Towards a large scale indoor localization service with crowdsensing indoor map generation

Zhang, Chi

2014

Zhang, C. (2014). Towards a large scale indoor localization service with crowdsensing indoor map generation. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/65068>

<https://doi.org/10.32657/10356/65068>

Towards A Large Scale Indoor Localization Service with Crowdsensing Indoor Map Generation

ZHANG CHI

School of Computer Engineering

Nanyang Technological University

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy (Ph.D)

2014

To my family, for their unconditional love and endless support.

Acknowledgement

First and foremost I offer my deepest thanks to my supervisor, Prof. Jun Luo, for his support, guidance, encouragement, and patience. It has been a great pleasure working with him, whose personality inspired me a lot as well as his knowledge did.

Special thanks to my co-authors, Prof. Jianxin Wu, Prof. Kalyan P. Subbu, Prof. Ying He, Dr. Liu Xiang, Dr. Feng Li for their invaluable suggestions, and for our fruitful collaborations.

My deepest gratitude goes to my family for their continuous love and support throughout my life. I owe to them every step of my progress.

I would also like to express my sincere thanks to my friends, for their lovely company and encouragement. I always appreciate the precious moments we shared together.

Last but not least, I thank my colleagues and technicians in CeMNet for their patient and kind help.

Abstract

Knowing self location matters a lot in people’s daily life. While Global Positioning System (GPS) provides almost perfect solution for outdoor area, it would not work in indoor areas because of no line-of-sight to satellites. However, since human tend to spend more and more time in complexly constructed buildings, helping people localize themselves in indoor space become a critical problem. Raising localization accuracy and reducing deployment cost are two main objects in indoor localization problem. High localization accuracy ensures usability of service, while low deployment cost lessens the effort people must take to use localization service. Numerous technologies have been proposed to tackle the problem. However, practical indoor localization that can provide high localization accuracy with minimum cost is still a vacancy.

In the first part of this thesis, we devote ourself to explore the possibility of fingerprint based localization. Although a large number of fingerprinting based indoor localization systems have been proposed, our field experience with Google Maps Indoor (GMI), the only system available for public testing, shows that it is far from mature for indoor navigation. Motivated by the obtained insights from field studies with GMI, we propose GROPING as a self-contained indoor navigation system independent of any infrastructural support. GROPING relies on Ambient Magnetic Field fingerprints, which is formed by “twisted” geomagnetic field by building structures, that are far more stable than WiFi fingerprints, and it exploits crowdsensing to construct floor maps rather than expecting individual venues to supply digitized maps. Based on our experiments with 20 participants in various floors of a big shopping mall, GROPING is able to deliver a sufficient accuracy for localization and thus provides smooth navigation experience.

In our experiments with ambient magnetic field fingerprint, we see that scalability of ambient magnetic field based approach is not satisfactory comparing to WiFi based approaches. By further exploration, we find that dual properties naturally existed in ambient magnetic field fingerprint and WiFi fingerprint. Therefore based on GROPING, we present MaWi - a

---

dual-sensor enabled indoor localization system in the second part of this thesis. Central to MaWi is a novel framework combining two self-contained but complementary localization techniques: Wi-Fi and Ambient Magnetic Field. Combining the two techniques, MaWi not only achieves a high localization accuracy, but also effectively reduces human labor in building fingerprint databases: to avoid war-driving, MaWi is designed to work with low quality fingerprint databases that can be efficiently built by only one person. Our experiments demonstrate that MaWi, with a fingerprint database as scarce as one data sample at each spot, outperforms the state-of-the-art proposals working on a richer fingerprint database.

Although MaWi is designed to use minimum human effort to collect fingerprints, the initial spot surveying is still an inevitable burden for all fingerprint-based localization systems. To ultimately reduce human effort in initial phase, we try to find a localization solution in a model-based methodology. In the last part of this thesis, we focus on exploiting “multipath” phenomenon in wireless signal propagation and utilize it to fully or partially reconstruct the geometry of the indoor space, as well as locate signal source. Whereas a few physical layer techniques have been proposed to locate a signal source indoors, they all deem multipath a “curse” and hence take great efforts to cope with it. We, on the contrary, deem multipath a “blessing” and thus innovatively exploit the power of it. Essentially, with minor assumption (or knowledge) of the geometry of an indoor space, each signal path may potentially contribute a new piece of information to the location of its source. As a result, it is possible to locate the source with very few sensors (most probably just one hand-held device). At the same time, the extra information provided by multipath effect can help to fully or partially reconstruct the geometry of the indoor space, which enables a floor plan generation process missing in most of the indoor localization systems. To demonstrate these ideas, we instrument a USRP-based radio sensor prototype named iLocScan; it can simultaneously scan an indoor space (hence generate a plan for it) and position the signal source in it. Through iLocScan, we mainly aim to showcase the feasibility of harnessing multipath in assisting indoor localization, rather than to rival existing proposals in terms of localization accuracy. Nevertheless, our experiments show that iLocScan can offer satisfactory results on both source localization and space scanning.

Keywords

Indoor Localization, Indoor Navigation, Geomagnetism, Mobile Crowdsensing, Floor Plan Generation

Contents

Acknowledgement	iii
Abstract	iv
List of Figures	x
List of Tables	xiii
1 Introduction	1
1.1 Background	1
1.2 Scope of Research	3
1.3 Contribution	4
2 Literature Review	5
2.1 Fingerprint Based Localization	5
2.1.1 Fingerprint Quality	5
2.1.2 Spot Survey and Crowd-sourcing	6
2.2 Model-based Localization	7
2.3 Dead-reckoning Localization	7
2.4 Device Free Localization	8
2.5 Summary	9
3 GROPING: Geomagnetism and cROwdsensing Powered Indoor NaviGation	10
3.1 Introduction	10
3.2 Studies on Google Maps Indoor	12
3.2.1 A Field Study on GMI	12
3.2.2 Energy Efficiency Evaluation	15

3.2.3	Summary	15
3.3	GROPING System Overview	16
3.3.1	Map Building	17
3.3.2	Localization and Navigation	19
3.3.3	User Interface	19
3.4	System Components	22
3.4.1	Map Builder: A Joint Venture of The Crowd	22
3.4.2	Location Estimator: A Bayesian Approach	27
3.4.3	Navigation: Client–Server Interactions	31
3.5	System Evaluation Design	32
3.5.1	Experiment Setting	33
3.5.2	Incentives for Crowdsensing	33
3.6	Experiment Results	34
3.6.1	Why GROPING is Needed	34
3.6.2	Efficiency of GROPING Map Construction	35
3.6.3	Accuracy of GROPING Location Estimation	36
3.6.4	User Feedback on GROPING Navigation	38
3.7	More on The Map Generation	39
3.8	More on The Ambient Magnetic Field	44
3.9	Summary	46
4	A Dual-Sensor Enabled Indoor Localization System with Crowdsensing Spot Survey	47
4.1	Introduction	47
4.2	Fingerprint Studies	49
4.2.1	Wi-Fi Signal	49
4.2.2	Ambient Magnetic Field	52
4.3	Zone Discrimination	56
4.3.1	Problem Formulation	56
4.3.2	Circle Algorithm (CA): A Naive Approach	57
4.3.3	Similarity Voronoi Algorithm (SVA)	58
4.3.4	Comparing CA with SVA	58
4.4	A Hybrid Localization Approach	59

4.4.1	How to Utilize Fingerprints?	59
4.4.2	Revised Particle Filter: A Basic Version	60
4.4.3	Particle Updating with Zone	61
4.4.4	Confidence Coefficient	62
4.4.5	Particle Re-Sampling	63
4.4.6	Algorithm Description	63
4.5	MaWi System Overview	63
4.5.1	Spot Survey	64
4.5.2	Redundant Data	65
4.5.3	Data Flow Patterns	66
4.6	Evaluation	66
4.6.1	Experiment Setup	66
4.6.2	Light-Weight Survey Phase	67
4.6.3	Further Evaluations on SVA	67
4.6.4	Performance Comparisons	68
4.6.5	Localization Time Limit	70
4.6.6	Wi-Fi Access Point Number	70
4.7	Summary	71
5	iLocScan: Harnessing Multipath for Simultaneous Indoor Source Localization and Space Scanning	73
5.1	Introduction	73
5.2	Applications and System Architecture	76
5.2.1	Finding Signal Sources Indoors	76
5.2.2	Our Solution: iLocScan	78
5.3	ACMod – Measuring All AoAs At Once	82
5.3.1	Preliminary on AoA Estimation	82
5.3.2	Fine-Tuning the MUSIC Algorithm	83
5.3.3	Antenna Patterns	85
5.4	ALMod – AoA Reasoning	88
5.4.1	The Case of A Rectangular Area	88
5.4.2	A Rectangular Area with Open Side(s)	90
5.4.3	The L-Shaped Area and Beyond	90

5.5	LSMod – Autonomous Scenario Modeling and Problem Solving	91
5.6	Implementation Details	93
5.6.1	A Multi-Input Radio System	93
5.6.2	Phase Calibration	94
5.6.3	Detecting WiFi Preamble	95
5.7	Experimental Evaluations	96
5.7.1	Experiment Settings	96
5.7.2	A Concrete Example	97
5.7.3	Accuracy Evaluations	98
5.7.4	AoA Detection under Varying TX Powers	100
5.7.5	Reflections on Different Materials	100
5.8	Summary	102
6	Conclusion	104
6.1	Research Contribution	104
6.2	Future Directions	105
	References	107
A	Apply MUSIC to WiFi Signals	115

List of Figures

3.1	Screenshots taken on GMI.	13
3.2	GMI localization errors in five shopping malls.	14
3.3	The number of satisfied and unsatisfied cases for 11 users.	14
3.4	Energy consumption comparisons.	16
3.5	GROPING system architecture.	17
3.6	Floor map of a shopping mall and the walking trajectories contributed by Alice and Bob.	18
3.7	Virtual map generation using three trajectories and the associated magnetic fingerprints.	18
3.8	The user interface of a GROPING client.	20
3.9	More about GROPING client user interface.	21
3.10	Three components of a floor plan.	23
3.11	Segmenting a trajectory using gyroscope reading g_y	24
3.12	The same ambient magnetic field sensed by different smartphones.	30
3.13	The outcome of AP clustering.	30
3.14	Types of landmarks remembered by people.	35
3.15	Floor maps and the corresponding GROPING maps of 3-floor shopping mall.	36
3.16	Localization accuracy of GROPING. Comparisons are made in a large test site (b) and a small one (c).	37
3.17	Snapshots of the GROPING navigation.	39
3.18	Navigation satisfactory level from 1 to 10.	39
3.19	The floor map of a shopping mall. The slight difference between this map and that shown in Fig. 3.15 is caused by a renovation in-between these two sets of experiments.	40

3.20 Virtual map generation using eight trajectories and the associated magnetic fingerprints.	42
3.21 Statistical evaluation of the map generation performance.	43
3.22 Ambient magnetic field reading, in μ -Tesla (or μ T), taken by a smartphone at five different locations. For each location, we use West, South, East, and North to indicate the quadrants (centered at the smartphone) within which the “interfering” person is moving.	45
4.1 Test region for fingerprint studies.	50
4.2 Fingerprint similarity with one sample per spot given different numbers of APs.	51
4.3 Wi-Fi locational discrimination.	51
4.4 Localization error CDF.	52
4.5 Temporal stability comparison.	53
4.6 Magnetic field confusion matrix.	54
4.7 Locational fingerprint similarity.	55
4.8 Magnetic signal sequences collected by a walking user holding smart phone in hand in two corridors.	56
4.9 Wi-Fi spacial discrimination.	59
4.10 $\mathcal{A}$ of SVA and CA when $p = 0.8$ and safety is tuned to 0.8. Appearing frequency of fingerprint is plotted in background.	60
4.11 MaWi Localization Illustration.	61
4.12 System architecture.	65
4.13 Floor plans of a library (a) and a shopping mall (b).	66
4.14 MaWi localization on a client. Sizes of red dots denote particles’ weights, and time sequence is indicated at the upper-left corner of each figure. Obviously, the weight eventually concentrates at a single particle, which happens to be the user’s ground truth location.	67
4.15 SVA under different p and θ . Best result of SVA is achieved when θ changes linearly with p . At the same time Safety and Effectiveness are both relative high.	69
4.16 Localization Error under different p and θ	70
4.17 Comparing localization accuracy among MaWi, pure magnetic field, and RADAR.	70
4.18 Comparing localization accuracy between MaWi and Horus under fingerprint data of different quality.	71

4.19	Localization error evaluations	71
5.1	Multipath contains useful information.	74
5.2	Searching for a person-of-interest indoors	77
5.3	iLocScan architecture.	79
5.4	Four antenna patterns.	80
5.5	Indoor space model for iLocScan.	81
5.6	Detecting the AoA of a signal path using a two-antenna array. The antennas are aligned with the x -axis, while the y -axis indicates the forward direction, i.e., the detected AoA represents the signed angle counter-clockwise from y -axis to the signal's steering direction.	83
5.7	Comparing two angle spectra.	85
5.8	Circular antenna array. The blue points denoting the antennas are uniformly placed on a circle with radius R . The signal AoA is θ	86
5.9	Angle spectra of different arrays.	87
5.10	Accuracy of angle measurement by different arrays.	88
5.11	AoA patterns. We denote iLocScan and the signal source by blue dot and red star, respectively. We also mark the coordinate system of the antenna array by red arrows, and the detected AoAs by green arrows.	90
5.12	L-shaped indoor structure.	91
5.13	Illustrating automatic problem formulation.	92
5.14	System schematic and outlook of iLocScan.	95
5.15	An illustration of iLocScan evaluation. This set of experiments is conducted in an 800 m ² research lab. We fix three signal sources at location T1 , T2, T3. Our iLocScan chooses three spots to perform AoA measures. Each column of the figures corresponds to one iLocScan spot. The top row marks the measurement spots and also shows the estimated source locations and the floor plans. Another three rows plot the AoA measures for individual signal sources.	97
5.16	Evaluation of measurement accuracy.	99
5.17	More spots yield higher chance of locating a source.	99
5.18	More observations lead to higher accuracy.	100
5.19	Angle spectra under different tx powers of the signal source.	101
5.20	Reflections on concrete wall, glass window, and metal board.	102

List of Tables

3.1	User perceptions on indoor navigation solutions (average $\pm$ standard deviation).	34
3.2	Map construction times with different number of explorers.	35
4.1	Test Sites and Survey Times	68
5.1	Angle Resolution of Different Antenna Arrays.	86

Chapter 1

Introduction

Since its inception in early this century, indoor localization has been one of the most important research topics in wireless system community. This is obviously driven by need from our real-life experience: finding where other people are and even oneself in large scale indoor facilities (e.g., shopping malls or airport terminals) is becoming increasingly difficult due to the ever growing of our lives and thus their facilities. On one hand, due to the large indoor spaces and the amazing number of buildings constructed and will be constructed, scalability is an important issue that practical indoor localization solution should address. On the other hand, high localization accuracy is necessary in many kinds of indoor applications, such as shopping mall navigation. These challenging aspects have been and are stimulating research process in this field.

1.1 Background

As increasing urbanization forces people to stay more often at indoor environments, locating and navigating people in complex structures (e.g., airports and shopping malls) becomes a critical problem. Furthermore, government and business also benefit from accurate user location information in precise information pushing. However, one promise to locating people in indoor spaces is the existing of floor plan. In most indoor applications, localization is meaningless if we do not have floor plan of the indoor areas. Although some public buildings supply their floor plan, digitizing those floor plan is still a big workload to indoor localization service suppliers. Another cost before localization lies in initialization process which is necessary for most existing localization services. For example, fingerprint based approaches [1, 2] recruit

spot surveyor to collect fingerprints around the areas where localization service is supplied; model-based systems [3, 4, 5, 6] require deployment of dedicated devices acting as anchors in specific locations. Considering the size of indoor areas that need to be covered, we see the cost hinders the deployment of any practical indoor localization services.

The very natural metric to evaluate a localization system is localization accuracy, which is the difference between ground-truth and the location given by localization systems. High localization accuracy guarantees localization system’s applicability under special application scenarios, but usually requires more human work and/or resources comparing to lower accuracy systems. Therefore how to get tolerable localization result with affordable deployment cost become the central problem that recent proposed localization approaches try to deal with. Because spot survey (deemed necessary for achieving high accuracy in some fingerprint-based systems) hampers scalable deployments, recent proposals [7, 8, 9] suggest to distribute the intensive labors through crowd-sourcing, in which fingerprint databases are collected opportunistically by a large population. Whereas these technologies may have potential to make large deployment possible, opportunistic spot survey cannot warrant fingerprint quality: insufficient samples at a given spot and low density sampling spots remain threat to localization accuracy.

Recently, another kind of approaches using physical layer information to improve ranging-trilateration has become a new trend [10, 11, 12]. In particular, both ArrayTrack [11] and CUPID [12] apply an array of antennas to estimate the Angle-of-Arrival (AoA) of the direct path, while CUPID [12] takes one step further by using Channel State Information (CSI) to get more accurate estimation of the path length. Synthesizing the estimations (AoAs or even path lengths) from a few sensors would allow for an accurate location estimation for the signal source. Interestingly, both ArrayTrack and CUPID treat reflection signal paths as “noises” and make great efforts to remove them, while they in fact contain valuable information. Whereas the AoA of the direct path (θ_d) is what is sought by both ArrayTrack and CUPID, the AoA of the reflection path (θ_r), which was thrown away by the existing approaches, indicates the locations of a mirrored image of the signal source with respect of one wall. Given a known distance from the sensor (an antenna array)¹ to the wall, the source location can be estimated with θ_d and θ_r . In reality, multiple reflection paths do exist in an indoor space. Exploiting all these paths may allow us to learn not only the location of the source but also the geometry of

---

¹“sensor” and “antenna array” are used interchangeably in this thesis, as they refer to the same thing in our context.

the space, whereas this latter piece of information is missing in almost all indoor localization systems: a floor plan often needs to be known in advance.

