the variations in the RSS.

When three true, non-collinear locations (either AP or mobile users) are known, the equations of the system are uniquely solvable. All the other locations can be calculated. Note that prior knowledge of the RF environment is not required but the system occasionally needs GPS signals in order to obtain these three true locations. Overall, the EZ system reduces the survey cost significantly. The researchers deploy the system in two different buildings wich yields a median localization error of $2m$ in a small building and $7m$ in a large building respectively.

RCILS [40], a Robust Crowd-sourcing-based Indoor Localization System, is one of the most recent proposed IPS with crowd-sourcing. The system automatically constructs a WiFi radio map of a building using data collected from nearby smartphones. RCILS also reduces the influence of the variations of the RSS values by implementing a sequence-base radio map. The system is based on two key observations: the indoor map constrains people's activities and trajectories in the environment and the collected RSS vectors are continuous. RCILS matches the coordinates of these trajectories with the RSS values and provides them with location information. After their experiments they can conclude that the changing trend of the same path at different times are similar, even when, due to the environmental changes, the RSS values are different at different times. Another conclusion is that the changing trends of the RSS values collected by different types of smartphones are also similar. RCILS improves the robustness of crowd-sourcing-based indoor localization systems. The system has a median error of approx. 1.6m inside a medium sized academic building ($2750m^2$) [40].

Not only obtaining fingerprint data is important in a Wi-Fi fingerprinting localization system, it is also important to update the database information in the dynamically changing environment [41]. Due to various factors, discussed in Section 3.2, the current signal strengths may vary significantly compared to the signal strengths saved in the database. Therefore, the database becomes outdated, resulting in large estimation errors during localization. Because sporadically conducting new site surveys is not cost-effective, recent research also focuses on adapting the fingerprint database to these signal variations.

Recent research has two approaches to address this problem: one deploys external infrastructures to monitor the signal variations (infrastructure-based schemes) [42, 43] or uses algorithmic adaptation to fingerprint signal noise (non-infrastructure-based schemes) [44, 45].

[42] utilizes extra Wi-Fi monitor beacons (sniffers) for monitoring environmental dynamics. This approach is feasible because most current beacons (such as Wi-Fi and ZigBee stations) have both transmitting and receiving capabilities. These beacons observe the RSS values from other beacons (inter-beacon measurement). Because of this, gathered radio maps are calibrated on the fly without the need for additional hardware.

[43] implements a modified Bayesian regression algorithm to estimate the current RSS values probability distribution in the building space. These estimations are based on the observations from APs in the online phase. The system assumes Gaussian prior centered RSS values over a logarithmic pass loss mean. [43] expresses the signal strength, $s_{\mathbf{x}}$ at a location $\mathbf{x}$ as

$$s_{\mathbf{x}} = f(\mathbf{x}) + \epsilon \tag{3.4}$$

where ϵ is an additive zero-mean Gaussian noise and f is the estimated process output function for random values of x. Given N reference locations $\mathbf{X}$ and fingerprints $\mathbf{S}$, let the N-by-N matrix $\mathbf{K}$ be the covariance matrix between these samples. An element $k(\mathbf{x_i}, \mathbf{x_j})$ in $\mathbf{K}$ is usually given by an exponential kernel. In Bayesian analysis, instead of learning the weights like in neural networks [54], Gaussian Regression learns the kernel (covariance of training data) [55]. Based on a Gaussian process [43] with covariance $\sigma_n^2 I$, the predicted mean RSS values $\mu_{\mathbf{x}^*}$ at this location $\mathbf{x}^*$ is given by

$$\mu_{\mathbf{x}^*} = m(\mathbf{x}^*) + k(\mathbf{x}^*, \mathbf{X})(\mathbf{K} + \sigma_n^2 I)^{-1}(\mathbf{S} - m(\mathbf{X})) \tag{3.5}$$

where σ_n are the hyper-parameters of the Gaussian Process. The mean function $m(\mathbf{x}^*)$, is given by either kernel regression (using Bayesian inference) or propagation model regression. Training of these machine learning algorithms is usually computationally expensive, but experiments of the proposed algorithm shows that the dynamic radio map provides a 2-3m accuracy. This is comparable to results of an up-to-date offline radio map [43]. The experiments also show the consistency of the estimated location measured with the actual positioning location.

Instead of using sniffers for adapting the fingerprint database, [44] uses a Manifold co-Regularization, which is a machine learning technique for building a mapping function between data. Hereby it is assumed that RSS values from nearby positions have more similar values than those far away. So observations from a different time period should correspond to the same locations if no big changes do occur in the environment. The mapping function between the signal space and the physical location space is adapted dynamically by several labeled data from reference points and a few unlabeled data in a new time period. However, the system's algorithm relies on the stored fingerprints and aims at approaching small signal variation. Therefore, it cannot adapt to large environmental changes e.g. when APs are changed in transmission power, removed or added.

To adapt the fingerprint database to these large environmental changes user feedback or crowd-sourcing is needed [45], as already addressed in this section. In some cases, it may be inconvenient to prompt users to upload their collected signal data. Moreover, feedback from users is even not always reliable. So, error filtering is probably needed before updating the database.

Next to construct and adapt the fingerprint database, it is also important to keep in mind that users in the online phase are possibly using heterogeneous smartphones, and thus also obtain fingerprint data with different network interface cards. This will predominantly cause problems when RSS data is collected during crowd-sourcing for optimizing the fingerprint database. [56] tries to find a solution for these incorrect RSS values. They have evaluated the use of the difference in uploaded RSS values by various devices. [56] conclude that their proposed algorithm is more robust to device variations. Disregarding the amount of contributing devices, the system preserves the localization accuracy.

Much research also focuses on the localization algorithm itself used in the online phase. An interesting direction has been made by [57]. Their system implements an improved Spearman-distance-based K-Nearest-Neighbor (KNN) scheme. It is assumed that the ranking of the received AP signals is more likely the same or similar even if the absolute RSS values are quite different. The improved Spearman-distance-based KNN algorithm results in a localization error under 2.7m for 80% of the test samples.

Most reasearch implements a positioning algorithm that is either a deterministic or a probabilistic algorithm. These algorithms use real-time matching of RSS fingerprint data or the probability distribution of the signal strengths to obtain a user's location respectively.

Probabilistic Positioning describes the signal strengths with a probability distribution function and use a Bayesian system, most commonly together with clustering, to estimate the location of the user. Some examples of probabilistic positioning systems are: [58] which implements a Bayesian hierarchical model for positioning and [59] which utilizes the maximum likelihood function for positioning.

The most common techniques for deterministic positioning are data mining and machine learning algorithms [29, 57, 60, 61] such as KNN, Support Vector Machine (SVM) and artificial neural network. The KNN algorithm is an easy-to-use machine learning algorithm and is widely applied in indoor positioning systems because of its simplicity and high performance. The algorithm calculates the position based on the distance between the reference fingerprint and the fingerprints in the database. Various formulas can be used for distance calculation, e.g.: Manhattan distance or Euclidean Distance. It finds the K best matching fingerprints based on their mutual distance and calculates the position as the average of the positions of these K fingerprints. The WKNN [60] is a variant of KNN, this algorithm uses weights to calculate the position instead of just averaging the positions of the best matching fingerprints.

A large number of artificial intelligence technologies can be used as an approach for positioning algorithms and finding the best matching fingerprint from the database. Thus, many researchers use artificial neural networks in their positioning systems, as it is one of the most important methods in machine learning. Recent research implements multilayer perceptron [62], multi-layer neural networks [63] or regression neural networks [64] for their positioning algorithm.

Research also uses SVM [65,66] as the positioning algorithm. The SVM method is also based on the theory of statistical learning and is commonly used for data classification and regression. SVM classifies data by finding the optimal hyperplane, i.e. the plane with the largest margin between two classes, that separates all data points of one class from another class. Therefore, optimal nearest neighbor fingerprints can be obtained by SVM classification of the fingerprint database.

4 Proposed System Architecture

This chapter provides an overview of the implemented algorithm for the proposed indoor Wi-Fi fingerprint positioning system.

4.1 Introduction

A global overview of the proposed system architecture is shown in Figure 4.1. Obtaining the fingerprints is done through an Android application. Matching the target fingerprint with the database of collected fingerprints is done remotely on a webserver written in Erlang [67]. Erlang is a functional language with a declarative style of function declaration. This allows the program implementation to be done in a short time. Tuples are used to represented data structures and lists are very efficient to manipulate data in Erlang. One tries to find the device's location based on a best match algorithm by sending a target fingerprint via a HTTP request to the server. In the offline phase the database is built. The online phase is designed for localization.

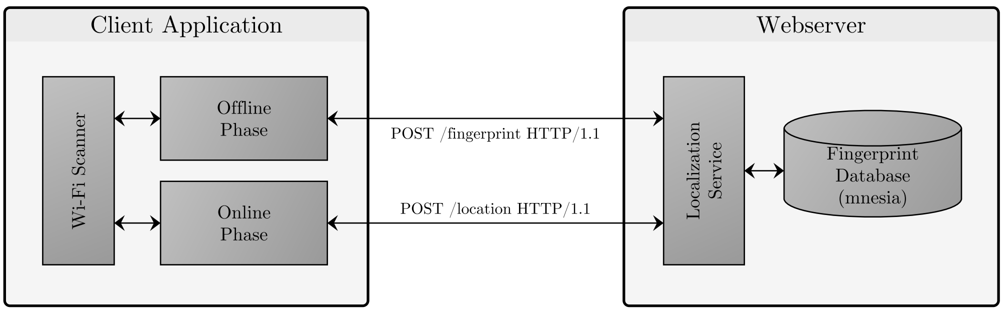
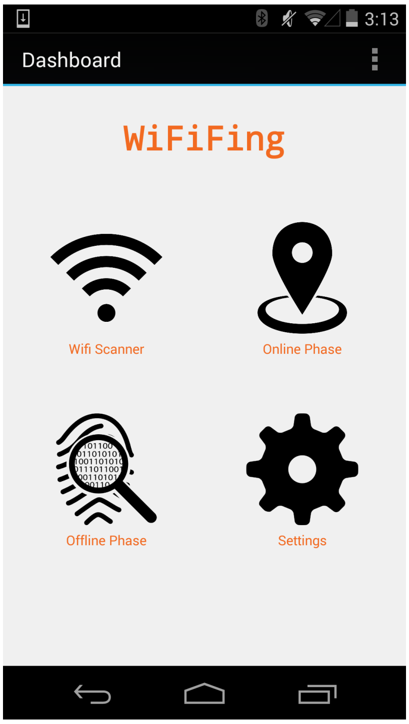
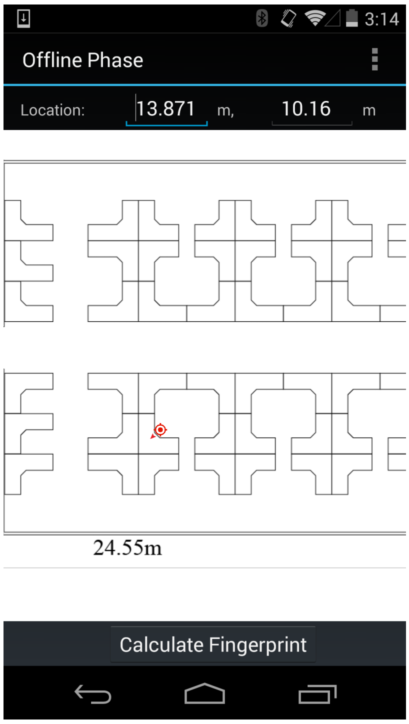


