Modeling and Performance Analysis of Hybrid Localization Using Inertial Sensor, RFID and Wi-Fi Signal

by

Guanxiong Liu

A Thesis

Submitted to the Faculty

of the

WORCESTER POLYTECHNIC INSTITUTE

In partial fulfillment of the requirements for the

Degree of Master of Science

in

Electrical and Computer Engineering

by

May 2015

APPROVED:

Professor Kaveh Pahlavan, Major Thesis Advisor

Professor Yehia Massoud, Head of Department

Abstract

The development in wireless technology, mobile smart devices and Internet of Things has gave birth to a booming era for the wireless indoor geolocation. This technology have been increasingly used within our daily life and help people to build up the tracking system which could be used by fulfillment centers and grocery stores. To achieve higher localization accuracy with wireless geolocation, we need a higher density of deployment which involves high deployment and maintenance cost. To balance the accuracy and the cost, people have begun using wireless localization employing inertial navigation system (INS) which provide speed and direction of movement. When we combine Radio Frequency (RF) localization with INS, we have a hybrid INS/RF localization system which can achieve high localization accuracy with low cost.

In this thesis, we use accelerometers and magnetometers in an Android smart phone to build a hybrid INS/RF system and use two different technologies for RF localization: Radio Frequency Identification Device (RFID) and Wi-Fi. Using this system, we conducted measurements of the hybrid localization system and evaluate its performance. The specific contributions of the thesis are: (1)Empirical performance evaluation of the INS/RFID localization system. It relates the localization error to the number and position of RFID tags. (2)Model the effect of metallic objects on accuracy of magnetometer. The model shows the relation between direction error and distance to metallic component. (3)Model shadow fading in close proximity of RF transmitter. It builds a distance dependent shadow fading model. (4)Model based performance evaluation of hybrid localization. The test bench uses our models to simulate the hybrid localization data.

Acknowledgements

In this thesis I describe the research I conducted in pursuit of my Master of Science Degree in Electrical & Computer Engineering in Worcester Polytechnic Institute.

Firstly, I would like to offer my sincerest gratitude to my research advisor and life mentor, Professor Kaveh Pahlavan, for leading me into the academic research and teaching me his life experience. Professor Pahlavan is a kindly advisor and intelligent mentor with a deep understanding in not only research but also many other aspects of the life. It is my greatest honor to have him as my advisor.

Secondly, I want to thank Yishuang Geng for his help during my master level of study. All my research in CWINS lab is also under his guidance. He offered a lot of help when I meet some technical problems. The cooperation with him gives me a valuable experience and will benefit myself in the future.

I am really grateful that Dr. Allen H. Levesque, Prof. Emmanuel O. Agu and Prof. Andrew Clark are willing to be my committee members. Thank you for the valuable comments and reviewing of my thesis. I would also want to thank Dr. Dan Liu (DLOU) and Dr. Shihong Duan (USTB) for their help, guidance and teaching during my master level study.

Moreover, I want to thank all the members in CWINS lab, Mingda Zhou, Luyao Niu and Yingyue Fan. Thank these people so much for offering me a nice atmosphere in the lab like a big family.

Finally, I would like to dedicate my thesis to my beloved parents and grandparents, who offer me selfless support and infinity love.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

In modern life, an accurate indoor geolocation technology is becoming a necessary and novel emerging component to commercial, public safety and military applications. It can help people solving different kinds of problems under variety of situations [1]. Combined with the LANs, it can help people to build up the tracking system which could be used by medical centers, fulfillment centers and grocery stores. Also, there are some routing problems and social network applications that use the indoor geolocation technology through the WANs to achieve indoor navigation in the museums, airports and other large public areas [2] [3].

In the indoor environment, it's almost impossible to receive signal from the global positioning system (GPS). Even if a successful reception of GPS signal occurs, the several meters average localization error in the GPS result is too high to be used by the indoor localization applications. Besides the inapplicable GPS technology, the Time-of-Arrival (TOA), Phase-of-Arrival (POA) and RSS based approaches are more often used in the indoor localization area [4] [5]. Among these technologies,

TOA and POA could offer a high localization accuracy in optimal situation (LOS and lower environment complexity) [6] [7] [8]. However, they are not so robust in the real world since the complicated indoor environment with heavy multipath can easily result in a huge DME when these technologies are being used. Compared with the TOA and POA technology, the RSS based approach doesn't suffer from the multipath and is becoming the most common choice in implementation of the indoor localization application [9]. However the RSS based technology is highly depend on the reference database, it always requires a large and up-to-date database which cost huge maintenance consumption [10]. Therefore, the RSS base indoor localization technology also meets the bottleneck when people wants to build commercial application on top of it.

To overcome the shortage of the RF technologies used in the indoor geolocation and conduct a robust solution that could be wildly used in the commercial application, the hybrid INS/RF system is becoming popular these days [11]. The development of the mobile smart device allows people to integrate different varieties of embedded sensors inside it. Therefore, we could easily use some of these sensors to build a traditional inertial navigation system and combine them with the existing wireless localization technologies [12]. Finally, the hybrid system could achieve a higher localization accuracy with a lower resource consumption. Also, the system takes the advantage of every standalone technology components to become an applicable solution for commercial using.

1.2 Motivation

With the appearance of hybrid INS/RF system, people start to build commercial application on top of it and get some fairly acceptable localization results. However,

to further the algorithm design and application optimization, the detailed analysis in hybrid INS/RF system is necessary. In previous works, people always over simplify the performance analysis of hybrid INS/RF system and overlook the error discussion. Actually, the performance of the hybrid INS/RF system is highly depend on the deployment of reference points (RPs). Therefore, a performance model of the system could be helpful to optimize the algorithm design. Besides that, the error model of the INS and RF part is also necessary. By using these models, people could come up with a simulation test bench that will be helpful in the future study in this area.

1.3 Contribution

In this thesis, there are three major sections and the contribution could be described into three main parts as below:

1. Empirical performance evaluation of the INS/RFID localization system
2. Model the effect of metallic objects on accuracy of magnetometer
3. Model shadow fading in close proximity of RF transmitter

As mentioned before, we use an Android smart phone to build the hybrid INS/RF system and choose RFID as the RF part technology. With the help of this system, we measured the DME in a typical indoor environment and analyzed sources of inaccuracy in hybrid INS/RF system. To quantitatively describe the error, we introduce a statistical model for hybrid INS/RF system DME based on the measurement results. The model separates the impact factor into number of RFID and deployment of RFID. The deployment of RFID is further divided into the tag density and tag to path-corner distance.

When we model the error of the hybrid INS/RF system, we separate it into two parts (INS and RF). For the inertial navigation part, the main parameter is the direction estimation θ and the distance estimation d. Based on our measurement, the direction estimation has an unstable performance and could cause large error in the result. Therefore, we conduct a series of measurements with the direction estimation under different scenarios and come up with a mathematical model to describe it. This model gives the direction estimation error based on the distance between the mobile device and typical large metal component in indoor environment (door and elevator). Since the property of RSS based localization technology, the error source of the estimation is the fluctuation of the RSS value. Therefore, based on the Friis transmission equation, we analysis the shadow fading model in the indoor environment [13]. In this thesis, we conduct the measurement with a VNA and build up a distance dependent shadow fading model which related the shadow fading and the distance between the transmitter and receiver in LOS situation.

1.4 Thesis Outline

The remainder of the thesis is organized as follow: Chapter 2 gives the basic background and development introduction about indoor geolocation technologies. Chapter 3 presents the analysis on the performance of hybrid INS/RF system with a mathematical model. In this chapter, we use the RFID calibration tags as the RF technology. Chapter 4 introduces the inertial navigation system we used. The introduction includes the user interface of the inertial navigation tool, step detection algorithm and direction estimation. Chapter 5 discusses error model of the embedded magnetometer. The error model provides a distance related error for the embedded magnetometer. Chapter 6 shows the measurement with the VNA and

build a distance dependent shadow fading model. Last but not the least, Chapter 7 presents the conclusion of this thesis and discussion of the future work.

Chapter 2

Background of Indoor Geolocation

2.1 Introduction

Recent advancements in mobile smart devices have enabled the building of indoor geolocation system not only on top of the tablet but also upon the smartphone or robot. The variety of sensors inside these smart devices make them a good implementation platform to achieve the geolocation oriented cyber physical system. With the development of Internet of Things, especially the idea of the smart city, the indoor geolocation technology is becoming a essential and major component to commercial business, public safety and even military applications. Recently, the Kiva Systems is using the indoor geolocation technology on top of their warehouse labor robot. Their robot can be located by a centralized system in real time. Therefore, the system can achieve an automatically control on a large number of robots and build the unmanned package transmission center on top of it. Also, there are other companies like Skyhook which use the indoor geolocation technology to build a indoor navigation application which is a useful supplementation to the existing map application which is unavailable in the indoor environment.

2.2 Signal Based Localization

In indoor geolocation area, the TOA and POA are popular choices [14]. The general idea in these technologies is utilizing the speed of the signal and the propagation time to estimate the distance between the transmitter and receiver [15]. TOA is using the first peak in the signal to estimate the propagation time. And POA is using the phase of the receiving signal to determine the propagation time. However, in the real world scenario, these technologies usually suffers from the significant error under non-line-of-sight (NLOS) situation. Under this situation, the directed propagation signal could be easily weaken or even blocked. Therefore, these technologies, which are highly depending on the directed path, usually lose accuracy.

2.3 RSS Based Localization

At the same time of using signal base localization, some people choose to use RSS based localization technology. Instead of highly dependent on the properties of the receiving signal, RSS based localization simply make the use of signal strength [16]. And the general idea here could be separated into two group. One of them is using the relation between the propagation distance and the receiving signal strength to locate the target in a mesh network with multiple signal source. The other one is using the pre-collected RSS database and the statistical model to estimate the location based on the real time RSS observation. Since the wide availability of the wireless signal from large number of Access Points (APs) in modern indoor environment, using RSS based technology to solve the indoor geolocation problems is applicable. The RSS based technology is relatively robust in NLOS scenario since RSS doesn't significantly changed when directed path is blocked. However, it always requires a large database which will incur large maintenance expenditure.

2.4 Hybrid INS/RF Localization

Instead of using standalone RF localization technology to solving indoor geolocation problem, the hybrid INS/RF system is becoming popular these days for its robust and low cost. The development of the mobile smart device allow people to integrate different embedded sensors inside it. Therefore, we could easily use some of these sensors to build an inertial navigation system and combine them with the existing wireless localization technologies. The system takes the advantages of every standalone technology component to achieve a higher localization accuracy with a lower resource consumption. This hybrid INS/RF localization technology can be implemented in different platforms while robots and smartphones are two popular choices [17].

As the technology develops, robots have the ability to take more and more tasks such as labor work, public service or even disaster assistance. To achieve these goals, localization and mapping are the core technologies. Either of these two missions could be easily solved if they are separated. However, dealing with both of them at the same time is a difficult problem. Such problem blocked the advantage of location based services for a long while until Wi-Fi simultaneous localization and mapping (Wi-Fi SLAM) was proposed. With Wi-Fi SLAM, the robot should be able to move in indoor environment and trace itself through the hybrid INS/RF system.

Upon smartphone, hybrid INS/RF system are always used to replace the RSS based localization application. The embedded accelerometer and magnetometer can build the inertial navigation system for the user. With the development of the RF technologies, not only Wi-Fi but also some Bluetooth based RFID could be used to achieve RF localization part. The hybrid system allows the smartphone to use indoor navigation inside the building where RF signal isn't fully covered.

Chapter 3

Smart Phone based Inertial Navigation System

3.1 Introduction

With the development of the Micro-Electro-Mechanical System (MEMS), people can now integrate different varieties of embedded sensors inside the mobile smart devices that we use every day [18]. Therefore, we could easily use the embedded magnetometers, accelerometers and gyroscopes to build the traditional inertial navigation system on a smart phone [19] [20] [21]. This kind of system is widely used to mitigate the inaccuracy in the standalone wireless localization results and to improve the reliability in the absence of RF Signal. One of the well-known application is the Wi-Fi localization assisted inertial navigation system, which is notable for its low cost, high availability and extraordinary accuracy. In this chapter, we are going to introduce a self-created inertial navigation system based on an Android smart phone.

The introduction of this system will include three main parts. The first part is

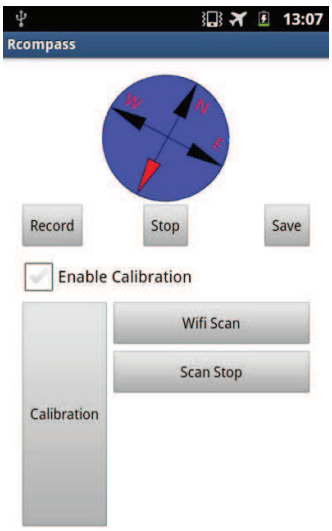

Figure 3.1: Smart Phone based Inertial Navigation System.

the user interface and function of the system. After that, we will show the detail of the step detection algorithm which is based on an open source code from the Git. The last section will be used to show details of the implementation of the direction estimation inside the system.

3.2 User Interface and Function Design

The application user interface that is showed in the Fig. 3.1 is our smart phone based inertial navigation system. The entire system is a self-designed android application on Samsung Exhibit II SGH-T679 smart phone, which runs on any android system above version 2.2 and is embedded with accelerometer and magnetometer for step detection and direction estimation respectively.

This system can be divided into two major functional components. In first component, we build a traditional inertial navigation system through using the embedded accelerometer and magnetometer. The accelerometer is used to build the

step detection algorithm which could detect the user's step after a training process. And the magnetometer could return the direction of the user at the moment that a step is detected through the algorithm. After clicking on the start button, the step detection algorithm will automatically monitor the accelerometer data. Once a step is detected by the algorithm, the system time stamp will be recorded inside the memory along with the magnetometer data. When user want to stop the recording, he just have to click the stop button. However, the data inside the data will not be saved until the save button is clicked. This is designed in case of some situations that user don't want to save one of his measurement.

The second part of the functional parts of the system is used for the RF simulation or data collection. This part is controlled by the Enable Calibration Checkbox. If it's not checked, the system will work as a pure inertial navigation system. When user check this checkbox, the system will sensing and store extra RF part data based on the user's choice. The larger button on the left with Calibration on it is used for the simulation of the RFID calibration tags. Once the user clicked this button, a calibration mark will be created and stored with the nearest preceding detected step. Therefore, we could rebuild the process of go across a RFID calibration tags. The other buttons on the right side is used for the Wi-Fi data collection. If it's used during the walking process, the smart phone will repeat sampling of the Wi-Fi data at the time of each step is detected. When we use it alone, it could help us to build up a RSS database for reference point.

3.3 Step Detection Algorithm

A major part of our smart phone based inertial navigation system is the step detection algorithm. The implementation of this algorithm is based on an open sourced

code uploaded by Paul Smith in Git (https://github.com/COMSYS/FootPath.git) [22]. In his implementation, it provides a StepDetection.java which accesses the accelerometer and the compass to detect a step. It use the highest sampling rate to collect the sensor data and then a low pass filter is conducted to filter out the major movements inside the data.

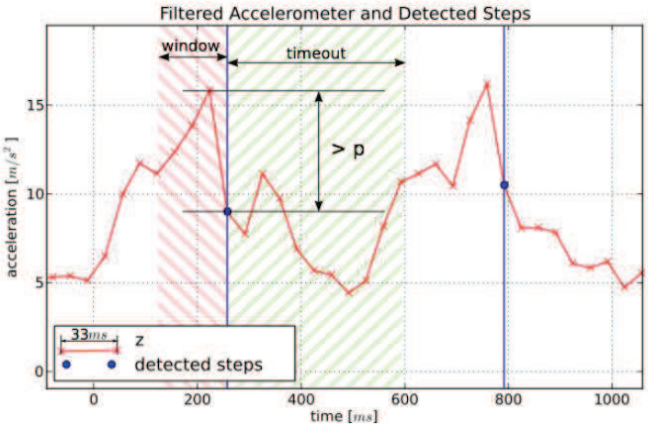

Figure 3.2: Sensor Data and Algorithm Design.

The step can be detected through the characteristic of change in acceleration. When the user is walking with the device holding in their hands, the vertical movement of human body will transfer to the smart phone. Previous work shows that the scale of movement depends on the user and the holding position. As it shows in the Fig. 3.2, based on the standard holding position of Android smart phone, our step detection algorithm use $-0.7ms^2$ within 5 samples as the vertical axis acceleration threshold to detect the step. After a successful detection of the step, the algorithm will apply a 400ms timeout to prevent multi-detection of the same step. The algorithm diagram is showed in the Fig. 3.3

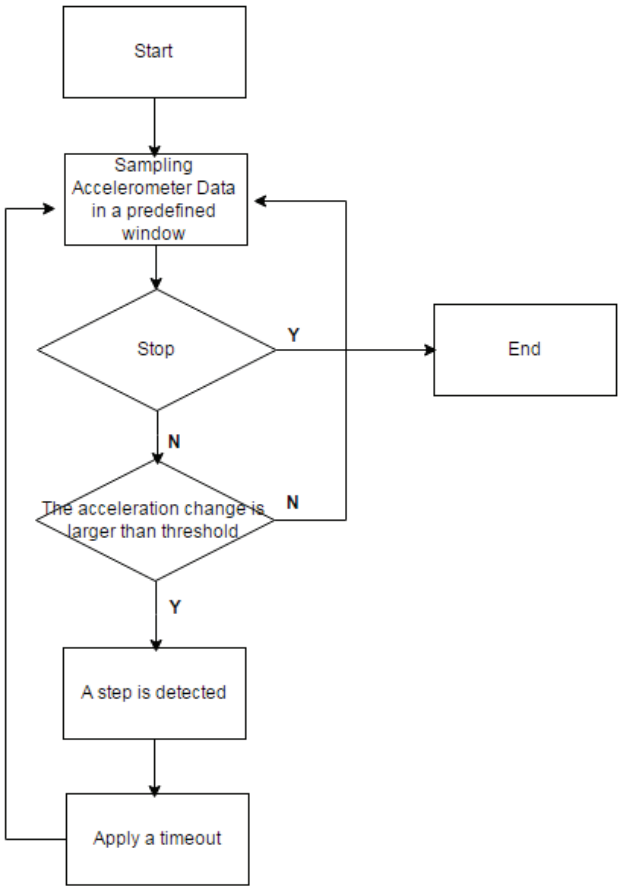

Figure 3.3: Step Detection Algorithm Diagram.