1.2 Scope of Research

In this thesis, our focus lies in inventing scalable mobile indoor localization system that produces high locating accuracy. To this aim, I propose three approaches with different localization methodology: GROPING and MaWi base on fingerprint matching; iLocScan bases on multi-path signal propagation. All the three systems can be initiated with very low cost, including generate floor plan of deployment area, and accurately locate user in the floor plan.

Geomagnetism and cROwdsensing Powered Indoor NaviGation (GROPING) is proposed as a completely smart phone based self-contained, lightweight, and practical prototype for indoor localization/navigation. GROPING encapsulates three functions, namely *map building*, *localization* and *navigation*, into one unit. It first builds a map using user contributed sensor data and semantic labels; it then performs localization based on the magnetic fingerprints, and finally it runs a navigation service on top of these two functions: it computes navigational routes using the early constructed map and the real-time location information. In this way, GROPING eliminates *infrastructure dependence*: it needs neither wireless infrastructure nor digitized floor maps. Our intensive experiments with GROPING demonstrate its usability and also show that it compares favorably with typical WiFi-based localization systems in supporting indoor navigation.

In our experiments with ambient magnetic field fingerprint, we find that ambient magnetic field based localization does not perform well enough in large scale deployment areas as it performs in small scale areas. By further exploration, we find that dual properties naturally existed in ambient magnetic field fingerprint and WiFi fingerprint. Therefore on the basis of GROPING, we propose MaWi, a smart phone based indoor localization system relying on both magnetic field and Wi-Fi as fingerprints. These two fingerprints are used in a “duet” manner such that they complement each other. Through this smart combination, MaWi achieves a scalable deployment due to its low demand on the fingerprint database, while getting very competitive localization accuracy compared to state-of-the-art systems. We analyze properties of magnetic field and Wi-Fi fingerprints and we identify their complementarity in indoor localization.

In experiments, Wi-Fi signal is not stable enough for accurate localization. However, the multipath features in Wi-Fi signal propagation can be actually put into use in assisting indoor localization. Basing on this, we construct an antenna array system, iLocScan. iLocScan simultaneously samples the signals from its multiple antennas, and the samples are infused into a computation module running a fine-tuned version of MUSIC [13], in order to obtain a set of estimated AoAs (including both direct path and multiple reflection paths). iLocScan then uses a logic module to i) tell which AoA belongs to which path, if a sufficient number of AoAs have been gathered, and otherwise ii) suggest a possible new location to gather more AoAs. Finally, all the acquired information is put together to form a least squares problem that computes the estimations of various variables (including both source location and space geometry) as those best fitting the known parameters.

1.3 Contribution

In this thesis, the following contributions have been made:

- GROPING is innovatively proposed as a completely smart phone based self-contained, lightweight, and practical prototype for indoor localization/navigation.
- We analyze properties of magnetic field and Wi-Fi fingerprints and we, for the first time, identify their complementarity in indoor localization, and present a scalable indoor localization system, MaWi.
- We design a system, iLocScan, that can simultaneously locate a signal source and sketch the plan of the floor where the source is located at; we engineer iLocScan to exploit the power of multipath rather than to simply avoid it, which enables us to utilize far more information embedded in the radio signals propagating indoors.

Chapter 2

Literature Review

When human build more and more large indoor construction where we spend most of our time, indoor localization become a pressing problem to solve. In order to achieve perfect indoor localization technology like GPS in outdoor area, numerous approaches are proposed by researchers around the world. We can divided them into four kinds: fingerprint-based approaches, model-based approaches, dead-reckoning approaches, and device free approaches. In this chapter, we will review the representative researches in the four classes proposed in last decade.

2.1 Fingerprint Based Localization

A Fingerprint based approach (a.k.a. empirical approach) [1, 2, 7, 8, 9, 14, 15, 16, 17, 18, 19, 20] collects fingerprints around the deployment area in the initial phase to assist later (empirical) localization. Generally speaking, the central idea of fingerprint based localization approaches is matching data collected at unknown location to a fingerprint database and find a best matched entry, which indicates user's location. Fingerprint's quality, in other words, how the fingerprint can uniquely represent locations, is crucial to performance of a fingerprint based approach.

2.1.1 Fingerprint Quality

Theoretically, all data that can be collected by mobile sensors can be used as localization fingerprint. But some kinds of fingerprint can identify locations better than others. For example, cellular signal [14], Wi-Fi signal [1, 16], ambient sound [15], light [21], and magnetic fields

[19] are considered as fingerprint that can uniquely identify locations and are utilized in recent indoor localization systems. There are two properties that need to be considered when we judge whether a fingerprint is eligible for localization: temporary stability and locational discrimination. Because fingerprint database are previously collected before we can use it to locate users, the fingerprints collected at the same location at different time must be constant, or at least similar, which is called temporary stability. To accurately locate object in a large area, fingerprint collected from each location should be universally unique, in other words, different locations must generate different fingerprints. This property is called locational discrimination. The most popular fingerprint that used in recent indoor localization systems is Wi-Fi signal [1, 2, 7, 17]. The widely deployment of Access Points(AP) and the unique identification BSSI in each AP bring great locational discrimination to Wi-Fi signal fingerprint. However, Wi-Fi's poor temporary stability is caused by frequent redeployment of APs and subtle RF signal propagation pattern.

2.1.2 Spot Survey and Crowd-sourcing

As shown by [17], fingerprint based approach can deliver high accurate localization. However the labor-intensive survey procedure (a.k.a. war-driving) has greatly hampered a wide deployment of these systems. In war-driving, system builders are required to collect hundreds of fingerprint sample at each spot, and those spots are distributed intensively around deployment area. This process usually takes several days to several weeks, depending on the scale of deployment area. Besides the labor-intensive spot survey, constant maintenance of fingerprint database even increases the workload of building such systems. To this end, crowd-sourcing is adopted by recent proposals. Redpin [22] takes a folksonomy-like approach that allows users to identify location themselves when they are wrongly located and then to correctly associate fingerprints to these locations. OIL [9] applies a similar approach to Redpin, but it further handles spatial uncertainty and labeling errors made by users. Zee [7] uses particle filter and dead reckoning to identify user's walking trace and enriches the fingerprint database with the WiFi data collected along the trace. ARIEL [20] differentiates rooms through clustering on WiFi fingerprints collected by randomly moving users to achieve a room level localization accuracy. Unloc [23] uses distinct patterns from accelerometer, WiFi RSSI, and magnetic fluctuations detected by smartphones as organic landmarks to help locating users. Loci [24] improves semantic location service through user feedback, in which user inputs are used to correct place

detections by the service. Walk&Sketch [25] attempts to create floor maps using high resolution cameras mounted on users' backpacks. Whereas crowd-sourcing distributes survey load among the crowd, it provides no quality assurance for the generated fingerprint databases.

2.2 Model-based Localization

A model-based approach [3, 4, 5, 6, 11, 26, 27, 28] employs specific devices to measure an object's relative location to anchors or to other objects and then infers the object's absolute location using such information; it usually entails a pre-deployed infrastructure. The category includes any methods that involves measuring distance. Earlier ranging techniques are mostly RSSI-based and were used for outdoor environment [29]. They were later adapted to indoor localization [26]: as indoor signal propagation is far more complicated than outdoors, they proposed to deploy a set of calibrated anchors to better characterize the relation between the RSSs. The same approach was later improved by replacing the anchor system by a mobile node that may sporadically get a GPS location fix indoors [4]. Time-of-Flight (ToF) can be a good indicator of distance, but extremely accurate clock is needed to measure RF ToF [30], unless one replaces and complements RF with acoustic signal or ultrasound [27, 31, 32]. Although Time-of-Arrival (ToA) and Time Difference of Arrival (TDoA) can also be used for ranging, earlier technology can only allow them to be measured for acoustic signal or ultrasound [33]. In fact, TDoA can be, in theory, translated to Angle of Arrival(AoA) through the induced phase difference, but a system that derives AoA from WiFi signal has been implemented only recently [34].

2.3 Dead-reckoning Localization

Dead reckoning systems [35, 36, 37, 38, 39] typically use the accelerometer data for computing the displacement, the compass (through the magnetic field sensor) to obtain the direction or heading information and perform tracking using probabilistic algorithms. However, indoor spaces induce magnetic field variations due to the presence of ferromagnetic building structures like pillars. These variations cause the compass to fluctuate leading to inaccurate heading. To overcome this issue, compasses are used in conjunction with gyroscopes which are immune to magnetic field variations and they measure the angular rotation of the device [23]. As the inertial sensors used by a dead reckoning system are often measuring differential quantities,

their inherent drift can cause large errors in the final estimation (drift is the increase of error in displacement due to the double integration of noisy acceleration data). Therefore, a typical drawback of dead reckoning systems is the need for complicated algorithms to cope with such errors. As auxiliary algorithm, dead-reckoning is used in many existing localization systems [7, 40]. However, the most valuable usage of dead-reckoning algorithm lies in crowdsensing map building. Given floor plan, Zee [7] locates user through the trajectory user walk through, and collect WiFi fingerprint to build fingerprint database. UnLoc [8] detects relative position of landmarks through dead-reckoning, which provide for the information of map shape. PiLoc [16] collects WiFi fingerprint while user walking through corridors, then it combines trajectories with similar WiFi signal to build floor plan and fingerprint database at the same time. Jigsaw [41] records the relative position of places of interest to build general map shape with help of vision algorithms. Travi-Navi [42] allows user to record walking traces as well as vision information captured by phone camera in order to navigating people to specific locations. All these approaches utilize dead-reckoning to assist other methodologies and build powerful standalone map building and localization systems.

2.4 Device Free Localization

Device free localization is different from the former three classes because the localization object become human body itself, rather than mobile devices carried by people. A device free localization approach [43, 44, 45] utilizes pre-deployed infrastructure (usually RF signal transmitter/receiver) to detect the influence of human body to the surrounding environment.

In device free localization, human bodies are viewed as moving obstacles. WiVi [43] locates moving people by utilizing a three-antenna MIMO (two for transmitting and one for receiving) to collect WiFi signal reflected on human body. By delicately nulling the reflection on static objects, it can tell people's position even behind a wall. One disadvantage of WiVi is that it can only detect location of moving people, therefore it cannot work on stationary people. SCPL [44] is designed to synchronously locate up to four people inside a certain area through profiling the influence of Receive Signal Strength(RSS) caused by human bodies at every possible locations. [45] propose a new localization approach that does not require profiling RSS.

All device free localization approaches share two common shortcomings comparing to the other three classes of approaches:

- They cannot recognize people's identity.
- Very limited number of people that can be synchronously localized in certain area.

The first shortage constrains the application of device free localization to a quite small scope such as people cardinality. More attractive applications, e.g., Location Based Service(LBS), Self Navigation, would require the identity of people as premise. The second shortage limits the large deployment as well as applicability of a device free localization: modern buildings may contain hundreds of people inside a hall at the same time; in such scenario, current device free localization systems will not perform properly.

2.5 Summary

From the analysis of the four types of indoor localization approach, we can see that Fingerprint-based and Model-based approaches are more suitable for practical localization service, whereas Dead-reckoning can be used as a good compliment to the two technologies. Limited by the nature of technology itself, Device Free localization cannot be used in a large scale practical localization service. Therefore in this thesis, we focus on building indoor localization systems using Fingerprint-based and Model-based approaches with the help of Dead-reckoning technology.

Chapter 3

GROPING: Geomagnetism and cROWdsensing Powered Indoor NaviGation

In this chapter, we propose GROPING, a self-contained indoor navigation system independent of any infrastructural support. GROPING relies on geomagnetic fingerprints that are far more stable than WiFi fingerprints, and it exploits crowdsensing to construct floor maps rather than expecting individual venues to supply digitized maps and fingerprint database. Based on our experiments with 20 participants, GROPING is able to achieve satisfying accuracy for localization and thus can provide smooth navigation experience.

3.1 Introduction

Successful indoor navigation requires computing location information and visualizing that information on a map *in real-time*. Though commercial products (e.g., [46, 47]) and innumerable academic solutions (e.g., [1, 17, 23, 48]) have been developed for indoor localization, indoor navigation still appears to be a challenging issue. On one hand, wireless signal (e.g., WiFi and GSM), the most exploited source for inferring location [1, 7, 49, 50], may not be suitable for navigation purposes. On the other hand, presuming the availability of floor maps is common in most existing proposals, but digitized floor maps are not easily available due to proprietary and privacy issues.

It is well known that RF signals suffer from instability, which implies that achieving a

satisfactory location accuracy demands heavy computations [1]. Moreover, RF sensing is notoriously energy consuming. As both factors go against navigation that entails a continuous and real-time location estimation, a fully functional navigation service seems to demand a lightweight localization scheme efficient in both computation and energy consumption.

The dependence of navigation on digitized maps is not as strong as we often expect. As described in [51], people build cognitive maps by subconsciously remembering landmarks and moving between them to reach their destinations. Therefore, the imperceptible signs contained in a digitized map may not be that relevant; a more practical solution could be to involve human users themselves to collectively construct a map and also to provide semantic landmark information. Specifically, people carrying smartphones loaded with sensors can either volunteer or be recruited to gather information from the ambient environment for both map construction and landmark identification. This form of information collection through human participation is indeed a type of *mobile crowdsensing* [52].

Localization commonly requires a fingerprint library against which certain newly sampled signal may compare and hence determine the location. However, the localization function required by indoor navigation differs in two main aspects from a pure localization scheme that pinpoints the current position of a user. On one hand, it requires real-time and constant location computations. This means that it demands very stable fingerprints, as it may not afford comparing with a library in which a single location is associated with a large number of fingerprints (e.g., WiFi fingerprints [50, 53]). On the other hand, it does not require a very high accuracy, as the navigation service only needs to lead a user to a point within the visual range of the actual destination. This makes it unnecessary to have a meter level accuracy achieved by, for example, dead reckoning systems [35] at the cost of handling directional/drift errors and performing calibrations/computations with multi-sensor data on resource constrained devices. Therefore, our design applies the magnetometer and exploits *geomagnetism* as the location indicating fingerprint: it is lightweight (only a 3D vector) and very stable, and it is completely independent of any kind of wireless infrastructure.

To better motivate our design philosophy, we first report a study on Google Maps Indoor (GMI) [47], the only indoor navigation system available for *public* testing, as well as on basic properties of both WiFi and geomagnetism in location estimation; this study reveals issues pertaining to the aforementioned ones. In response to these issues, we propose *Geomagnetism and cROwdsensing Powered Indoor NaviGation* (GROPING) as a completely self-contained,

lightweight, and practical prototype for indoor navigation. GROPING encapsulates three functions, namely *map building*, *localization* and *navigation*, into one unit. It first builds a map using user contributed sensor data and semantic labels; it then performs localization based on the magnetic fingerprints, and finally it runs a navigation service on top of these two functions: it computes navigational routes using the early constructed map and the real-time location information. In this way, GROPING eliminates *infrastructure dependence*: it needs neither wireless infrastructure nor digitized floor maps. Our intensive experiments with GROPING demonstrate its usability and also show that it compares favorably with typical WiFi-based localization systems in supporting indoor navigation.

3.2 Studies on Google Maps Indoor

In spite of the huge numbers of proposals on indoor localization, the only system that is available for public testing is Google Maps Indoor (GMI) [47]. Therefore, we organize a group of 11 people to perform a detailed study on it. Given that GMI appears to a user as a blackbox, our study is separated into two parts. The first part is a field study in five big-scale shopping malls (above 10000 m²) to test the accuracy of GMI, as well as to make sure if WiFi is used by GMI (which appears to be true). The second part reports an evaluation of the energy efficiency of WiFi-based localization systems, leveraging on the energy profiles obtained as a by-product of the earlier studies.

3.2.1 A Field Study on GMI

As GMI works only for venues that contribute floor maps to Google, we are confined in choosing test sites (Fig. 3.1(a) shows two of them). In fact, only 11 shopping malls in Singapore have GMI support available. The mobile phones we use include Samsung Galaxy S2/S3, Sony Xperia S, and HTC One X. In this study we mainly want to answer the following three questions.

- **Q1:** What is the accuracy of GMI's localization?
- **Q2:** Does GMI's navigation work well?
- **Q3:** Does GMI heavily rely on WiFi infrastructure?

3.2.1.1 Location Accuracy

The team members unanimously agree that GMI usually produce unsatisfactory localization accuracy. We first show a few screenshots taken on GMI in Fig. 3.1(b), in which both actual locations (pinpointed by the users on-site) and the locations indicated by GMI (the blue arrows) are shown.

(a) Two shopping malls as examples of our GMI test site.

(b) Three examples of inaccurate localization

Figure 3.1: Screenshots taken on GMI.

To quantify GMI's localization accuracy, we perform tests at 30 randomly chosen positions in each of the five malls. The accuracy results are shown in Fig. 3.2 (with the number of average available WiFi APs at test points shown alongside the names of the malls): four malls have average localization error of around or above 20 meters, which can be hardly usable for

indoor localization. Only test cases in ION exhibit reasonable errors: half of them are less than 10 meters. This is partially due to the smaller size of ION and hence a much denser WiFi deployment there.