Figure 4.1: Overview of System Architecture

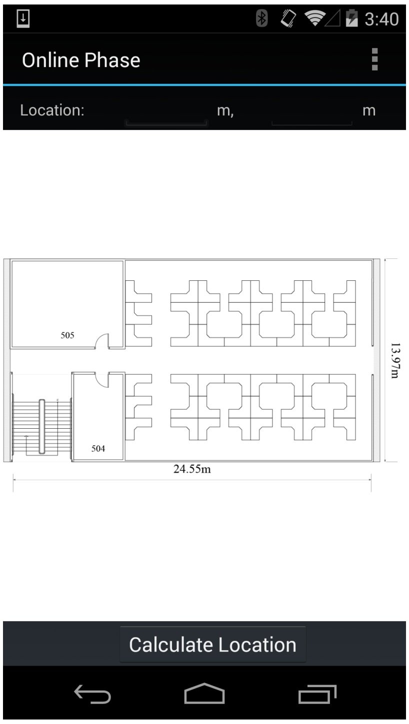
Figure 4.2 depicts the view of the Android application, named WiFiFing. The



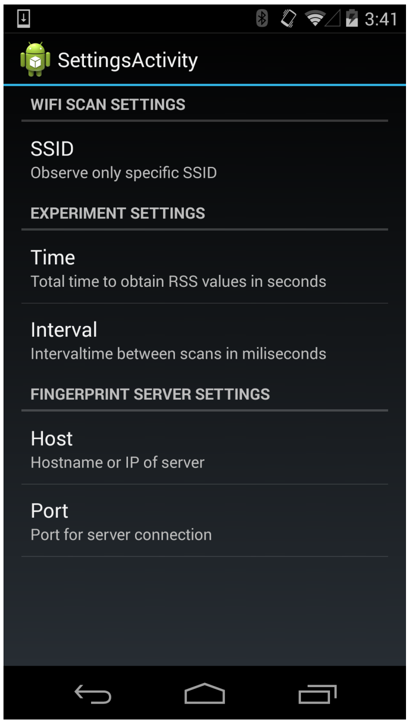
(a) Dashboard screen



(b) Offline Phase Activity



(c) Online Phase Activity



(d) Settings Activity

Figure 4.2: View of WiFiFing application

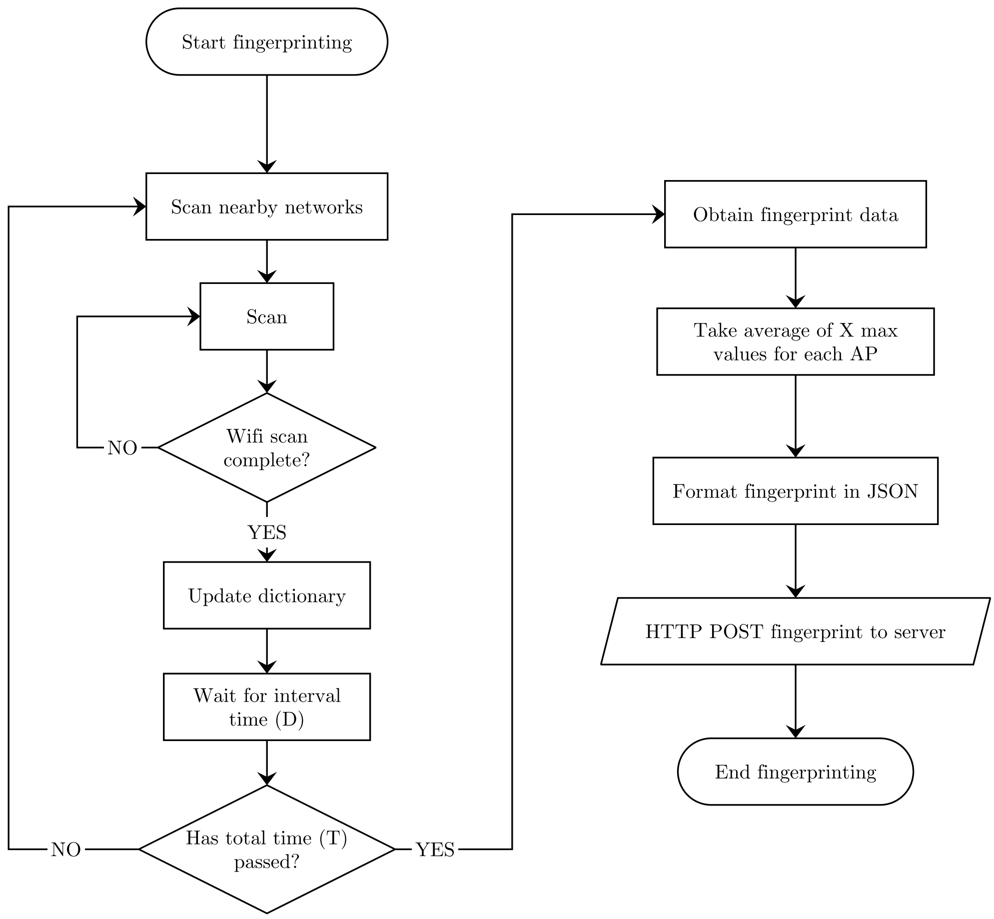


Figure 4.3: Offline phase flowchart

application is composed of 4 main parts: 4.2a the dashboard menu, 4.2b the offline phase, 4.2c the online phase and 4.2d the settings page. Next sections focus on the offline and online activity of the application that corresponds to the offline phase and online phase of the Wi-Fi fingerprint system respectively. The purpose of the dashboard and settings page in the application speaks for itself.

4.2 Offline Phase

4.2.1 Initialization

Figure 4.3 depicts the backend workflow when the user makes a fingerprint in the offline phase. This is initiated by pressing on the provided button in the application after the current location is set. The current location is determined by the user by pressing on the location in the map or by typing the location in meters

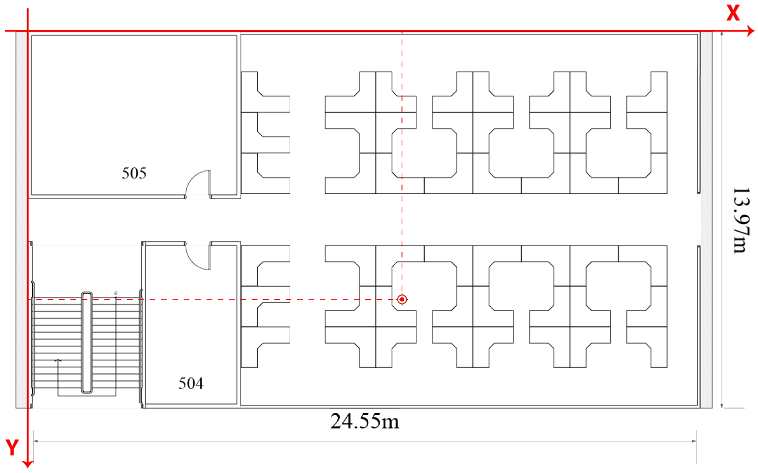


Figure 4.4: Cartesian coordinate system on experiment floor map with a marked location at (14m, 10m)

in the textviews. A Cartesian coordinate system is used with the origin at the top left corner of the map, with inverted y-axis, as is shown in Figure 4.4.

4.2.2 Obtaining fingerprint

After initialization, the first scan results are provided by the Android WifiManager class. At this moment the total scan takes T milliseconds with a sample rate of D milliseconds, T and D can be set in the settings of the application. After every scan, the dictionary containing the scan results is updated. The scanner only saves the acces points (APs) with the SSID equal to the provided SSID from the settings page by the user. The user can hereby select the fixed APs in the environment. From the moment that the total scan time is reached, all the fingerprint data are retrieved. The value for every BSSID key in the dictionary is reduced to the average of the X maximum values observed during the scan, based on the algorithm using the average of a number (X) of selected maximum RSSI observations [36]. Before taking the average of X RSS values from the obtained RSS vector, the received RSS values are sorted by descending order. Because of this the occasional poor sampled RSS values are removed before saving the fingerprint in the database. RSS vectors whose amount of RSS values is less then X, are ignored when building the fingerprint, the observed BSSID is not included in the final fingerprint. This also reduces the fingerprint size, saves storage in the database and also indirectly ensures faster response time from the server. The final fingerprint saved

```
{
"x" : 550,
"y" : 400,
"rssvals":[
        {"BSSID":"80:f6:2e:14:b5:90/ZJUWLAN","RSSI":-64},
        {"BSSID":"80:f6:2e:14:b5:30/ZJUWLAN","RSSI":-74},
        {"BSSID":"80:f6:2e:14:b4:f0/ZJUWLAN","RSSI":-78},
        ...
        {"BSSID":"80:f6:2e:14:fc:f0/ZJUWLAN","RSSI":-81}
]
}
```


Figure 4.5: Encoded JSON example

in the database consist of the location of the reference point and every obtained RSS value in dBm together with the BSSID of the signal at that point.

4.2.3 Saving fingerprint

Next, the dictionary is formatted to a JavaScript Object Notation (JSON) string and sent with a HTTP request to the server on the host. The host address is set in the settings activity of the application. Figure 4.5 shows an example of a possible JSON string send by the application to the server.

The server will listen to incoming requests. When a fingerprint HTTP POST request is received, the received JSON is decoded and the fingerprint is saved in the database. The server implements a Mnesia database [68]. Mnesia is a distributed, soft real-time database management system written in the Erlang programming language. If the fingerprint is succesfully saved, the server sends a HTTP 200 status-code [69] back to the application. From the moment that the Android application receives this message, the fingerprinting process is finished.