3.4 Direction Estimation

In this section, we introduce the direction estimation of our smartphone base hybrid localization system. The content includes the defination of world frame and body frame and the direction measurement.

3.4.1 World Frame

To introduce the direction estimation in our inertial navigation system, we should introduce the axis definition inside the Android system. According to the documentation from the Android, the coordinates used to describe the position on the

physical world is called world frame and its definition is showed as blow: [23]

1. X_w is defined as the vector product $Y_w.Z_w$ (It is tangential to the ground at the device's current location and roughly points East)

2. Y_w is tangential to the ground at the device's current location and points towards the magnetic North Pole

3. Z_w points towards the sky and is perpendicular to the ground

where w is used to denote the axis is used in the world frame, Fig. 3.4 also shows the world frame definition.

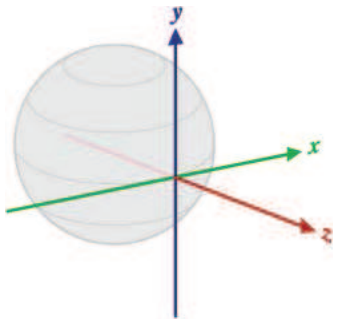

Figure 3.4: World Frame Defination in Android.

3.4.2 Body Frame

Besides the world frame which is used to describe the world, there is also another set of definition called body frame. Since in using the holding position of smart phone is different, it's hard and even impossible to use the world frame in some application, especially when it's related to the embedded sensor data. Therefore, the body frame is defined by the phone itself and used to model some embedded sensor's data which includes accelerometer, magnetometer and etc. The original definition is showed in blow: [24]

1. x_b: horizontal axis from the left to the right to the device

2. y_b: vertical axis from the bottom to the top of the device

3. z_b: axis from the backside to the front side of the device

where b is used to denote the axis is used in the body frame, Fig. 3.5 also shows the world frame definition.

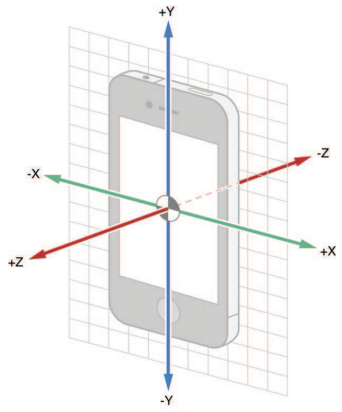

Figure 3.5: Body Frame Defination in Android.

3.4.3 Direction Measurement

In our implementation of the direction estimation, the data is collected from the embedded magnetometer. Based on the definition inside the Android documentation, the magnetometer will return a vector of orientation data which is defined as below: [25]

1. Azimuth (degrees of rotation around the z axis). This is the angle between magnetic north and the device's y axis. For example, if the device's y axis is aligned with magnetic north this value is 0, and if the device's y axis is pointing south this value is 180. Likewise, when the y axis is pointing east this value is 90 and when it is pointing west this value is 270.

2. Pitch (degrees of rotation around the x axis). This value is positive when the positive z axis rotates toward the positive y axis, and it is negative when the positive z axis rotates toward the negative y axis. The range of values is 180 degrees to -180 degrees.

3. Roll (degrees of rotation around the y axis). This value is positive when the positive z axis rotates toward the positive x axis, and it is negative when the positive z axis rotates toward the negative x axis. The range of values is 90 degrees to -90 degrees.

In our design, the user should holding the smart phone with its z_b axis points to the sky. Therefore, the Azimuth returned by the magnetometer could represent the angle between magnetic north and user's moving direction. With the pre-measured north direction, we could use the transfer equation to calculate the moving direction of the user.

Chapter 4

Modeling the Performance of INS/RF Hybrid Navigation System

4.1 Introduction

The hybrid localization system applications nowadays not only mitigate the inaccuracy of standalone RF localization approach, but also increase the reliability in the absence of supporting RF infrastructure [26]. One of the outstanding hybrid approaches is the RFID assisted inertial navigation system, which is notable for its low cost, simple implementation and extraordinary accuracy [27]. Previous work on such hybrid systems failed to find the correlation between the deployment of the multiple calibration points and the indoor localization accuracy. In this chapter, we use the Android smart phone to build a hybrid localization platform and conduct measurements with multiple RFID calibration tags. Based on the measurement results, we define a mathematical model which includes the calibration point number, RFID

tag density and the RFID tag-to-corner distance to describe the deployment effect on the localization accuracy. This model facilitates the future study of algorithm design, system evaluation and application development.

4.1.1 Background

Nowadays, the rapid advancement of wireless access and localization technology not only provides high data rate wireless communication, but also helps to complete the precise indoor localization. With the well-known RSS, TOA and POA based approaches, location information can be obtained even in the indoor area that GPS could never cover [28] [29] [30]. Since wireless indoor localization (1)requires the knowledge of reference points and (2)suffers from the multipath phenomenon caused by complicated indoor environments, the traditional pedestrian dead reckoning method maintains its priority in the absence of supporting infrastructure. To further improve the localization accuracy for critical cases such as localizing patients, prisoners and first responders, current research has proposed cooperative localization applications to combine multiple approaches [31].

4.1.2 Related Work

Among all available hybrid localization technologies, RFID assisted approach is wildly noticed for its low energy consumption, low cost characteristic and simple implementation. The trend of using RFID tags in cooperative localization has been maximized as the concept of smart building spreads [32]. Ni et al. proposed 'LANDMARK', the first prototype system using active RFID technology to obtain location information in indoor environment [33]. More recently, Ruiz et al. reported a pedestrian navigation system using tightly coupled foot-mounted IMU and RFID ranging to achieve accurate indoor localization [34]. These applications utilize RFID tag

as transceivers and require a ranging process. Other research studies regard the RFID tag as a calibration point with small enough coverage to serve as a landmark for pass-by objects. These studies have used the RFID-assisted localization system for first responders, self-calibrated RFID tags in smart factories and even location estimation system for construction materials [35] [36] [37]. Note that in this work, we focus on the second category of RFID assisted indoor localization and employ the RFID tag for calibration purpose only.

Current literature on RFID-assisted localization system generally focused on system implementation, algorithm optimization and robustness under the multipath phenomenon. However, many of these researchers did not look into the influence of RFID tag number or geometrical deployment of the tags. Ruiz et al. graphically illustrated the effect of tag numbers but failed to quantitatively model it; Hameed et al. tried to sparsely distribute the tags in the smart factory and evaluated multiple deployment topology but did not report any mathematical model regarding tag location. Although it is intuitive that avoiding linearly-aligned, close-range calibration tags may help to get a better localization result [38], an optimized way to deploy tags is still essential and urgently demanded for both academia and industry.

4.1.3 Our Work

In this chapter, the performance of RFID assisted inertial navigation system has been measured in a typical office environment with different numbers of calibration RFID tags and various combinations of tag locations. The effect of calibration tag number and tag locations have been investigated with empirical data. In order to quantitatively describe these effects, a mathematical model for average localization error has been built using tag number, tag-to-corner distance and tag density as parameters. The proposed model can be uniformly used in cases when given the

floor layout, which may benefit future tag deployment optimization and system performance evaluation at a large scale.

The remainder of this chapter is organized as follows. In section 4.2, the measurement scenario and measurement system have been introduced and necessary definitions have been provided for further analysis. In section 4.3, the effect of tag deployment has been analyzed from the perspective of calibration tag number, tag-to-corner distance and tag density. In section 4.4, regression fitting has been applied to empirical data and a mathematical model has been built to illustrate the influence of tag deployment on localization accuracy. In section 4.5, we summarize this chapter and discuss future work.

4.2 Scenario and System Setup

In this section, the measurement scenario, system setup as well as necessary parameter definitions have been discussed. We conducted on-line measurement for pedestrians dead reckoning (PDR) indoor navigation in a typical office floor and the assistance of RFID calibration tags is achieved by off-line software simulation. Since this work focus on small range calibration tags with directional antenna, we assume that the RFID calibration has a 0.1m accuracy so that the calibration process can be simplified as moving the PDR particle with state $(\hat{x}_i, \hat{y}_i, \hat{\theta}_i)$ back to a random location within 0.1m radius to the calibration tag location. For PDR measurements, $\hat{x}_i$, $\hat{y}_i$ denotes to the coordinate estimate for i^{th} particle and $\hat{\theta}_i$ denotes to the heading.

4.2.1 Measurement Scenario

On-line measurement for PDR navigation is performed at the 3rd floor of the Atwater Kent Laboratory, the office building of ECE department, Worcester Polytechnic

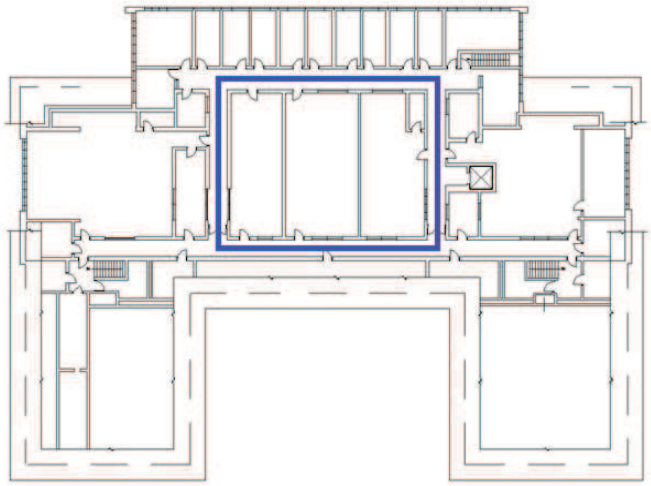

Figure 4.1: 2D floor layout for Atwater Kent building, WPI.

Institute, Worcester, MA, USA. As shown in Fig. 4.1, there is a rectangular path along the main corridor of $27.6 \times 20.4m$. The objective first and foremost goes through a training process to obtain the average step length l and then walks along the main corridor on a constant speed, holding the smart phone in hand. The measurement starts from the bottom left corner of the path and lasts for three entire cycles. Note that constant walking model is not a limitation on this work, preliminary results shows that following discussion still applied to random walking situation and it will be mentioned in future publications.

4.2.2 System Setup

The entire measurement process is performed using a self-designed android application on Samsung Exhibit II SGH-T679 smart phone, which runs on any android system above version 2.2 and is embedded with accelerometer and compass for step detection and heading measurement respectively. Every time a step is detected, a PDR particle is recorded in the format of $(\hat{x}_i, \hat{y}_i, \hat{\theta}_i)$ based on coordinate and heading

measurement on previous step. The update process of particles can be given as

$$\hat{x}_{i+\delta i} = \hat{x}_i + (l + n_l)\cos(\delta\hat{\theta} + n_\theta) \tag{4.1}$$

$$\hat{y}_{i+\delta i} = \hat{y}_i + (l + n_l)\sin(\delta\hat{\theta} + n_\theta) \tag{4.2}$$

$$\hat{\theta}_{i+\delta i} = \hat{\theta}_i + \delta\hat{\theta} + n_\theta \tag{4.3}$$

where δi is the step counts from the last known particle, $\delta\hat{\theta}$ denotes the heading change, n_l and n_θ are noise terms drawn from the step length and heading uncertainty models respectively. The iterative process indicates that the beginning state is required for this approach. Note that due to the metallic structure of the building and dense RF devices in some of the rooms, the heading measurement could be erroneous. Inaccuracy on steps length is also possible since the training performance is limited to some scale. To represent the above mentioned uncertainty, the localization error for specific PDR particle is defined as

$$\epsilon_i(d) = \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2} \tag{4.4}$$

where x_i, y_i are the ground true coordinate for i^{th} PDR particle.

4.2.3 Parameter Definitions

Apart from the state information of PDR particles, calibration tag locations is also recorded during our experiments. RFID tags are by nature attached to the walls and very frequently appears close to corridor corners for the sake of convenience and aesthetics. Also, the ultimate goal is to track objectives in the public area instead of private room. We consider RFID tags deployed along the path in the main corridor. The candidate set for calibration point number in this work can

be given by $m \in \mathcal{M} = \{1, 2, ..., M\}$ where m denotes the index of m^{th} RFID calibration tag and $M = 8$ in this work. Since dramatic heading change can be detected at the corners along the path, tags attached to the corner may intuitively limited the inaccuracy caused by heading measurement. To represent how far are the calibration tags located from the corners, we define the tag-to-corner distance d_{corner} as the average of distances between each tag and its nearest corner on the floor layout, which can be given as

$$d_{\text{corner}} = \frac{1}{M} \sum_{m \in \mathcal{M}} [\min_{j \in \mathcal{J}} (\sqrt{(x_m - x_j)^2 + (y_m - y_j)^2})] \tag{4.5}$$

where x_m, y_m are the coordinate for m^{th} RFID calibration tag and x_j, y_j are the coordinate for j^{th} corner. In this specific measurement scenario, candidate set for corner index is $j \in \mathcal{J} = \{1, 2, 3, 4\}$. Moreover, to represent the influence of homogeneous density of tag deployment, we define tag density d_{dense} as the average of distance between adjacent tags, which is given by

$$d_{\text{dense}} = \frac{1}{M - 1} [\mathcal{P} - \max_{m \in \mathcal{M}} (\gamma_{m, m+1_{\text{Mod}(M)}})] \tag{4.6}$$

where $\mathcal{P}$ is the total length of the target trajectory, which is along the rectangular path in this work, $\gamma_{m,m+1_{\text{Mod}(M)}}$ denotes the distance between adjacent tags along trajectory and is given as

$$\gamma_{m,m+1_{\text{Mod}(M)}} = \int_{C_{m,m+1}} f ds = \int_{\text{tag}_m}^{\text{tag}_{m+1_{\text{Mod}(M)}}} f(r(t)) |r'(t)| dt \tag{4.7}$$

where $C_{m,m+1}$ is the trajectory between m^{th} and $m + 1^{th}$ calibration tag, f is the scaler field and is constant 1 for 2D case, $\text{tag}_m = (x_m, y_m)$ denotes the coordinate for m^{th} calibration tag, $r(t)$ represents the sample point for line integral. Since we

have to take the $\gamma_{M,1}$ into consideration in the max(•) operation in (6), Mod(M) calculation has been included. Typical test case has been illustrate in Fig. 4.2, in which the tag number and distance between adjacent tags has been specified. Note that the density employed in this chapter is a linear density instead of spatial density due to the fact that the ultimate goal is to track objectives in the public area instead of private room. It is worth mentioning that with the prior knowledge of floor layout, the line integral can be easily simplified into summation of distance along the trajectory, reducing the computational complexity to the minimum.

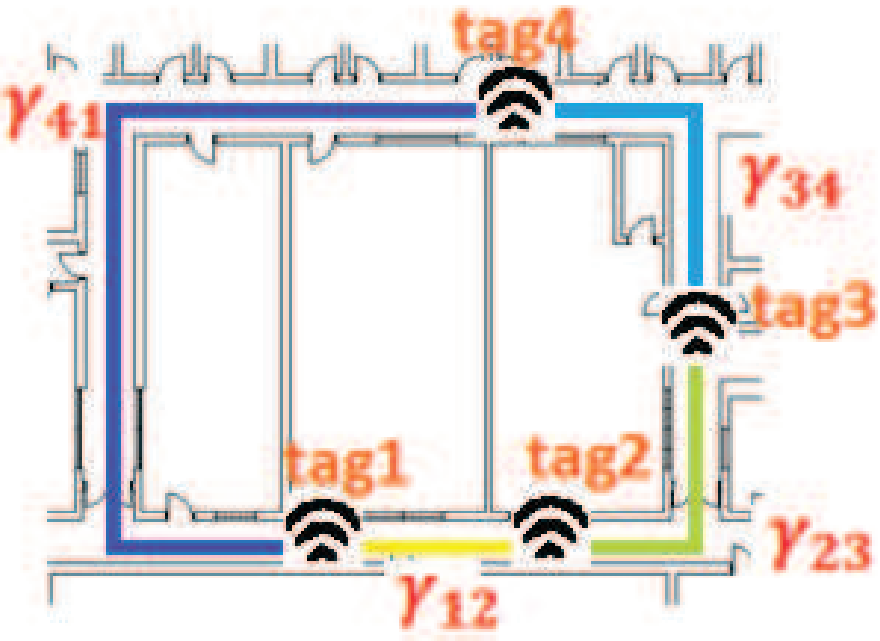

Figure 4.2: Sample calibration tag deployment when $n = 4$.

Based on the pre-defined parameters, with given calibration tag deployment, the localization error for specific target position can be defined as

$$\epsilon = \sqrt{(\hat{x} - x)^2 + (\hat{y} - y)^2} \tag{4.8}$$

where $\hat{x}$, $\hat{y}$ denote to the location estimate given specific calibration tag number and tag deployment. Throughout the off-line software simulation for various tag locations, we include all possible combination of n, d_{corner} and d_{dense}.