Figure 3.2: GMI localization errors in five shopping malls.

We also use Fig. 3.3 to show the satisfactory level of users. We ask eleven users to evaluate their personal localization experience in a specific shopping area. Users can visit the area any times they want to during a certain period. A user is *satisfied* with a GMI location indicator if he feels that the indicator helps to locate himself (i.e., location errors within visual range is tolerable); otherwise *unsatisfied*. Obviously, the satisfactory levels are generally low. As

Figure 3.3: The number of satisfied and unsatisfied cases for 11 users.

mentioned before, out of the five shopping malls, ION has denser WiFi access points (APs) than others. So quite some satisfactory cases are obtained there.

3.2.1.2 Navigation and WiFi Reliance

GMI's navigation function does not appear to be useful due to the unsatisfactory location accuracy (see the above discussions). Note that even if the initial location is satisfactory, a few unsatisfactory location estimations on the way may ruin the navigation.

All our results have evidently confirmed GMI's heavy reliance on WiFi infrastructure: when either a phone's WiFi interface is switched off or the WiFi signals become very weak (in a basement level where WiFi hotspots are not installed), the GMI's location indicator is often expelled outside of the building, suggesting that some sort of cellular-based location estimation is applied.

3.2.2 Energy Efficiency Evaluation

We record the energy consumption (in terms of battery percentage by Android's battery meter) of our Samsung Galaxy S2 (with a 1650mAh battery) during the field studies on GMI and GROPING. During our lab tests, we further monitor the energy consumption for four configurations, namely idle (i.e., no sensor running), sampling magnetometer and gyroscope at 5Hz (GROPING), WiFi scanning at 0.3Hz, and lastly a combination of WiFi scanning and accelerometer sampling both at 0.3Hz. The sampling frequency of 0.3Hz comes from our observation that GMI updates its location estimation in about every three seconds. For all the tests, the 1650mAh battery is fully charged before continuously operating for 300 minutes, and the drop in battery life is recorded every 20 minutes. To focus on the energy consumption of sensing, we deduct the energy consumed under the idle configuration from all other configurations, and the results are shown in Fig. 3.4.

According to Fig. 3.4, GMI consumes much more energy than GROPING, but both consume more than pure sensing (possibly due to the use of 3G). Fig. 3.4(b) further shows that sensing configurations involving WiFi use 24% to 28% of the battery in 5 hours. As a sharp contrast, using two inertial sensors (as the case with GROPING) consumes only 3% of the battery for the same period, exhibiting roughly 10 times battery savings than other lab settings.

3.2.3 Summary

We summarize the key insights on GMI that motivate our designs of GROPING in the following:

Figure 3.4: Energy consumption comparisons.

- GMI's implementation of WiFi-based localization has not worked accurately yet. This may attribute to the instability of WiFi signals, sparse WiFi deployments, and insufficient fingerprints.
- GMI cannot be very helpful in indoor navigation due to unsatisfactory location accuracy, as navigation requires consistent location estimations.
- Localization over the whole map area is not helpful for navigation purpose, as errors in location estimation may render the user location off a pathway and hence reduce the chance of successful navigation.¹
- High energy consumption is another major drawback of WiFi-based indoor localization systems, and this issue is exacerbated under navigation due to its need for constant location updates.

3.3 GROPING System Overview

GROPING provides services that caters to users' location and navigation requests in various indoor facilities, and it relies on the regular occupants of a certain indoor facility to assist in building floor maps. Basically, the end users include *map explorers* and *strayed users*. Map explorers are recruited due to their familiarity with a particular building. They walk along various pathways and upload their trajectories (consisting of sensed data) to the server. Strayed

¹Google Maps (outdoor) only works for places where road systems come across.

users are those who are unclear about their locations and hence require localization or/and navigation services. We illustrate the architecture of GROPING in Fig. 3.5. The system consists

Figure 3.5: GROPING system architecture.

of smartphone clients and a server. Each client provides a user interface for collecting data, as well as visualizing the constructed map, the current (estimated) location, and the navigation routes. The server is a cloud service; it consists of modules that build floor maps, estimate locations, and deliver real-time navigation. We shall briefly discuss these components in this section.

3.3.1 Map Building

We hereby illustrate by an example how GROPING utilizes the contributions from map explorers to gradually build an indoor map. Alice and Bob are regular visitors to a shopping mall shown in Fig. 3.6. One day Alice installs GROPING but finds no map exists for the mall yet. She decides to create one by making the first contribution. Starting from position **A**, Alice walks toward an arbitrary direction and records the ambient magnetic field by her smartphone running GROPING. After walking for a while, she sees a three-way conjunction point **B** ahead of her. She could rely on the gyroscope in her phone to tag such a junction, but she may also choose to tag it manually (see Sec. 3.3.3.2). Such tags help GROPING to partition trajectory into segments. Eventually, Alice stops at junction **E** and uploads the trajectory data (top-right of Fig. 3.6) to the GROPING server.

Figure 3.6: Floor map of a shopping mall and the walking trajectories contributed by Alice and Bob.

The next day Bob comes to the same mall and finds the incomplete map contributed by Alice. So he decides to complete it, which first results in a trajectory shown in the mid-right of Fig. 3.6. GROPING server uses the similarity in magnetic fingerprints to infer the overlapping segments among the trajectories and sticks them together. After a few seconds, Bob receives a map (Fig. 3.7(a)) shown on his screen, waiting for him to either confirm or revert. Bob feels satisfied with the map, so he confirms and starts another trajectory recording procedure, which eventually end up with a complete map shown in Fig. 3.7(b).

Figure 3.7: Virtual map generation using three trajectories and the associated magnetic fingerprints.

3.3.2 Localization and Navigation

Based on the constructed map, the localization and navigation functions are integrated in GROPING, in the sense that user mobility facilitates localization that in turn drives further mobility (i.e., navigate a user). This is achieved by a revised Monte Carlo Localization (MCL) algorithm whose details are explained in Sec. 3.4.2. To continue our illustrating example, let us consider a strayed user Cindy. She starts her GROPING client, chooses the already constructed map, and requests a navigation service by either providing a semantic label (a shop name) or pinpointing a location on the map. Without knowing the initial location of Cindy, GROPING recommends a tentative route. While Cindy is walking along the route, her GROPING client keeps updating the sampled magnetic field information to the server. This allows GROPING to refine the location estimation for Cindy and also updates the route accordingly, until Cindy reaches her destination.

3.3.3 User Interface

A GROPING client has a simple interface as shown in Fig. 3.8. The starting screen, Fig. 3.8(a), requires each user to select a map. For a map explorer, one may choose to build a new map or to reinforce an existing map. For a strayed user, one needs to choose a map from a list. As the GSM location information is attached to every map when it is first built, the map list shown to a user is confined to the region close to the user's estimated location by GSM and is sorted in increasing distances. As the GSM localization is just an ancillary function and it runs only when GROPING starts up or a user switches to a different building, the incurred overhead is negligible.

After choosing an existing map or initiating a new one, the "Start" button allows an explorer to start data collection and map generation, as shown by Fig. 3.8(b). Otherwise a strayed user may switch to the "Loc/Nav" panel to find his/her location and/or to obtain navigation guidance towards a certain destination, as shown by Fig. 3.8(c). We provide more details on these two panels in the following.

3.3.3.1 Map Panel

Map view allows explorers to collect trajectory data and upload them for map construction. The data collected along each walking trajectory includes both magnetic fingerprints and gyroscope readings. While GROPING uses the fingerprints to represent individual pathways, it

Figure 3.9: More about GROPING client user interface.

3.3.3.2 Navigation and Localization Panel

This panel first presents the location of the user with a yellow dot on the selected floor map. This location may not be accurate, but the server will gradually refine it after the user starts to move. If the user chooses a destination (red dot), the navigation route to the destination from current location is depicted in green color on the map, shown in Fig. 3.9(c). To define a destination, the user can either pinpoint it on the map or perform a semantic label searching. Label searching may cause all related labels being highlighted for further selection. For example, when a user searches for “cafe”, all labels containing “cafe” will be highlighted. The navigation route is computed as the shortest path between the current location and the chosen destination. Because the current location can be updated by the server (especially at the beginning), the route may experience some changes initially but should stabilize soon. If a user diverts from the specified route due to missing a correct turning point, a new route will be highlighted accordingly. To better assist the navigation, walking instructions such as “go straight” and “turn right” are given either regularly or before a certain event.

Remarks: As illustrated by Fig. 3.8(c) and 3.9(c), our map differs significantly from those of GMI. This follows from the rationale that only the “road system” is necessary for navigation,

adding other components such as rooms or cubicles may confuse users, given that the location estimation cannot be perfect. Our GROPING is meant to be a navigation service, so it may not provide a comprehensive localization function over the whole floor, as promised by GMI and other existing proposals [7, 23, 53]. Therefore, GROPING is rather a complement to the existing WiFi or dead reckoning based localization systems than a competitor to them, and it can be combined with other systems to perform both lightweight navigation and accurate localization.

3.4 System Components

In this section, we dive into the technical details of the three components comprising GROPING: map builder, location estimator, and navigation.

3.4.1 Map Builder: A Joint Venture of The Crowd

Map builder is the most unique part for GROPING compared with the existing indoor localization literature [7, 35, 40, 53, 54] (where a known map is always assumed). The principle behind GROPING map builder is that, when a certain number of explorers walk indoors, there is a high possibility of their trajectories overlapping. Merging these overlapping trajectories results in a floor map that comprises of only the indoor route structure. This simplifies the information content (pertaining to the floor map), making it easy for users to follow the map. Moreover, each map is enriched with semantic information, i.e., landmarks provided as labels, to facilitate navigation.

3.4.1.1 Virtual Map Terminologies

We consider three components of a floor plan, namely hallway, conjunction points, and semantic labels. As shown in Fig. 3.10, the blue areas are hallways, the red area is a conjunction point, and the numbered blank spaces are semantic labels attached to hallways. The objective of GROPING virtual map generation is to re-construct the map to the extent as illustrated by the yellow skeleton, using sensor data collected by the users. Based on the idea of crowdsensing, we let a group of users to arbitrarily pick up walking trajectories and use their smartphones to collect sensor data while walking. To endow the map with semantics, each user is supposed to *label* a couple of rooms (by names or numbers) along each trajectory.

Figure 3.10: Three components of a floor plan.

We define a virtual map M as one that contains route structure information, semantic labels l , and magnetic fingerprints F . Route structure information include segments (pathways), conjunctions/linkages between segments, and time spent on each segment. Semantic labels are stored as texts but are associated with respective locations in terms of segment percentages. Fingerprints of a segment are the magnetic field signals collected along that segment. Multiple fingerprints from different trajectories are allowed to be associated with the same (overlapping) segment. In particular, the map library $\mathcal{M}$ contains a set of virtual maps $\{M_1, M_2, \dots, M_k\}$, where each $M_i = \{C_i, E_i\}$ is represented as a graph with vertex set C_i and edge set E_i , with each vertex $c_{ij} \in C_i$ indicating a conjunction and each edge $e_{ij} \in E_i$ representing a segment. Moreover, each c_{ij} is associated with a set of angles (obtained from gyroscope readings), and each edge e_{ij} is associated with a set of fingerprints $\{F_{ij}^1, F_{ij}^2, \dots, F_{ij}^n\}$ and a set of labels $\{l_{ij}^1, l_{ij}^2, \dots, l_{ij}^m\}$. We explain in the following the three steps taken by GROPING to form a map.

3.4.1.2 Trajectory Segmentation

To identify hallways, we need to partition a user's walking trajectory (represented by the sensor data collected on the way) into segments. This is done by two approaches. In the first approach, we integrate the gyroscope reading g_y within a sliding window $W_{\text{turn}} = 5$ seconds. If the value goes beyond a threshold (20 degree in our setting), a conjunction point is detected, as shown in Fig. 3.11. The total turning angle is estimated by gradually enlarging the integration window

Figure 3.11: Segmenting a trajectory using gyroscope reading g_y .

until the result of integration stops increasing. As a result, the corresponding data segment is marked as T (i.e., turning point) and the total turning angle becomes the fingerprint associated with this segment. The second approach explores the human sensing ability, which we term *tagging*. Specifically, users manually tag the sensor data with T upon a conjunction and then tag P on the data upon returning to a hallway (see Sec. 3.3.3.1).

The first approach is automatic without the need for human intervention, but it fails to detect a conjunction point if the user goes straightly through it. The second approach works perfectly if a user remembers to tag all conjunctions. In practice, both approaches work fine if we have sufficient number of trajectories, as we can anyway drop those containing conjunctions that we fail to detect due to either a straight going through or a user's oblivion of tagging. We mainly use the first approach but optionally augmented by the second one. At the end of the segmentation phase, each trajectory T consists of at least one segment. A hallway segment (marked as P) contains magnetic fingerprints of the corresponding hallway, and a conjunction segment (marked as T) is associated with its angle. Also, each segment is sporadically labeled with room numbers or names.

3.4.1.3 Segment Matching

GROPING makes use of the overlaps between trajectories to stitch them together. Given a sufficient amount of trajectories that cover the whole floor and that overlap with each other, the skeleton of the floor map can be re-generated. To this end, we need to identify overlapping segments of an arbitrary pair of trajectories. However, the segments of magnetic fingerprints can be time misaligned since the walking speeds vary across users collecting data. Therefore, we use the DTW algorithm [55] to compute the similarity. DTW is well known to handle

sequences that follow a similar trend but vary across the time axis. The main idea behind DTW is to compress or stretch the time axis of one (or both) sequences for getting a better alignment.

Consider two segments of magnetic fingerprints, $F_1 = \{f_1, f_2, \dots, f_K\}$ and $F_2 = \{f'_1, f'_2, \dots, f'_L\}$. The goal is to find the best match between these two segments by an alignment w^* called *optimal warping path*. A warping path is given by $w = w(1), w(2), \dots, w(N)$, in which $w(n) = [i(n), j(n)]$ is the set of matched samples, where $i(n)$ and $j(n)$ belong to the index sets of F_1 and F_2 , respectively. The optimal warping path w^* minimizes the overall cost function given by $\sum_{n=1}^N \delta(w(n))$, where $\delta(w(n))$ is the distance measure computed using the inverse of cosine similarity given as:

$$\delta(i(n), j(n)) = \cos^{-1} \left(\frac{f_{i(n)} \cdot f'_{j(n)}}{\|f_{i(n)}\| \|f'_{j(n)}\|} \right). \quad (3.1)$$

Given a pair of segments, their minimized cost function $\sum_{n=1}^N \delta(w^*(n))$ characterizes their similarity: a lower function value indicates a higher similarity.

3.4.1.4 Map Formation

One major difference between our map formation and photo stitching is that we face a much more complicated topology: topologies involved in photo stitching often contain no loop. Our idea is to start the map from a single trajectory, then sequentially invoke **Algorithm 1** to gradually stitch incoming trajectories to the existing map.

The stitching process is based on Bayes filter [56]. It associates with an end point, $s_f.bEnd$, of $s_f \in E$ (an existing segment) the probability P_{coin} of s_f coinciding with an incoming segment $s \in T$, and it keeps updating P_{coin} while scanning sequentially through all segments in T (lines 2 to 28). For a conjunction segment s_f , the probability is updated according to the similarity in angle between s and s_f (computed by $\text{simA}(s, s_f)$, a function of the absolute difference between the two angles) multiplied by the probability associated with the hallway segment preceding s_f (lines 7 and 10), where we use $\propto$ to indicate that probabilities are to be normalized to satisfy unitarity. The situation is slightly more complicated for a hallway segment s_f , as all the preceding (conjunction) segments should be counted (lines 12 and 17). The similarity evaluation done by $\text{simM}(s, s_f)$ follows what was discussed in Section 3.4.1.3. For both cases, the similarity should be computed from both directions (lines 8 and 16). Since the first segment has no preceding one, we bootstrap it with a small probability.