4.3 Location determination

4.3.1 Initialization

As is shown in Figure 4.6, the initialization for the online phase is very similar to that of the offline phase. In fact, making the fingerprint is completely handled by

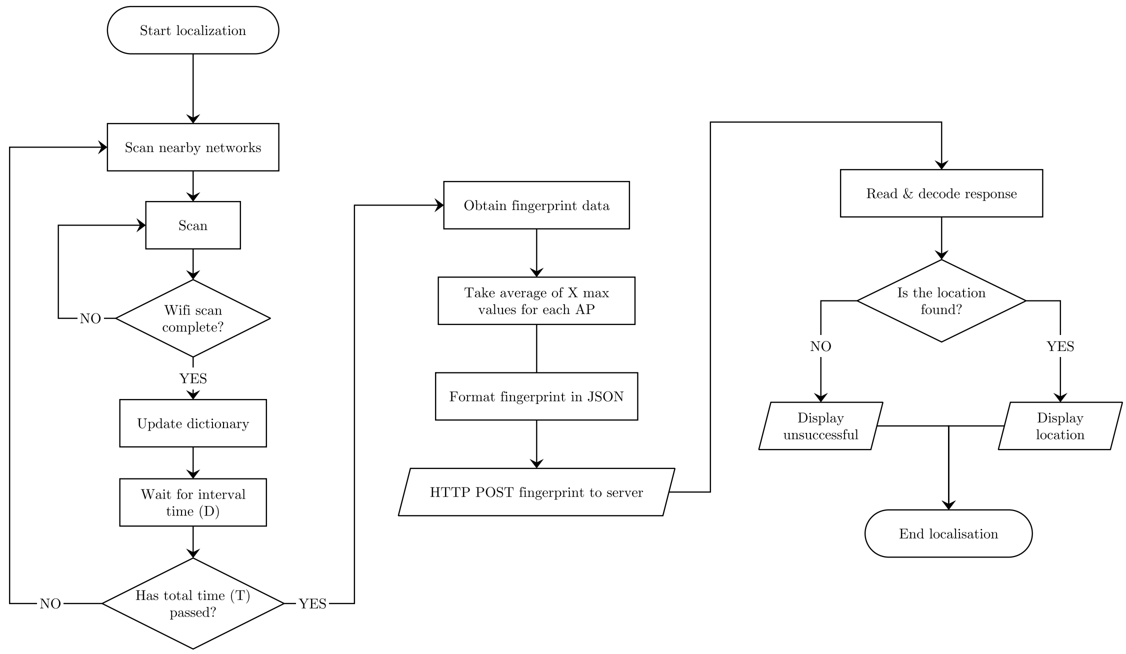


Figure 4.6: Online phase flowchart

the same code as in the offline phase. Ofcourse, the HTTP POST request doesn't contain a value for the x and y location.

The server responds with a HTTP 200 status-code containing a message and the calculated location, if found by the matching algorithm. The application displays the calculated location on the map or displays an unsuccessful message for the user, the localization process is finished.

4.3.2 Obtaining location

To match the target fingerprint with the fingerprints in the mnesia database, a best match algorithm, the modified WKNN [60] algorithm, is implemented on the server. By using Erlang as the programming language, this modified WKNN algorithm is implemented based on the Spearman Distance, the Euclidean distance, omitted RSS values and their mutual ranking.

4.3.2.1 Spearman Distance

First the database is filtered based on the best Spearman distance [57], resulting in fingerprints with the same ranking as the target fingerprint. Even if the absolute RSSI values of the discovered APs might be quite different, their ranking is more likely to be similar. This is based on the assumption that the RSSI values monotonically decrease when the distance between the source and APs increases. The received signal strengths are ranked based on their value. If two of more signals share the same signal strenght, they areăeach assignedăfractional ranksăequal to the average of their positions in the ascending order of the values.

To evaluate this ranking, the Spearman rank correlation coefficient [70] is utilized which is a measure of statistical dependence between two variables. This Spearman's coefficient evaluates how well the relationship between two variables can be described using a monotonic function and is appropriate for both continuous and discrete variables, including ordinal variables.

The Spearman's Distance can be calculated with the following formula [57]:

$$\rho = 1 - \frac{6 * (\sum d_i^2 + c)}{n * (n^2 - 1)} \tag{4.1}$$

where $\sum d_i^2$ is the sum of the d-squared values for the fingerprint with d the difference between target (T) and reference (R) in ranking for the i^{th} record in the fingerprint:

$$d_i = r_{Ti} - r_{Ri} \tag{4.2}$$

with r_{Ti} and r_{Ri} the ranking value of the the i^{th} target and reference record in the fingerprint respectively. n is the number of records in the fingerprint and c is a correction factor to be summed with the sum because the fingerprint can contain tied rankings. Tied ranks occur where two items in a column have the same rank because they share the same value. c can be calculated with the following formula [70]:

$$c = \sum^{ties} \frac{m * (m^2 - 1)}{12} \tag{4.3}$$

where m is the size of the tied rank.

Example The following example makes it clear how the ranking is evaluated and how the Spearman distance is calculate in the proposed algorithm. Looking at Table 4.1, the target fingerprint and the reference fingerprint can be seen. The reference fingerprint is a fingerprint obtained from the database. Notice that not every BSSID from the target is included in the reference fingerprint, and vice versa.

Thus, the first step is to sort and filter the 2 lists based on their BSSID value so that they match each other, as is shown in resulting Table 4.2. The algorithm also keeps track of how many times a signal is ommited from the reference fingerprint and from the target fingerprint. This number, called the *dropped* value, is used later to

Target		**Reference**	
BSSID	RSS (dBm)	BSSID	RSS (dBm)
80:f6:2e:14:b5:30	/	80:f6:2e:14:b5:90	-55
80:f6:2e:14:b5:90	-50	80:f6:2e:14:fe:10	/
80:f6:2e:14:b5:70	-73	80:f6:2e:14:b5:30	-64
80:f6:2e:14:b4:d0	-63	80:f6:2e:14:b5:70	-70
80:f6:2e:14:b4:f0	-79	80:f6:2e:14:5f:50	-87
80:f6:2e:14:b5:20	-63	80:f6:2e:14:b4:f0	/
80:f6:2e:14:b5:80	-55	80:f6:2e:14:b5:80	-52
80:f6:2e:14:fe:10	-79	80:f6:2e:14:b4:d0	-64
80:f6:2e:14:5f:50	-86	80:f6:2e:14:b5:20	-63

Table 4.1: Example of target and reference fingerprint data. Non-received signals are indicated by /.

determine the best fingerprint results. Then, the next step is to give each record its ranking. For the ranking the target fingerprint's RSS values, a tied rank can be seen. One can not know wich record should get the 3rd or 4rd rank. Therefore both records get as rank the mean of the tied ranks. The last 2 columns of Table 4.2 calculate the difference of the ranking values and their squared values respectively.

The sum of the d-squared values is calculated:

$$\sum d_i^2 = 1 + 0 + 0.25 + 0.25 + 1 + 0 = 2.5 \tag{4.4}$$

There is only 1 tied rank with size 2, so c is calculated:

$$c = \frac{2 * (2^2 - 1)}{12} = 0.5 \tag{4.5}$$

Now the Spearman Rank Correlation is calculated:

$$\rho = 1 - \frac{6 * (\sum d_i^2 + c)}{n * (n^2 - 1)} = 1 - \frac{6 * (2.5 + 0.5)}{6 * (6^2 - 1)} = 0.8571428571 \tag{4.6}$$

BSSID	Target (dBm)	Reference (dBm)	r_{Ti}	r_{Ri}	d	d^2
80:f6:2e:14:b5:90	-50	-55	1	2	-1	1
80:f6:2e:14:b5:70	-73	-70	5	5	0	0
80:f6:2e:14:b4:d0	-63	-64	3.5	4	-0.5	0.25
80:f6:2e:14:b5:20	-63	-63	3.5	3	0.5	0.25
80:f6:2e:14:b5:80	-55	-52	2	1	1	1
80:f6:2e:14:5f:50	-86	-87	6	6	0	0

Table 4.2: Target and reference after match and sort

The Spearman Rank Correlation for this set of fingerprints is thus 0.8571428571, which is not a very good correlation for the ranking. In the best case the Spearman Rank Correlation is 1 or at least greater than 0.9.

The algorithm is provided in pseudo-code, see Algorithm 1. Based on a given threshold (e.g. 0.9), the algorithm returns a list of all the fingerprints with a Spearman's Distance value above or equal to the threshold. The Spearman Distances for all fingerprints is sorted in descending order.

4.3.2.2 Euclidean Distance

After the calculation of the Spearman Distance, the Euclidean Distance is calculated between the target and the fingerprints with a good Spearman Distance value. In mathematics, the Euclidean distance or Euclidean metric is the ordinary distance between two points that one would measure with a ruler, and is given by the Pythagorean formula [71].

In Cartesian coordinates, if $p = (p_1, p_2, ..., p_n)$ and $q = (q_1, q_2, ..., q_n)$ represent two points in a plane, then the distance from p to q is given by the following equation:

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(p_i - q_i)^2} \tag{4.7}$$

The algorithm is provided in pseudo-code, see Algorithm 2. Based on a given threshold (e.g. 5), the *euclidean* procedure returns a list of all the fingerprints with a Euclidean Distance value below or equal to the threshold, in ascending order.

4.3.2.3 Best matching fingerprint

Before calculating the estimated location, the K nearest neighbors of the target needs to be found. Therefore, the fingerprints are ranked based on the resulting rank of the Spearman Distance $\mathbf{S_i}$, the amount of dropped signals $\mathbf{D_i}$ and the Euclidean Distance $\mathbf{E_i}$. The total ranking $\mathbf{R_i}$ for each fingerprint can be calculated as:

$$\mathbf{R_i} = w_S * \mathbf{S_i} + w_D * \mathbf{D_i} + w_E * \mathbf{E_i} \tag{4.8}$$

w_S, w_D and w_E can be considered as tuning parameters in the algorithm. Best results are currently obtained with $w_S = 0.30$, $w_D = 0.25$ and $w_E = 0.45$.

Based on this final ranking, the best results are the K best ranked fingerprints. Ofcourse the choice of K is also very critical. If there are n fingerprints found that closely matches the target fingerprint, then is K [72]:

$$K = \sqrt{n} \tag{4.9}$$

Instead of just averaging the K best ranked fingerprints, the algorithm will give each K best ranked fingerprints their weight based on the Euclidean distance calculated in the previous step. This follows the formula:

$$\begin{cases} (\hat{x}, \hat{y}) = \sum_{i=1}^{K} w_i(x_i, y_i) \\ \sum_{i=1}^{K} w_i = 1 \end{cases} \tag{4.10}$$

where $(\hat{x}, \hat{y})$ is the estimated location and (x_i, y_i) represents the i-th reference location. In the formula above, w_i is the weight of the i-th nearest neighbor point and can be calculated by the following formula:

$$w_i = \frac{\frac{1}{d_i^2}}{\sum_{j=1}^{K} \frac{1}{d_j^2}} \tag{4.11}$$

with d_i the Euclidean distance between the i-th reference fingerprint and the target fingerprint. When using formula (4.11), smaller Euclidean distances will result in larger values for the weight. The final estimation for the location can be calculated using formula (4.10). After the calculation, this location is returned by the localization API.