4.3 Effect of Tag Number, Tag-to-Corner Distance and Tag Density

General localization error for the RFID assisted inertial navigation system has been plotted in Fig. 4.3 to Fig. 4.7. In Fig. 4.3, with no calibration, it is very obvious that localization result keeps drifting away from the ground truth. Such drifting phenomenon agree with the pure PDR system performance bound reported in previous works. When RFID calibration tag gets involved, localization performance gets improved to a large degree and the drifted result is pulled back to ground truth even though the uncertainty still exists. Conclusion can be drawn from Fig. 4.4 to Fig. 4.7 is that with the increment of calibration point number, localization accuracy becomes better. To better illustrate the statistics of the empirical result, cumulative density function of $\bar{\epsilon}$ for localization accuracy with different number of calibration tags has been depicted in Fig. 4.8, from which we can see that for a given floor layout, a sufficient number of calibration tag can be implied. If the tag number is more than adequate, the extra tags have very limited contribution on the performance improvement. In this work, sufficient number would be 4 tags. When using 8 tags to calibrate the PDR system, performance improvement is highly limited.

As for the influence of tag-to-corner distance, the experimental result shows that RFID calibration tags in the corner contribute more on the improvement of localization performance. Statistic for the effect of tag-to-corner distance has been illustrated in Fig. 4.9, in which the CDF for $\bar{\epsilon}$ has been plotted with four calibration

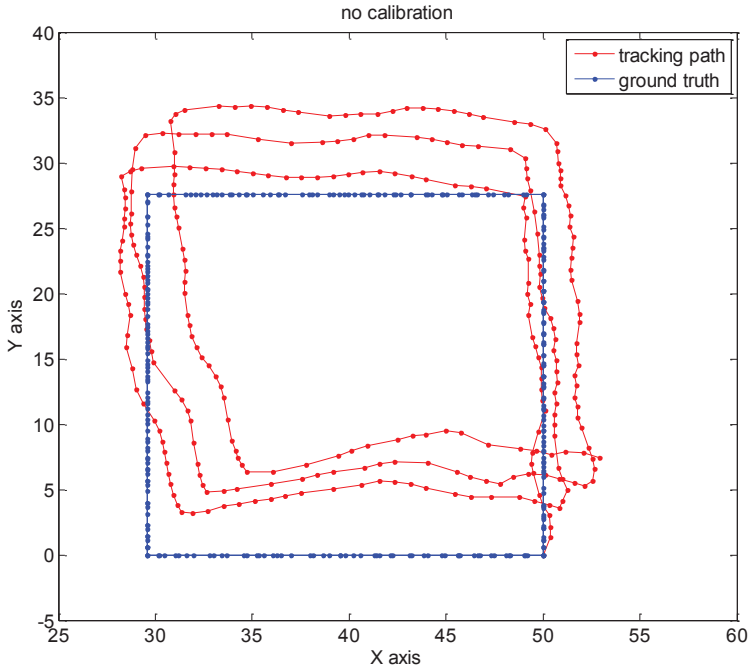

Figure 4.3: Navigation result of pure inertial navigation without RFID calibration tags.

tags, same tag density but various tag-to-corner distance. Identical tag density can be guaranteed by initiate the four calibration tags at the four corner and step by step move the tags along same direction. As long as the step size is constant, tag density will not change. Statistics show that with the increment of d_{corner}, localization error becomes larger, indicating that it is preferred to deploy calibration tags as close to corners as possible.

The effects of tag density cannot be ignored since in most of the previous work of INS with multiple calibration, people always tried to figure out a way to uniformly deploy the calibration points. Four tag calibration case has been again plotted in Fig. 4.10 with fixed tag-to-corner distance and various tag density. It shows that with uniformly deployed calibration tags, best localization performance can be achieved. The integrated analysis on the effect of tag number and tag locations propose a tough

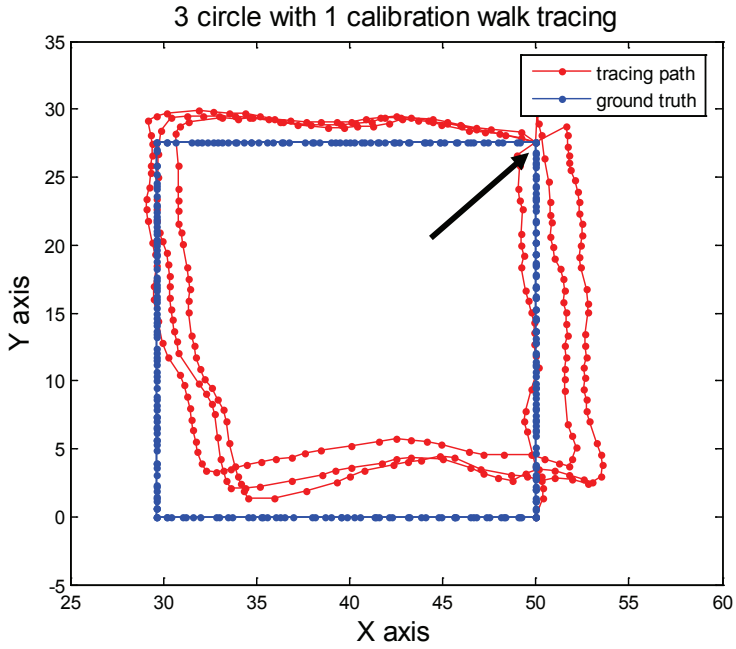

Figure 4.4: Navigation result of inertial navigation with one RFID tag on top right corner.

question on the optimized deployment, that is, how can we manage the trade-off among tag number, tag-to-corner distance and tag density. To answer that question, we proposed a mathematical model to support deployment optimization.

4.4 Modeling the Effect of Tag Deployment

To quantitatively describe the trade-off among n, d_{corner} and d_{dense}, mathematical model on localization error has been proposed in this section. The empirical data shows that localization error $\epsilon_i(d)$ can be modeled as a uniformly distributed random variable and as long as we have at least one calibration point, the variance of $\epsilon_i(d)$ is limited in a small range. Based on that observation, we focus on the average localization error $\bar{\epsilon}$ and do not take the variance into consideration. By further

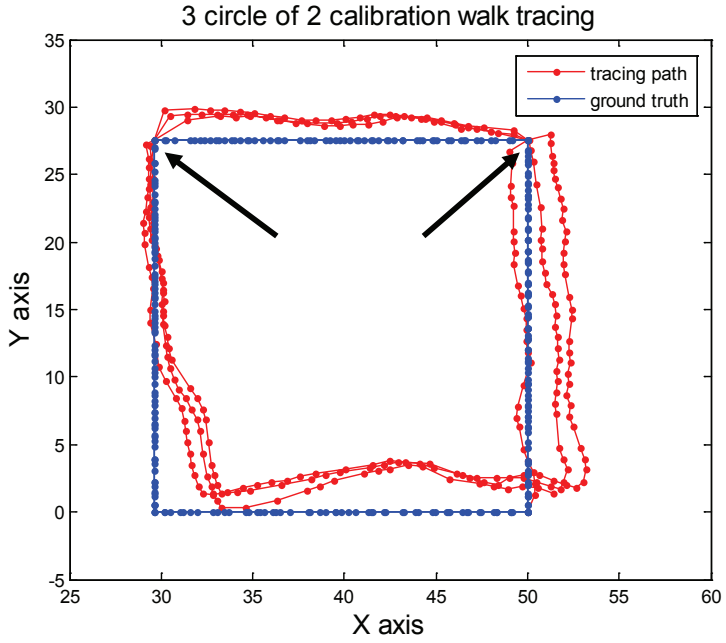

Figure 4.5: Navigation result of inertial navigation with two RFID tags on top left and top right corners.

examining the empirical measurement results, we notice that for a given n and d_{corner}, linear relationship between $\bar{\epsilon}$ can be obtained as

$$\bar{\epsilon} = A_{n,d_{\text{corner}}} \times d_{\text{dense}} + B_{n,d_{\text{corner}}} \tag{4.9}$$

where $A_{n,d_{\text{corner}}}$ and $B_{n,d_{\text{corner}}}$ are intermediate coefficients depends on calibration number n and tag-to-corner distance d_{corner}. Case specific fitting result for $\bar{\epsilon}$ has been plotted in Fig. 4.11 in which four calibration tags are employed. Exploiting $A_{n,d_{\text{corner}}}$ and $B_{n,d_{\text{corner}}}$ for different calibration tag number, we can further model the relationship between intermediate coefficients and tag-to-corner distance as

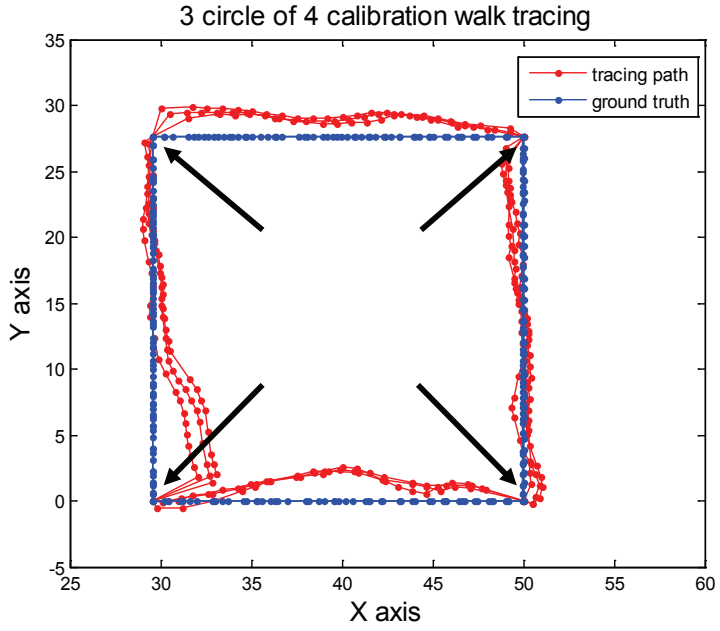

Figure 4.6: Navigation result of inertial navigation with four RFID tags on each corner.

$$\begin{cases} A_{n,d_{\text{corner}}} = C_n \times d_{\text{corner}} + D_n \\ B_{n,d_{\text{corner}}} = B_{d_{\text{corner}}} = E \times d_{\text{corner}} + F \end{cases} \tag{4.10}$$

where C_n and D_n are coefficients depends on tag number n, E and F are constant coefficients. We could further fit C_n and D_n by n, however, since the tag number could be integers only and massive curve fitting influences the accuracy of error model, Table. I has been provided for C_n and D_n regarding different tag number n. Fitting result for $A_{n,d_{\text{corner}}}$ and $B_{n,d_{\text{corner}}}$ have been depicted in Fig. 4.12 and Fig. 4.13 respectively and the final model for averaging localization error can be given as

$$\bar{\epsilon} = (C_n \times d_{\text{corner}} + D_n) \times d_{\text{dense}} + E \times d_{\text{corner}} + F$$

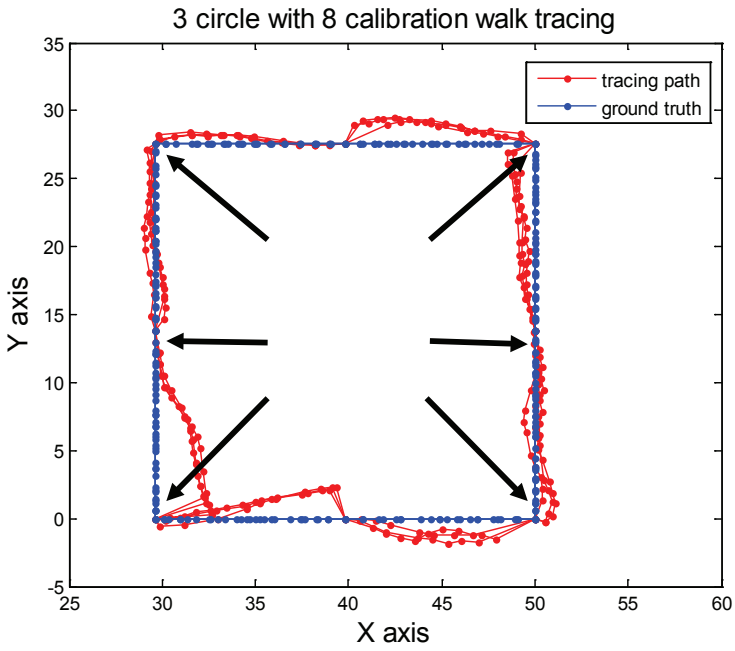

Figure 4.7: Navigation result of inertial navigation with eight RFID tags on each corner and middle of path.

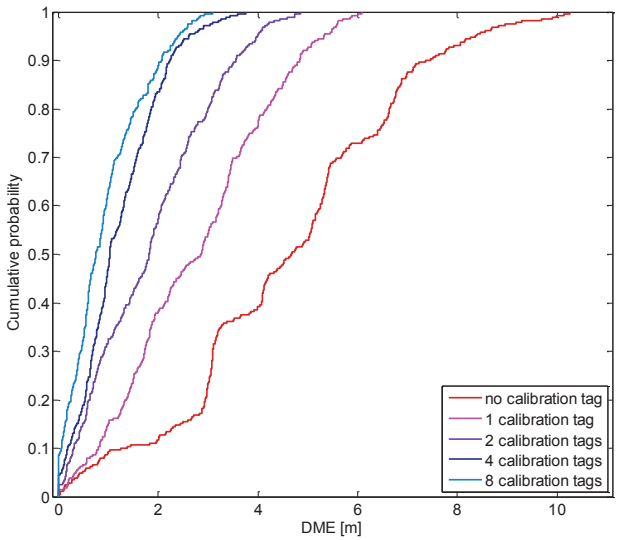

Figure 4.8: CDF plot for the effect of tag number.

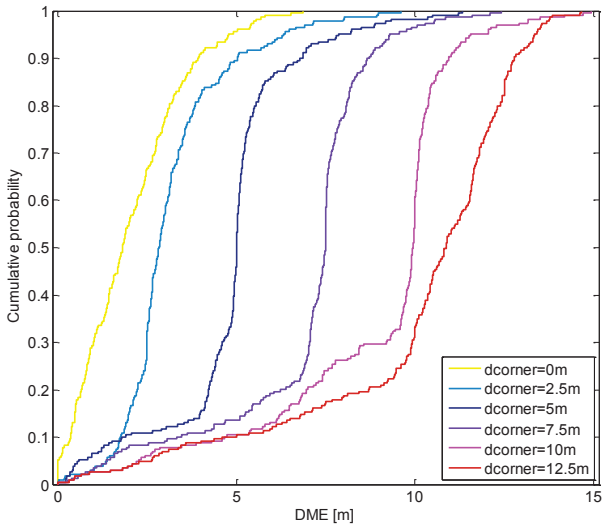

Figure 4.9: CDF plot for the effect of calibration tag-to-corner distance.

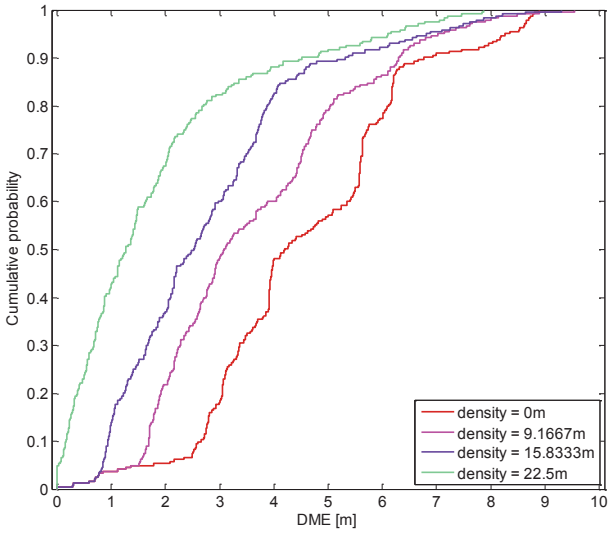

Figure 4.10: CDF plot for the effect of calibration tag density.

$$= C_n \times d_{\text{corner}} \times d_{\text{dense}} + D_n \times d_{\text{dense}} + E \times d_{\text{corner}} + F \tag{4.11}$$

where C_n, D_n, E and F can be found in Table. 5.3.

To further validate the proposed average localization error model, we select typical measurement cases and compare it with the software simulated localization error

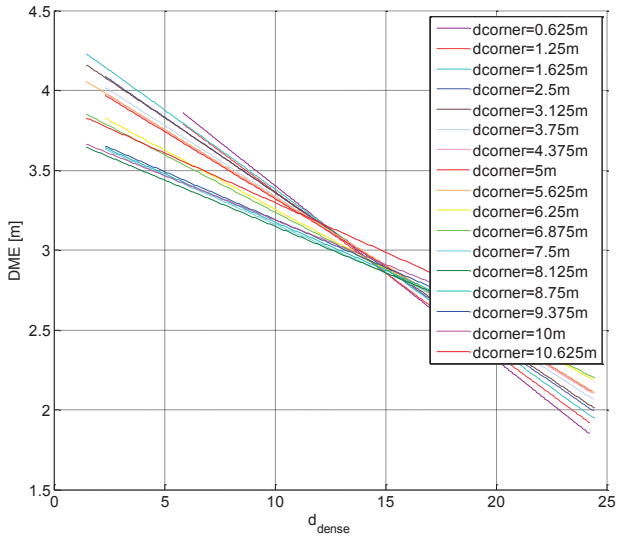

Figure 4.11: Regression fitting result (localization error VS tag density parameter).

model. As shown in these figures, given $n = 4$, the empirical measurement result in Fig. 4.14 has a highly agreement with simulated error model in Fig. 4.15, which further prove the validity of the proposed average localization error model.