Algorithm 1: Trajectory Stitching

Input: New trajectory T , current map $M = \{C, E\}$

```

1 foreach  $c \in C$  do  $P_{\text{temp}}(c) \leftarrow 0$ ;  $P_{\text{coin}}(c) \leftarrow 0$ ;
2  $s \leftarrow T.\text{firstSeg}$ 
3 while  $s \neq \text{NULL}$  do
4   foreach  $s_f \in E$  do
5     if  $s.\text{tag} = \text{T} \wedge s_f.\text{tag} = \text{T}$  then
6        $s_p \leftarrow s_f.\text{prevSeg}$ ;  $p_{\text{coin}} \leftarrow P_{\text{coin}}(s_p.\text{bEnd})$ 
7        $P_{\text{temp}}(s_f.\text{bEnd}) \propto \text{simA}(s, s_f) \times p_{\text{coin}}$ 
8        $s'_f \leftarrow \text{reverseSeg}(s_f)$ 
9        $s_p \leftarrow s'_f.\text{prevSeg}$ ;  $p_{\text{coin}} \leftarrow P_{\text{coin}}(s_p.\text{bEnd})$ 
10       $P_{\text{temp}}(s'_f.\text{bEnd}) \propto \text{simA}(s, s'_f) \times p_{\text{coin}}$ 
11    else if  $s.\text{tag} = \text{P} \wedge s_f.\text{tag} = \text{P}$  then
12      foreach  $s_p \in s_f.\text{prevSeg}$  do
13         $p_{\text{coin}} \leftarrow p_{\text{coin}} + P_{\text{coin}}(s_p.\text{bEnd})$ 
14      end
15       $P_{\text{temp}}(s_f.\text{bEnd}) \propto \text{simM}(s, s_f) \times p_{\text{coin}}$ 
16       $s'_f \leftarrow \text{reverseSeg}(s_f)$ 
17      foreach  $s_p \in s'_f.\text{prevSeg}$  do
18         $p_{\text{coin}} \leftarrow p_{\text{coin}} + P_{\text{coin}}(s_p.\text{bEnd})$ 
19      end
20       $P_{\text{temp}}(s'_f.\text{bEnd}) \propto \text{simM}(s, s'_f) \times p_{\text{coin}}$ 
21    end
22  end
23   $[c_{\text{max}}, p_{\text{max}}] \leftarrow \text{maxProb}(P_{\text{temp}})$ 
24  if  $p_{\text{max}} > \text{defiThreshold}$  then
25     $\text{mergeSeg}(s, c_{\text{max}}.\text{endSeg})$ ;  $\text{backBProp}(T, s)$ 
26  end
27  if  $\text{formLoop}(M)$  then  $\text{relaxLoop}(M)$ ;
28   $P_{\text{coin}} \leftarrow P_{\text{temp}}$ ;  $s \leftarrow s.\text{nextSeg}$ 
29 end

```

If a certain coincidence probability (associated with an end point v_{max}) becomes larger than defiThreshold (i.e., *definiteness threshold*) (lines 23 and 24), $s \in T$ is merged with the segment in M whose end point is v_{max} , and a backward belief propagation $\text{backBProp}(T, s)$

is applied to trace back the stitching history such that all the previous segments are properly merged into M (line 25). If a loop is formed after stitching, $relaxLoop(M)$ is invoked to adjust the geometry of the loop such that the graph M can be embedded into a 2D plane.

The resulting map M , on one hand, contains all the fingerprints that have been collected, and each segment is associated with a set of fingerprints collected from the corresponding hallway or conjunction. On the other hand, it has the same topology as the original floor map, as well as a similar geometry (the length of a hallway can be estimated by the number of points contained in a corresponding segment fingerprint). In fact, the index of a sample point in a segment fingerprint also indicates a rough location on the corresponding hallway. For example, if a point is the 100-th point out of 1000 samples of a segment, then the location is at the 10% length of the whole hallway. Of course, indicating location in this way may lead to error, but it is within the tolerable range of the applications targeted by GROPING. Later in Section 3.4.2.1, we shall abuse the terminology by using $l \in M$ to denote that l belongs to the index set of the sample points in M ; in other words, l is a location on our virtual map M .

3.4.2 Location Estimator: A Bayesian Approach

The basis of GROPING’s location estimator is a classification process similar to other WiFi-based localization schemes (e.g., [23, 50]), where a user’s sensor data are compared with the existing fingerprints to obtain a list of similarity indices, and the location is suggested by the highest similarity index. However, the ambient magnetic field that we rely on offers less information than the WiFi-based infrastructure: the former is just a 3D vector field (magnetic field strength in X, Y, and Z directions) but the latter, given a sufficient amount of available WiFi hotspots, may produce fingerprints in a much higher dimensional space. As a result, we have to resort to a filtering technique that spans the temporal dimension to gain more information for achieving a sufficiently accurate location estimation.

3.4.2.1 Revised Monte Carlo Localization

To involve the temporal dimension, a sequential estimation technique is needed. This motivates us to use the *Monte Carlo Localization* (MCL) approach [57]. Under a Bayesian framework, MCL recursively computes the posterior distribution of the location l_t (a.k.a. *belief*) $B(l_t) = p(l_t | \mathbf{m}_{1:t})$ at time t , considering different measurements $\mathbf{m}_{1:t}$ (collected sensor readings) from

time 1 up to time t . We briefly walk through the algorithm below, while emphasizing on our revisions. Using the Bayes rule, we have

$$B(\mathbf{l}_t) = \gamma p(\mathbf{m}_t | \mathbf{l}_t) p(\mathbf{l}_t), \quad (3.2)$$

where γ is the normalizing constant.

While a user keeps walking (and collecting new sensor readings), the belief is recursively updated as follows:

$$B(\mathbf{l}_t) = \gamma p(\mathbf{m}_t | \mathbf{l}_t) \sum_{\mathbf{l}_{t-1} \in M} p(\mathbf{l}_t | \mathbf{l}_{t-1}) B(\mathbf{l}_{t-1}), \quad (3.3)$$

where M refers to the virtual map that we build using the techniques presented in Section 3.4.1.4.

After a certain period t , an MLE estimator is applied to select the location with the highest posterior probability, giving a location estimation:

$$\hat{\mathbf{l}} = \operatorname{argmax}_{\mathbf{l}_t \in M} [B(\mathbf{l}_t)]. \quad (3.4)$$

In general, a larger t leads to a higher estimation accuracy. However, as we shall show in Section 3.6.3, the accuracy is sufficiently high after only a few tens of seconds. To implement (3.3), we need both $p(\mathbf{m}_t | \mathbf{l}_t)$ (observation model) and $p(\mathbf{l}_t | \mathbf{l}_{t-1})$ (motion model). In the following, we discuss how we tailor these two models to accommodate the features of GROPING. As both models are time-invariant, we drop the subscript t hereafter.

Observation Model: We evaluate $p(\mathbf{m} | \mathbf{l})$, the observation model, in the following way. For a new measurement $\mathbf{m}$, we compare it with all sample points in M . This comparison is again based on the cosine similarity between a sample point $\mathbf{s} \in M$ and $\mathbf{m}$. We have a few sample points sharing the same index (i.e., at the same location $\mathbf{l} \in M$), as a result of the clustering procedure explained in Sec. 3.4.2.2). So we take the maximum cosine similarity value to build the observation model:

$$p(\mathbf{m} | \mathbf{l}) \propto \max_{\mathbf{s} \in \mathcal{S}_{\mathbf{l}}} \cos(\mathbf{s}, \mathbf{m}), \quad (3.5)$$

where $\mathcal{S}_{\mathbf{l}}$ is the set of sample points indexed by $\mathbf{l}$. The operator $\propto$ is again for normalization purpose.

In fact, what estimated by $\max_{\mathbf{s} \in \mathcal{S}_{\mathbf{l}}} \cos(\mathbf{s}, \mathbf{m})$ is rather $p(\mathbf{l} | \mathbf{m})$. However, according to Bayes rule, $p(\mathbf{m} | \mathbf{l}) \propto p(\mathbf{l} | \mathbf{m})$ if we assume non-informative priors for $\mathbf{l}$ and $\mathbf{m}$ (i.e., $p(\mathbf{l})$ and $p(\mathbf{m})$ both follow a uniform distribution).

Motion Model: The motion model is represented by a Markov transition matrix, in which nearest locations in both directions from the current location have non-zero transition probabilities, and all other probabilities are zero. In the most ideal case (where the walking speed of the current user coincides with that implied by the normalized length of the segment fingerprints), only two transitions are possible: forward and backward, shown by an example within one segment as follows:

$$p(\mathbf{l}^+ | \mathbf{l}) = \begin{matrix} & \ell_1 & \ell_2 & \ell_3 & \cdots & \ell_n & \cdots \\ \begin{matrix} \ell_1 \\ \ell_2 \\ \ell_3 \\ \vdots \\ \ell_n \\ \vdots \end{matrix} & \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & \cdots \\ 0.5 & 0 & 0.5 & \cdots & 0 & \cdots \\ 0 & 0.5 & 0 & \cdots & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \\ 0 & 0 & 0 & \cdots & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \end{pmatrix} \end{matrix},$$

where $\ell_i \in M$ are indices of sample points. In our implementation, we assume that the actual walking speed of a user can be at most α (≤ 2) times faster than the normalized one. Therefore, there might be up to 2α possible transitions from each location. Our motion model differs from the traditional one assuming a continuous transition distribution, simply because the virtual map M is a discretized version of the original map.

There is yet another issue that we need to handle before proceeding to actual localization. As incoming segment fingerprints increase linearly with user participation, this tends to increase the complexity of location estimation, since more and more fingerprints need to be compared against the newly sampled sensor data. To this end, we apply a clustering algorithm to obtain the representatives among the fingerprints for a given segment in the following.

3.4.2.2 Clustering with Affinity Propagation

In our experiment we see heterogeneous device models may obtain different (albeit involving similar features) magnetic field strength readings, as shown in Fig. 3.12. However, if we kept all the data associated with a segment as its fingerprints, the complexity of location estimation would keep increasing. Our idea here is to classify the fingerprints for each segment, and choose one fingerprint for each cluster to represent it. The outcome is that only a few fingerprints need to be compared during the location estimation procedure.

Figure 3.12: The same ambient magnetic field sensed by different smartphones.

Obviously, typical clustering algorithms such as k -means do not work, as we do not know k a priori, and those algorithms may not return existing values in a data set. Therefore, we apply the *Affinity Propagation* (AP) algorithm [58] to obtain a few representatives out of the fingerprint set. AP is a message passing algorithm, where the magnitude of each message passed showcases the current belief or affinity one data point (segment fingerprint in our case) has for choosing another data point as its exemplar among a set of points pertaining to a particular cluster. AP does not assume a priori knowledge of k , i.e., the number of clusters. It proceeds iteratively using a similarity matrix containing the similarity score between each pair of fingerprints and updated by the messages passed.

Figure 3.13: The outcome of AP clustering.

In Fig. 3.13, the background (light blue curves) shows about 100 fingerprints associated with a certain hallway segment, whereas the foreground (red curves) are the seven represen-

tatives chosen by AP. This significantly reduces the complexity of executing (3.5). Note that we cannot use DTW to compute the similarity scores, as the outcome of DTW is not a metric. Therefore, we first apply the DTW-based time-normalization procedure [59], in order to normalize all fingerprints associated with a certain segment to the same length (the median length) and variance. Then the similarity scores are computed as the negative Euclidean distances among these normalized fingerprints.

3.4.3 Navigation: Client–Server Interactions

We first sketch the client-side navigation by pseudo-codes in **Algorithm 2**. While navigating, the client periodically queries location from the server (line 2). If the current location is sufficiently close to the destination, the navigation is completed (line 3). Otherwise if the current location is off the route, a new route is queried from the server (line 5). At the end of each round, the route is rendered on the map and certain instructions are also shown. The client-side of GROPING only performs simple computations, while the heavy computations are offloaded to the server-side.

Algorithm 2: Client-side Indoor Navigation

Input: Destination point d , Map M

```

1 while navigationOn do
2    $c \leftarrow \text{currentLocation}()$ 
3   if inRange( $c, d$ ) then break ;
4   if notOnRoute( $c, rt$ ) then
5      $rt^+ \leftarrow \text{requestNavigation}(d, c, rt, M_i)$ 
6   end
7   renderRoute( $rt^+$ ); showInstruction( $c, rt^+$ )
8 end
```

Given a destination chosen by a user and the current location returned by the location estimator, the navigation manager (server-side) calculates the route to destination on a map and provides continuous instructions. As we have discussed, give a map $M_i = \{C_i, E_i\}$, we use the average number of sample points for the fingerprints associated with a segment in E_i to roughly represent the length of that segment. This allows the server to compute a shortest path on M_i from the current location to the destination.

As shown by **Algorithm 3**, given the current location c and destination d , the server treats two segments (the segment containing the current location e_c and that containing the destination e_d) differently. Basically, the server divides e_c into e_c^1 and e_c^2 by the current position r_c with respect to e_c , and e_d into e_d^1 and e_d^2 similarly. The length of the new edges are assigned proportionally, and the new (temporary) map M'_i is fed to the Dijkstra's algorithm to compute the shortest path between the current location and destination. In practice, a slight preference will be given to the current walking direction of the user. For each navigation request coming

---

Algorithm 3: Server-side Indoor Navigation

```

1 upon recvNavRequest( $c, d, rt, M_i$ )
2  $(e_c, p_c) \leftarrow \text{pointInMap}(c, M_i)$ 
3  $(e_d, p_d) \leftarrow \text{pointInMap}(d, M_i)$ 
4  $M_i.\text{removeEdge}(\{e_d, e_c\})$ 
5  $(e_c^1, e_c^2) \leftarrow e_c.\text{breakAt}(p_c); (e_d^1, e_d^2) \leftarrow e_d.\text{breakAt}(p_d)$ 
6  $M_i.\text{addEdge}(\{e_c^1, e_c^2, e_d^1, e_d^2\})$ 
7  $rt^+ \leftarrow \text{Dijkstra}(p_c, p_d, M_i)$ 
8 if  $rt^+ \subseteq rt$  then deliverToClient(NULL) ;
9 else deliverToClient( $rt^+$ ) ;
```

---

from a client, a shortest path to destination is calculated by navigation manager using **Algorithm 3** and is then delivered to client. Instead of sending only one navigation request at the beginning, the client actually generates such a request on a regular basis. However, the server does not send a new route back as long as the user's location is still on the previously determined route; a route update is sent back only if the server finds out that the user is off the previously determined route.

3.5 System Evaluation Design

We briefly explain how we perform user studies and performance evaluations on GROPING in this section, and also discuss some issues we have encountered for the user studies, as well as our current and future solutions.

3.5.1 Experiment Setting

We recruited 20 users to participate in our user study and evaluations. As our emphasis is rather on the functionalities of GROPING than on the usability of its interface, we only involve participants with CS background, but they are all first-time users of GROPING. Eight of them were selected to play the role of map explorers due to their familiarity with the test site, and the rest were strayed users. Their specific tasks include familiarizing with GROPING interface, collecting sensor data, labeling landmarks, and providing feedback. While walking, a user is required to hold the phone horizontally and point it ahead.

We have evaluated GROPING by three studies mainly in one test site (a shopping mall with 3 floors). In the first one, we measure the time needed to complete a map in each floor. In the second one, we focus on quantitative evaluations on the accuracy of the localization service. Finally, we qualitatively study the navigation service, and report the user experiences on it. We also implemented FreeLoc [2], a recently proposed WiFi-based indoor localization system, and we compare the localization accuracy of GROPING with both FreeLoc and GMI. Further comparisons with two canonical proposals RADAR [17] and Horus [1] are conducted in the smaller test site, as these proposals entail intensive WiFi fingerprinting.

As user acceptance is important to a navigation system [60], we try to understand user preferences before and experiences after using our navigation system, by employing user feedback. To understand user expectations, we first conduct a questionnaire-based on-line study about the users experiences of getting directions inside large buildings without a navigation service and on what they expect from a navigation system (in which we involve extra participants through Amazon Mechanical Turk (AMT) [61]). The user experience is the outcome of the aforementioned third study: each of our 20 (local) participants delivers a feedback on GROPING after using it.

3.5.2 Incentives for Crowdsensing

Getting a sufficient number and diversity of participants for our user studies has been a challenge, because one of the main tasks that we assign to our users is mobile crowdsensing (for map generation) using their individual smartphones. Crowdsensing data collection differs significantly from traditional crowdsourcing since it demands individuals' utilizing of time, energy (e.g., physical activities) and resource (e.g., smartphone usages), so the incentive to "entice" participants into providing high quality data may need to be very substantial. In other words,

S/N	Question	Our Test Site	General
1	How many times have you been to the mall within last year	10.61 ± 8.74	18.07 ± 11.04
2	How familiar are you with the indoor space (1-10)	5.11 ± 2.80	7.11 ± 2.17
3	How easy it is to navigate to a particular store based on you-are-here maps (1-10)	5 ± 1.43	7.03 ± 2.21
4	How helpful it will be to have a smartphone-based indoor navigation system (1-4)	3.5 ± 0.67	2.7 ± 0.77
5	How many landmarks do you usually recall every time you enter the mall	5 ± 2.66	4.27 ± 1.55

Table 3.1: User perceptions on indoor navigation solutions (average $\pm$ standard deviation).

some form of remuneration is necessary to encourage active participation in crowdsensing. Incentive mechanisms are often task dependent and can range from monetary incentives (cash, lottery tickets, gift cards, etc.) to valuable services (e.g., free WiFi access or storage spaces) [62, 63].

3.6 Experiment Results

Beside the comparisons made in Sec. 3.2, we further evaluate GROPING in this section.

3.6.1 Why GROPING is Needed

To show why a portable indoor navigation solution is needed, we design a questionnaire survey about people's indoor experience. The survey is done in two groups. The first group includes our 20 participants, and the 118 participants of the second group are involved by extending the survey to AMT [61] and the questions are raised towards a familiar mall. Table 3.1 shows the answers to the first five questions and Fig. 3.14 shows the outcome of the last question. Because the first group is restricted to choose our test site and the second group can choose any familiar shopping mall, participants of the second group show more confidence in navigating by you-are-here maps than those of the first group. Also, as our test site has a more complicated route structure (see Fig. 3.15), it is reasonable that the first group expresses more eagerness to have a portable navigation system. In fact, both groups can recall on average less than five locations,

Figure 3.14: Types of landmarks remembered by people.

so a handy navigation system may always help to avoid finding/checking you-are-here maps. The outcome shown in Fig. 3.14 has independent interests. Although each participant only remembers about five landmarks, those landmarks are well spread across different types. In other words, by asking map explorers to sporadically label landmarks, there is a fairly good chance that the labels would cover diverse landmarks in a mall.