Algorithm 1 Calculate the correlation between the target fingerprint and the reference fingerprint based on their ranking

```
procedure SPEARMAN(Target, References, Distance)
    BestMatchRanking = new empty list
    for each fingerprint FP in References do
        SpearmanDistance = spearman_correlation(Target, FP)
        if SpearmanDistance >= Distance then
            add FP.Location to BestMatchRanking
        end if
    end for
    return BestMatchRanking sorted descending on spearman distance
end procedure
procedure SPEARMAN_CORRELATION(Target, Reference)
    Match Target and Reference to each other (same size and order)
    Give Target and Reference ranking values
    Sum = 0
    Length = size(Target)
    for each rank value RT in Target and corresponding RR in Reference do
        Difference = (RT − RR)^2
        Sum = Sum + Difference
    end for
    for each tied rank in the ranking of Target or Reference do
        M = number of ties in specific rank
        Sum = Sum + M ∗ (M^2 − 1)/12
    end for
    return 1 − 6 ∗ Sum/(Length ∗ (Length^2 − 1))
end procedure
```

Algorithm 2 Calculate best Euclidean distances between target fingerprint and list of fingerprints, received after the Spearman distance calculation, based on given distance threshold for the euclidean distance

```
procedure EUCLIDEAN(Target, References, Distance)
    BestMatch = new empty list
    for each fingerprint FP in References do
        EuclideanDistance = euclidean_distance(Target, FP)
        if EuclideanDistance <= Distance then
            add FP.Location to BestMatch
        end if
    end for
    return BestMatch sorted ascending on euclidean distance
end procedure
procedure EUCLIDEAN_DISTANCE(Target, Reference)
    Sum = 0
    for each access point AP in Target do
        Key = AP.BSSID
        if Key exists in Reference then
            TargetLevel = AP.Level
            ReferenceLevel = Reference[Key].Level
            Sum = Sum + (TargetLevel − ReferenceLevel)^2
        end if
    end for
    return sqrt(Sum)
end procedure
```

5 Experimental Results

In this chapter the results of the experiments conducted to obtain the final goal of this thesis will be discussed. The effect of different parameters in the system is analyzed. These parameters are:

- the interval time between AP scans (D),
- the total scanning time (T),
- the amount of (maximum) values (X) for calculating the average RSS value.

Then, the positioning accuracy of the proposed system is examined on the basis of some experiments. Finally the positioning accuracy is compared with the results from recent state-of-the art Wi-Fi fingerprint-based positioning systems.

5.1 Analysing effect of various parameters

This section discusses the effects of various parameters in the system. Based on the next conclusions, the experimental settings for the rest of this chapter are determined.

5.1.1 Effect of D, the interval time between AP scans, on fingerprint

An optimal situation is when the interval time between AP scans (sample time) is as small as possible, to obtain the most RSS information of the nearby APs. This should result in a more accurate fingerprint. However, the proposed system is for this case limited by the operating principle of the Android *WifiManager* class [73]. This class provides the primary API for managing all aspects of Wi-Fi connectivity, including scanning for nearby Wi-Fi signals.

When a new AP scan is initiated, a new worker thread is started by the API. However, the results of the scan are of course not immediately available, but are made available after the API throws an asynchronous event. Receiving this event shows that the scan is complete and results are available.
This means that the system is limited when choosing a good value for the interval time. It is possible to start hundreds of AP scans in 20 seconds, but it is more likely that only 15-20 scans will complete in 20 seconds. So a compromise must be found for D to avoid creating unnecessary worker threads in the Android application.
An experiment has been conducted to examine the effect of the interval time on the fingerprint. For this experiment the total scanning time (T) is set to 15s, the amount of (maximum) values (X) before averaging is equal to 10 and various values for D are alternated for 3 measurements each time. Results of this experiment are shown in Table 5.3 to 5.7 for D = 100ms, D = 500ms, D = 1000ms, D = 1500ms and D = 2000ms respectively. The variance $\mathrm{Var}(X)$ of the RSS values is calculated for the AP's RSS values that are received in every measurement with the following formula:

$$\mathrm{Var}(X) = \frac{1}{3}\sum_{i=1}^{3}(RSS_i - \mu)^2 \tag{5.1}$$

where RSS_i is the received signal strength of the i^{th} measurement and average $\mu = \frac{1}{3}\sum_{i=1}^{3} RSS_i$. If some signals are not detected by all of the three measurements, the variance is not calculated. Based on the variances, it is clear that D = 1500ms also has the smallest average variance value (0.4928), as is shown in Table 5.1. 1500ms is thus a good value for the interval time of the system (D). It results in a scan with more similar discovered APs.

5.1.2 Effect of T, the total scanning time, on fingerprint

For the total scanning time T, a similar experiment has been performed. In this case the system is not limited by the Android API. However, it is not wise to choose a high value for T, resulting in a long scanning time. A long waiting time to obtain the fingerprint would definitely not be user-friendly for the user who wants to know his location, as well as for the person(s) creating the database in the offline phase. Therefore, a trade-off between the system's ease of use and performance needs to be made. A maximum waiting time for the user of 20 seconds is still doable. Therefore, the experiment examines the effect of T for the following values: T = 2s, T = 5s, T = 10s, T = 15s and T = 20s. In this part of the experiments, D is set to 1500ms and

	D = 100	**D = 500**	**D = 1000**	**D = 1500**	**D = 2000**
BSSID 1	1.127	2.074	4.153	0.452	0.276
BSSID 2	0.469	0.363	0.265	0.667	-
BSSID 3	4.259	-	6.79	0.889	4.12
BSSID 4	2.735	-	-	0.222	3.389
BSSID 5	-	-	-	0.234	1.102
BSSID 6	-	0.222	-	-	-
BSSID 7	-	-	-	-	-
BSSID 8	-	0.222	-	-	-
BSSID 9	-	-	-	-	-
BSSID 10	-	-	-	-	-
Average	**2.1475**	**0.72025**	**3.736**	**0.4928**	**2.22175**

Table 5.1: Variances of the equal received AP RSS values with varying value of the interval time (D)

X is equal to 10. Results of this experiment is shown in Table 5.10 to 5.14 for the above-mentioned values for T.

As in the previous experiment, the variances of the received RSS values which have been received by each measurement can be calculated to obtain a better understanding of the results. From Table 5.8 it is clear that the system does not benefit from a short scanning time. A total scanning time of 2 seconds (T = 2000) results in unstable fingerprints. There are only 2 APs that are received by each measurement and moreover it results in a large variance for one of these two APs. The other values for T overall result in at least 4 measurments without obmitted signals, with the exception of a total scanning time of 20s, resulting in 6 measurements of equal received APs. A total scanning time of 10s and 20s are a good choice for the system based on the results of this experiment. In fact, the difference in average variance is liquidated by the difference in the amount of equal received signals in this case, making a total scanning time of 20s the choice for the system. This is also confirmed in Figure 5.1 that shows the RSS value for each measurement of the 80:f6:2e:14:b5:90/ZJUWLAN signal. This signal results on the place of the experiment as the best received signal strength in comparison with the other signals, therefore it should be the most stable signal. According to this figure,

#	BSSID/SSID
BSSID 1	80:f6:2e:14:b5:90/ZJUWLAN
BSSID 2	80:f6:2e:14:b4:d0/ZJUWLAN
BSSID 3	80:f6:2e:14:b5:30/ZJUWLAN
BSSID 4	80:f6:2e:14:b5:80/ZJUWLAN
BSSID 5	80:f6:2e:14:b5:70/ZJUWLAN
BSSID 6	80:f6:2e:14:b5:20/ZJUWLAN
BSSID 7	80:f6:2e:14:b4:f0/ZJUWLAN
BSSID 8	80:f6:2e:14:b4:c0/ZJUWLAN
BSSID 9	80:f6:2e:14:b4:f0/ZJUWLAN
BSSID 10	80:f6:2e:14:fc:f0/ZJUWLAN

Table 5.2: BSSID/SSID Legend

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-61.2	-61.3	-59
BSSID 2	-72	-71.33	-70.33
BSSID 3	-65	-66.86	-70
BSSID 4	-66	-66.86	-63
BSSID 5	-74.56	-	-73
BSSID 6	-	-68.33	-69.33
BSSID 7	-	-	-74.6
BSSID 8	-	-	-
BSSID 9	-	-	-
BSSID 10	-	-	-

Table 5.3: D = 100 ms with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-58.7	-62	-61.43
BSSID 2	-71.17	-70	-69.8
BSSID 3	-67	-	-68.5
BSSID 4	-	-70	-
BSSID 5	-73.29	-73.14	-
BSSID 6	-72	-71	-71
BSSID 7	-	-	-
BSSID 8	-75	-74	-74
BSSID 9	-	-	-74
BSSID 10	-75	-	-

Table 5.4: D = 500 ms with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-62.71	-61.78	-58
BSSID 2	-68.17	-69.17	-68
BSSID 3	-63.71	-69.71	-68.6
BSSID 4	-66	-	-66
BSSID 5	-	-	-
BSSID 6	-	-72	-74
BSSID 7	-74.67	-	-
BSSID 8	-74	-	-
BSSID 9	-	-	-
BSSID 10	-	-	-

Table 5.5: D = 1000 ms with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-59.43	-58.5	-60.14
BSSID 2	-71	-70	-69
BSSID 3	-70	-68	-70
BSSID 4	-66	-67	-67
BSSID 5	-72.43	-73.4	-73.5
BSSID 6	-	-	-74
BSSID 7	-75.6	-	-
BSSID 8	-	-	-
BSSID 9	-	-	-
BSSID 10	-	-	-

Table 5.6: D = 1500 ms with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-59.4	-59.2	-60.4
BSSID 2	-71	-70	-
BSSID 3	-69	-68.33	-64.4
BSSID 4	-67	-64.5	-69
BSSID 5	-73.4	-73	-71
BSSID 6	-74	-73	-
BSSID 7	-	-	-72
BSSID 8	-	-	-
BSSID 9	-	-	-
BSSID 10	-	-	-

Table 5.7: D = 2000 ms with 3 RSS measurements.

	T = 2000	T = 5000	T = 10000	T = 15000	T = 20000
BSSID 1	2	7.58	0.168	0.62	0.107
BSSID 2	9.556	2.969	0.389	0.842	1.529
BSSID 3	-	6	-	0.691	0.809
BSSID 4	-	1.556	0.667	2.668	-
BSSID 5	-	-	-	-	0
BSSID 6	-	-	-	-	0.82
BSSID 7	-	-	0	-	0.007
Average	**5.778**	**4.5263**	**0.306**	**1.2052**	**0.5453**

Table 5.8: Variances of the received AP RSS values that did not have obmitted values with varying value of the total scanning time (T)

T = 10s and T = 20s is again the best choice, resulting in the most stable RSS values over the three measurements.

Note that a calculation of the fingerprint in 20s is not the best choice when the user is walking. A person's steady walking pace is around $0.7m/s$ [74], this would result in an update of the location every 14m at best. This is not ideal. The system should detect if the person is walking or not and based on that use 10s or 20s for walking and standing still modes respectively.

5.1.3 Effect of X, the amount of (maximum) values before averaging, on fingerprint

As mentioned in Chapter 3, fading and shadowing creates negative fluctuations in the RSS values due to walls, people and other structures in the indoor environment. This leads to multipath propagation. Therefore, it is important to average the RSS values over a certain period.

Table 5.21 shows the RSS values calculated by the system for just (a) one measurement and (b) a complete measurement with D = 1.5s and T = 20s. Note that the results are quite similar. This is due to the quiet environment during the short experiment. However, at the time of the measurement there was just regular movement and activity by other students in the lab. Therefore there are some notable differences in the results, which would certainly lead to erroneous results in the localization phase.