4.5 Conclusion and Future Work

The major contribution of this chapter is that we analyzed the effect of metal component and the distance to these components on magnetometer's direction estimation error for smart-phone based inertial navigation system. A simple error model has been proposed for indoor inertial navigation performance analysis and it also facilitates the future work on algorithm design, system integration and application development.

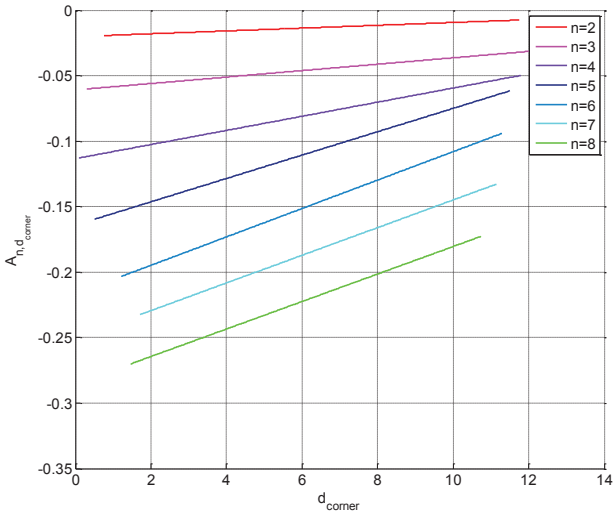

Figure 4.12: Regression fitting result (coefficients A VS tag-to-corner distance).

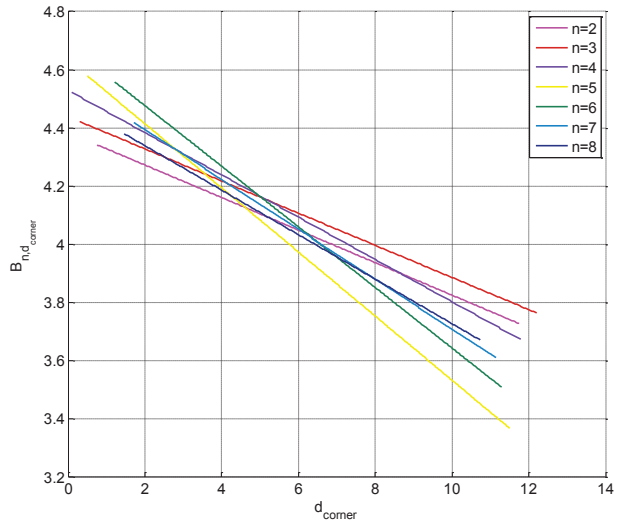

Figure 4.13: Regression fitting result (coefficients B VS tag-to-corner distance).

Table 4.1: Coefficients for the proposed Model.

n	C_n	D_n	E	F
2	0.00187	-0.02493		
3	0.00374	-0.06587		
4	0.00603	-0.1140		
5	0.00709	-0.1586	-0.0821	4.541
6	0.00902	-0.2044		
7	0.01001	-0.2458		
8	0.01174	-0.2942		

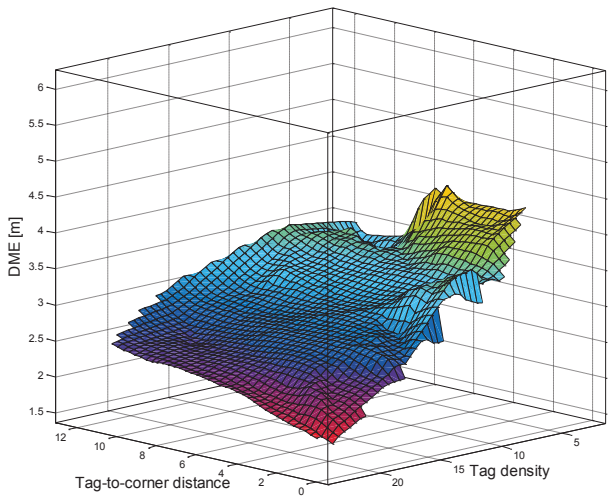

Figure 4.14: Empirical data.

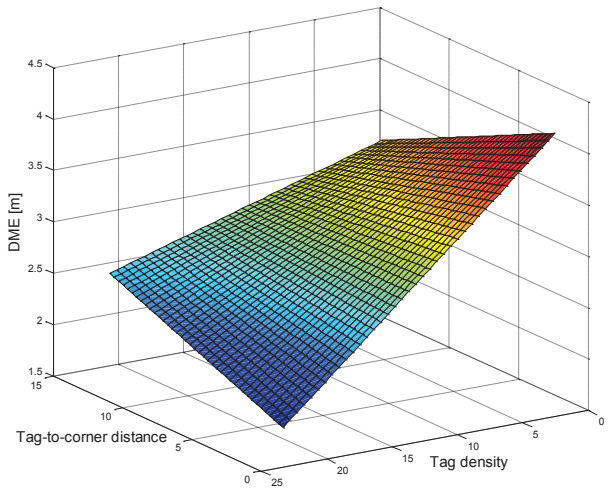

Figure 4.15: Proposed Model.

Chapter 5

Modeling the Magnetometer Error

5.1 Introduction

With the development of the MEMS technology, it's becoming usual to integrate magnetometers, accelerometer and gyroscope inside the smart-phone. Therefore, the inertial hybrid localization technology is widely used to mitigate the inaccuracy in the standalone wireless localization results and improve the reliability in the absence of Radio Frequency Signal. One of the well-known application is the Wi-Fi localization assisted inertial navigation system, which is famous for its low cost, high availability and extraordinary accuracy. In this chapter, we build a hybrid localization platform on the Android smart-phone and conduct measurements with different indoor scenarios. From these results, it shows that there are two error distribution existed in the direction estimation. Based on these two error distributions, we simulate performance of the hybrid localization system and compare with the typical Wi-Fi localization. The result gives the general idea of direction estimation reliability in different environment. The proposed error model and its effect on the hybrid localization system could help performance analysis and error calibration

with prior knowledge of the environment.

5.1.1 Background

Nowadays, the increasing needs of indoor localization technology not only demand an accurate localization solution, but also require the reliability of localization system [39]. Since wireless indoor localization requires the prior knowledge of the reference points and suffers from the multipath phenomenon caused by complicated indoor environment, the traditional pedestrian dead reckoning method maintains its priority in the absence of supporting infrastructure. To achieve the improvement of localization accuracy and reliability, researchers proposed hybrid localization applications to combine wireless localization with traditional localization method. One of the most common use is to implement such hybrid localization system on smart-phone. The rapid development of the smart-phone requires accuracy localization to support the social application and mobile P2P network [40]. This demand promotes the development of a hybrid localization system based on smart-phone, which utilize data from its embedded sensors. Compared with other existing indoor localization technologies, the hybrid localization approach is well known for its widely usability, higher localization accuracy and stronger reliability.

5.1.2 Related Work

In past few years, this kind of localization system has becoming a hot topic in research area. Christian Lukianto et al. built the earliest prototype system that combine the inertial and wireless hybrid localization based on smart-phone and a inertial measurement unit (IMU) [41]. A step further, Wonho Kang et al. achieve to integrated the whole system by smart-phone [42]. Instead of these works which have the assumption that the smart-phone is placed face up and head to the moving

direction, Fan Li et al. build the system to handle the situation that the smartphone is in the pocket [43]. Note that in this chapter, our measurement is based on the first system setup.

Based on our literature search, all of the works we mentioned above generally focused on platform implementation, algorithm optimization and system integration. However, they merely talk details about the fluctuation among movement direction estimation. Wonho Kang et al. noticed the existence of unreliability of magnetometer in complex indoor environment, however, their work lacks of providing detailed mathematical model of the direction estimation error. Although we can see in an intuitive way that the existence of the metal component will affect the performance of the magnetometer, it is essential and necessary to give the academic and industry an mathematical model of magnetometer's error.

5.1.3 Our Work

Among this chapter, we firstly measure the magnetometer's performance based on several simple scenarios which include open space, metal doors and elevators. Based on the empirical data, the effect of the metal component to the direction estimation error is investigated and a statistical model is introduced. Then we take a further step to measure the direction estimation error in some real-world environments like grocery store and typical office building. In analysis of these measurement results, we combine it with the environment layout to validate the model we bring in previously. After the validation process is successfully finished, the model of the magnetometer's performance can be uniformly used in general scenario which given the knowledge of environment. Therefore it could be used in simulation for future algorithm design, system performance analysis and application optimization about hybrid localization system [44] [45].

The remainder of this chapter is organized as follows. In section 5.2, the measurement system and measurement scenario has been introduced and necessary definitions has been provided for further analysis. In section 5.3, the effect of metal component to the direction estimation in different scenarios has been analyzed. In section 5.4, the measurement has been repeated in the real-world environment and previous analysis has been applied to explain the magnetometer's performance. In section 5.5, we summarize this chapter and discuss future work.

5.2 Scenario and System Setup

In this section, the measurement scenario, system setup as well as necessary parameter definitions have been discussed. We firstly conducted a series of isolated component measurements to analyze the relationship of direction estimation error and distance between metallic component and smart-phone. After that, we implement an inertial based system on the smart-phone to investigate the effect of metallic components on indoor pedestrians dead reckoning (PDR) localization. Finally, an INS/RF hybrid localization system has been developed to expand the performance analysis work towards nowadays practical localization approaches.

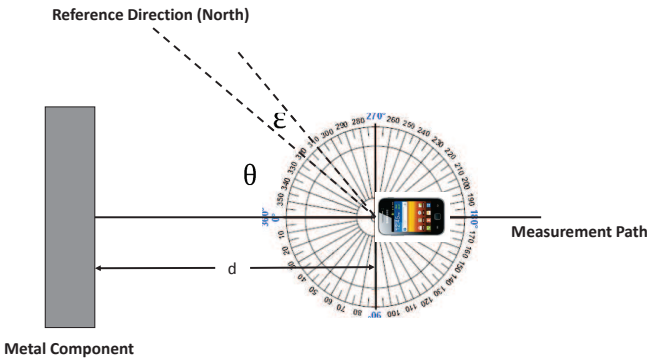

Figure 5.1: Isolated Component Measurement. Measurement path is perpendicular to the metal component.

5.2.1 Isolated Component Measurement

During isolated component measurement, we choose a series of scenarios including open space, door and elevator. Both the door and elevator can be seen as regular metal component in the indoor environment while the open space measurement is used as a controlled trial. Like it shows in the Fig. 5.1, we locate the isolated components and get measurement path's direction through Google Earth as the ground truth.

Since all measurements are done at the Worcester, MA, USA, 1° changing in longitude is corresponding to 83Km in distance. Therefore, all of our measurements have the assumption that the reference direction (North) will not change during the process. The entire measurement process is performed using a self-designed android application on Samsung Exhibit II SGH-T679 smart phone, which runs on any android system above version 2.2 and is embedded with accelerometer and magnetometer for step detection and direction estimation respectively.

Then we do the measurements along that path at varieties of distances to the target component including 200cm, 100cm, 50cm, 40cm and 30cm. In each measurement points, we sample the data of the magnetometer with the frequency of 10Hz for 30 seconds. The definition of the direction estimation error is given as

$$\hat{\theta}_i = \theta_i + \epsilon_i \tag{5.1}$$

where the ϵ_i represent the i^{th} direction estimation error.

5.2.2 Inertial Navigation

PDR navigation measurement in real-world environment is conducted in varieties of places. The open space measurement is done on the playground of the Worcester

Figure 5.2: Direction Error Measurement in AK.

Polytechnic Institute (WPI), the grocery store environment is inside a grocery store and the typical office building is represented by 3^{nd} floor of the Atwater Kent Laboratory, the office building of ECE department at the WPI as shown in Fig. 5.2. In all four scenarios, rectangular trajectory has been selected for measurement and the four edges of the rectangle is defined as 1^{st} to 4^{th} sub-path in order. The objective first and foremost goes through a training process to obtain the average step length l and then walks along the main corridor on a constant speed, holding the smart phone in hand. The measurement starts from one randomly selected corner in the path and lasts for three entire cycles. Note that constant walking model is not a limitation on this work, preliminary results shows that following discussion still applied to random walking situation and it will be mentioned in future publications.

In our inertial based localization system, the pre-defined relative coordinate sets the origin position at (50, 0) and ground truth direction of the 1^{st} sub-path is relative North. The data is collected at every time a step is detected and PDR particle is recorded in the format of $(\hat{x}_i, \hat{y}_i, \hat{\theta}_i)$ where $\hat{x}_i$ and $\hat{y}_i$ is the position in relative coordinate and $\hat{\theta}_i$ is the angle between moving direction and relative North.

The update process of particles is showed as below

$$\hat{x}_{i+1} = \hat{x}_i + (l + \epsilon(l)) \sin(\hat{\theta}_i) \tag{5.2}$$

$$\hat{y}_{i+1} = \hat{y}_i + (l + \epsilon(l)) \cos(\hat{\theta}_i) \tag{5.3}$$

$$\hat{\theta}_{i+1} = \hat{\theta}_i + \delta\hat{\theta} + \epsilon(\theta) \tag{5.4}$$

where $\delta\hat{\theta}$ denotes the direction change, $\epsilon(l)$ and $\epsilon(\theta)$ are error terms drawn from the step length and direction estimation respectively. The iterative process indicates that the beginning state is required for this approach. Note that due to the training process of the step detection algorithm and the constant speed walking, the step length error is constrained at 0.2 meter. To evaluate the localization accuracy, we define the localization error for a specific PDR particle as

$$\epsilon_i = \sqrt{(\hat{x}_i - x_i)^2 + (\hat{y}_i - y_i)^2} \tag{5.5}$$

where x_i, y_i are the ground truth coordinate for i^{th} PDR particle.

5.2.3 Hybrid Localization

To catch up with the latest technology and investigate the effect of direction estimation error on the performance of INS/RF hybrid localization systems, we combine the previously mentioned PDR approach with wireless localization approach, in which we use Kernel Method with the Gaussian kernel function and the reference point data is collected at the four corners at the 3^{nd} floor AK Building. The equation

of the Gaussian kernel method is showed as [46]

$$p(o|l) = \frac{1}{n}\sum_{i=1}^{n} K(o : o_i) \tag{5.6}$$

$K(o : o_i)$ denotes the Gaussian kernel function with observation o and the i^{th} training data o_i. $p(o|l)$ represents the probability that at location l we get the observation o which is calculated by take the equally weighted Gaussian kernel results. The detailed Gaussian kernel function is shown below

$$K_{Gauss}(o : o_i) = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(o - o_i)^2}{2\sigma^2}) \tag{5.7}$$

where σ is the adjustable parameter that is used to determines the width of the kernel.

To combine the inertial navigation result with the wireless localization, we use the Kalman filter to do the off-line statistical signal processing. In the state prediction part, the Kalman filter takes the inertial navigation data and calculates the position result corresponding to the previous PDR particle update equations.

$$\bar{p}_t = f(p_{t-1}, \mu_t) = Ap_{t-1} + B\mu_t \tag{5.8}$$

$$\bar{p}_t = \begin{bmatrix} \bar{x}_t \\ \bar{y}_t \end{bmatrix} \quad p_{t-1} = \begin{bmatrix} x_{t-1} \\ y_{t-1} \end{bmatrix} \quad \mu_t = \begin{bmatrix} \sin\theta_t v_t \\ \cos\theta_t v_t \end{bmatrix} \tag{5.9}$$

$$A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad B = \begin{bmatrix} t & 0 \\ 0 & t \end{bmatrix} \tag{5.10}$$

where $\bar{p}_t$ is the t^{th} position prediction, p_{t-1} is the $t - 1^{th}$ position and μ_t is the

t^{th} inertial navigation data. Since the wireless localization also gives the position result directly, the Kalman filter's sensor prediction process is just use unit matrix to transfer the state prediction to sensor prediction

$$\bar{z}_t = C\bar{p}_t \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.11}$$

and the working process of the Kalman filter is showed as

$$\begin{cases} \bar{p}_t = Ap_{t-1} + B\mu_t \\ \bar{E}_t = AE_{t-1}A^T + E_p \\ K_t = \bar{E}_t C^T (C\bar{E}_t C^T + E_z)^{-1} \\ p_t = \bar{p}_t + K_t(z_t - C\bar{p}_t) \\ E_t = (I - K_t C)\bar{E}_t \end{cases} \tag{5.12}$$

where E_p is the error in the inertial navigation result, E_z is the error in the wireless localization result, E_t is the t^{th} error in the whole Kalman filter result and K_t is the so-called Kalman filter gain.

5.3 Effect of the Various Metal Component on Direction Estimation Accuracy

The general direction estimation result along the measurement path toward a metal door is showed in the Fig. 5.4. As for the bar figure of the direction estimation, it is very obvious that result keeps drifting away from the ground truth when the measurement position keeps forward to the metal door. Such drifting phenomenon

agree with the pure inertial navigation system performance showed in other's works. When the smart-phone is at 200cm away from the metal door, there are nearly no error between the direction estimation and the ground truth. The error become 10° when the measurement is located at 100cm and it's speeding up to drift away from ground truth in the same direction when it gets close to metal door. Finally, the error reaches 30° at distance of 30cm.