3.6.2 Efficiency of GROPING Map Construction

We perform a field study in a shopping mall with three floors shown in Fig. 3.15 (top row). The most intriguing aspect of this mall is its complicated indoor route structure, which makes indoor navigation an actual necessity (most of our participants often get confused whenever they enter this mall). We also show the constructed map as the screenshots on our phone in Fig. 3.15 (bottom row).

We summarize the time needed to complete the map construction for individual floors with different numbers of explorers in Table 3.2. A map is completely constructed if the topology of the route system is fully captured; we do not count the time to completely label all shops. The

	8 explorers	4 explorers	2 explorers
Floor 1	14 minutes	34 minutes	1.5 hours
Floor 2	12 minutes	24 minutes	1 hour
Floor 3	5 minutes	15 minutes	45 minutes

Table 3.2: Map construction times with different number of explorers.

results in Table 3.2 show that GROPING can construct a rather complicated map in less than

Figure 3.15: Floor maps and the corresponding GROPING maps of 3-floor shopping mall.

one hour. According to indoor map uploading manual [64], Google needs about 24 hours to refurbish the map contributed by a venue before it can be used by GMI. Note that this still does not include the collecting fingerprint process.

3.6.3 Accuracy of GROPING Location Estimation

As GROPING needs to first find out the current location of a strayed user before being able to navigate him/her, a sufficient accuracy in localization is very important. In Fig. 3.16, we report the statistics of the data accumulated during our field studies on the GROPING navigation service (reported later). Fig. 3.16(a) shows the localization errors as a function of the number of samples (i.e., the time a user spends on walking). Five exemplar traces were obtained by

Figure 3.16: Localization accuracy of GROPING. Comparisons are made in a large test site (b) and a small one (c).

different users from five distinct locations, and the GROPING location estimator starts to report location only after 20 samples (4 seconds). We can see that initially the errors can be large but approximately after 150 samples (30 seconds), the algorithm converges with errors less than 5 meters. There are also cases where the initial 20 samples are sufficient to obtain accurate location estimations.

Fig. 3.16(b) depicts the distribution of location errors for all our experiments. It shows that after 30 seconds, 90% of the errors are within 5 meters, coinciding well with Fig. 3.16(a). Only a very small fraction is between 10 to 15 meters. As GROPING is a navigation service, such an accuracy is sufficient and the sporadic large errors can be visually corrected, because two adjacent units could well be spaced anywhere between 5 to 15 meters in large scale entertaining facilities. We further compare GROPING with GMI and our implementation of FreeLoc [2] in Fig. 3.16(b). While GMI always performs the worst, FreeLoc shows comparable accuracy with GROPING in the first 10 seconds, but much lower when it comes to 20 seconds.

As the comparisons with RADAR [17] and Horus [1] are done in a different test site, we present them separately in Fig. 3.16(c). While all the three systems achieve very good localization accuracy due to the small area of the test site (about 1200 m^2), GROPING still outperforms its competitors. In addition, GROPING has an accuracy similar to what are claimed in [7, 35, 53], without the need for a WiFi infrastructure or a map. Although UnLoc [23] performs better than GROPING, GROPING, using only two inertial sensors and requiring far less user interventions, is a lightweight system consuming much lower energy, as we have discussed earlier.

3.6.4 User Feedback on GROPING Navigation

After the maps were constructed, we let the remaining 12 participants (except the 8 map explorers) to install the GROPING client on their own smartphones (which include Samsung Galaxy S2/S3/Ace Plus, Sony Xperia S, and HTC One X). The whole evaluation process has lasted for several weeks with participants visiting our test site sporadically and performing hundreds of tests (each test involving an arbitrary source-destination pair). In Fig. 3.17, we show a participant walking under the navigation guidance, as well as the screenshot of his phone at that moment.

The feedback provided by a participant after each test included two points:

1. Was the navigation process successful or not?
2. A mark on the satisfactory level.

A navigation succeeds if it guides a participant to be within the visual range of the destination in 10 minutes. Participants all give a mark (1 to 10) on the satisfactory level to represent their navigation experience, and they also provide us with comments to explain their marks.

The outcome shows that *all* the tests ended up successfully, and Fig. 3.18 illustrates the distribution of the satisfactory levels. Apparently, users are rather satisfied with their navigation experiences. There are a few cases where the satisfactory level falls between 1 to 6, which are often caused by the initial “jumping” of the current locations and also by our rudimentary user interface that does not allow map rotation. While the interface issue can be easily handled, our temporary solution for preventing location jumping is to delay the display of the current location (hence the navigation route). However, users may still feel unsatisfied as they have to walk “blindly” for tens of seconds. In our future work, we could combine a (one-time)

Figure 3.17: Snapshots of the GROPING navigation.

WiFi-based localization with GROPING, such that GROPING may quickly obtain an accurate location to start with, while still retaining the benefit of geomagnetism-based navigation.

Figure 3.18: Navigation satisfactory level from 1 to 10.

3.7 More on The Map Generation

Unlike prior proposals that often perform evaluations on simple building floors [7, 54], we choose venues with fairly complicated floor maps containing loops, as shown in Fig. 3.19 for the first floor of our large test site. While the left figure in Fig. 3.19 shows the ground truth

Figure 3.19: The floor map of a shopping mall. The slight difference between this map and that shown in Fig. 3.15 is caused by a renovation in-between these two sets of experiments.

floor map, the right figure is the skeleton (virtual) map we are aiming at generating by GROPING.¹ The floor has a polygonal shape with hallways forming many loops. It also contains conjunction points of either open area type (bottom-left corner) or turns with arbitrary degrees (bottom-right corner) instead of exact right angles in many existing tests. We first illustrate how GROPING’s map generation module works on this floor map in Fig. 3.20, and then we summarize the performance of map generation in Fig. 3.21, along with discussions on its implications. In Fig. 3.20(a) to (h), we use dashed lines to illustrate the actual walking trajectories of users and solid lines to represent the virtual trajectories “seen” by GROPING (through interpreting the sensor data). There are two major differences between these two set of trajectories. First, the virtual trajectories tend to be less straight, which is mainly due to the small angle estimation errors at the conjunction points. Second, the virtual trajectories are often shorter; the reason is that, as we only use the gyroscope readings collected from the conjunction points to estimate

¹One hallway (the hatched area in the right figure), though shown by the real map, is identified by our first team as blocked for renovation. Should such a “stale” map be used for indoor localization [7, 40, 54], it may incur large location errors. However, GROPING’s crowdsensing map generation can handle such situations automatically: after the renovation was finished (see Fig. 3.15), the map were updated as some users are bound to pass through the new hallway.

the angle, we assume the length of each conjunction to be negligible for now. The two errors are handled by *relaxLoop*(M) at a later stage.

Figure 3.20: Virtual map generation using eight trajectories and the associated magnetic fingerprints.

The stitching procedure is shown by Fig. 3.20(i) to (p). The procedure runs pretty smooth from (i) to (k), but certain distortion can be observed in (l) (fourth step). Now the whole outer loop has been explored, but it is yet to be determined whether the two end points actually coincide. When the sixth trajectory is introduced, the loop is closed but it is geometrically distorted (which results from the aforementioned two errors). Consequently, $relaxLoop(M)$ kicks in to make proper adjustments. The adjustment shown by Figure 3.20(n) is that i) each conjunction point is expanded based on the number of sample points involved, which results in the detection of the open space, and ii) each angle is computed as the average among all associated fingerprints. For brevity, the plots stop at Fig. 3.20(p) with some open loops, but they are closed with a couple of new trajectories in our experiments.

As a probabilistic algorithm, GROPING's map generation is prone to erroneous stitching. In Fig. 3.21, we use *precision* and *recall* as two metrics to evaluate the performance of the map

Figure 3.21: Statistical evaluation of the map generation performance.

generation. As discussed in Section 3.4.1.4, the performance of GROPING's map generation is controlled by `defiThreshold`, the definiteness threshold. Fig. 3.21 shows that, when increasing the threshold, we have a higher precision but lower recall. As a higher precision implies lower *false positive* rate and a higher recall suggests a lower *true negative* rate, we prefer to have a large value of `defiThreshold`, simply because true negative (i.e., overlapped segments are not detected) is almost harmless apart from wasting data. Therefore, we set `defiThreshold = 0.7` to achieve a precision of 90%, which in turn wastes about 60% of trajectories.

The remaining 10% of false positives may affect the virtual map of GROPING in two ways.

First, it associates fingerprints that do not belong to a segment with that segment. Second, it creates hallways that do not exist on the real map. In reality, we may remove those non-existent hallways (hence the corresponding trajectories, along with the fingerprints) using crowdsensing (again). The basic idea is that non-existent hallways will never be passed by any future user, while existing hallways are bound to have some users walk along them. Therefore, GROPING keeps monitoring the user appearance on individual hallways, and it removes a hallway (and the corresponding trajectory) if no one appears on it for a long time. This also helps to detect a newly renovated hallway as illustrated by comparing Fig. 3.19 with Fig. 3.15.

3.8 More on The Ambient Magnetic Field

In this section we present more results in understanding the stability of the ambient magnetic field. Since the application environments of GROPING may involve constant human movements around the smartphone client, we want to know how such movements affect the magnetometer readings taken by the smartphone. To this end, we measure the magnetic field at five different locations. At each location, the phone has a fixed orientation, but we let one person to move arbitrarily within each of the four quadrants centered at the phone location, namely West, South, East, and North. The distance between the person and the phone is limited within 2 meters. A reading is taken by the phone for each of such quadrant-limited movements.

We plot the 3D magnetic field vectors in Fig. 3.22. Fig. 3.22(a) shows the readings collected at these five locations altogether. Apparently, the differences in magnetometer readings caused by human movements within different quadrants are negligibly small so that they do not affect the location discriminating ability of the ambient magnetic field. To give a closer look at these differences, we plot the reading for individual locations in Fig. 3.22(b)-(f), respectively. Clearly, the variances of the ambient magnetic field may reach $100\mu\text{T}$ (see the scale of Fig. 3.22(a)), whereas those caused by human movements never go beyond $3\mu\text{T}$ (see the y-axis of Fig. 3.22(d) as an example). These experiments allow us to firmly conclude that human movements have very limited influence on the ambient magnetic field, so they do not affect the localization accuracy achievable by GROPING.

(a) Ambient magnetic field readings.

(b) Readings at location 1

(c) Readings at location 2

(d) Readings at location 3

(e) Readings at location 4

(f) Readings at location 5

Figure 3.22: Ambient magnetic field reading, in μ -Tesla (or μ T), taken by a smartphone at five different locations. For each location, we use West, South, East, and North to indicate the quadrants (centered at the smartphone) within which the “interfering” person is moving.

3.9 Summary

Whereas a plethora of proposals on WiFi-based indoor localization systems have been proposed, we believe that an indoor navigation service may require features that are not provided by these existing localization systems. Motivated by the incompetent navigation service of Google Maps Indoor (GMI), we aim to eliminate the heavy reliance on a WiFi infrastructure and also on contributory floor maps can be beneficial to indoor navigation.

To this end, we proposed GROPING, an all-in-one system that includes map generation, localization, and navigation. GROPING relies on the geomagnetic field to characterize indoor locations. This allows GROPING to i) utilize crowdsensing for magnetic fingerprinting and for constructing a floor map from an arbitrary set of walking trajectories, and ii) to perform lightweight localization and hence navigation based on magnetic fingerprints and the constructed maps. Evaluations and user studies in a large shopping mall with 20 participants have demonstrated the high usability of GROPING's navigation service.

Whereas WiFi-based indoor localization systems show disadvantage in energy efficiency and fingerprint stability, the higher dimensionality of WiFi fingerprints, if properly used, may still offer better location discriminability than magnetic fingerprints. Therefore, we are considering the possibility of a hybrid system combining both technologies in our future work. Moreover, we also plan to make GROPING more autonomous by minimizing the required user interventions.

Chapter 4

A Dual-Sensor Enabled Indoor Localization System with Crowdsensing Spot Survey

In the study of ambient magnetic field fingerprint, we are surprised by the fact that scalability of ambient magnetic field based approach is not satisfied comparing to WiFi based approaches. Much Fewer dimensions in geomagnetic field fingerprint comparing to WiFi fingerprint hinders the discrimination of locations in a large area. By further exploration, we find that interesting dual properties naturally existed in ambient magnetic field fingerprint and WiFi fingerprint. This fact motivate us to design a new system that combine the two techniques. In this chapter, we present MaWi, a dual-sensor enabled indoor localization system that combine two self-contained but complementary localization techniques: Wi-Fi and Ambient Magnetic Field. MaWi is designed to achieve high localization accuracy with minimum human effort.

4.1 Introduction

As increasing urbanization forces people to stay more often at indoor environments, locating and navigating people in complex constructions (e.g., airports and shopping malls) becomes a crucial problem. Furthermore, government and business also benefit from accurate user location information in precise information pushing. Luckily, rapid spread of high performance smart phones and wide 3G/Wi-Fi network access have caused an explosion of mobile sensing applications. Since smart phone has become an indispensable device in people's daily life,

an indoor localization service through smart phone sensing can be handily deployed without extensive efforts.

To address the challenge of indoor localization, many systems have been proposed in recent years [1, 2, 3, 4, 5, 6]. However, smart phone based systems achieving both scalable and high accuracy are still missing. On one hand, while high accuracy has been achieved by some fingerprint-based (or empirical) systems [1, 2], they share the common prerequisite to entail heavy labor in building fingerprint database (a set of spots with associated signal fingerprints). Such a laborious spot survey process hinders the scalability in deployment. On the other hand, model-based systems [3, 4, 5, 6] avoid the spot survey by using propagation model to infer distances and trilateration to locate users. Unfortunately, these systems have either low accuracy [3] or high computational/infrastructural demand: [4] requires the knowledge of the distances between all users and all Wi-Fi access points (APs) to compute a single user's location; [5] requires software access to all APs; [6] relies on a deployment of dedicated acoustic beacon.

Because spot survey (deemed necessary for achieving high accuracy) hampers scalable deployments, recent proposals [7, 8, 9, 65] suggest to distribute the intensive labors through crowd-sourcing, in which fingerprint databases are collected opportunistically by a large population. Whereas these technologies may have potential to make large deployment possible, opportunistic spot survey cannot warrant fingerprint quality: insufficient samples at a given spot and low density in sampled spots are both potential problems.

Combining different localization approaches [27] could be a way to build a ideal solution. However, it is hard to find the approaches which can supplement the weakness as well as augment the strength. For example, in smart phone based localization, one straight-forward combination is Wi-Fi and cellular signals. But the intent of this combination is more to supplement the vacancy of Wi-Fi signal in specific area, instead of increasing accuracy. Because both Wi-Fi and cellular signal share the same property of wireless transmission model, and therefore suffer from the same weakness of instable signal strength. If we can find two localization approaches with complementary properties, we could be one step closer to a ideal localization solution.

In this work, we aim to build a highly accurate and yet scalable localization system by relying only on a light-weight fingerprint database. To this end, we study performance of fingerprints in two aspects: locational discrimination and temporal stability. As defined in 2.1.1, *Locational discrimination* represents the fingerprint's ability to tell a location from others, and

temporal stability is the variance of fingerprint sampled from the same spot at different times. We find that two kinds of fingerprints, Magnetic Field and Wi-Fi signal, have interesting complementary performance in locational discrimination, and that magnetic field has outstanding temporal stability. Combining them would make highly accurate localization possible even with a light-weight fingerprint database.

Based on our studies, we propose MaWi, a smart phone based indoor localization system using Magnetic field and Wi-Fi as fingerprints. The two fingerprints are used in a “duet” manner such that they complement each other. Through this smart combination, MaWi achieves a scalable deployment due to its low demand on the fingerprint database, while getting very competitive localization accuracy compared to state-of-the-art systems. Furthermore, MaWi is light-weight as it requires no dedicated devices (e.g., [18]) or adaption of existing infrastructure (e.g., [4]). Finally, MaWi is friendly for widely deploying because it uses a smart phone as both the survey device and the localization client. Our major contributions in designing MaWi are:

- We analyze properties of Wi-Fi signal and ambient magnetic field, and we, for the first time, identify their complementarity in indoor localization.
- We present a scalable indoor localization system, MaWi, aiming to reduce the deployment cost for better scalability and to achieve high localization accuracy on normal smart phones.
- We deploy MaWi in an office building, a library, and a shopping mall with complicated floor plans. Our extensive experiments demonstrate a low deployment workload and high localization accuracy.

4.2 Fingerprint Studies

In this section, we present our studies on Wi-Fi and Magnetic field in both time and space dimensions. This serves as the major motivation of our proposal. Our test region is an 800 m² office area shown in Figure 4.1.

4.2.1 Wi-Fi Signal

According to [27], two main factors hamper localization accuracy of fingerprint-based systems: 1) similar signals at multiple locations, and 2) transient measurement of signals at the same

Figure 4.1: Test region for fingerprint studies.

location. High signal similarity reduces locational discrimination, and transient measurement weakens the ability of properly classifying on-line observed fingerprints. While the former is caused by particular deployment of APs and indoor structures, the latter mainly attributes to limitations of measuring devices. Our studies intend to reveal how exactly a Wi-Fi localization approach is affected by these two factors. To the best of our knowledge, no extensive study has been performed in this aspect. Following conventions, we define *Wi-Fi fingerprint* as a vector of RSSs (Receive Signal Strength) from a set of detectable APs, easily measurable by normal smart phones, and we measure *similarity* by negative Euclidean distance.