#	BSSID/SSID
BSSID 1	80:f6:2e:14:b5:90/ZJUWLAN
BSSID 2	80:f6:2e:14:b5:30/ZJUWLAN
BSSID 3	80:f6:2e:14:b4:d0/ZJUWLAN
BSSID 4	80:f6:2e:14:b5:80/ZJUWLAN
BSSID 5	80:f6:2e:14:b4:c0/ZJUWLAN
BSSID 6	80:f6:2e:14:b4:f0/ZJUWLAN
BSSID 7	80:f6:2e:14:b5:20/ZJUWLAN

Table 5.9: BSSID/SSID Legenda

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-61	-58	-58
BSSID 2	-73	-72	-66
BSSID 3	-	-71	-72
BSSID 4	-	-	-73
BSSID 5	-	-	-
BSSID 6	-	-	-
BSSID 7	-	-	-

Table 5.10: T = 2000 ms with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-61	-55	-55.33
BSSID 2	-71.67	-69	-67.5
BSSID 3	-68	-74	-71
BSSID 4	-70	-72	-73
BSSID 5	-	-73	-
BSSID 6	-	-72	-
BSSID 7	-	-	-

Table 5.11: T = 5000 ms with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-58	-57.43	-57
BSSID 2	-68	-67.5	-66.5
BSSID 3	-	-69.8	-
BSSID 4	-70	-71	-69
BSSID 5	-	-75	-75
BSSID 6	-	-74	-73
BSSID 7	-73	-73	-73

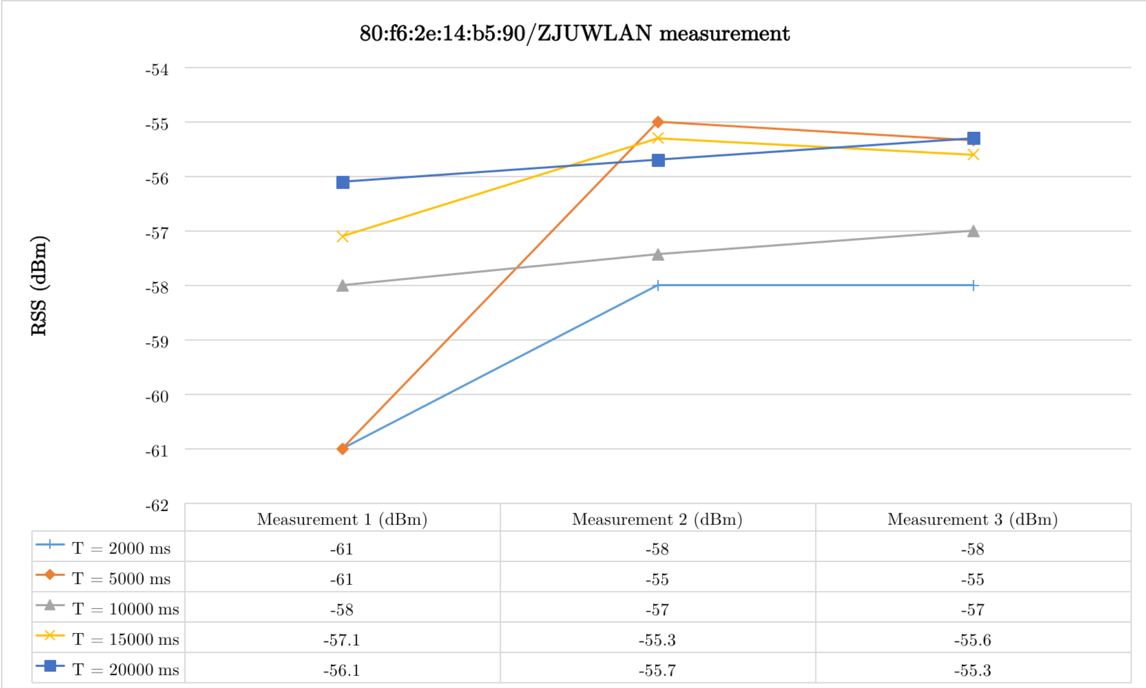
Table 5.12: T = 10000 ms with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-57.1	-55.3	-55.6
BSSID 2	-68	-68.9	-66.67
BSSID 3	-70	-68	-69.33
BSSID 4	-70.5	-70.43	-67
BSSID 5	-75	-75.67	-
BSSID 6	-	-	-70
BSSID 7	-73.25	-	-

Table 5.13: T = 15000 ms with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-56.1	-55.7	-55.3
BSSID 2	-71	-68.2	-68.6
BSSID 3	-72	-71	-69.8
BSSID 4	-70.25	-	-69.375
BSSID 5	-74	-74	-74
BSSID 6	-72	-69.9	-70.33
BSSID 7	-73.2	-73	-73.125

Table 5.14: T = 20000 ms with 3 RSS measurements.



	Measurement 1 (dBm)	Measurement 2 (dBm)	Measurement 3 (dBm)
T = 2000 ms	-61	-58	-58
T = 5000 ms	-61	-55	-55
T = 10000 ms	-58	-57	-57
T = 15000 ms	-57.1	-55.3	-55.6
T = 20000 ms	-56.1	-55.7	-55.3

Figure 5.1: RSS measurement for 80:f6:2e:14:b5:90/ZJUWLAN with T = 2s, T = 5s, T = 10s, T = 15s and T = 20s

	(a)	(b)
80:f6:2e:14:b5:80/ZJUWLAN	-73	-73.6
80:f6:2e:14:b5:30/ZJUWLAN	-67	-65
80:f6:2e:14:b5:20/ZJUWLAN	-73	-73
80:f6:2e:14:b5:90/ZJUWLAN	-67	-70
80:f6:2e:14:b4:d0/ZJUWLAN	-70	-70.8

Table 5.15: RSS value comparison between one measurement and measurement with D = 1.5s and T = 20s.

Hence parameter X is introduced in the system's design. Because T = 20s and D = 1.5s are set, the system only receives a maximum of 13 values as a RSS vector after the total scanning time of 20s. Table 5.17 to 5.20 depicts the fingerprints for various values for X, e.g.: X = 4, X = 7, X = 10 and X = 13. Just like in the previous sections, a conclusion is made after the calculation of the variances of the three-times received

#	BSSID/SSID
BSSID 1	00:21:27:36:e8:04/Simon@VLSI5
BSSID 2	bc:d1:77:c8:c3:e8/TP-LINK_142857
BSSID 3	bc:85:56:dc:f3:78/360_5M
BSSID 4	80:f6:2e:14:b5:90/ZJUWLAN
BSSID 5	80:f6:2e:14:b5:30/ZJUWLAN
BSSID 6	80:f6:2e:14:b5:80/ZJUWLAN
BSSID 7	80:f6:2e:14:b4:d0/ZJUWLAN
BSSID 8	80:f6:2e:14:b5:70/ZJUWLAN
BSSID 9	80:f6:2e:14:b4:c0/ZJUWLAN
BSSID 10	80:f6:2e:14:b5:20/ZJUWLAN
BSSID 11	80:f6:2e:14:b4:f0/ZJUWLAN

Table 5.16: BSSID/SSID Legenda

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-44.5	-44	-44
BSSID 2	-43.75	-44	-43.5
BSSID 3	-52.75	-50.25	-49.5
BSSID 4	-68.75	-63	-64.75
BSSID 5	-65.25	-64.25	-64.25
BSSID 6	-70.5	-70	-70.25
BSSID 7	-65	-67	-70
BSSID 8	-67.25	-68.5	-70.25
BSSID 9	-75	-	-
BSSID 10	-	-	-
BSSID 11	-	-	-

Table 5.17: X = 4 with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-44.71	-44.71	-43
BSSID 2	-44.57	-45	-45.43
BSSID 3	-50.57	-51.71	-50.29
BSSID 4	-63.86	-64	-68.29
BSSID 5	-65	-65	-62.43
BSSID 6	-70	-72	-75
BSSID 7	-68.5	-67.67	-67
BSSID 8	-68	-68	-68.71
BSSID 9	-75	-	-75
BSSID 10	-75	-	-
BSSID 11	-	-	-

Table 5.18: X = 7 with 3 RSS measurements.

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-43.5	-43.9	-43.8
BSSID 2	-48.3	-48.2	-48.5
BSSID 3	-52.2	-52.1	-51.6
BSSID 4	-66.6	-65.4	-67.3
BSSID 5	-64	-63.3	-65.6
BSSID 6	-71	-72.78	-70.5
BSSID 7	-68.78	-70.89	-67
BSSID 8	-69.2	-68.9	-69.6
BSSID 9	-74	-75.33	-
BSSID 10	-	-	-
BSSID 11	-	-	-

Table 5.19: X = 10 with 3 RSS

	1 (dBm)	2 (dBm)	3 (dBm)
BSSID 1	-43.31	-43.67	-44.23
BSSID 2	-49.23	-48.17	-48
BSSID 3	-53.46	-53	-54.08
BSSID 4	-71.38	-72.83	-72.46
BSSID 5	-71	-64.44	-65.38
BSSID 6	-69.83	-70	-68
BSSID 7	-66.9	-66.67	-66.14
BSSID 8	-68.73	-68.57	-69.58
BSSID 9	-74	-	-75.14
BSSID 10	-	-	-
BSSID 11	-	-73	-73.75

Table 5.20: X = 13 with 3 RSS

signals, as shown in Table 5.21. Resulting from this experiment, a good choice for X is 10 values. X = 10 results in less variantion over the same fingerprint location.

	X = 4	**X = 7**	**X = 10**	**X = 13**
BSSID 1	0.055	0.653	0.0289	0.144
BSSID 2	0.042	0.122	0.015	0.3
BSSID 3	1.93	0.381	0.069	0.195
BSSID 4	5.792	4.222	0.616	0.377
BSSID 5	0.222	1.469	0.927	8.377
BSSID 6	4.167	4.222	0.955	0.821
BSSID 7	4.222	0.376	2.527	0.100
BSSID 8	1.514	0.113	0.082	0.198
Average	**1.727**	**1.445**	**0.652**	**1.314**

Table 5.21: Variances of the equal received AP RSS values with varying value of the number of (maximum) values (X) before averaging.

5.2 Positioning Accuracy

5.2.1 Experiment Settings

In order to evaluate the performance of the proposed system, a test environment has been prepared on the fifth floor of the Laoshengyi (老生仪) building at Zhejiang University. Figure 5.2 shows the experimental site as a floorplan that is used for this experiment. 32 reference points have been chosen in an area of approx. $115m^2$. These points are chosen in accessible places for pedestrians. Tables 5.22 and 5.23 provides an overview of the location coordinates of the used reference and test points respectively. Following parameters are set during the experiment:

- T = 20s,
- D = 1.5s,
- X = 10.