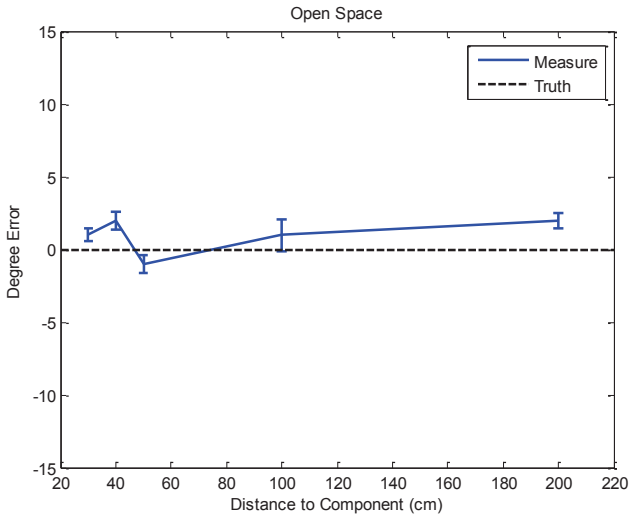

Figure 5.3: Direction Estimation Error Distribution Fitting (Open Space).

We repeat the same measurement process at the open space and in front of elevator and show the bar figure of the result. As we seen from the Fig. 5.5, the direction estimation error with elevator in front also has the similar property that error may increase when the smart-phone is getting close to the elevator. Compare with the open space measurement result in the Fig. 5.3, we can see that metal component has significant effect on the direction estimation. Finally, we come up with a mathematical model of the distance to metal component and the magnetometer's estimation error in degree.

$$\epsilon = \alpha \times e^{-\beta d} \tag{5.13}$$

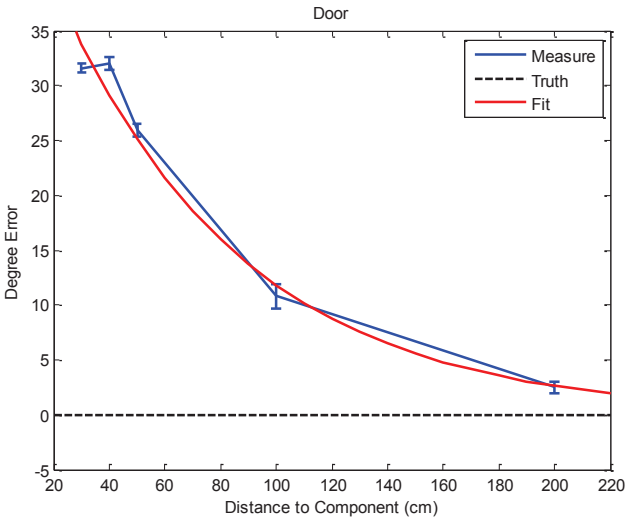

Figure 5.4: Direction Estimation Error Distribution Fitting (Door).

where ϵ is the direction estimation error, d is the distance to component and (α, β) is coefficient.

Component	α	β
Door	53.05	0.015
Elevator	59.41	0.021
Metal Shelf	51.63	0.015

5.4 Data Analysis and Performance Simulation

In this section, we first show the direction estimation error among the PDR localization with the empirical data collected in real world scenarios. Through the process, our direction estimation error model is validated. Second, we use our direction estimation model to simulate the inertial navigation data and to evaluate the performance of the hybrid localization system under different inertial navigation

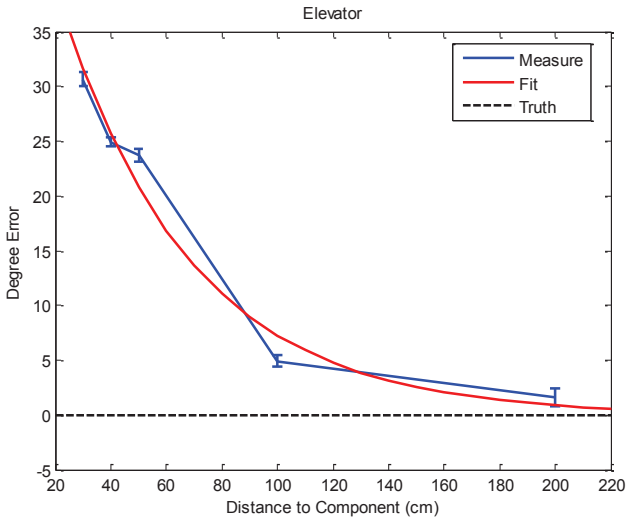

Figure 5.5: Direction Estimation Error Distribution Fitting (Elevator).

errors.

5.4.1 Direction Estimation Error Among PDR Localization

Since we have shown that the metal component can cause a significant error in direction estimation measured by the magnetometers, in this section we analysis the direction estimation error in PDR localization. In real-world environment, the corridor in building is usually within the width of 2 to 3 meters. For INS user who is walking through corridor, the distance between the user and wall should around 1 meter. When the corridor has several metal doors or elevators in the wall, there will be continues direction estimation error in user's inertial navigation result. Therefore, we conducted inertial navigation measurements in several different real-world scenarios to test its performance. During the measurement, we use INS to trace a rectangle shaped walking path and each side of the path is defined as Sub-path 1 to Sub-path 4 according to visiting order. The results are showed in the Fig. 5.6 and Fig. 5.7.

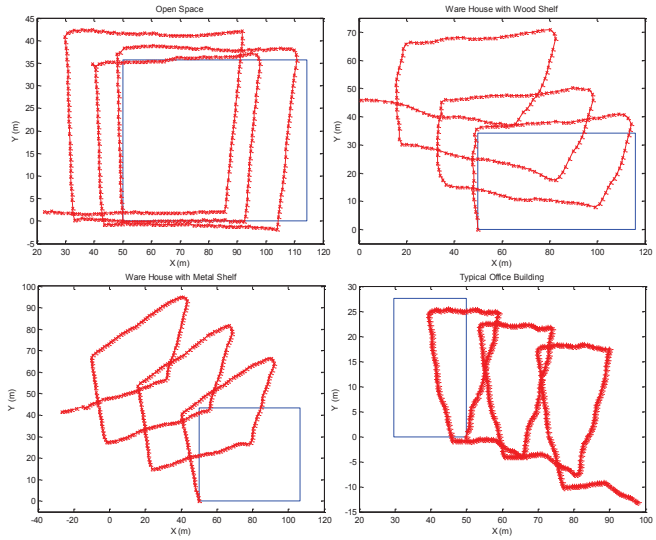

Figure 5.6: Real-world Environment Measurement. (a) Open Space; (b) Grocery Store with Wood Shelf; (c) Grocery Store with Metal Shelf; (d) Typical Office Building

Intuitively from Fig. 5.6, we can see that INS has the best accuracy in the open space environment. Each Sub-path of tracing result has similar direction with its corresponding ground truth. From curve fitting of the error distribution in Fig. 5.7, the INS result shows a relatively poor performance under other three remaining scenarios. Especially in metal shelf warehouse and office building, the bias mean direction estimation error is dominating. For the further analysis of direction estimation error in these indoor localization scenarios, we isolate the error according to each Sub-path and repeat the curve fitting to show the error distribution.

In our measurement environment in metal shelf warehouse, shelves are placed with 1 meter in between. When we do the measurement, the distance between the INS user and shelf is 40 to 60 centimeters. According to the introduced model in previous section, the direction estimation error should located between 20 to 30 degree. Fig. 5.9 shows the direction estimation error of each Sub-path in metal shelf warehouse. All of them is bias mean Gaussian distribution and mean error agree

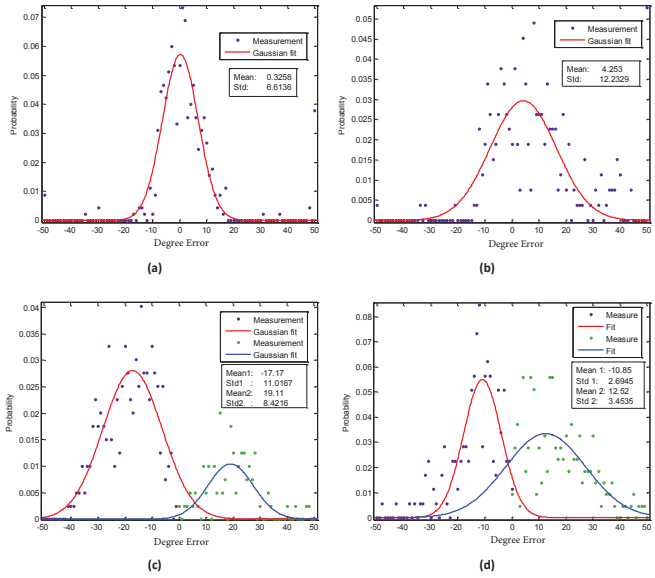

Figure 5.7: Direction Estimation Error Distribution. (a) Open Space; (b) Grocery Store with Wood Shelf; (c) Grocery Store with Metal Shelf; (d) Typical Office Building

with our mathematical model.

The measurement in typical office building is conducted at 3rd floor of AK building. The corridor width is 1.8 meters and there are continuous metal doors along Sub-path 1 and 3 of the corridor. Therefore, we take the Sub-path 2 and 3 to continue our analysis. In the Fig. 5.8, we show the direction estimation error distribution in Sub-path 2 and 3 along with environment snapshots. We can see that there are only one door in Sub-path and the error distribution is a clearly zero mean Gaussian distribution. Compare with that, there are lots of metal doors which are the doors for office rooms appear at the right hand side of the path. Highly agree with that, the error distribution in this Sub-path shows a significant bias mean Gaussian distribution. The 15 degree mean error indicates that the distance between door and user is about 80 centimeters which fits our situation that the user is walking through a 1.8 meters width corridor.

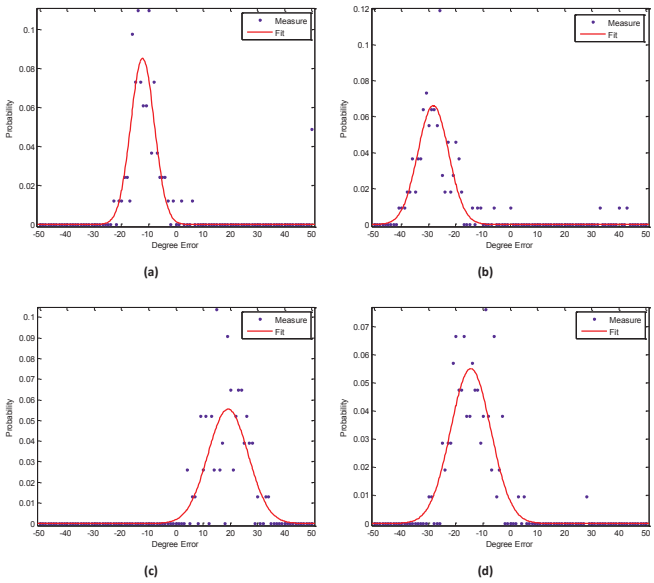

Figure 5.8: Direction Estimation Error Distribution in grocery store with metal shelf. (a) Sub-path 1; (b) Sub-path 2; (c) Sub-path 3; (d) Sub-path 4

5.4.2 Simulation of Hybrid Localization

In this section, we use the error model to simulate the performance of the hybrid localization system and compare it with pure Kernel result. The simulation generates inertial navigation data with zero/bias mean Gaussian Distributed Error and use Kalman filter to combine them with the wireless localization result. Fig. 5.10 5.11 is the CDF plot of DME.

If the direction estimation error is zero mean Gaussian distribution, the DME of hybrid localization approach is always better than pure Kernel algorithm. When standard deviation (std) of direction estimation error is 20^o, the hybrid approach is 2 meters superior to pure Kernel. Even if the std rise up to 50^o, the performance of hybrid approach is still comparable with pure Kernel. Whenever the error distribution became bias mean Gaussian, inertial based part can only provide limited support to hybrid localization. With the error std goes beyond 20^o, the performance

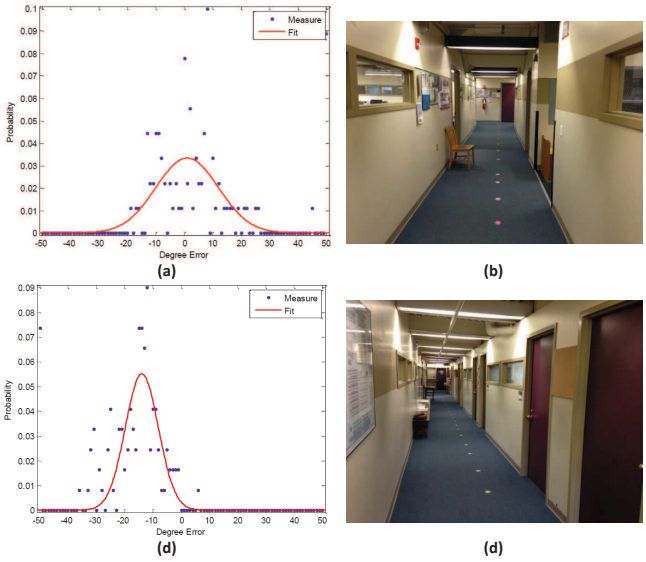

Figure 5.9: Direction Estimation Error Distribution in typical office building. (a) Sub-path 1; (b) Sub-path 2; (c) Sub-path 3; (d) Sub-path 4

of hybrid localization is even worse than pure Kernel approach.

These results show us that the hybrid localization is not robust in an environment where metallic component's effect to magnetometer is strong. In real application, there should be a threshold to prevent using hybrid localization in such environment or to trigger the calibration of magnetometers.

5.5 Conclusion and Future Work

The major contribution of this chapter is that we analyzed the effect of metal component on magnetometer's direction estimation error and introduced a simple error model. Then we validate our error model by conducting measurements in varieties of real-world scenarios. Finally, we utilize this model to simulate the performance of hybrid localization system in different kinds/scales of direction estimation error. In the future, we will work on building a more detailed model on top of it which

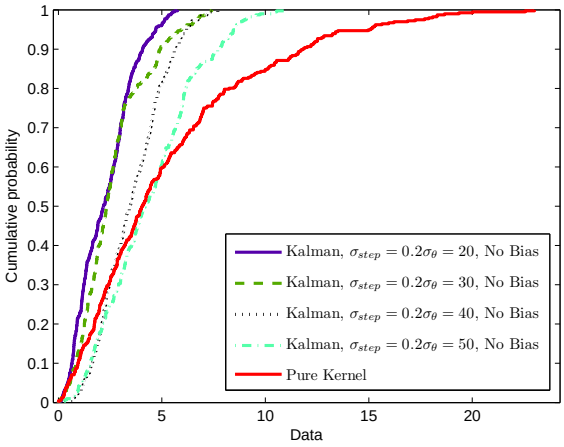

Figure 5.10: Simulation Result for Hybrid Localization (Zero Mean).

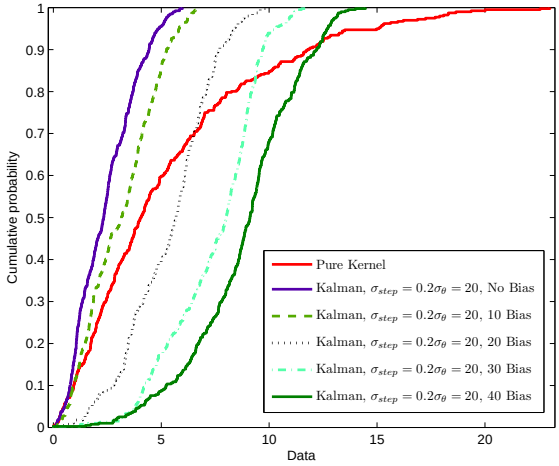

Figure 5.11: Simulation Result for Hybrid Localization (Bias Mean).

could be used as a simulation test bench.

Chapter 6

Modeling the Shadow Fading

6.1 Introduction

With the increasing interests about the indoor geolocation, wireless localization, as a important technology in this area, is becoming popular nowadays. Among different wireless localization technologies, RSS base localization, which includes RSS ranging and fingerprint algorithms, is known for its robustness in complex indoor environment and have been dominating. In previous work on RSS based localization algorithm, people always simply assume that the shadow fading as a static value. However, the shadow fading is used to reflect the complexity of the environment and should be sensitive to the changes during the experiment. In this chapter, we introduced a statistical RF signal shadow fading model based on the measurements in a typical office building. This model related the shadow fading to the distance and use two different distribution function to approximate the shadow fading between the breaking point. Since the ranging based RSS localization algorithm and the fingerprint localization algorithm are widely used these days, we use this distance related shadow fading model to the further analysis of its effect on these localization

algorithms.

6.1.1 Background

Nowadays, the increasing interests about the Internet of Things not only demand high data rate wireless communication, but also require the precise indoor localization. In the indoor environment, the signal from the global positioning system (GPS) is almost impossible to be received. Even if a successful reception of GPS signal occurs, the localization accuracy of the GPS technology is too low to be used for the indoor localization application. Besides the inapplicable GPS technology, the TOA, POA and RSS based approaches are well known in the indoor localization area. Among these technologies, TOA and POA could offer a high localization accuracy [47]. However, they are not so robust in the real world application since the complicated environment with heavy multipath can easily result in a huge DME in using these technologies [48]. Compared with the TOA and POA technology, the RSS based approach will not suffer from the multipath and is becoming the most common choice in implementation of the indoor localization application.

6.1.2 Related Work

In past few years, there are many different RSS based indoor localization algorithms has been proposed. To effectively investigate the performance of these algorithms, the simulation is an important part of the work. Therefore, a good mathematical model is necessary to ensure correctness and efficiency in the algorithm design, performance analysis and system optimization [49]. However, when modeling the radio propagation during the previous work of simulation, people always simply assume that the shadow fading as a static value. In reality, the shadow fading, which is used to reflect the complexity of the environment, should be sensitive to

the slight changes of the physical environment during the experiment. Therefore, the shadow fading should be fluctuated when the distance between the transmitter and receiver is changed.