4.2.1.1 Single Concurrent Sample per Spot

We evaluate the fingerprint similarity with respect to the number of APs in Figure 4.2. Color temperature is proportional to the fingerprint similarity between the location marked by the pentagram and the remaining area. Interfered by indoor construction, the area bearing similar fingerprints (the red area) for one AP appears like an irregular ring. With the increase of APs, the red area shrinks to a small zone nearby the pentagram. This experiment shows that given sufficient APs, similar fingerprints are prone to be observed in close-by location. Therefore, we may effectively tell two distant locations from each other using Wi-Fi fingerprints.

4.2.1.2 Multiple Samples per Spot

Since fingerprints in database and those collected on-line are sampled at different times, we also extend our study to the time dimension by sampling fingerprint at each spot for 30 minutes at a rate of 10 samples per minute. The five typical sampling spots are shown in Figure 4.1. We use confusion matrix in Figure 4.3(a) to illustrate fingerprint similarity of different locations

Figure 4.2: Fingerprint similarity with one sample per spot given different numbers of APs.

and times. The time dimension is represented by five arbitrary samples within the 300 samples at each location. As we expected, higher similarity exists between nearby locations (e.g., L4 and L5); it is so high that Wi-Fi fingerprint can hardly tell close-by locations from each other.

Figure 4.3: Wi-Fi locational discrimination.

To further demonstrate the relation between fingerprint similarity and location accuracy, we plot in Figure 4.3(b) the similarity between fingerprints collected at L1 and locations of varying distances from L1. In the worst case, fingerprint collected at 12 meter away can still

be mistaken classified as L1, shown by the overlapping in similarity ranges. Obviously, Wi-Fi fingerings cannot support a fine-grained location differentiation.

4.2.1.3 Can More Samples Help?

Figure 4.4: Localization error CDF.

Due to the temporal variance in Wi-Fi fingerprints sampled at the same spot, the boundary of fingerprints from close-by locations are blurred. To cope with this situation, existing systems sample many fingerprints at a spot and take either certain statistical quantities [1] or the most frequent fingerprint [2] as the *representative fingerprint*. As it is general believed that the larger sample size leads to better localization accuracy, we intend to experimentally examine their relation.

In Figure 4.4, we plot the localization error under different sample size. Most frequent fingerprint is chosen as the representative fingerprint for a given spot [2]. The figure shows that localization accuracy is improved only marginally as the sample data increase, and the error bound remains to be about 17 meters irrespective of sample size. This might be caused by high instability in Wi-Fi signal strength received and other influence from environment. Therefore, **a different strategy is needed to achieve finer-grained localization.**

4.2.2 Ambient Magnetic Field

Geomagnetic field exists ubiquitously; it is "twisted" indoor by building structures and forms unique *ambient magnetic field*, which can identify indoor locations [66, 67]. Ambient magnetic field signal can be sensed by magnetometer embedded in smart phones, hence can be used as

fingerprint for indoor localization. We define *magnetic fingerprint* as a tuple (x, y, z) , the three dimension vector of magnetic flux. *Similarity* of magnetic fingerprints is computed as cosine similarity. In this section, we analyze the complementary natures of ambient magnetic field to Wi-Fi signal, demonstrating its precious value in indoor localization.

4.2.2.1 Temporal Stability

In contrast to Wi-Fi, ambient magnetic field shows great stability in time dimension. We perform studies in the same office area to compare the temporal stability of Wi-Fi with that of magnetic field. We choose 10 locations and collect both Wi-Fi fingerprint and magnetic fingerprint for 5 minutes, and we repeat this for ten rounds spreading over five different days. For each round, we compute the mean and standard deviation of the magnitude of fingerprints vectors, and we use $\frac{\text{mean}}{\text{standard deviation}}$ as the *stability index*. In Figure 4.5, we compare the stability indices of Wi-Fi and magnetic fingerprints in terms of their average values over ten rounds. It is obvious that even the most unstable case of the magnetic fingerprint is far better than that of Wi-Fi fingerprint. Such a difference obviously stems from the sources that generates these two types of signals.

Figure 4.5: Temporal stability comparison.

4.2.2.2 Advantage in Differentiating Close-by Location

The superiority of magnetic field in temporal stability suggests that it may have an advantage in locational discrimination, as the distributions of fingerprints sampled at different locations would have far less chances to overlap. Moreover, the similarity of magnetic field is not correlated with distance, as shown by Figure 4.6, where similar magnetic fingerprints are observed at L1 and L4, rather than at L1 and L2. This can be somewhat explained by that tiny construction differences may drastically re-shape the magnetic field.

Figure 4.6: Magnetic field confusion matrix.

To further demonstrate the different characteristic of Wi-Fi signal and magnetic field in differentiating close-by locations, we apply spatial tessellation to visualize difference. We sample Wi-Fi signal and magnetic field from 50 spots (the black dots in Figure 4.7(a)) within the test region shown in Figure 4.1. Then we build a triangle mesh graph through Delaunay Triangulation with sample spots as vertices. For every edge (v_1, v_2) in the graph, we make a *confusion point* (the blue dots in Figure 4.7(a)) and assign its value as the percentage of cases where fingerprints sample from v_1 are wrongly classified to v_2 and vice versa. Color temperature in Figure 4.7(b) and 4.7(c) represents the values of those confusion points. We can see that magnetic fingerprint shows much lower confusion in close-by locations than Wi-Fi, which makes it a better method to differentiate such locations.

4.2.2.3 Fingerprint Similarity in Far Apart Location

Ambient magnetic field is naturally not scalable to large indoor areas. In a complex indoor structure, a few similarly constructed areas that are far apart from each other may lead to similar magnetic fields, which can attributes to that magnetic fingerprints only contain three components, as opposed to the many components of the multi-dimensional Wi-Fi fingerprints. Although employing sequence of magnetic sample data collected from a trajectory instead of a spot as fingerprint may mitigate the problem to some extent, it cannot eliminate all similar fingerprints from far apart locations. For example, Figure 4.8 shows two magnetic signal sequences collected from two far away corridors with similar structure. Moreover, using long

(a) Sample point and confusion point distribution

(b) Locational confusion of magnetic field

(c) Locational confusion of Wi-Fi signal

Figure 4.7: Locational fingerprint similarity.

fingerprint sequences as a new type of fingerprints requires a user to keep moving and also results in delay in localizations; both are not desirable for a localization service.

Remarks The intriguing complementary nature of Wi-Fi signal and ambient magnetic field motivates us to design an indoor localization system that use hybrid Wi-Fi and magnetic fingerprints as location discriminators. Such a system is bounded to be more accurate than existing system using either of the fingerprints alone, and it also has the potential to work along with

Figure 4.8: Magnetic signal sequences collected by a walking user holding smart phone in hand in two corridors.

light-weight fingerprint databases for scalable deployments.

4.3 Zone Discrimination

Inspired by our analysis in Section 4.2.1, we define *zone discrimination* problem in this section and propose an algorithm to solve the problem, fully exploiting the advantage of Wi-Fi fingerprint in coarse-grained indoor localization.

4.3.1 Problem Formulation

Consider an ideal fingerprint database containing very dense *survey spots* and a large number of fingerprints collected at each spot (created by conventional war-driving). When a localization algorithm accepts an fingerprint sampled on-line as its input, our results in Section 4.2.1 suggests that the algorithm may at the best return an area (or a set of spots the are close to each other) based on whether the on-line fingerprint exists in the sample set associated with a given spot or not. Formally speaking, for each incoming on-line fingerprint, there exists a

set of *matching fingerprints* in the ideal database. An algorithm aiming at *zone discrimination* returns a *similarity zone* $\mathcal{A}$ that contains p fraction of matching fingerprints. An optimal zone $\mathcal{A}_o$ is the one whose area is minimized. Obviously, given the ideal database and the *confidence coefficient* p , $\mathcal{A}_o$ is the best location indication we can expect.

To obtain such an ideal war-driving database, e.g., with 100 samples per survey spot, and survey spots arranged 1 meters apart in a $100 \times 100\text{m}^2$ area, one need to work without any sleep or rest for more than one month¹. To improve the deployment scalability, we cannot afford to obtain $\mathcal{A}_o$ that in turn entails an ideal database. So we need an algorithm working with a light-weight fingerprint database containing sparser survey spots and far fewer fingerprints sampled at each spot (even one fingerprint per spot in the extreme case). Now facing such a light-weight database, a well-performed zone discrimination algorithm needs to minimize the difference between $\mathcal{A}$ and the *optimal solution* $\mathcal{A}_o$, which translates into maximizing:

- Safety: the ratio of $\mathcal{A}_o$ that is included into $\mathcal{A}$:

$$\text{Safety} = |\mathcal{A}_o \cap \mathcal{A}| / |\mathcal{A}_o|$$

- Effectiveness: the ratio of $\mathcal{A}$ that is inside $\mathcal{A}_o$:

$$\text{Effectiveness} = |\mathcal{A}_o \cap \mathcal{A}| / |\mathcal{A}|$$

4.3.2 Circle Algorithm (CA): A Naive Approach

This naive approach finds a survey spot that contains the most similar fingerprint compared with the on-line sample, and defines $\mathcal{A}$ as a circle area centered at that spot. The radius of circle is a parameter set according to p , i.e., a bigger p induces a larger radius. This algorithm utilizes local similarity feature of Wi-Fi signal (see Section 4.2.1.1), but it is inflexible because the shape of $\mathcal{A}$ is defined universally rather than basing on environment. For example, an area that can never generate the observed fingerprint (e.g., isolated by thick walls) could be included into $\mathcal{A}$ by CA, resulting in low effectiveness. More illustration can be found later.

¹The highest Wi-Fi RSS sampling rate on average that can be achieved by a most up-to-day smart phone (e.g., Samsung Galaxy S3) is 0.33 Hz.

4.3.3 Similarity Voronoi Algorithm (SVA)

To better tackle the zone discrimination problem, we propose Similarity Voronoi Algorithm (SVA). SVA generates a Voronoi graph on the surveyed area with survey spots as Voronoi sites, and it returns $\mathcal{A}$ as a union of carefully chosen Voronoi cells. SVA takes *similarity threshold* θ as a parameter to decide which cell to be put into $\mathcal{A}$. **Algorithm 4** shows the main steps of SVA. Line 1 builds Voronoi graph with survey spots as sites. After that, SVA adds a cell V_s into $\mathcal{A}$ if the similarity between an on-line fingerprint F and the fingerprint $F(s)$ associated with site s goes beyond θ (line 4).

Algorithm 4: Similarity Voronoi Algorithm (SVA)

Input: Survey spot set $\mathcal{S}$, on-line fingerprint F , similarity threshold θ

Output: Similarity zone $\mathcal{A}$

```

1  $\{V_s\}_{s \in \mathcal{S}} \leftarrow \text{Voronoi}(\mathcal{S}); \mathcal{A} \leftarrow \emptyset$ 
2 forall the  $s \in \mathcal{S}$  do
3   | if  $\text{similarity}(F, F(s)) > \theta$  then  $\mathcal{A} \leftarrow \mathcal{A} \cup V_s$ ;
4 end
```

4.3.4 Comparing CA with SVA

We evaluate safety and effectiveness of both CA and SVA under different radius (CA), θ (SVA), and p . The results are shown in Figure 4.9. The intersection point of safety and effectiveness denotes the highest performance that algorithms can achieve under their parameters. Under both cases of $p = 0.5$ and $p = 0.8$, SVA significantly outperforms CA as SVA reaches up to 0.8 for both safety and effectiveness while CA only achieve less than 0.4. We choose one case to illustrate the similarity zone generated by SVA and circle algorithm in Figure 4.10. Color map shows the appearing frequency of the on-line fingerprint. Red dashed area is the similarity zone generated by SVA, which is much smaller than the black dashed circular area by CA, and is also more reasonable according to fingerprint appearing frequency.

For evaluation purpose, optimal solution $\mathcal{A}_o$ is obtained by an algorithm running on a real war-driving fingerprint database. To get the database, we select survey spots at 1 m interval within the test region and sample 1200 Wi-Fi fingerprints (in 1 hour) for each spot.

Figure 4.9: Wi-Fi spatial discrimination.

4.4 A Hybrid Localization Approach

Equipped with SVA as a coarse-grained localization method, we are now ready to present the design of MaWi localization algorithm in this section. In order to achieve high accurate localization using only a light-weight fingerprint database, we propose a revised particle filter to make the best use of the complementary nature of Wi-Fi and magnetic fingerprints.

4.4.1 How to Utilize Fingerprints?

When using particle filter [68] for locating an object, *particles*, designed to model the states of the object, are a set of tuples (l, v, w) , where l denotes location, v denotes velocity, and w denotes the weight of the particle. The particle weight stands for the probability that a particle correctly traces object, and it is periodically updated by similarity between *observations* (on-line fingerprints) and records in the (fingerprint) database.

Whereas of both Wi-Fi and magnetic fingerprints are collected on-line by MaWi, Wi-Fi's

Figure 4.10: $\mathcal{A}$ of SVA and CA when $p = 0.8$ and safety is tuned to 0.8. Appearing frequency of fingerprint is plotted in background.

poor temporal stability (as evaluated in Section 4.2.1.2) may harm the stability of the filtering process. Moreover, the survey spots sparsity also makes Wi-Fi fingerprints inadequate to serve as observations¹, as survey spots can be too sparse to infer a meaningful location of the object.

For these reasons, MaWi only takes magnetic fingerprints as direct observations, while applying the similarity zones generated by SVA (with Wi-Fi fingerprints as input) as constraints. Our revised particle filter updates particles according to 1) similarity between on-line magnetic fingerprints and records in database, and 2) whether the particles locate within certain similarity zones. Through this design, we exploit both the stability of ambient magnetic field and the Wi-Fi's coarse-grained location discrimination. In the following, we first present a basic version of our particle filter method in Section 4.4.2, then we introduce the full version involving Wi-Fi fingerprints in Section 4.4.3.

4.4.2 Revised Particle Filter: A Basic Version

Particle Generation Initially, particles are generated at magnetic survey spots with different velocity in both forward and backward directions. By taking both location and velocity as state of particles, MaWi does not make any assumption on whether the object moves or not. Details are summarized as follows:

1. At each magnetic survey spot l , generate five particles with five different velocities:

$$\{(l, v) | v \in \{2v', v', 0, -v', -2v'\},$$

v' denotes velocity taken in survey process.

¹As opposed to 0.33 Hz of Wi-Fi scanner, magnetometer has sampling rate up to 50Hz. Consequently, magnetic field survey spots are much denser than Wi-Fi survey spots even in a light-weight fingerprint database.

Figure 4.11: MaWi Localization Illustration.

2. Assign weight w uniformly for all particles.

Particle Updating Particle updating is activated when new on-line magnetic fingerprint (an observation) F_m arrives. In every updating, we update location of particles according to their last location l and velocity v , and we update weights of particles as:

$$w = \eta \times \text{similarity}(F_m, F_m(l)) \times w',$$

where $F_m(l)$ denotes the magnetic field in fingerprint database collected at location l , w' denotes the current weight, $\text{similarity}(F_m, F_m(l))$ computes the similarity between F_m and $F_m(l)$, and η is a normalization factor. After every updating, MaWi returns an estimated location l_e : the location of the particle with the highest weight. With increasing observations, l_e should gradually go close to object's ground truth location.

4.4.3 Particle Updating with Zone

We now upgrade the basic version by incorporating SVA into particle updating. Apart from F_m , an observation may also include F_w as an on-line Wi-Fi fingerprint. Taking F_w as an

input, SVA will returns a zone $\mathcal{A}(F_w)$. Given that particles are generated randomly in a large number, density of particles can be considered uniform. According to the definition of zone discrimination (Section 4.3.1), the sum of weight of particles inside $\mathcal{A}(F_w)$ should be p percent of the total weight, and those outside should be $1 - p$. So we update the particle weight by adding a new parameter δ :

$$w = \eta \times \text{similarity}(F_m, F_m(l)) \times w' \times \delta, \quad (4.1)$$

where $\delta = p/\omega_{in}$ if $l \in \mathcal{A}(F_w)$; otherwise $\delta = (1 - p)/\omega_{out}$. We use ω_{in} (resp. ω_{out}) to denote the sum of weights of particles located inside (resp. outside) $\mathcal{A}(F_w)$. Through the new updating equation, we involve Wi-Fi zone discrimination as a probabilistic constraint on the basic particle filter to complement the observations solely based on magnetic fingerprints, hence increasing localization accuracy.

In Figure 4.11 we illustrate the process of localization for both a moving object and a static object. Orange dots denote particles, and their sizes indicate weights. Both cases start with particles of uniform weight. In the moving case, $\mathcal{A}(F_w)$ (blue area) changes with the position of object, as F_w is changed by object motion. After a few rounds of updating with both magnetic fingerprint and $\mathcal{A}(F_w)$, a particle that is at object's ground truth location and shares the same velocity with the object gets the highest weight. In static case, despite of small changes in $\mathcal{A}(F_w)$ caused by the temporal variance of F_w , the static particle located at object's standing location will get the highest weight after a few updating rounds due to stable magnetic fingerprint.

4.4.4 Confidence Coefficient

From Section 4.4.3, we see that the confidence coefficient $p \in (0, 1]$ is a crucial parameter. On one hand, if $p = 1$, $\mathcal{A}(F_w)$ becomes very large, containing every possible location of the object. So MaWi becomes a particle filter localization approach relying almost solely on magnetic field. On the other hand, $p \rightarrow 0$ makes an empty $\mathcal{A}(F_w)$, since it is almost impossible to have an on-line fingerprint exactly the same as records in database, given the instability of Wi-Fi signal. All particles keep the weight updated by magnetic fingerprints, which makes MaWi, again, a localization approach relying solely on magnetic field. In order to make $\mathcal{A}(F_w)$ small enough to utilize Wi-Fi's coarse-grain location discrimination ability, while at the same time, large enough to tolerate the Wi-Fi's instability, we have to set p moderately. In our experiment

we find that $p = 0.8$ generates the best localization performance. Therefore we set $p = 0.8$ by default.