At the time of the experiment, there was busy activity in the lab by other students. Six test points for localization have been randomly chosen in the

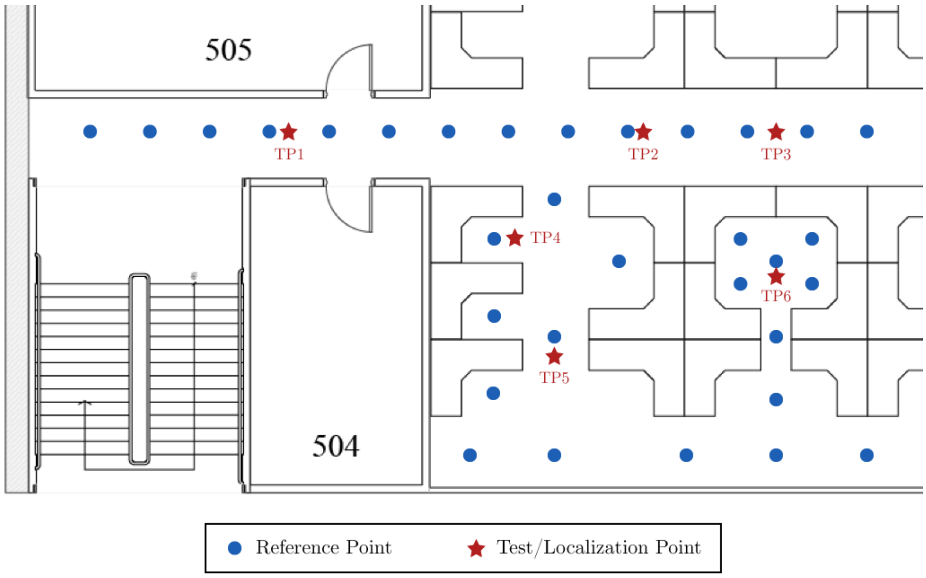


Figure 5.2: Overview of experimental site with marked reference and test points

experimental site, test point 6 is at the same location of a reference point. To build the fingerprint database, at every reference point a fingerprint has been calculated and saved in the database via the fingerprint server.

5.2.2 Results

At each test point the location is calculated every 30 seconds for five times. Every time the server responds with an estimated position, the positioning error, which is the distance between the true and estimated position, is noted. The positioning error ϵ is the distance between the real location coordinates (x_0, y_0) and the system's estimated location coordinates (x, y):

$$\epsilon = \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{5.2}$$

The results of this experiment with the average error in meters at each test point, are shown in Table 5.24.

The proposed system results in a remarkably good accuracy with approximately a 80 cm average positioning error. A good measure to show the accuracy of the proposed system is the cumulative distribution function (CDF) of the positioning errors obtained during the 30 measurements of the experiment. The CDF is shown in Figure 5.3. The horizontal axis is the positioning error in meters for the given probability function. The vertical axis is the probability in percent. So the CDF shows the probability for the positioning errors less than or equal to a specific error.

X(m)	Y(m)
13.14	6.81
5.98	6.83
13.97	6.83
16.70	6.83
5.30	6.86
12.11	6.88
16.57	6.88
8.98	6.94
12.21	6.94
15.79	6.94
7.87	6.96

X(m)	Y(m)
10.69	6.96
16.62	6.96
3.63	7.01
1.44	7.09
2.52	7.27
14.00	8.91
15.46	8.91
9.46	9.03
11.88	9.42
10.69	9.62
15.46	9.88

X(m)	Y(m)
14.02	10.01
9.41	10.47
10.42	10.90
14.78	10.93
15.64	12.08
9.26	12.80
10.62	12.95
13.01	13.21
16.59	13.26
15.64	13.38

Table 5.22: List of the location coordinates of the used reference points.

TP	X(m)	Y(m)
TP1	5.22	6.91
TP2	12.18	6.81
TP3	14.75	6.86
TP4	9.68	8.83
TP5	10.64	10.92
TP6	14.80	9.34

Table 5.23: List of the location coordinates of the used test points.

As can be seen from this figure, 95% of the estimated locations had a positioning error below 2m and 50% of the returned locations showed a positioning error below 1m. Note that for around $6,5\%$ of the test points a perfectly correct location is obtained from the positioning algorithm.

	Positioning Error (m)					Avg. Error (m)
Test Point 1	0.9	1.4	0.7	1.4	1	0.811
Test Point 2	0.8	0	2	2	1	0.831
Test Point 3	1.8	1	1.4	1	0.9	0.808
Test Point 4	0.9	1.6	1.2	1.1	2.3	1.221
Test Point 5	0.9	1	0.8	0.9	0.6	0.623
Test Point 6	0.7	1.1	0.5	1.3	0.8	0.708

Table 5.24: Positioning errors in meters for every 5 measurements and average error at each test point

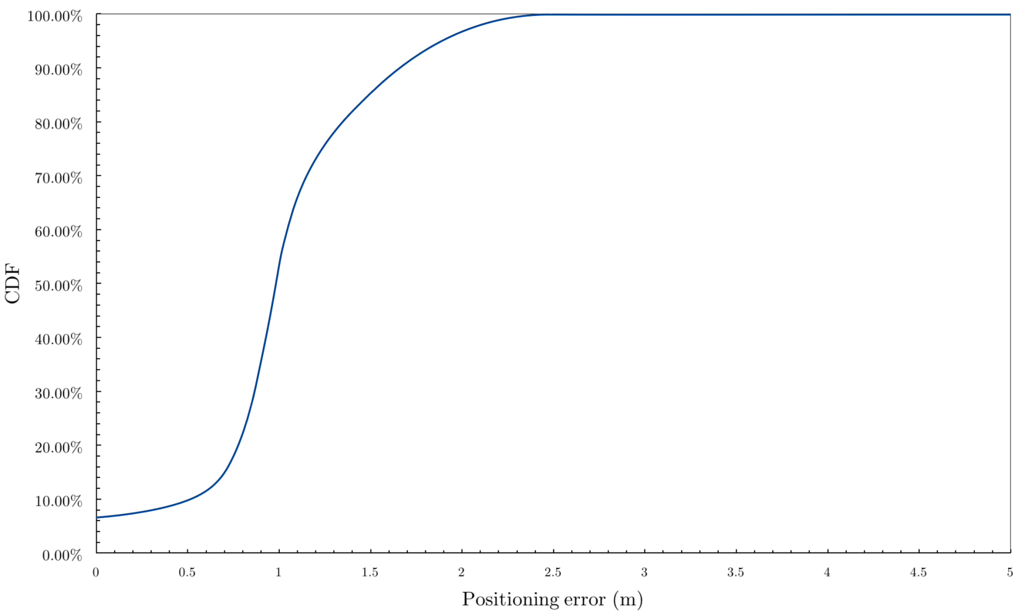


Figure 5.3: Cummulative Distribution Function of Error

The day after the experiment, the experiment was repeated for a second time to evaluate the robustness of the system, without making a new fingerprint database. There was no change in the experimental settings or the access point setup, only the activity of the neighboring students and arrangement of the student's desks are slightly different from the day before. However the system results in a higher average positioning error of approx. 1,2m, it is still a reasonable error to deal with in indoor positioning. This means that the proposed algorithm has a strong ability of robustness against minor changes in the environment and signal space.

	Positioning Error (m)					Avg. Error (m)
Test Point 1	1.3	1.2	1	2.8	2.7	1.285
Test Point 2	2.2	2.8	1.5	2.2	1	1.429
Test Point 3	0.5	4.2	1.5	1.5	0	1.288
Test Point 4	1.4	2.6	1.3	2.2	3	1.690
Test Point 5	0.7	1	1	1	1.2	0.742
Test Point 6	0.2	1	1.4	0.4	0.1	0.567

Table 5.25: Results of the later repeated experiment

5.3 Comparison of algorithms

It is very difficult to make a reliable comparison with other systems when these measurements are not done in exactly the same circumstances. Although based on published results from research papers, an approach of the accuracy comparison can be made. [75] shows a nice overview of the average error of the different localization techniques. Table 5.26 shows this overview, displaying the mean error for the classic KNN algorithm, Pearson Correlation Coefficient (PCC), a combination of KNN and PCC, a combination of KNN, PCC and Extented Kalman Filtering approach and the proposed WiFiFing positioning algorithm. Their experiment involves localization at 60 unknown locations.

Localization methods	KNN	PCC	KNN-PCC	KNN-PCC-EKF	WiFiFing
Mean error (m)	1.54	1.47	0.97	0.72	0.83

Table 5.26: Mean error comparison with different localization methods [75].

The proposed system certainly does not have a bad accuracy in comparison with other positioning methods and algorithms. It can certainly be compared regarding to the results of most state-of-the art Wi-Fi fingerprint-based positioning systems.

6 Conclusion

This chapter provides a summary of the implementation and the performance of the proposed Wi-Fi fingerprint-based indoor positioning system. Finally, a general conclusion is drawn about the research and further suggestions are provided for future adjustments and improvements.

6.1 Implementation and performance

This thesis explores the possibilities for indoor positioning based on a literature study. From this research, it has been clear that Wi-Fi fingerprinting is a technique that is used most frequently for this application. Despite the disadvantages of the unstable Wi-Fi signals, recent research does not differ from this method. The main causes of the unstable Wi-Fi signals are the body effect, fading, multipath, obstructions and interference. Nevertheless, recent research has succeeded in making highly accurate localization that is accurate enough for an indoor environment. Because no additional infrastructure investment is needed for Wi-Fi fingerprinting, the technology is extremely cost-effective, easy to apply and quick to deploy.

An important key element of a Wi-Fi fingerprint-based indoor positioning systems is the used positioning algorithm. This thesis implements the easiest-to-use machine learning algorithm for this, the WKNN algorithm. In addition, this algorithm has been modified for this thesis to achieve a more accurate result. The WKNN algorithm is based on the calculation of the Spearman's rank correlation coefficient for each fingerprint in the database. As a result, the ranking of the received signal strengths is also taken into account and not only the absolute received signal strength values itself. The proposed positioning algorithm also takes into account the number of signals being omitted when searching for the best matching fingerprints. This allows bad measurements to be filtered out of the results. Fingerprints with a large omitting value weigh less on the final calculation of the position. Then, the fingerprints are ranked on the Spearman's Distance, Euclidean Distance and the amount of omitted values. Based on this final ranking, the position is estimated by giving the K best fingerprints different weights according to the Euclidean distance between the reference fingerprint and this best ranked fingerprint from the database. Finally, the location is calculated based on these weights and sent back to the user.

By modifying the frequently-used KNN algorithm for the proposed positioning system, the positioning precision and accuracy is improved. Moreover, the proposed system achieves an exceptional accuracy with an average positioning error of approximately 80 cm using an up-to-date fingerprint database. In fact, 95% of the estimated locations resulted in a positioning error of less than 2m. Around 50% got a positioning error below or equal to 1m. As a result, the system is associated with the higher-scoring indoor positioning systems based on Wi-Fi fingerprinting proposed in other research. This all together means that the system is a good working prototype for Wi-Fi fingerprint-based indoor positioning systems.

6.2 Further improvements

In this section some suggestions are made to optimize the proposed system to obtain more accurate positioning estimations and make the overall speed of the system faster.