6.1.3 Our Work

The implementation of different RSS based indoor localization algorithms choose different technologies in transferring the RSS information to the location information. For example, in the RSS ranging algorithm, the Friis transmission equation is used to estimate the distance. And the fingerprint algorithm is based on the probabilistic model between the measurement RSS and the signature RSS. In these algorithms, the fluctuation of the shadow fading will cost different scales of DME. In the RSS ranging algorithm, the DME is directly coming from the RSS variation which is directly caused by the shadow fading. While, in the fingerprint algorithm, the error is the result of the confusion between true position and fake position and this kind of confusion is highly related to the shadow fading.

Therefore, a precise shadow fading model is strongly necessary in algorithm design, performance evaluation and application development. In this chapter, we firstly introduced a statistical shadow fading model based on the measurements in a typical office building. In this model, we relate the shadow fading to the distance between the transmitter and receiver and use two different distribution function to approximate the shadow fading between the two sides of the breaking point [50] [51]. Therefore, the shadow fading model can be easily used in different RSS based indoor localization algorithms with varieties of deployments. Then, we show the effect of this shadow fading model through introducing it into the analysis of RSS ranging algorithm and fingerprint algorithm.

The remainder of this chapter is organized as follows. In section 6.2, the mea-

surement scenario and measurement system has been introduced and necessary definitions has been provided for further analysis. In section 6.3, the measurement result of shadow fading has been analyzed and a regression fitting has been applied to it. In section 6.4, the statistical model of the shadow fading has been used to analysis its effect on different indoor localization algorithms. In section 6.5, we combine the shadow fading model with the previous direction estimation model to build an INS/RF hybrid system test bench and analyze the localization performance. Finally, in section 6.6, we give the conclusion and future work.

6.2 Scenario and System Setup

As it's showed in the Fig 6.1, the measurement system is composed by a vector network analyzer (Agilent E8363), a pair of 3G to 8G antenna (Skycross SMT-3TO10M) and low loss cables [52]. The transmitter and receiver antennas are fixed on the shelf which is 1.5 meters above the ground. During the test, vector network analyzer is used to measure the received signal from S21 channel in frequency domain with 1601 sample points [53].

The measurement was done in the corridor of Atwater Kent Laboratory, a typical office building located in the Worcester Polytechnic Institute, Worcester, MA, US. As shown in Fig 6.1 (a), the distance between the transmitter and receiver was changed from 10cm to 20m with 10cm step length under 1 meter and 1m step length from 1m to 20m.

From the Friis equation, the shadow fading is a random variable and the power gradient α is highly related to the transmission environment.

$$L_p = L_0 + 10\alpha \log d + X \tag{6.1}$$

Therefore, we repeat measurement 500 times in each location and use the variance to approach the shadow fading.

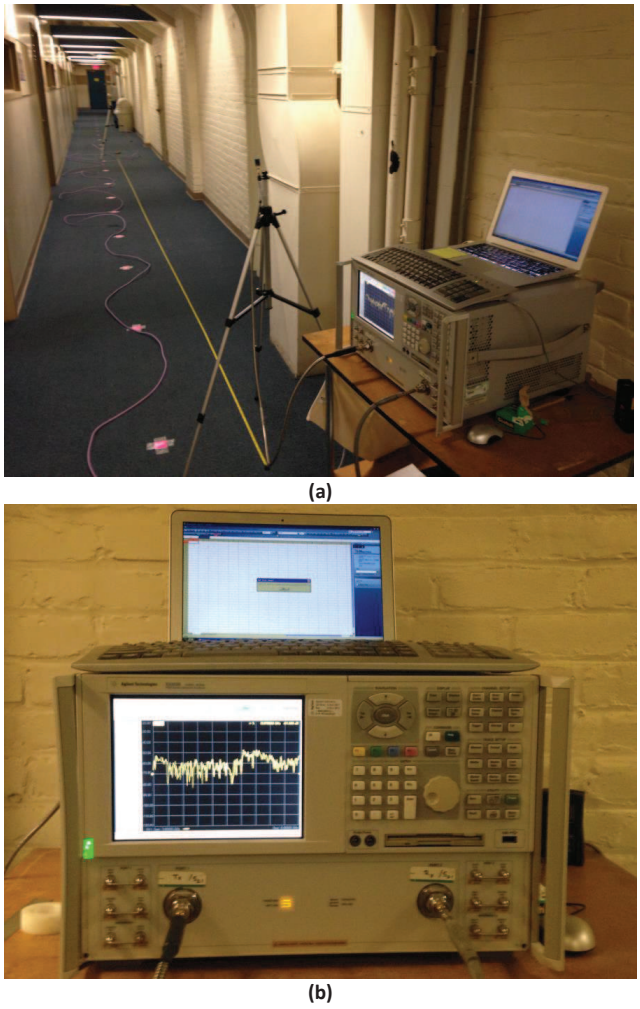

Figure 6.1: Measurement System. (a) System Setup. (b) Vector Network Analyzer.

6.3 Data Modeling and Validation

In this section, we use the measurement result to build a mathematical model which relates the shadow fading with the distance between the transmitter and receiver. With the distance dependent shadow fading model, we calculate the spatial average

shadow fading model which shows the relation between the average shadow fading and the coverage of the signal. The comparison of spatial average shadow fading and the IEEE 802.11 validates our shadow fading model.

6.3.1 Modeling Shadow Fading

From the Fig 6.2, the red curve, which is the measurement result, shows that the shadow fading in short distance is small and continuously increase with the distance in between. This is reasonable since in short distance the path loss is lower and the transmission environment is relatively simple. As a result, the RSS should be more stable than in longer distance.

To model the shadow fading, we choose to use the exponential function

$$\sigma = \lambda \times e^{-\beta \times d} + \gamma \tag{6.2}$$

We can see the curve fitting result from Fig 6.2. It shows that the exponential function is better to model the changing of shadow fading within the break point. For the shadow fading after 5 meters, the exponential function is becoming flat and finally approaching the γ. It meets the IEEE 802.11 standard and the value of γ will related to the environment which is the typical shadow fading value defined inside the IEEE 802.11.

Table 6.1: Fitting Parameters.

λ	β	γ
-4.28	0.9372	4.31

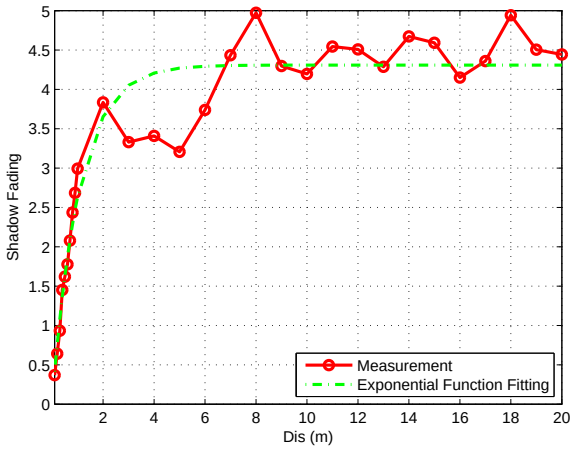

Figure 6.2: Curve Fitting Result.

6.3.2 Model Validation

After we got the relation between the shadow fading and distance, it's easily to be used to calculate the spatial average of the shadow fading. Assume there is a transmitter in the space, the location with equal shadow fading is circled around the transmitter. Like what we do in calculation of the spatial average date rate, we use integration to sum up the different shadow fading $2\pi\sigma(d)$ from 0 to the coverage R. Then the spatial average of shadow fading will become [13]

$$\sigma(R) = \frac{\int_0^R [2 \times \pi \times r \times \sigma(r)]dr}{\pi \times R^2} \tag{6.3}$$

From the spatial average of the shadow fading, we could get the shadow fading for different technologies based on their coverage.

Since we have two functions to model the shadow fading between the break point, we also have two spatial average function of the shadow fading which are Plug in to the previous equation, we could calculate the spatial average of the shadow fading

with our distance dependent shadow fading model

$$\sigma(R) = \frac{-\frac{2\alpha}{\beta^2}[(\beta R + 1)e^{-\beta R} + 1]}{R^2} + \gamma \tag{6.4}$$

And the result of the spatial average is showed in the Fig 6.3. It shows that for the typical RFID which coverage is around 1 meter, the shadow fading is less than 1 dB. The iBeacon and Wi-Fi signal, which usually has more than 10 meter coverage, the shadow fading is around 4.5 dB. As we said from the previous section, this value will change by the environment and is similar to the typical shadow fading value defined inside the IEEE 802.11.

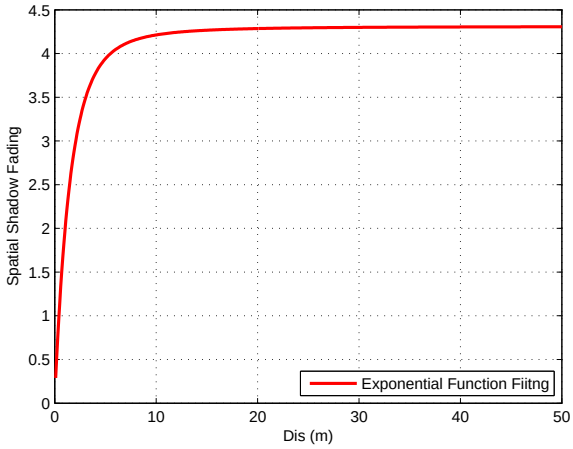

Figure 6.3: Spatial Average of Shadow Fading.

6.4 Effects on the Localization Algorithms

In this section, we use the shadow fading model to analyze its effect on several localization algorithms. Here, we choose the RSS ranging algorithm and the fingerprint algorithm as examples.

6.4.1 Effect on RSS Ranging

In indoor localization area, the Cramer-Rao Lower Bound (CRLB) is used to compare the performance of different localization algorithms. In itself, Cramer-Rao Lower Bound expresses a lower bound on the variance of estimators of a deterministic parameter. If we combine the location estimation algorithm with the Cramer-Rao Lower Bound, it could measure the spread of the error associated with the location estimation [54].

In RSS Ranging algorithm, it tries to use the receiving signal strength, power gradient and the Friis equation to generate the distance between the transmitter and receiver. The observation of this algorithm is

$$O = g(d) + \eta = P_r = P_0 - 10\alpha \log d + X \tag{6.5}$$

Based on the conditional probability, we could come up with probability density function of getting the observation O at the distance d.

$$f(O/d) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{[O-g(d)]^2}{2\sigma^2}} \tag{6.6}$$

Since the shadow fading is changed with the distance which means that $\sigma = \sigma(d)$, the natural logarithm of probability density function could be expanded to

$$\ln f(O/d) = \ln \frac{1}{\sqrt{2\pi}} - \ln \sigma(d) + \frac{[O - g(d)]^2}{2\sigma^2} \tag{6.7}$$

Finally, the Fisher information matrix and Cramer-Rao Lower Bound will be

$$\begin{aligned}\boldsymbol{F} &= -\boldsymbol{E}[\frac{\partial^2 \ln f(O/d)}{\partial d^2}] = \boldsymbol{E}[\frac{\partial \ln f(O/d)}{\partial d}]^2 \\ &= \boldsymbol{E}[\frac{[O-g(d)]\{[O-g(d)]\sigma'(d) - g'(d)\sigma(d)\}}{\sigma^3(d)} - \frac{\sigma'(d)}{\sigma(d)}]^2 \\ &= [\frac{\sigma'(d)-g'(d)}{\sigma(d)} - \frac{\sigma'(d)}{\sigma(d)}]^2 = \frac{[g'(d)]^2}{\sigma^2(d)}\end{aligned} \tag{6.8}$$

$$\sigma_p \geq \sqrt{CRLB} = \sqrt{\boldsymbol{F}^{-1}} = \sqrt{\frac{(\ln 10)^2\sigma^2(d)}{100\alpha^2}d^2} = \frac{\ln 10\ \sigma(d)}{10\alpha}d \tag{6.9}$$

To compare the changes in the Cramer Rao Lower Bound of the RSS ranging algorithm, we choose to bring in the shadow fading model and compare it with traditional CRLB which assumes that shadow fading equals to 5. From the result in Fig 6.4, we can easily figure out that the CRLB with the distance dependent shadow fading model is highly different from the tranditional one in short distance. It shows that the traditional CRLB is tow times of the CRLB calculated with distance dependent shadow fading model within 2 meters. This tells us that the RSS ranging algorithm could has a better performance in short distance which means that the iBeacon or Wi-Fi signal could be used as the "RFID" within a short distance.

6.4.2 Effect on Fingerprint

Generally in the fingerprint algorithm, the RSS information is collected in different positions, which is the so-called reference point, to build a database. The localization process is to find one of these positions that has the highest probability based on the information sampled by the device. The selection of the (x, y) coordinates corresponds to the reference point RSS vector with the smallest signal distance to the sampled RSS vector as the estimated location. Therefore, the core of the fingerprint is the function to calculate the signal distance. Commonly, people will

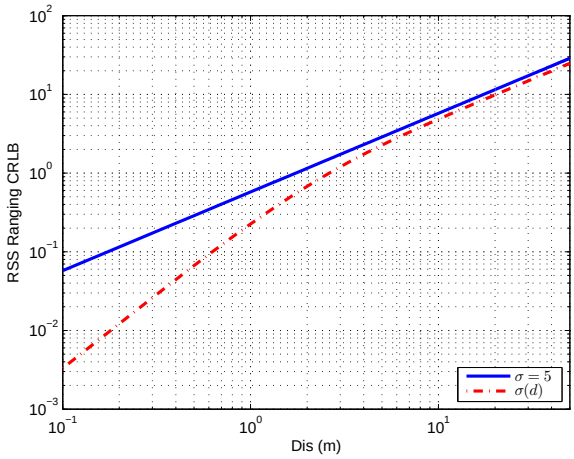

Figure 6.4: CRLB of RSS Ranging.

use the Euclidean distance to represent the signal distance and the distance metric is shown as

$$Z = [\sum_{i=1}^{N} (\rho_i - r_i)^2]^{\frac{1}{2}} = [\sum_{i=1}^{N} q_i^2]^{\frac{1}{2}} \tag{6.10}$$

where p_i is the element of RSS vector at reference point and r_i is the element of RSS vector at the measurement position [10].

In real world scenario, the RSS from each APs are the individual random variable within each reference point RSS vector. Moreover, according to the Friis equation, the RSS will have the normal distribution. Assume we have the random variable $X = Z^2$. If the measured RSS vector is at the same position as traning process, X should have the central chi-squared distribution with N degree of freedom, where N represent the number of RSS inside the vector.

$$p_{X_N^2}(x) = \frac{1}{\sigma^N 2^{\frac{N}{2}} \Gamma(\frac{N}{2})} e^{-\frac{x}{2\sigma^2}} x^{\frac{N}{2}-1} \tag{6.11}$$

If the measured RSS vector is at the different position as traning process, X will

have the non-central chi-squared distribution with a non-centrality parameter

$$\lambda = \sum_{i=1}^{N} \rho_i^2 = \sum_{i=1}^{N} [\boldsymbol{E}\{\rho_i\} - r_i] \tag{6.12}$$

$$p_{X_N^2}(x, \lambda) = e^{-\frac{(\lambda+x)}{2\sigma^2}} \frac{1}{2\sigma^2} (\frac{x}{\lambda})^{\frac{N-2}{4}} I_{\frac{N-2}{2}} (\frac{\sqrt{\lambda x}}{\sigma^2}) \tag{6.13}$$

Here, the $I_\alpha(x)$ is the αth-order modified Bessel function of the first kind.

In evaluation of the shadow fading model's effect on the fingerprint algorithm, we choose a very common scenario that used in the simulation of the fingerprint algorithm. Assume there are 4×4 APs that uniformly distributed in a 20m×20m area and we collect the reference point RSS vector in the middle of the area which is $(10, 10)$. Based on the distance between the APs and reference point, we divide APs into three groups. As it shows in the Fig 6.5, we define that $G_1 = \{X_1, X_2, X_3, X_4\}$ as the group one (blue points), $G_2 = \{X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}\}$ as the group two (yellow points) and $G_3 = \{X_{13}, X_{14}, X_{15}, X_{16}\}$ as the group three (red points).

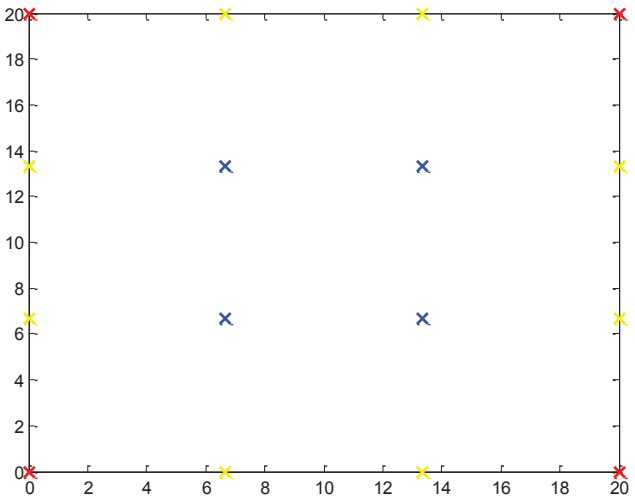

Figure 6.5: Layout of the simulation scenario.