4.4.5 Particle Re-Sampling

To trace an object when it changes velocity, and also to remove particles of very low weight (i.e., no need to be updated), we need to periodically re-sample particles. In re-sampling, we randomly select a certain number of locations according to the weight of previous particle that are located around. In other words, locations near a higher weight particle get more chance to be selected. New particles share weight with previous particles at the same location; other steps are the same as the particle generation phase (see Section 4.4.2). Through re-sampling process, we can model object's varying moving state, and also give MaWi a chance to reduce the unnecessary computation cost.

4.4.6 Algorithm Description

We summarize MaWi localization in **Algorithm 5**. MaWi initializes particles according to particle generation process in Section 4.4.2, and puts them into set $\mathcal{P}$ (lines 1 to 3). An update process is triggered by new observation (line 5), and similarity zones are computed accordingly (line 6). During the updating process (lines 7 to 17), MaWi first updates particles' locations according to the movement model, then it adjusts all particle weights according to Equation (4.1). Periodically, the quality of the particles is verified, and if necessary, MaWi performs a particle re-sampling based on existing particles (line 18). The updating process continues until certain convergence condition is satisfied (line 19). The convergence condition can be based on a threshold on the particle weight (i.e., triggered by an event that certain particle weight goes beyond that threshold), or it can be triggered by a timeout (e.g., a few seconds). Upon completion, MaWi returns the location of the particle whose weight is the largest to indicate an estimated location (line 20).

4.5 MaWi System Overview

In this section, we briefly summarize MaWi system architecture and also discuss some implementation details. As illustrated in Figure 4.12, MaWi has two types of user: Surveyor who supplies fingerprint data, and Strayer who needs self-locating. Fingerprints from surveyor are

Algorithm 5: Localization Algorithm**Input:** Wi-Fi survey spot set $\mathcal{S}_w$, magnetic survey spot set $\mathcal{S}_m$, confidence coefficient p **Output:** Estimated location l_e

```

1 forall the  $l \in \mathcal{S}_m$  do
2    $\mathcal{P}.add([(l, 2 \times \text{backward}, \frac{1}{5|\mathcal{S}_m|}), (l, \text{backward}, \frac{1}{5|\mathcal{S}_m|}), (l, \text{static}, \frac{1}{5|\mathcal{S}_m|}),$ 
    $(l, \text{forward}, \frac{1}{5|\mathcal{S}_m|}), (l, 2 \times \text{forward}, \frac{1}{5|\mathcal{S}_m|})])$ 
3 end
4 while true do
5    $F_w \leftarrow \text{getOnlineWiFi}(); F_m \leftarrow \text{getOnlineMagn}()$ 
6    $\mathcal{A} \leftarrow \text{SVA}(\mathcal{S}_w, F_w, p); \omega_{in} \leftarrow 0; \omega_{out} \leftarrow 0$ 
7   forall the  $o \in \mathcal{P}$  do
8      $o.\text{updateLocation}()$ 
9      $o.w \leftarrow o.w \times \text{similarity}(F_m, F_m(o.l))$ 
10    if  $o.l \in \mathcal{A}$  then  $\omega_{in} \leftarrow \omega_{in} + o.w$ ;
11    else  $\omega_{out} \leftarrow \omega_{out} + o.w$ ;
12  end
13  forall the  $o \in \mathcal{P}$  do
14    if  $o.l \in \mathcal{A}$  then  $o.w \leftarrow \frac{o.w \times p}{\omega_{in}}$ ;
15    else  $o.w \leftarrow \frac{o.w \times (1-p)}{\omega_{out}}$ ;
16  end
17   $\mathcal{P} \leftarrow \text{normalize}(\mathcal{P})$ 
18  if needResample then  $\mathcal{P} \leftarrow \text{resample}(\mathcal{P})$ ;
19  if converge then
20     $o_e \leftarrow \arg \max_o (o.w | o \in \mathcal{P}); \text{return } l_e \leftarrow o_e.l$ 
21  end
22 end

```

---

labeled by Fingerprint Labeling Module, and then stored into Fingerprint Databases. On receiving a localization request, Revised Particle Filter (discussed in Section 4.4) draws fingerprints from databases and estimates a strayer's location based on on-line fingerprints.

4.5.1 Spot Survey

Conventionally, high accuracy localization approaches (e.g., Horus [1]) require a rich fingerprint database which could takes days even with the collaboration of multiple surveyors. To

Figure 4.12: System architecture.

enable a scalable deployment, MaWi has a simple spot survey mechanism. A surveyor indicates the trace he/she will take to collect fingerprint before survey starts. During the survey, smart phone records the fingerprints while surveyor walks. Recording process finishes when surveyor reaches the end of the trace. Assuming surveyor walking in uniform speed, MaWi uniformly arranges along the trace all the survey spots associated with fingerprints. Wi-Fi fingerprints are collected at 0.33 Hz, which is the highest frequency achievable by normal smart phones. Magnetometer records ambient magnetic field at 5 Hz. With a normal human walking speed of 1 m/s, Wi-Fi and magnetic field is recorded every 3 m and 0.2 m respectively.

Because MaWi has very loose requirements on fingerprint databases that it work with, it is compatible with legacy databases generated for other indoor localization approaches, such as [1, 2, 17]. Furthermore, MaWi's spot survey can be further enhanced (hence being more time efficient) through crowd-sourcing, a method already proposed in [7, 9, 65].

4.5.2 Redundant Data

In a fingerprint database (especially a legacy database), multiple fingerprints can be collected at the same location, or locations very closed to each other. We call those samples redundant data. Redundant data arte usually less valuable. It wastes computation and storage resources, especially when MaWi runs in smart phone devices with very limited resources. To address this problem, we pre-process fingerprint database by merging close fingerprint samples into one by, for example, averaging, and we set the new location to the gravity center of original sample locations (if they are slight different from each other).

4.5.3 Data Flow Patterns

MaWi offers two working modes of localization: server-side and client-side. The former performs computation on a backend server. In this mode the smart phone client consumes network bandwidth to upload observed fingerprint. The latter performs computation on a phone client, which may be confined by the limited computational resource on a smart phone. MaWi chooses the client-side mode with priority to avoid network delay in continuous localization, but users can change it to server-side mode if computation becomes a bottleneck.

4.6 Evaluation

We evaluate the performance of MaWi in this section, and we also compare it with well known indoor localization systems (including RADAR [17] and Horus [1]) in terms of localization accuracy.

4.6.1 Experiment Setup

We deploy MaWi in three test sites: an office area (Figure 4.1), a library (Figure 4.13(a)), and a shopping mall (Figure 4.13(b)). The dimensions of these sites and the number of available APs at each site are summarized in Table 4.1. We develop an Android application as the MaWi client (interface illustrated in Figure 4.14). In both survey and localization phases, we employ two smart phone models: Samsung Galaxy S2 and S3. Our backend server is a PC with Intel Xeon CPU E5-1650 3.20 GHz and 16 GB memory. We show a real localization process on

Figure 4.13: Floor plans of a library (a) and a shopping mall (b).

Figure 4.14: MaWi localization on a client. Sizes of red dots denote particles' weights, and time sequence is indicated at the upper-left corner of each figure. Obviously, the weight eventually concentrates at a single particle, which happens to be the user's ground truth location.

our MaWi client for five seconds in Figure 4.14. User stands still during the whole procedure at the lower side of the library (indicated by the particle at the 5-th second).

4.6.2 Light-Weight Survey Phase

We count the time we need to build a fingerprint database that can satisfy the requirement of MaWi. We employ only one surveyor holding smart phone to walk around the deployment area, while recording the fingerprints of both Wi-Fi and magnetic field at passed location. The results in Table 4.1 show that, even for a large area of 22500 m², we only need no more than 1 hour to get a usable database for localization.

4.6.3 Further Evaluations on SVA

As SVA takes confident coefficient p as input, we need to know how to derive a proper similarity threshold θ according to it for better performance. Generally, safety decreases with p and θ (Figure 4.15(a)), and effectiveness increases with p and θ (Figure 4.15(b)). Besides safety and effectiveness, we consider another metric *deviation*, which is the size of difference area

Table 4.1: Test Sites and Survey Times

Area	Size	Wi-Fi APs	Time (min)
Office	800 m ²	18 ± 5	6
Library	3100 m ²	43 ± 9	28
Shopping Mall	22500 m ²	27 ± 7	55

between optimal solution and SVA: $|\mathcal{A}_o \cup \mathcal{A}_{sva} - \mathcal{A}_o \cap \mathcal{A}_{sva}|$. We plot the relation between deviation, p , and θ in Figure 4.15(c). In all the three figures, we see that the best performance emerges at a strip area indicated by black dashed lines. In other words, θ should be set linearly according to p .

To analyze the relationship between SVA and MaWi’s performance, we run MaWi under different p and θ and plot the localization errors in Figure 4.16. Comparing it with Figure 4.15, we see obvious correlation: higher localization accuracy is achieved when SVA has smaller deviation, and higher safety and effectiveness (e.g., $p = 0.8, \theta = -50$). In our experiment, average localization error of MaWi is lower than 2.5m when deviation is below 0.3. Therefore, a good performance of SVA is crucial to the location accuracy of MaWi.

4.6.4 Performance Comparisons

In this section, we compare MaWi with other indoor localization approaches to demonstrate MaWi’s advantage in localization accuracy.

4.6.4.1 Magnetic Field and Wi-Fi

To show the power of combining Wi-Fi and Magnetic Field, we compare MaWi with a pure magnetic field localization and a pure Wi-Fi fingerprint approach (i.e., RADAR [17]) in terms of localization accuracy. According to [17], empirical approach performs better than model-based approach. Therefore we tune RADAR to run in its empirical mode, in which it take as the estimated location the survey spot containing the most similar fingerprint comparing with an on-line fingerprint. The pure magnetic field algorithm can be considered as a special case of MaWi where magnetic fingerprints are used to update particle weights. The performance comparisons are shown in Figure 4.17. MaWi clearly achieves the best localization accuracy. Pure magnetic field approach, although suffers from the limitation of magnetic field in locational discrimination (see Section 4.2.2.3), still performs better than RADAR, the pure Wi-Fi

Figure 4.15: SVA under different p and θ . Best result of SVA is achieved when θ changes linearly with p . At the same time Safety and Effectiveness are both relative high.

localization approach. In this and all the subsequent experiments, we set $p = 0.8$ and $\theta = -50$.

4.6.4.2 Comparison with Rich Database

While the previous comparisons are based on the same fingerprint database, we want to compare MaWi with a proposal relying on rich databases. We choose Horus [1] as our rival; it is said to be the most accurate fingerprint-based localization system [69]. Horus utilizes multiple samples to model the distribution of Wi-Fi RSS as a Gaussian distribution for every survey spot and takes it as fingerprint. To collect such rich fingerprints for the database, we regularly choose survey spots and employ several surveyors to collect fingerprints for 10 minutes at every spot. We compare MaWi working on low quality database with Horus working on richer ones in Figure 4.18. Even when sample number increases to 100 (5 minutes survey time) at each survey spot, Horus's performance is still worse than MaWi.

Figure 4.16: Localization Error under different p and θ .

Figure 4.17: Comparing localization accuracy among MaWi, pure magnetic field, and RADAR.

4.6.5 Localization Time Limit

In **Algorithm 5**, we can use timeout to force the convergence of the localization process. Longer time limit gives MaWi more chances to make particle weight converge, therefore should result in higher accuracy. We set the timeouts as 5 seconds, 8 seconds, and 12 seconds, respectively, and plot CDF of localization errors in Figure 4.19(a). MaWi's performance improves as the time limit increasing, which confirms our expectation.

4.6.6 Wi-Fi Access Point Number

Generally MaWi makes use of all detectable AP, but the number of detectable APs is determined by a particular venue. In Section 4.2.1.1, we shows that a better locational discrimination can be achieve with more APs. So the question is how much the performance of MaWi is affected by the number of APs. We run MaWi in environments with 5 to 20 detectable APs and

Figure 4.18: Comparing localization accuracy between MaWi and Horus under fingerprint data of different quality.

(a) MaWi localization errors under different timeouts. (b) MaWi localization errors under different numbers of APs.

Figure 4.19: Localization error evaluations

plot the localization error in Figure 4.19(b). Surprisingly, we find that median localization error decreases only slowly with number of APs (only decreasing 0.1 meter with 15 APs added), but the variance is much better controlled with more APs. So MaWi benefits more in stability than median error from increasing APs.

4.7 Summary

In this chapter, we propose MaWi, a smart phone based indoor localization system for high accurate localization and scalable deployment. Combining Wi-Fi signal and ambient magnetic field as its fingerprints, MaWi is able to work with a light-weight fingerprint database while

achieving high localization accuracy, hence unifying the two objectives that have long be contradicting each other. We implement MaWi on smart phones and deploy it in multiple venues; all the experiments have strong confirmed its promising performance.

Chapter 5

iLocScan: Harnessing Multipath for Simultaneous Indoor Source Localization and Space Scanning

In chapter 4 we present MaWi as a low human interference system that requires minimum deployment effort. However initial spot surveying is still an inevitable burden for all fingerprint based localization systems, and it would increase as the enlarging of deployment area. To ultimately reduce human work in initial phase, we try to find a localization solution through a model based methodology. Comparing to fingerprint based approach, a model based localization system does not require history data, thus avoid fingerprint collecting process. However, the need for infrastructural support is the main shortcoming of proposed model based approaches. In this chapter we present iLocScan to simultaneously scan an indoor space (hence generate a plan for it) and position the signal source in it using WiFi Access Points, which are universally deployed in modern buildings.

5.1 Introduction

Recently, there is a new trend of extracting physical layer information to improve both of the aforementioned approaches [10, 11, 12]. In particular, both ArrayTrack [11] and CUPID [12] apply an array of antennas to estimate the Angle-of-Arrival (AoA) of the direct path, while CUPID [12] takes one step further by using Channel State Information (CSI) to get more accurate estimation of the path length. Synthesizing the estimations (AoAs or even path lengths) from a

few sensors would allow for an accurate location estimation for the signal source. Interestingly, both ArrayTrack and CUPID treat reflection signal paths as “noises” and make great efforts to remove them, while they in fact contain valuable information. As illustrated in Figure 5.1(a), whereas the AoA of the direct path (θ_d) is what is sought by both ArrayTrack and CUPID, the

Figure 5.1: Multipath contains useful information.

AoA of the reflection path (θ_r), which was thrown away by the existing approaches, indicates the locations of a mirrored image of the signal source with respect of one wall. Given a known distance from the sensor (an antenna array)¹ to the wall, the source location can be estimated with θ_d and θ_r . In fact, multiple reflection paths do exist in a normal indoor space, as shown in Figure 5.1(b). Exploiting all these information may allow us to figure out not only the location of the source but also the geometry of the space, whereas this latter piece of information is missing in almost all indoor localization systems: a floor plan often needs to be known in advance.

Aiming at demonstrating that all these aforementioned multipath features can be actually put into use in assisting indoor localization, we construct an antenna array system, iLocScan, using multiple USRP2 Software Defined Radios (SDRs). iLocScan simultaneously samples the signals from its multiple antennas, and the samples are infused into a computation module running a fine-tuned version of MUSIC [13], in order to obtain a set of estimated AoAs (including both direct path and multiple reflection paths). iLocScan then uses a logic module

¹We use “sensor” and “antenna array” interchangeably in this thesis, as they refer to the same thing in our context.

to i) tell which AoA belongs to which path, if a sufficient number of AoAs have been gathered, and otherwise ii) suggest a possible new location to gather more AoAs. Finally, all the acquired information are put together to form a least squares problem that computes the estimations of various variables (including both source location and space geometry) as those best fit the known parameters. Although our iLocScan prototype is designed to work with 2.4GHz WiFi signals, it has the potential to track most microwave signal sources, including mobile phones, ZigBee, and Bluetooth. In summary, we are making the following major contributions in instrumenting iLocScan and in understanding the features of indoor radio signal:

- We engineer iLocScan to fully exploit the power of multipath rather than to simply avoid it; this enables us to utilize far more information embedded in the radio signals propagating indoors.
- We, for the first time, design a system that can simultaneously locate a signal source and sketch the plan of the floor where the source is located at.
- We perform detailed experimental investigations on the performance of various antenna arrays and the properties of indoor multipath propagations of radio signal; the results not only guide us in designing iLocScan but also have the potential to benefit future developments.
- We implement iLocScan using several USRP2 units, and we perform extensive experiments on it in various indoor spaces. The results strongly confirm the feasibility of exploiting multipath for assisting indoor localization, as well as the benefit of automatically constructing floor plans.

In a real application scenario, this signal source can be a friend or a colleague using a WiFi device (or even a missing device to be found), and the corresponding indoor space can be an airport terminal or a conference venue. The signal source may also be a terrorist remotely controlling a bomb but hiding itself in, say, a shopping mall. These application scenarios may be rather different from the conventional ones assumed by the majority of the research proposals (i.e., locating the user itself), but they are equally significant. Most importantly, iLocScan said require any support from an already deployed infrastructure (e.g., a set of WiFi APs), which makes it very useful in venues where no infrastructure is available. Whereas the current prototype of iLocScan is rather bulky due to the large size of individual USRP2 units,

our long term vista is to have it integrated into a hand-held device, so that a user with this device may find a signal source even in a previously unknown indoor space.