Although the accuracy of the system is already quite good, it is always better to do more research to obtain an even better accuracy. Research into other positioning algorithms with other machine learning algorithms is a direction that can be explored and examined. Although other machine learning algorithms will probably increase the complexity of the system, it might be a good way to achieve higher accuracy. Combining the proposed system with existing PDR systems, Bluetooth fingerprinting or other techniques, could also improve the accuracy of the system.

Currently, the proposed system has a major disadvantage of speed. The speed is largely imposed by the observations of the RSS values. In fact, measuring the nearest signals takes currently around 20s (T). As a result, the system is at this moment not real-time and is therefore not suitable to navigate while walking. Obtaining RSS fingerprints with a sliding-window technique could solve a part of this problem but will also not make the system real-time. Further research may thus include obtaining a smaller total scanning time and trying to make the system achieving real-time responses.

The thesis did not investigate whether the system is robust against major changes in the environment, e.g.: movement of large metal objects, influence by a lot of people with high activity, etc. The system was only tested in a lab environment and should also be tested in very large open spaces, e.g.: malls, library, etc. Future research should investigate the influence of the placement of the AP, the number of APs, the impact of changes in the environment, etc. on the accuracy. It is also usefull to investigate the impact of the RP density on the accuracy of the system. In addition, it can also be investigated to what extent the offline database changes at different times of the day/week.

Bibliography

[1] Z. Li, C. Liu, J. Gao, and X. Li, "An Improved WiFi / PDR Integrated System Using an Adaptive and Robust Filter for Indoor Localization," *ISPRS International Journal of Geo-Information*, vol. 5, no. 12, 2016.

[2] Q. Chang, S. Van de Velde, W. Wang, Q. Li, H. Hou, and H. Steendam, "Wi-Fi Fingerprint Positioning Updated by Pedestrian Dead Reckoning for Mobile Phone Indoor Localization," *Lecture Notes in Electrical Engineering*, vol. 3, no. 342, pp. 729–739, 2015.

[3] Xamarin Software, "Mobile App Development & App Creation Software - Xamarin," 2017. [Online]. Available: https://www.xamarin.com

[4] R. Want, A. Hopper, V. Falcão, and J. Gibbons, "The active badge location system," *ACM Transactions on Information Systems*, vol. 10, no. 1, pp. 91–102, 1992.

[5] J. Chung, M. Donahoe, C. Schmandt, I.-J. Kim, P. Razavai, and M. Wiseman, "Indoor location sensing using geo-magnetism," in *Proceedings of the 9th international conference on Mobile systems, applications, and services - MobiSys '11*, 2011, p. 141. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1999995.2000010

[6] H. Xie, T. Gu, X. Tao, H. Ye, and J. Lv, "MaLoc: A practical magnetic fingerprinting approach to indoor localization using smartphones," in *UbiComp '14 Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2014, pp. 243–253.

[7] V. Radu and M. K. Marina, “HiMLoc: Indoor smartphone localization via activity aware pedestrian dead reckoning with selective crowdsourced WiFi fingerprinting,” in *2013 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2013*, no. October, 2013, pp. 28–31.

[8] R. Smith, M. Self, and P. Cheeseman, “Estimating uncertain spatial relationships in robotics,” in *Proceedings. 1987 IEEE International Conference on Robotics and Automation*, vol. 4, 1990, pp. 850–850. [Online]. Available: http://ieeexplore.ieee.org/document/1087846/

[9] J. J. L. Durrant and H. F. Whyte, “Mobile robot localization by tracking geometric beacons.pdf,” in *IEEE/RSJ International Conference on Intelligent Robot Systems*, 1989, pp. 376–382.

[10] N. Karlsson, E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. E. Munich, “The vSLAM algorithm for robust localization and mapping,” in *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2005, no. April, 2005, pp. 24–29.

[11] U. Bandara, M. Hasegawa, M. Inoue, H. Morikawa, and T. Aoyama, “Design and implementation of a Bluetooth signal strength based location sensing system,” in *Proceedings. 2004 IEEE Radio and Wireless Conference (IEEE Cat. No.04TH8746)*, no. October, 2004, pp. 3–4.

[12] S. S. Chawathe, “Beacon Placement for Indoor Localization using Bluetooth,” in *Proceedings of the 11th International IEEE Conference on Intelligent Transportation Systems*, 2008, pp. 980–985.

[13] M.-s. Choi and B. Jang, “An Accurate Fingerprinting based Indoor Positioning Algorithm,” *International Journal of Applied Engineering Research*, vol. 12, no. 1, pp. 86–90, 2017.

[14] S. Yang, P. Dessai, M. Verma, and M. Gerla, “FreeLoc : Calibration-Free Crowdsourced Indoor Localization,” in *2013 Proceedings IEEE INFOCOM*, 2013, pp. 2481–2489.

[15] C. Wu, S. Member, and Z. Yang, “WILL : Wireless Indoor Localization without Site Survey,” in *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 4, 2013, pp. 839–848.

[16] J. Korhonen, T. Ojala, M. Klemola, and P. Väänänen, “mTag-Architecture for Discovering Location Specific Mobile Web Services Using RFID and Its Evaluation with Two Case Studies,” in *Advanced Int’l Conference on Telecommunications and Int’l Conference on Internet and Web Applications and Services (AICT-ICIW’06)*, 2006, pp. 191–200. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=1602324

[17] N. Alsindi and K. Pahlavan, “Cooperative localization bounds for indoor ultra-wideband wireless sensor networks,” *EURASIP Journal On Advances In Signal Processing*, vol. 2008, no. 852509, p. 13, 2008.

[18] N. A. Alsindi, B. Alavi, and K. Pahlavan, “Measurement and modeling of ultrawideband TOA-based ranging in indoor multipath environments,” *IEEE Transactions on Vehicular Technology*, vol. 58, no. 3, pp. 1046–1058, 2009.

[19] a. Mandal, C. Lopes, T. Givargis, A. Haghighat, R. Jurdak, and P. Baldi, “Beep: 3D indoor positioning using audible sound,” in *Second IEEE Consumer Communications and Networking Conference, 2005. CCNC. 2005*, 2005, pp. 348–353.

[20] K. Liu, X. Liu, and X. Li, “Guoguo: Enabling Fine-Grained Smartphone Localization via Acoustic Anchors,” *IEEE Transactions on Mobile Computing*, vol. 15, no. 5, pp. 1144–1156, 2016.

[21] DraMCo, “Nauwkeurigheid van dynamische RSS-based indoor plaatsbepalingssystemen,” pp. 1–11, 2008. [Online]. Available: https://iiw.kuleuven.be/onderzoek/dramco/research/ladi-witepa/Technieken.pdf

[22] M. B. Nabil Ali Alrajeh and B. Shams2, “Localization Techniques in Wireless Sensor Networks,” *International Journal of Distributed Sensor Networks*, vol. 2013, pp. 1–9, 2013.

[23] M. Sauter, “3.7.1 Mobility Management in the Cell-DCH State,” in *From GSM to LTE: An Introduction to Mobile Networks and Mobile Broadband*, 2010, p. 160.

[24] Cisco, “Location Tracking Approaches,” in *Wi-Fi Location-Based Services 4.1 Design Guide*, 2014.

[25] S. Yiu, M. Dashti, H. Claussen, and F. Perez-Cruz, “Wireless RSSI fingerprinting localization,” *Signal Processing*, vol. 2017, no. 131, pp. 235–244, 2017. [Online]. Available: http://dx.doi.org/10.1016/j.sigpro.2016.07.005

[26] K. Kaemarungsi and P. Krishnamurthy, “Properties of Indoor Received Signal Strength for WLAN Location Fingerprinting,” in *Proceedings of the First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous’04)*, 2004. [Online]. Available: https://pdfs.semanticscholar.org/c87f/ca49f8e3dacbcd5ab11261508aa79a03ca8a.pdf

[27] Y. Chapre, A. Ignjatovic, A. Seneviratne, and S. Jha, “CSI-MIMO : Indoor Wi-Fi fingerprinting system,” in *39th Annual IEEE Conference on Local Computer Networks*, no. January 2016, 2014, pp. 202–209.

[28] H. Wang, A. Elgohary, and R. R. Choudhury, “No Need to War-Drive : Unsupervised Indoor Localization,” in *Proceedings of the 10th international conference on Mobile systems, applications, and services (MobiSys ’12)*, 2012, pp. 197–210.

[29] P. Bahl, V. Padmanabhan, and A. Balachandran, “RADAR: An in-building RF based user location and tracking system,” in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, vol. 2, no. c, 2000, pp. 775–784. [Online]. Available: http://research.microsoft.com/en-us/groups/sn-res/infocom2000.pdf

[30] B. Li, J. Salter, A. Dempster, and C. Rizos, “Indoor positioning techniques based on wireless LAN,” in *Proceedings of First IEEE International Conference Wireless Broadband and Ultra Wideband Communication*, 2006. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.72.1265

[31] K. D’hoe, G. Ottoy, P. Keersebilck, J. P. Goemaere, and L. De Strycker, “Indoor room location estimation,” *Advances in Electrical and Computer Engineering*, vol. 8, no. 2, pp. 78–81, 2008.

[32] M. Shchekotov, “Indoor Localization Method Based on Wi-Fi Trilateration Technique,” in *Proceeding of the 16Th Conference of Fruct Association*, 2014, pp. 177–179. [Online]. Available: https://fruct.org/publications/abstract16/files/Shc1.pdf

[33] Z. Farid, R. Nordin, and M. Ismail, “Recent Advances in Wireless Indoor Localization Techniques and System,” *Journal of Computer Networks and Communications*, vol. 2013, p. 12, 2013.

[34] E. C. C. E. w. t. E. C. o. P. (CEPT) and T. Administrations, “The European table of frequency allocations and applications in the frequency range 8.3 kHz to 3000 GHz (ECA Table),” 2016. [Online]. Available: http://www.erodocdb.dk/docs/doc98/official/pdf/ERCRep025.pdf

[35] W. Wong, J. Ng, and W. Yeung, “Wireless LAN Positioning with Mobile Devices in a Library Environment,” in *25th IEEE International Conference on Distributed Computing Systems Workshops*, 2005, pp. 633–636. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1437236

[36] W. Xue, W. Qiu, X. Hua, and K. Yu, “Improved Wi-Fi RSSI Measurement for Indoor Localization,” *IEEE Sensors Journal*, vol. 17, no. 7, pp. 1–1, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7835628/

[37] V. Moghtadaiee and A. G. Dempster, “Design protocol and performance analysis of indoor fingerprinting positioning systems,” *Physical Communication*, vol. 13, no. PA, pp. 17–30, 2014. [Online]. Available: http://dx.doi.org/10.1016/j.phycom.2014.02.004

[38] P. Bahl, V. Padmanabhan, and A. Balachandran, “A software system for locating mobile users: Design, evaluation, and lessons,” 2000. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.43.3784{&}rep=rep1{&}type=pdf

[39] K. Chintalapudi, “Indoor Localization Without the Pain,” in *MobiCom ’10, Proceedings of the sixteenth annual international conference on Mobile computing and networking*, 2010, pp. 173–184. [Online]. Available: http://research.microsoft.com/pubs/135721/ez-mobicom.pdf

[40] B. Zhou, Q. Li, Q. Mao, and W. Tu, “A Robust Crowdsourcing-Based Indoor Localization System,” *Sensors*, vol. 17, no. 4, p. 864, 2017. [Online]. Available: http://www.mdpi.com/1424-8220/17/4/864

[41] S. He, S. Member, S. G. Chan, and S. Member, “Wi-Fi Fingerprint-Based Indoor Positioning : Recent Advances and Comparisons,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 466–490, 2016.