Based on these groups, we conduct three sets of sampled APs where set one

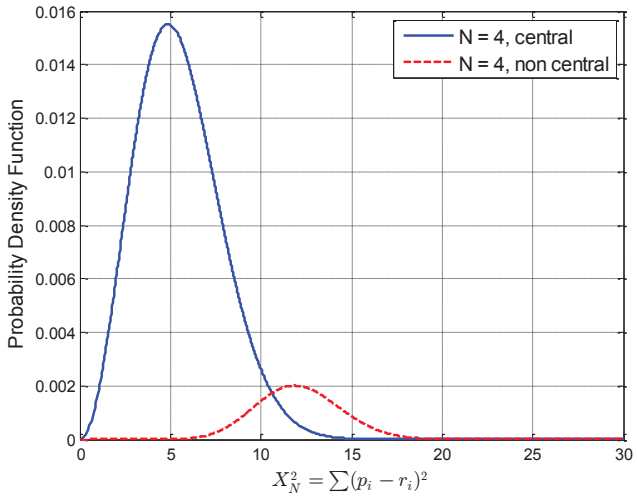

Figure 6.6: Probability Density Function of Power Distance Matrix (Set One).

$S_1 = \{G_1\}$, set two $S_2 = \{G_1, G_2\}$ and set three $S_3 = \{G_1, G_2, G_3\}$. Each time, we take one of three sets to sample the reference point RSS vector and plot the corresponding central chi-squared distribution to compare it with the non-central chi-squared distribution with non-centrality parameter equals to 10.

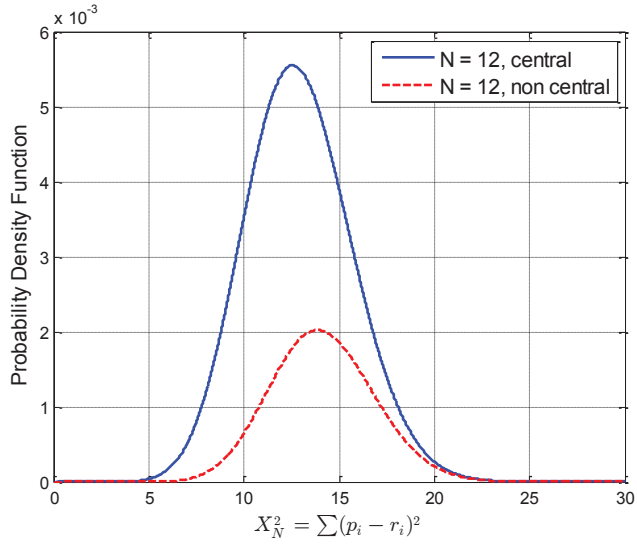

Figure 6.7: Probability Density Function of Power Distance Matrix (Set Two).

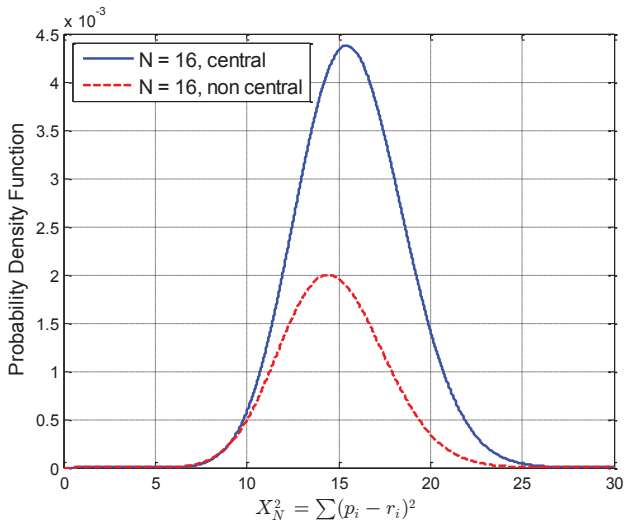

Figure 6.8: Probability Density Function of Power Distance Matrix (Set Three).

From the Fig 6.6, we can see that the distance metric square of true position has a mean on 5 while the fake position's distance metric square is centralized at 12. Based on the fingerprint algorithm's property, it will have an accurate localization result. However, when we try to bring more far away APs into our RSS vector, the performance has a significant change. The result from Fig 6.7 shows that the true position and fake position has a near mean value which will increase the probability that fingerprint algorithm returns fake position coordinate. Finally, from the Fig 6.8, the mean value fake position becomes smaller than the true position and the fingerprint algorithm's performance is even worse than before according to our previous analysis.

This result tells us that for the fingerprint algorithm, the RSS vector database is not the larger the better. Actually, if we bring in some far away APs into the RSS vector, its unstable or larger shadow fading could easily make the performance worse. Therefore, in the algorithm design, it's worth to set up some requirements to evaluate the AP that will be considered to collect as RSS vector.

6.5 Performance Analysis in an INS/RF Hybrid Localization Test Bench

During the implementation of the hybrid localization system, the simulation is a major part of work. Meet with the complex working environment for indoor geolocation application and varieties of technologies to choose, people need an effective way to compare the performance of different implementation and make a decision on top of that. Therefore, a test bench is becoming necessary.

In the previous part, we have introduced a distance dependent shadow fading model which could reflect more details about the shadow fading in indoor environment. Compare with the existing model, it is more geolocation oriented which means it could be used to simulate the indoor RF propagation for the geolocation technology. Combine with the previous direction estimation model, we could build an INS/RF hybrid localization test bench to complete the performance analysis [55].

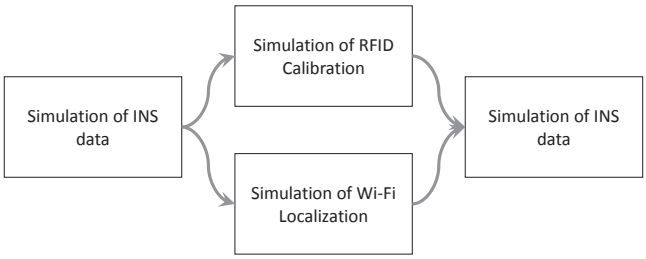

Figure 6.9: System Setup of the Hybrid Localization Test Bench.

In the Fig. 6.9, it shows the working process of our test bench. At first, it generates the inertial navigation result based on the direction estimation error. These data could be combined with RFID calibration data or Wi-Fi localization result through Kalman Filter to generate the hybrid localization result. During the process, we could control the direction estimation error and simulate the RF localization

based on the deployment information. This test bench will also generate a set of ground truth which could help the calculation of DME.

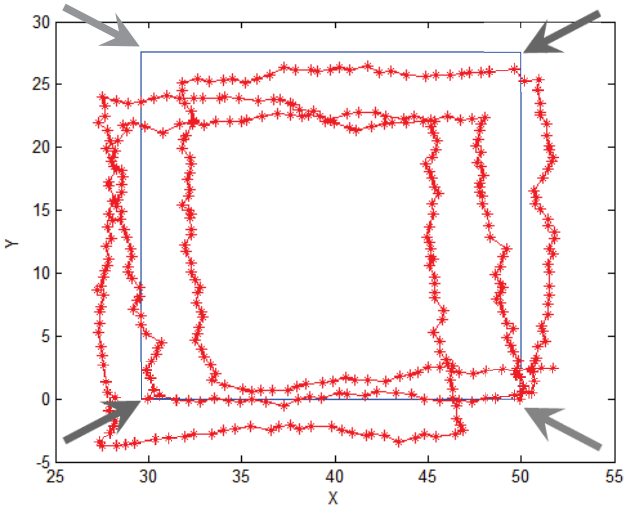

Figure 6.10: Simulation Environment Setup.

With the help of hybrid localization test bench, we compare the performance of the INS/RF and INS/Wi-Fi hybrid system under different direction estimation error. The environment setup is showed in the Fig. 6.10. The INS part traced a three circle rectangular shaped walking and we seperate the INS into zero mean and bias mean error. Along the path, the RFID calibration points and Wi-Fi localization RPs is located at four corners. Using the Monte Carlo method, we could compare the performance of two hybrid localization system with original INS result.

From the Fig. 6.11, we can see that the INS/RFID and INS/Wi-Fi hybrid system has an alternative performance. When the std of zero mean direction error is lower, INS/RFID hybrid system has a better performance. However, when the std is increasing, the different between two system is becoming close. Finally, when the std is go beyond 30, INS/Wi-Fi achieves the better performance. This is because that Wi-Fi has a large coverage than RFID and it could continuously reduce the drift error

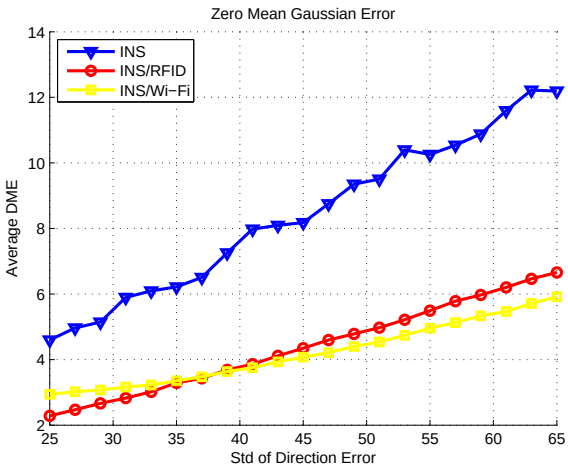

Figure 6.11: Hybrid Localization Performance with Zero Mean INS Error.

in INS. Therefore, when the std is lower, INS/RFID is better since the calibration accuracy is higher and when the std becomes higher, INS/Wi-Fi is becoming better since its continuously correct the localization result.

6.6 Future Work

In this chapter, we conducted the measurement of shadow fading in the LOS situation and present a statistical model for it. Then, we calculate the spatial average of the shadow fading based on this model and compare it with the IEEE 802.11 model to validate it. Later, we combine this model with the CRLB for RSS ranging and the PDF of power distance matrix for fingerprint algorithm to analyze its effect. Finally, we use this shadow fading model and the previous direction estimation error model to build an INS/RF hybrid localization test bench and analyze the system performance. From the result, this shadow fading model and the hybrid localization test bench shows the ability in helping performance analysis and are necessary in the algorithm design, performance evaluation and application development. In the

future, we will work on collecting and modeling the shadow fading under the NLOS situation to improve the simulation accuracy.

Chapter 7

Conclusion and Future Work

This chapter summaries the contribution and conclusion of the thesis as well as presents the plan for future work.

7.1 Summary and Conclusion

In this master thesis, we introduced a self-created hybrid INS/RF system which is built on an Android smart phone. This hybrid system uses an open source step detection code and the embedded magnetometers to form a traditional inertial navigation system to trace the user's walking. It could also collect different RF signal data during the measurement. We use the data collected by this system to complete the off-line research about hybrid INS/RF localization.

With the help of this system, we study the relation between the performance of hybrid INS/RFID localization and the deployment of the RFID calibration tags. We abstract three parameters to represent the deployment of RFID tags which includes number of tags, density of tags and the distance between tag and corner of path. An empirical model is introduced which shows us a mathematical equation to calculate the average DME with these parameters.

After that, we conduct the direction estimation error measurement of the embedded magnetometers in variety of scenarios. The result shows that metallic component in indoor environment have a significant effect on magnetometers and the mathematical relation between the mean value of estimation error and distance (between different metal component and magnetometers) is proposed and validated.

Last but not least, we measure the shadow fading model in LOS situation within a typical office building. Based on the measurement result, we build a distance dependent shadow fading model that shows the continuous change in the shadow fading. Utilizing the shadow fading model, we analyze the performance of several traditional RSS based localization algorithms. It shows that the CRLB of the RSS ranging has a better performance in short distance and the performance of fingerprint algorithm decreases when signature of far away AP is collected. Finally, we create the simulation test bench of hybrid INS/RF system on top of this model and the model of magnetometer error. With the help of this test bench, we compare the INS/RFID with INS/Wi-Fi hybrid system and the result shows that INS/RFID has better performance when INS error is low.

7.2 Future Work

In the future, there are several aspects that we could work on to continue the research of hybrid INS/RF system:

1. Conduct diversified measurement of the direction estimation error to build a more detailed model on top of the existing one.

2. Improve the existing simulation test bench for the hybrid INS/RF localization system.

With the help of this test bench, we could simulate the performance of the hybrid system or any parts of it and the only prior knowledge is the measurement environment, which includes the deployment of the RF source and the position of metallic components along the path. This test bench will be useful to the future algorithm design, performance analysis and application optimization since it offers a economical way to repeat the measurement.

Appendix A

Wi-Fi Compass Source Code

A.1 MainActivity.java

```
package com.example.rcompass;


import java.util.ArrayList;
import java.util.List;
import com.example.rcompass.service.SaveData;


import android.hardware.Sensor;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.net.wifi.ScanResult;
import android.net.wifi.WifiManager;
import android.os.Bundle;
import android.os.Handler;
import android.os.SystemClock;
```

```
import android.view.View;
import android.view.animation.RotateAnimation;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.CheckBox;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.AdapterView.OnItemSelectedListener;
import android.app.Activity;
import android.content.Context;

public class MainActivity extends Activity {

        private float angle = 400f;
        private int stepNum = 0;
        private float accelerate_y = 0f;
        private static long TimeStamp;

        private ImageView IV_compass;
        private boolean record = false;
        private ArrayList<String> recordData = new ArrayList<String>();
        private ArrayList<String> fixData = new ArrayList<String>();
        private ArrayList<String> TrainingData = new ArrayList<String>();
        private ArrayList<String> ScanData = new ArrayList<String>();
        private ArrayList<String> angleData = new ArrayList<String>();
        private WifiManager wifiManager;
```

```
private WiFiTrainning Trainning = new WiFiTrainning();
private float linear_acceleration;
private long timeout = 0;
private float compass_angle = 0f;
private float preAngle = 0;
private float[] angle_buffer = new float[20];
private int last = 0;
private CheckBox Enable;
private static final String[] TrainingNum = { "1", "2", "3", "4",
    "5", "6", "7", "8" };
private TextView view;
private Spinner TrainingNumSpinner;
private ArrayAdapter<String> adapter;
private String TrainingPoint;
private long total_train_time = 0;

Handler handler = new Handler();

public void Record(View view) {
        stepNum = 0;
        record = true;
        fixData.add(String.valueOf(SystemClock.elapsedRealtime()));
        fixData.add("\r\n");
}

public void Stop(View view) {
        record = false;
```

```
}

public void Save(View view) {
        if (record) {
                record = false;
        }

        boolean result = SaveData.saveCompassData(recordData,
            fixData, ScanData);

        if (result) {
                System.out.println("Successful!");
        } else {
                System.out.println("Fail!");
        }
}

public void Calibration(View view) {
        if (Enable.isChecked()) {
                fixData.add(String.valueOf(SystemClock.elapsedRealtime()));
                fixData.add("\r\n");
        }
}

public void WifiScan(View view) {
        TrainingData.clear();
        total_train_time = 0;
```

```
        handler.postDelayed(Trainning, 10000);
}

public void ScanStop(View view) {
        // handler.removeCallbacks(Trainning);

        boolean result = SaveData.saveTrainingData(TrainingData,
            TrainingPoint);

        if (result) {
                System.out.println("Successful!");
        } else {
                System.out.println("Fail!");
        }
}

public void SingleScan(View view) {
        new ScanWifi().run();
        boolean result = SaveData.saveSingleScan(ScanData);

        if (result) {
                System.out.println("Successful!");
        } else {
                System.out.println("Fail!");
        }
}
```

```
public void CompassRecord(View view) {
        total_train_time = 0;
        angleData.clear();
        handler.post(new CompassRecord());
}

/*
 * Record the angle data in 30s, sampling frequency is 2Hz
 *
 * @author Guanxiong Liu
 */
class CompassRecord implements Runnable {

        @Override
        public void run() {
                // TODO Auto-generated method stub
                angleData.add(SystemClock.elapsedRealtime() + " " +
                    angle);
                angleData.add("\r\n");
                total_train_time += 100;
                System.out.println(total_train_time);

                if (total_train_time <= 30000) {
                        handler.postDelayed(this, 100);
                } else {
                        System.out.println("stop");
                        handler.removeCallbacks(this);
```

```
                    boolean result =
                        SaveData.saveCompassRecord(angleData);

                    if (result) {
                        System.out.println("Compass Record
                            Successful!");
                    } else {
                        System.out.println("Compass Record
                            Fail!");
                    }
                }
            }

}

/**
 * Get the WiFi training data from the reference posints.
 *
 * @author Guanxiong Liu
 *
 */
class WiFiTrainning implements Runnable {

        @Override
        public void run() {
            wifiManager.startScan();
            /**
```

```
                 * write wifi data
                 */
                List<ScanResult> tempResult =
                    wifiManager.getScanResults();
                for (ScanResult instance : tempResult) {
                        MainActivity.TimeStamp =
                            SystemClock.elapsedRealtime();
                        TrainingData.add(instance.BSSID + " " +
                            String.valueOf(instance.level) + " " +
                            String.valueOf(MainActivity.TimeStamp));
                        TrainingData.add("\r\n");
                }
                total_train_time += 1000;

                if (total_train_time <= 60000) {
                        handler.postDelayed(this, 1000);
                } else {
                        System.out.println("stop");
                        handler.removeCallbacks(this);
                }
        }

}

/**
 * Get the recent readable WiFi data.
 *
```

```
 * @author Guanxiong Liu
 *
 */
class ScanWifi implements Runnable {
        public void run() {
                wifiManager.startScan();
                /**
                 * write wifi data
                 */
                List<ScanResult> tempResult =
                    wifiManager.getScanResults();
                for (ScanResult instance : tempResult) {
                        ScanData.add(instance.BSSID + " " +
                            String.valueOf(instance.level) + " " +
                            String.valueOf(stepNum));
                        ScanData.add("\r\n");
                }
        }
}

@Override
protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        /**
         * set the listener to the orientation sensor
```

```
 */
SensorManager sensorManager = (SensorManager)
    getSystemService(SENSOR_SERVICE);
Sensor orientation_sensor =
    sensorManager.getDefaultSensor(Sensor.TYPE_ORIENTATION);
sensorManager.registerListener(new
    OrientationSensorListener(),orientation_sensor,
    SensorManager.SENSOR_DELAY_GAME);

/**
 * set the listener to the accelerometer sensor
 */
Sensor accelerometer_sensor =
    sensorManager.getDefaultSensor(Sensor.TYPE_LINEAR_ACCELERATION);
sensorManager.registerListener(new
    AccelerometerSensorListener(),accelerometer_sensor,
    SensorManager.SENSOR_DELAY_GAME);

/**
 * set the listener to the gyroscope sensor
 */
Sensor gyroscope_sensor =
    sensorManager.getDefaultSensor(Sensor.TYPE_GYROSCOPE);
sensorManager.registerListener(new
    GyroscopeSensorListener(),gyroscope_sensor,
    SensorManager.SENSOR_DELAY_GAME);
```

```
        /**
         * set the WiFi manager to collect the AP info
         */
        wifiManager = (WifiManager)
            getSystemService(Context.WIFI_SERVICE);

        IV_compass = (ImageView) findViewById(R.id.IV_compass);
        Enable = (CheckBox) findViewById(R.id.CB_calibration);

        /**
         * set the Training number spinner
         */
        view = (TextView) findViewById(R.id.spinnerText);
        TrainingNumSpinner = (Spinner)
            findViewById(R.id.SP_trainingNum);

        adapter = new ArrayAdapter<String>(this,
            android.R.layout.simple_spinner_item, TrainingNum);
        adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_

        TrainingNumSpinner.setAdapter(adapter);
        TrainingNumSpinner.setOnItemSelectedListener(new
            SpinnerSelectedListener());
        TrainingNumSpinner.setVisibility(View.VISIBLE);

    }
```

```
/**
 * SpinnerSelectedListener
 *
 * @author Guanxiong Liu
 *
 */
private class SpinnerSelectedListener implements
    OnItemSelectedListener {

        @Override
        public void onItemSelected(AdapterView<?> arg0, View arg1,
            int arg2, long arg3) {
                TrainingPoint = TrainingNum[arg2];
                view.setText("For Training Point" + TrainingPoint);
        }

}

/**
 * OrientationSensorListener: Measure the angle between the y axis
    and the
 * North
 *
 * @author Guanxiong Liu
 *
 */
```

```
private class OrientationSensorListener implements
    SensorEventListener {
        @Override
        public void onSensorChanged(SensorEvent arg0) {


                float[] value = arg0.values;


                if (Enable.isChecked()) {
                        if (angle > 360) {
                                angle = value[0];
                        } else {
                                if (Math.abs(angle - value[0]) < 80) {
                                        angle_buffer[last] = value[0];
                                        last++;
                                        if (last == 20) {
                                                float sum = 0;
                                                for (float temp :
                                                    angle_buffer) {
                                                        sum += temp;
                                                }
                                                angle = sum / 20;
                                                last = 0;
                                        }
                                } else {
                                        angle = value[0];
                                        last = 0;
                                }
```

```
                }
            } else {
                angle = value[0];
            }


            compass_angle = value[0];
            RotateAnimation rotate = new
                RotateAnimation(preAngle, -compass_angle,
                RotateAnimation.RELATIVE_TO_SELF, (float) 0.5,
                RotateAnimation.RELATIVE_TO_SELF, (float) 0.5);
            rotate.setDuration(100);
            IV_compass.setAnimation(rotate);
            preAngle = -compass_angle;
        }
}

/**
 * AccelerometerSensorListener: Used to detect the Z axis and Y
    axis
 * accelerometer, also using the step detection with Z axis
    accelerometer
 *
 * @author Guanxiong Liu
 *
 */
```

```
private class AccelerometerSensorListener implements
    SensorEventListener {

        @Override
        public void onAccuracyChanged(Sensor sensor, int accuracy)
            {
                // TODO Auto-generated method stub

        }

        @Override
        public void onSensorChanged(SensorEvent event) {
                float[] value = event.values;

                linear_acceleration = value[2];
                accelerate_y = value[1];

                if (SystemClock.elapsedRealtime() - timeout >= 400
                    && record) {
                        if (ZAxis.max() - linear_acceleration >= 1.4
                            && record) {
                                /**
                                 * write record data data type:
                                     [theta omiga a time_stamp]
                                 */
                                stepNum += 1;
```

```
                                TimeStamp =
                                    SystemClock.elapsedRealtime();
                                recordData.add(String.valueOf(angle)
                                    + " " + String.valueOf(stepNum) +
                                    " " +
                                    String.valueOf(accelerate_y) + "
                                    " +
                                    String.valueOf(MainActivity.TimeStamp));
                                recordData.add("\r\n");

                                new ScanWifi().run();

                                timeout =
                                    SystemClock.elapsedRealtime();
                            }
                            ZAxis.enqueue(linear_acceleration);
                        }
                    }

}

private static class ZAxis {
        private static int head = 0;
        private static float[] data = new float[5];
        private static int size = 0;

        public static void enqueue(float in) {
```

```
                if (size < 5) {
                        data[size] = in;
                        size++;
                } else {
                        data[head] = in;
                        head++;
                        head = head % 5;
                }
        }

        public static float max() {
                float max = 0;
                for (int i = 0; i < size; i++) {
                        if (data[(head + i) % 5] > max) {
                                max = data[(head + i) % 5];
                        }
                }
                return max;
        }
    }

}
```