[42] C. C. Lo, L. Y. Hsu, and Y. C. Tseng, “Adaptive radio maps for pattern-matching localization via inter-beacon co-calibration,” *Pervasive and Mobile Computing*, vol. 8, no. 2, pp. 282–291, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.pmcj.2012.01.001

[43] M. M. Atia, A. Noureldin, and M. J. Korenberg, “Dynamic online-calibrated radio maps for indoor positioning in wireless local area networks,” *IEEE Transactions on Mobile Computing*, vol. 12, no. 9, pp. 1774–1787, 2013.

[44] S. Pan, J. Kwok, Q. Yang, and J. Pan, “Adaptive Localization in a Dynamic WiFi Environment through Multi-view Learning.” in *National conference on artificial Intelligence*, vol. 1, 2007, pp. 1108–1113. [Online]. Available: http://www.aaai.org/Papers/AAAI/2007/AAAI07-176.pdf

[45] R. Hansen, R. Wind, C. S. Jensen, B. Thomsen, S. L. Vej, and D.-A. Ø, “Algorithmic Strategies for Adapting 802.11 Location Fingerprinting to Environmental Changes,” in *2010 International Conference on Indoor Positioning and Indoor Navigation*, no. September, 2009, pp. 15–17.

[46] X. Wang, L. Gao, S. Mao, and S. Pandey, “CSI-based Fingerprinting for Indoor Localization: A Deep Learning Approach,” *IEEE Transactions on Vehicular Technology*, vol. 66, no. 1, pp. 763–776, 2017. [Online]. Available: http://arxiv.org/abs/1603.07080

[47] C. Wu, Z. Yang, Z. Zhou, K. Qian, Y. Liu, and M. Liu, “PhaseU: Real-time LOS identification with WiFi,” in *Proceedings - IEEE INFOCOM*, vol. 26, no. 978, 2015, pp. 2038–2046.

[48] X. Zhang, Z. Yang, C. Wu, W. Sun, Y. Liu, and K. Xing, “Robust trajectory estimation for crowdsourcing-based mobile applications,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 7, pp. 1876–1885, 2014.

[49] X. Cao, Y. Chen, and K. J. R. Liu, “Time-Reversal Indoor Positioning With Centimeter Accuracy Using Multi-Antenna Wifi,” *Icassp 2016*, vol. 1, no. November, pp. 6220–6224, 2016.

[50] P. Bolliger, "Redpin - Adaptive, Zero-Con guration Indoor Localization through User Collaboration," in *Melt'08*, 2008, pp. 55–60.

[51] J.-g. Park, B. Charrow, D. Curtis, J. Battat, E. Minkov, J. Hicks, S. Teller, and J. Ledlie, "Growing an organic indoor location system," in *Proceedings of the 8th international conference on Mobile systems, applications, and services - MobiSys '10*, 2010, p. 271. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1814433.1814461

[52] B. Ferris, D. Fox, and N. Lawrence, "WiFi-SLAM Using Gaussian Process Latent Variable Models." in *Intl. Joint Conf. on Artificial Intelligence (IJCAI)*, 2007. [Online]. Available: http://www.aaai.org/Papers/IJCAI/2007/IJCAI07-399.pdf

[53] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee : Zero-Effort Crowdsourcing for Indoor Localization Categories and Subject Descriptors," in *MobiCom'12*, 2012.

[54] I. Aizenberg, *Complex-Valued Neural Networks with Multi-Valued Neurons*, 1st ed. Springer, 2011.

[55] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2005.

[56] C. Laoudias, D. Zeinalipour-Yazti, and C. G. Panayiotou, "Crowdsourced indoor localization for diverse devices through radiomap fusion," in *2013 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2013*, no. July, 2013, pp. 28–31.

[57] Y. Xie, Y. Wang, A. Nallanathan, and L. Wang, "An Improved K-Nearest-Neighbor Indoor Localization Method Based on Spearman Distance," *IEEE Signal Processing Letters*, vol. 23, no. 3, pp. 351–355, 2016.

[58] D. Madigan, E. Einahrawy, R. P. Martin, W.-H. Ju, P. Krishnan, and A. S. Krishnakumar, "Bayesian indoor positioning systems," in *In Proceedings of the IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 2, no. C, 2005, pp. 1217–1227. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs{_}all.jsp?arnumber=1498348{%}5Cnhttp://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.83.8028{&}rep=

rep1{&}type=pdf{%}5Cnhttp://ieeexplore.ieee.org/xpl/login.jsp?reload=true{&}tp={&}arnumber=1498348{&}url=http://ieeexplore.ieee.org/

[59] N. Patwari, A. O. Hero, M. Perkins, N. S. Correal, and R. J. O'Dea, "Relative location estimation in wireless sensor networks," *IEEE Trans. Signal Proc.*, vol. 51, no. 8, pp. 2137–2148, 2003.

[60] B. Shin, J. H. Lee, T. Lee, and H. S. Kim, "Enhanced Weighted K-Nearest Neighbor Algorithm for Indoor Wi-Fi Positioning Systems," in *In Proceedings of the 2012 8th International Conference on Computing Technology and Information Management (ICCM)*, Seoul, 2012, pp. 574–577.

[61] Z. Guowei, X. Zhan, and L. Dan, "Research and I mprovement on Indoor Localization B ased on RSSI Fingerprint Database and K-Nearest Neighbor Points," in *In Proceedings of the 2013 International Conference on Communications, Circuits and Systems (ICCCAS)*, no. 2, Chengdu, 2013, pp. 68–71.

[62] M. Nowicki and J. Wietrzykowski, "Low-effort place recognition with WiFi fingerprints using deep learning," *Advances in Intelligent Systems and Computing*, vol. 550, no. November, pp. 575–584, 2017.

[63] S.-h. Fang and T.-n. Lin, "Indoor Location System Based on Discriminant-Adaptive Neural Network in IEEE 802 . 11 Environments," *Control*, vol. 19, no. 11, pp. 1973–1978, 2008.

[64] A. Kushki, K. N. Plataniotis, and A. N. Venetsanopoulos, "Kernel-Based Positioning in Wireless Local Area Networks," *Mobile Computing, IEEE Transactions on*, vol. 6, no. 6, pp. 689–705, 2007.

[65] Z. Wu, J. Liu, and B. Liu, "Particle Filter and Support Vector Machine Based Indoor Localization System," p. 1, 2016. [Online]. Available: https://www.microsoft.com/en-us/research/wp-content/uploads/2015/10/zhengwu.pdf

[66] A. S. Abdou, "An Efficient Indoor Localization System based on Affinity Propagation and Support Vector Regression," in *In Proceedings of the 2016 Sixth International Conference on Digital Information Processing and Communications (ICDIPC)*, Beirut, 2016, pp. 1–7.

[67] H. M. Joe Armstrong, Bjarne Däcker, Thomas Lindgren, "Open-source Erlang - White Paper," 2011.

[68] H. Mattsson, H. Nilsson, and C. Wikstrom, "Mnesia - A Distributed Robust DBMS for Telecommunications Applications," in *PADL '99: Proceedings of the First International Workshop on Practical Aspects of Declarative Languages*, 1998, pp. 152–163. [Online]. Available: http://www.erlang.se/publications/mnesia{_}overview.pdf{%}5Cn/home/steven/Document/jabref/DC/Mattsson1998.pdf

[69] T. Berners-Lee, R. Fielding, and H. Frystyk, "Hypertext Transfer Protocol – HTTP/1.0," 1996. [Online]. Available: http://www.rfc-editor.org/info/rfc1945

[70] A. Lehman, *Jmp For Basic Univariate And Multivariate Statistics: A Step-by-step Guide.* NC: SAS Press, 2005.

[71] E. Deza and M. M. Deza, *Encyclopedia of Distances.* Springer, 2009.

[72] S. Thirumuruganathan, "A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm," 2017. [Online]. Available: https://saravananthirumuruganathan.wordpress.com/2010/05/17/a-detailed-introduction-to-k-nearest-neighbor-knn-algorithm/

[73] Android, "WifiManager," 2017. [Online]. Available: https://developer.android.com/reference/android/net/wifi/WifiManager.html

[74] J. E. BERTRAM and A. RUINA, "Multiple Walking Speed – frequency Relations are Predicted by Constrained Optimization," *Journal of Theoretical Biology*, vol. 209, no. 4, pp. 445–453, 2001. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0022519301922799

[75] J. Yang, Y. Li, and W. Cheng, "An improved neighbor-correlation-extended-Kalman-filter fusion method for indoor navigation," *International Journal of Distributed Sensor Networks*, vol. 13, no. 5, p. 155014771771165, 2017. [Online]. Available: http://journals.sagepub.com/doi/10.1177/1550147717711651

Auteursrechtelijke overeenkomst

Ik/wij verlenen het wereldwijde auteursrecht voor de ingediende eindverhandeling:
Study of Wi-Fi Fingerprint-Based Indoor Positioning on a smartphone

Richting: **master in de industriële wetenschappen: elektronica-ICT**
Jaar: **2017**

in alle mogelijke mediaformaten, - bestaande en in de toekomst te ontwikkelen - , aan de Universiteit Hasselt.

Niet tegenstaand deze toekenning van het auteursrecht aan de Universiteit Hasselt behoud ik als auteur het recht om de eindverhandeling, - in zijn geheel of gedeeltelijk -, vrij te reproduceren, (her)publiceren of distribueren zonder de toelating te moeten verkrijgen van de Universiteit Hasselt.

Ik bevestig dat de eindverhandeling mijn origineel werk is, en dat ik het recht heb om de rechten te verlenen die in deze overeenkomst worden beschreven. Ik verklaar tevens dat de eindverhandeling, naar mijn weten, het auteursrecht van anderen niet overtreedt.

Ik verklaar tevens dat ik voor het materiaal in de eindverhandeling dat beschermd wordt door het auteursrecht, de nodige toelatingen heb verkregen zodat ik deze ook aan de Universiteit Hasselt kan overdragen en dat dit duidelijk in de tekst en inhoud van de eindverhandeling werd genotificeerd.

Universiteit Hasselt zal mij als auteur(s) van de eindverhandeling identificeren en zal geen wijzigingen aanbrengen aan de eindverhandeling, uitgezonderd deze toegelaten door deze overeenkomst.

Voor akkoord,

Vandenabeele, Thomas

Datum: **22/06/2017**