Appendix B

Full Publication List

B.1 Related to this Thesis

1. Guanxiong Liu, Yishuang Geng and Kaveh Pahlavan, "Effects of Calibration RFID Tags on performance of Inertial Navigation in Indoor Environment", International Conference on Computing, Networking and Communications (ICNC) 2015, Feb 2015, Anaheim, California, USA

2. Guanxiong Liu, Yishuang Geng and Kaveh Pahlavan, "Direction Estimation Error Model of Embedded Magnetometer in Indoor Environment", International Workshop on Wireless Sensor Network and Positioning Services (WSNPS) 2015, Aug 2015, Beijing, China

3. Luyao Niu, Yingyue Fan, Kaveh Pahlavan, Guanxiong Liu and Yishuang Geng, "On the Accuracy of Wi-Fi Localization using Robot and Human Collected Signatures", IEEE 26th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC) 2015

4. Guanxiong Liu, Yishuang Geng and Kaveh Pahlavan, "A Distance Dependent Shadow Fading Model for Wireless Geolocation in Indoor Environment", IEEE 82nd Vehicular Technology Conference (VTC) 2015 Fall, Sep 2015, Boston, Massachusetts, USA

B.2 Other Publication

1. Dan Liu, Yishuang Geng, Guanxiong Liu, Mingda Zhou and Kaveh Pahlavan, "WBANs-Spa: An Energy Efficient Relay Algorithm for Wireless Capsule Endoscopy", IEEE 82nd Vehicular Technology Conference (VTC) 2015 Fall, Sep 2015, Boston, Massachusetts, USA

Bibliography

[1] Jie He, Qin Wang, Qianxiong Zhang, Bingfeng Liu, and Yanwei Yu. A practical indoor toa ranging error model for localization algorithm. In *International Symposium on Personal Indoor and Mobile Radio Communications*, 2011.

[2] Kaveh Pahlavan, Xinrong Li, and J-P Makela. Indoor geolocation science and technology. *IEEE Communications Magazine*, 2002.

[3] Gan Yan, Yuxiang Lv, Qiyin Wang, and Yishuang Geng. Routing algorithm based on delay rate in wireless cognitive radio network. *Journal of Networks*, 2014.

[4] Ismail Guvenc and Chia-Chin Chong. A survey on toa based wireless localization and nlos mitigation techniques. *IEEE Communications Surveys & Tutorials*, 2009.

[5] Fernando Seco, A Jimenez, Carlos Prieto, Javier Roa, and Katerina Koutsou. A survey of mathematical methods for indoor localization. In *IEEE International Symposium on Intelligent Signal Processing*, 2009.

[6] Neal Patwari, Joshua N Ash, Spyros Kyperountas, Alfred O Hero, Randolph L Moses, and Neiyer S Correal. Locating the nodes: cooperative localization in wireless sensor networks. *IEEE Journal on signal processing magazine*, 2005.

[7] Quentin H Spencer, Brian D Jeffs, Michael A Jensen, and A Lee Swindlehurst. Modeling the statistical time and angle of arrival characteristics of an indoor multipath channel. *IEEE Journal on Selected Areas in Communications*, 2000.

[8] KC Ho and YT Chan. Solution and performance analysis of geolocation by tdoa. *IEEE Transactions on Aerospace and Electronic Systems*, 1993.

[9] Ahmad Hatami and Kaveh Pahlavan. A comparative performance evaluation of rss-based positioning algorithms used in wlan networks. In *IEEE Wireless Communications and Networking Conference*, 2005.

[10] Kamol Kaemarungsi and Prashant Krishnamurthy. Modeling of indoor positioning systems based on location fingerprinting. In *Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, 2004.

[11] F. Evennou and F. Marx. Advanced integration of wifi and inertial navigation systems for indoor mobile positioning. *Eurasip journal on applied signal processing*, 2006.

[12] R. Harle. A survey of indoor inertial positioning systems for pedestrians. *IEEE Communications Surveys & Tutorials*, 2013.

[13] Kaveh Pahlavan and Prashant Krishnamurthy. *Networking fundamentals: wide, local and personal area communications*. John Wiley & Sons, 2009.

[14] Yongtao Ma, Liuji Zhou, Kaihua Liu, and Jinlong Wang. Iterative phase reconstruction and weighted localization algorithm for indoor rfid-based localization in nlos environment. *Sensors Journal, IEEE*, 2014.

[15] Deliang Liu, L Kaihua, Yongtao Ma, and Jiexiao Yu. Joint toa and doa localization in indoor environment using virtual stations. *IEEE communication letters*, 2014.

[16] Yang Zhao, Kaihua Liu, Yongtao Ma, and Zhuo Li. An improved k-nn algorithm for localization in multipath environments. *EURASIP Journal on Wireless Communications and Networking*, 2014.

[17] Guanxiong Liu, Yishuang Geng, and Kaveh Pahlavan. Effects of calibration rfid tags on performance of inertial navigation in indoor environment. In *International Conference on Computing, Networking and Communications*, 2015.

[18] Yishuang Geng and Kaveh Pahlavan. Enlighten wearable physiological monitoring systems: On-body rf characteristics based human motion classification using a support vector machine. *IEEE Transactions on Mobile Computing*, 2015.

[19] C. Lukianto and H. Sternberg. Steppingsmartphone-based portable pedestrian indoor navigation. *Archiwum Fotogrametrii, Kartografii i Teledetekcji 22*, 2011.

[20] C Ascher, C Kessler, M Wankerl, and GF Trommer. Dual imu indoor navigation with particle filter based map-matching on a smartphone. In *International conference on Indoor positioning and indoor navigation*, 2010.

[21] A. Serra, D. Carboni, and V. Marotto. Indoor pedestrian navigation system using a modern smartphone. In *International conference on Human computer interaction with mobile devices and services*, 2010.

[22] T. Konrad and P. Wolfel. Wifi compass: Wifi access point localization with android devices. Master's thesis, St. Polten University of Applied Sciences, 2012.

[23] Google. *Android 2.2 APIs: public abstract class SensorManager*.

[24] Google. *Android 2.2 APIs: Sensors Overview.*

[25] Google. *Android 2.2 APIs: Position Sensors.*

[26] Yishuang Geng and Kaveh Pahlavan. On the accuracy of rf and image processing based hybrid localization for wireless capsule endoscopy. In *IEEE Wireless Communications and Networking Conference (WCNC)*, 2015.

[27] Yongtao Ma, Kaveh Pahlavan, and Yishuang Geng. Comparison of poa and toa based ranging behavior for rfid application. In *2014 IEEE 25th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2014.

[28] Jie He, Yishuang Geng, and Kaveh Pahlavan. Modeling indoor toa ranging error for body mounted sensors. In *2012 IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2012.

[29] R. Ouyang and A. Wong. Received signal strength-based wireless localization via semidefinite programming: Noncooperative and cooperative schemes. *IEEE Transactions on Vehicular Technology*, 2010.

[30] Jie He, Yishuang Geng, Fei Liu, and Cheng Xu. Cc-kf: Enhanced toa performance in multipath and nlos indoor extreme environment. *IEEE Sensors Journal*, 2014.

[31] Bird, Jeff, and Dale Arden. Indoor navigation with foot-mounted strapdown inertial navigation and magnetic sensors [emerging opportunities for localization and tracking]. *Wireless Communications, IEEE*, 2011.

[32] Y. Ma, L. Zhou, K. Liu, and J. Wang. Iterative phase reconstruction and weighted localization algorithm for indoor rfid-based localization in nlos environment. *IEEE Sensors Journal*, 2014.

[33] Lionel M Ni, Yunhao Liu, Yiu Cho Lau, and Abhishek P Patil. Landmarc: indoor location sensing using active rfid. *Wireless networks*, 2004.

[34] Antonio Ramón Jiménez Ruiz, Fernando Seco Granja, Jose Carlos Prieto Honorato, and Jorge I Guevara Rosas. Accurate pedestrian indoor navigation by tightly coupling foot-mounted imu and rfid measurements. *IEEE Transactions on Instrumentation and Measurement*, 2012.

[35] JR Guerrieri, MH Francis, PF Wilson, T Kos, LE Miller, NP Bryner, DW Stroup, and L Klein-Berndt. Rfid-assisted indoor localization and communication for first responders. In *First European Conference on Antennas and Propagation*, 2006.

[36] Bilal Hameed, Farhan Rashid, Frank Dürr, and Kurt Rothermel. Self-calibration of rfid reader probabilities in a smart real-time factory. In *Pervasive Computing*, 2012.

[37] S. Razavi and C. Haas. Using reference rfid tags for calibrating the estimated locations of construction materials. *Automation in Construction*, 2011.

[38] I. Sharp, K. Yu, and M. Hedley. On the gdop and accuracy for indoor positioning. *IEEE Transactions on Aerospace and Electronic Systems*, 2012.

[39] Yishuang Geng, Jie He, and Kaveh Pahlavan. Modeling the effect of human body on toa based indoor human tracking. *International Journal of Wireless Information Networks*, 2013.

[40] Degui Zeng and Yishuang Geng. Content distribution mechanism in mobile p2p network. *Journal of Networks*, 2014.

[41] C. Lukianto, C. Honniger, and H. Sternberg. Pedestrian smartphone-based indoor navigation using ultra portable sensory equipment. In *Indoor Positioning and Indoor Navigation (IPIN), 2010 International Conference on, vol., no*, 2010.

[42] Wonho Kang, Seongho Nam, Youngnam Han, and Sookjin Lee. Improved heading estimation for smartphone-based indoor positioning systems. In *International Symposium on Personal Indoor and Mobile Radio Communications*, 2012.

[43] Fan Li, Chunshui Zhao, Guanzhong Ding, Jian Gong, Chenxing Liu, and Feng Zhao. A reliable and accurate indoor localization method using phone inertial sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, 2012.

[44] Jie He, Yishuang Geng, Yadong Wan, Shen Li, and Kaveh Pahlavan. A cyber physical test-bed for virtualization of rf access environment for body sensor network. *IEEE Sensors Journal*, 2013.

[45] Xinchao Song and Yishuang Geng. Distributed community detection optimization algorithm for complex networks. *Journal of Networks*, 2014.

[46] Teemu Roos, Petri Myllymäki, Henry Tirri, Pauli Misikangas, and Juha Sievänen. A probabilistic approach to wlan user location estimation. *International Journal of Wireless Information Networks*, 2002.

[47] Eiman Elnahrawy, Xiaoyan Li, and Richard P Martin. The limits of localization using signal strength: A comparative study. In *Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004. 2004 First Annual IEEE Communications Society Conference on*, 2004.

[48] Jie He, Yishuang Geng, and Kaveh Pahlavan. Toward accurate human tracking: modelling time-of-arrival for wireless wearable sensors in multipath environment. *IEEE Sensors Journal*, 2014.

[49] Davide Dardari, Andrea Conti, Ulric Ferner, Andrea Giorgetti, and Moe Z Win. Ranging with ultrawide bandwidth signals in multipath environments. *Proceedings of the IEEE*, 2009.

[50] Yishuang Geng, Yadong Wan, Jie He, and Kaveh Pahlavan. An empirical channel model for the effect of human body on ray tracing. In *2013 IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2013.

[51] Yishuang Geng, Jie He, Haokun Deng, and Kaveh Pahlavan. Modeling the effect of human body on toa ranging for indoor human tracking with wrist mounted sensor. In *2013 16th International Symposium on Wireless Personal Multimedia Communications (WPMC)*, 2013.

[52] Yishuang Geng, Jin Chen, and Kaveh Pahlavan. Motion detection using rf signals for the first responder in emergency operations. In *2013 IEEE 24th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2013.

[53] Jie He, Yishuang Geng, Cheng Xu, and al. et. Height dependent toa ranging error model for near ground localization applications. In *2014 IEEE 25th International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, 2014.

[54] Neal Patwari, Alfred O Hero, Matt Perkins, Neiyer S Correal, and Robert J O'dea. Relative location estimation in wireless sensor networks. *Signal Processing, IEEE Transactions on*, 2003.

[55] Guanxiong Liu, Yishuang Geng, and Kaveh Pahlavan. Direction estimation error model of embedded magnetometer in indoor environment. In *International Workshop on Wireless Sensor Network and Positioning Services*, 2015.