5.2 Applications and System Architecture

We first explain the applications scenarios that drive our system design, then we give a general overview of the system architecture of iLocScan. We briefly discuss the design challenges behind each individual component, but leave the details to the later sections.

5.2.1 Finding Signal Sources Indoors

In most of the metropolitan areas, people keep building tremendous business and entertaining facilities (such as shopping malls, airport terminals, convention centers), and they do spend plenty of time inside such indoor spaces for both working and entertaining. However, the ever growing complexity of these indoor structures makes it increasingly troublesome for people to find where themselves are and also **where a person/object-of-interest is**. While the majority of the indoor localization proposals aim to handle the former problem (i.e., locating users themselves), we set about tackling the latter issue (i.e., finding someone or something of interest to a user). Imaging the scenario of a user walking into an indoor facility (say an airport terminal), where he/she wishes to find another person (who could be a colleague/friend of his/hers, but could also be a criminal hiding somewhere), as show in Figure 5.2, thus our goal is to develop a device that helps the user to achieve his/her objective.

Obviously, an existing indoor localization system can be a candidate solution, as the system keeps track of the location of every user in order to respond to the location queries from individual users. Unfortunately, it is not necessarily the best solution due to the following reasons. First of all, there might not be such a system deployed in the facility, probably because it is not cost-effective to do so. Secondly, temporarily deploying such a system consisting of a collection of networked sensors¹ (e.g., [11, 12]) is way too costly and may disturb other people (including the person-of-interest who may just wish to hide). Thirdly, even if an indoor localization system is in place, the person/object-of-interest may not want to register to the system due to, for example, privacy concerns. Last but not least, majority of the proposed indoor localization systems require the floor plans to be ready.

---

¹Fingerprint-based systems (e.g., [1, 70]) are not very useful in this circumstance, as they often has a very long lead time required for constructing the fingerprint map.

Figure 5.2: Searching for a person-of-interest indoors

If this person-of-interest is simply hiding silently, we would have no way of finding him/her. However, the pervasive availability of wireless gadgets almost always makes every person a signal source: he/she might have a mobile phone, or might even have a device (the phone or a iPad) connected using WiFi. As a result, tracking the radio signals emitted from their sources may reveal the location of the source. This is in some way similar to shooter localization (e.g., [71]), but positioning radio source indoors can be fundamentally different from locating acoustic source outdoors: the former has its specific challenges given the complicated indoor structures and the totally different signal propagation features between sound and radio signal. In particular, we may not afford to deploy a networked sensing system given the reason discussed earlier.

The major obstacle that prevents us from building a simple yet fast source localization system is the limited information acquired by a single sensor: for example, ArrayTrack [11] only obtains the AoA of the direct path signal with respect to each sensor. Consequently, locating a signal source is possible only if a networked sensing system with multiple synchronized sensors is in position. Fortunately, radio signals propagating indoors contain far more information than those that have been utilized. As illustrated in Figure 5.1, information buried in multipath, which used to be filtered but if properly utilized, may potentially suggest the location of the source, as well as the floor plan on which the source is. However, designing such a “two-bird

one-stone” system is far from trivial. Whereas antenna arrays have been used recently to detect the AoA (and even length) of the direct path [11, 12], how to handle the reflection paths is still an open issue. Moreover, existing designs unanimously take a linear antenna array, but which antenna pattern suits the best for exploiting multipath is yet to be investigated. Finally, the system has to handle the situation where information at one spot is not sufficient, possibly due to the complicated indoor structures.

5.2.2 Our Solution: iLocScan

We hereby present iLocScan as a prototype for simultaneously indoor source Localization and space Scanning. In designing iLocScan, we intend to deliver some preliminary results to demonstrate the feasibility of harnessing multipath for liberating indoor localization from the reliance on any pre-deployed infrastructure. Although iLocScan may not achieve the centimeter level of localization accuracy as reported in [11], the edge of iLocScan is very clear: it requires a single sensor (instead of multiple networked ones) and it does not demand the knowledge of the floor plans. The general system architecture is presented in Figure 5.3. Basically, it has a *multi-input radio system with an antenna array* at its physical layer (the module at the top). The *AoA Computation Module* (ACMod) takes the signals gathered by all the antennas to derive a set of AoAs. Then the *AoA Logic Module* (ALMod) attempts to separate direct path from the remaining reflection paths. If it succeeds given a sufficient number of observed AoAs, the *Least Squares Module* (LSMod) will be invoked to estimate the source location, as well as the geometry of the space. Otherwise the logic unit will indicate a new spot, which may potentially yield sufficient AoA observations. In the following, we briefly discuss the challenges in designing these modules.

5.2.2.1 Antenna Pattern and AoA Estimation

As the physical component of iLocScan, the antenna array is crucial to the performance of our system. In particular, we are concerned with what antenna pattern to be used for detecting AoAs. Recent proposals only apply a linear array, but the aim of those proposals is only to identify the AoA of the direct path. As iLocScan needs to detect the AoAs of all directions along which the signal gain is significant, we need to compare various antenna patterns in terms of their ability in discriminating these AoAs. Another reason we try different antenna patterns is that linear array can only identify AoA in 180 degree. We need 2D antenna array which

Figure 5.3: iLocScan architecture.

can identify AoA in 360 degree to build a more adaptable system. Assuming a system with 7 antenna, the following Figure 5.4 shows four meaningful patterns that we shall investigate.

Several mature algorithms can be applied to synthesize the readings gathered by multiple antennas and thus to estimate AoAs. Among them MUSIC [13] is popular as it entails a rather straightforward implementation. Although directly using MUSIC has been shown to be effective in identifying the AoA of the direct path [11, 12], certain fine-tuning has to be applied to make MUSIC suitable for the ACDet of iLocScan. Essentially, as the original MUSIC algorithm appears to be designed under an (implicit) assumption that the number of antennas is much larger than the number of incoming signals, applying it directly is fine for detecting only the direct path AoA, but may not be adequate for estimating all potential AoAs.

Figure 5.4: Four antenna patterns.

5.2.2.2 Extracting Information from AoAs

After obtaining a set of AoAs, the next crucial step is to tell the AoA of the direct path from others or, in the worst case, to tell whether this AoA exists or not. Existing approaches rely on either the stability of AoAs [11] or the mobility-induced AoA variance [12] to identify the direct path, but they are not suitable for the application scenarios targeted by iLocScan. On one hand, the stability criteria works only if there is a line-of-sight path between the signal source and iLocScan, which may not be the case initially. On the other hand, requiring mobility is not practical in our applications as the signal source is not controlled by our system. Therefore, iLocScan needs the logic module, ALMod, to reason about the geometry relations among different paths.

As illustrated by Figure 5.5(a), most indoor spaces have a rather regular layout. In order to facilitate the reasoning of AoAs, iLocScan assumes a simple yet powerful model for indoor spaces: an *axis-aligned* polygon shown in Figure 5.5(b). Under such a circumstance, ALMod mainly needs to reason about two typical situations: a rectangular area and an L-shaped area, as shown by the hatched areas in Figure 5.5(b): it determines which AoA is that of the direct path in the former case, while it simply senses the latter case and then suggests a new spot so that iLocScan may potentially get better readings in terms of AoA by moving there.

(a) Floor plan of a typical airport.

(b) An axis-aligned model.

Figure 5.5: Indoor space model for iLocScan.

5.2.2.3 Simultaneous Localization and Mapping

After having a sufficient amount of information on the AoAs of various paths, LSMod puts these constraints together to make an overdetermined equation system and try to estimate the variables (source location and space geometry). Solving this overdetermined equation system by minimizing the sum of the squares of the errors is a rather standard procedure, but we need LSMod to autonomously build the optimization problem and then solve it. Whereas building an equation system automatically in general is far from trivial, we can take the advantage of the special structure of our problem and hence allow LSMod to derive the model by itself. As an iLocScan user may need to collect information at different spots, the locations of these spots (relative to the initial spot) should be the input to LSMod. We let the person who operates our iLocScan prototype to bring a mobile phone for this purpose: it measures the displacements using the dead-reckoning method reported in [38, 72].

5.3 ACMod – Measuring All AoAs At Once

It is well known that microwave signal (including, for example, 3G, WiFi, and ZigBee) can be reflected by obstacles and thus creates multipath in an indoor space. Our iLocScan differentiates itself from existing proposals in its attempt to exploit the information inferred by multipath instead of simply filtering the reflection paths. In this section, we present the technical details of ACMod (iLocScan’s module on estimating AoAs) by answering the challenges raised in Section 5.2.2.1. As some algorithm details have been discussed in [11], we focus on our fine-tuning of the algorithm, as well as experimental evaluation of the antenna patterns.

5.3.1 Preliminary on AoA Estimation

Most wireless communication systems are using QAM to modulate their signal, so our design is based on the assumption that each symbol carried by wireless signal has an I-Q representation. Typically, for a complex symbol with *amplitude* a and *frequency* f , it can be represented as

$$ae^{j(2\pi ft + \varphi)} = a \cos(2\pi ft + \varphi) + ja \sin(2\pi ft + \varphi),$$

where φ is the phase of the modulating symbol. On the I-Q plane, the symbol can be considered as a point rotating counter-clockwise around the origin, and $\Phi = 2\pi ft + \varphi$ denotes the *instantaneous phase*.

Denoting the distance from the source to the first antenna by d , the phase of the signal arriving at the receiver will be

$$\varphi_1 = 2\pi d\lambda^{-1} + \varphi$$

where $\lambda = c/f$ is the wave length of the signal, with c being the speed of light. Obviously, varying the distance d will change the phase of the received signal. Now let us take a second antenna whose distance towards the first antenna is $\tilde{d}$. Assuming the AoA of the signal with respect to the two-antenna array is $\theta \in (-\pi/2, \pi/2]$ and, without loss of generality, $d \gg \tilde{d}$, we may derive the distance from the transmitter to the second antenna as $d + \tilde{d} \sin \theta$. Consequently, there is a constant offset $\Delta\Phi = 2\pi \sin \theta \tilde{d} \lambda^{-1}$ between the instantaneous phases of the two antennas. To eliminate the ambiguity in computing this phase offset, the phase offset should be less than π . Typically, one may space the two antennas by half of the wavelength, i.e., $\tilde{d} = \lambda/2$, and thus $\Delta\Phi = \pi \sin \theta$. As a result, with a measurement of $\Delta\Phi$, we can derive the AoA as $\theta = \arcsin(\Delta\Phi \pi^{-1})$. Apparently, the $\Delta\Phi$ is independent of the signal amplitude a and symbol phase φ , so any symbol transmitted by WiFi can be used for the purpose of

Figure 5.6: Detecting the AoA of a signal path using a two-antenna array. The antennas are aligned with the x -axis, while the y -axis indicates the forward direction, i.e., the detected AoA represents the signed angle counter-clockwise from y -axis to the signal's steering direction.

estimating AoA. Without loss of generality, we hereafter assume $a = 1$ and $\varphi = 0$ to simplify the exposition.

5.3.2 Fine-Tuning the MUSIC Algorithm

In reality, what each antenna receives is actually the superposition of several signals (from both direct and reflection paths) with different AoAs. To handle this, several algorithms have been proposed and among them MUSIC [13] is the most popular one. However, our experiments show that the original MUSIC algorithm does not perform very well when the number of antennas is not far beyond the number of signal paths. Therefore, we shall first introduce the basics of MUSIC, and then present our fine-tuning to the algorithm.

5.3.2.1 MUSIC Primer

Assume there are N antennas in the array, and signals from M directions are received by the antenna array. Obviously, N should be more than M for eliminating multi-path ambiguity. Since the signals are time varying, the input of the MUSIC algorithm is a set of signal samples taken at the N antennas at the same time. Denote by $\mathbf{s} = [s_1, s_2, \dots, s_M]^T$ the incident signal from M directions, $\mathbf{r} = [r_1, r_2, \dots, r_N]^T$ the signal vectors received by N antennas, and $\mathbf{w} = [\omega_1, \omega_2, \dots, \omega_N]^T$ the noise vector at the antenna array. The incoming signal of the i -th antenna

can be defined as the combination of the M signals from different directions plus the noise

$$r_i = \sum_{k=1}^M g_i(\theta_k) s_k + \omega_i, \quad (5.1)$$

where $g_i(\theta_k)$ is the *gain* of the k -th signal received by the i -th antenna. Overall, $\mathbf{r}$ can be characterized by the following linear model

$$\mathbf{r} = \mathbf{G}\mathbf{s} + \mathbf{w} \quad (5.2)$$

where $\mathbf{G} = [\mathbf{g}(\theta_1), \mathbf{g}(\theta_2), \dots, \mathbf{g}(\theta_M)]$ and $\mathbf{g}(\theta_k) = [g_1(\theta_k), g_2(\theta_k), \dots, g_N(\theta_k)]^T$ is the *steering vector* for the k -th signal.

In order to estimate $\{\theta_k\}_{k=1, \dots, M}$ from the observed signal vector $\mathbf{r}$, MUSIC exploits the fact that the correlation among signals received at different antennas contains the information about the directions indicated by the steering vectors. Denote by $\mathbf{C}_r$ the $N \times N$ *correlation matrix* of the received signal, we have, according to [13],

$$\mathbf{C}_r = \mathbf{G}\mathbf{C}_s\mathbf{G}^* + \sigma_w^2 \mathbf{I}$$

where $*$ represents a conjugate transpose, $\mathbf{C}_s$ denotes the correlation matrix of the incident signals, and σ_w^2 is the variance of the (zero mean and i.i.d.) noise. Apparently, $\mathbf{G}\mathbf{C}_s\mathbf{G}^*$ is singular and has a rank of M if the number of the incident signal is less than the number of the antennas (i.e., $M < N$). Therefore, if $\{\tau_1 \geq \tau_2 \geq \dots \geq \tau_N\}$ and $\{\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_N\}$ are eigenvalues and corresponding eigenvectors of $\mathbf{C}_r$, we have $\tau_{M+1} = \tau_{M+2} = \dots = \tau_N = \sigma_w^2$, and

$$\{\mathbf{u}_{M+1}, \mathbf{u}_{M+2}, \dots, \mathbf{u}_N\} \perp \{\mathbf{g}(\theta_1), \mathbf{g}(\theta_2), \dots, \mathbf{g}(\theta_M)\}. \quad (5.3)$$

In other words, the noise space $\mathbf{U}_n = [\mathbf{u}_{M+1}, \mathbf{u}_{M+2}, \dots, \mathbf{u}_N]$ is orthogonal to the column space of $\mathbf{G}$ in an ideal case. However, this orthogonality may not hold strictly. Therefore, the original MUSIC algorithm proposes to scan the angle spectrum by computing

$$P_{\text{MU}}(\theta) = \frac{1}{\mathbf{g}(\theta)^* \mathbf{U}_n \mathbf{U}_n^* \mathbf{g}(\theta)} \quad (5.4)$$

The rationale is the following: if $\theta \in \{\theta_k\}_{k=1, \dots, M}$, P_{MU} would be rather large due to the orthogonality stated above.

5.3.2.2 MUSIC for iLocScan

As our later comparisons will show, running MUSIC to detect all AoAs leads to rather unstable results, i.e., the angle spectrum P_{MU} may differ significantly in time, thus affecting the accuracy of AoA estimations. In fact, according to the orthogonality condition in (5.3), we also have $\mathbf{g}(\theta_1), \mathbf{g}(\theta_2), \dots, \mathbf{g}(\theta_M)$ exactly lying in the signal space $\mathbf{U}_s$ spanned by $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_M$. In particular, $\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_M$ are indicating the most correlated directions. Therefore, we redefine the angle spectrum as

$$P_{\text{iLocScan}}(\theta) = \mathbf{g}(\theta)^* \mathbf{U}_s \mathbf{U}_s^* \mathbf{g}(\theta) \quad (5.5)$$

Ideally, P_{MU} and $P_{\text{iLocScan}}(\theta)$ should be exactly the same. In reality, if $N - M < M$, $\mathbf{U}_s$ contains more information; or the other way around if $N - M > M$. In practice, an antenna array may not contain more than 8 antennas due to size limit while there could be up to 5 incident signals in an indoor environment, $P_{\text{iLocScan}}(\theta)$ (instead of $P_{\text{MU}}(\theta)$) should be used to improve the accuracy of AoA estimations. The two figures in Fig. 5.7 compares the two angle spectrums under the same location arrangement: the spectrum generated by $P_{\text{iLocScan}}(\theta)$ is apparently more stable and hence yields a clearer indication of the AoAs.

Figure 5.7: Comparing two angle spectra.

5.3.3 Antenna Patterns

A crucial component of iLocScan is its antenna array, as it determines the quality of information the system may acquire. Although several patterns are possible (shown in Figure 5.4), linear antenna array is often used in the literature for detecting the AoA of direct

path. As our objective is to identify all possible AoAs, it is necessary to investigate the performance of these patterns experimentally. We apply our fine-tuned MUSIC algorithm to each of these patterns by adapting its steering vector. For example, the steer vector for linear array is $\left[e^{-j2\pi \sin \theta \bar{d} \lambda^{-1}} \right]_{i=0,1,\dots,N-1}^T$ (according to Section 5.3.1), and that for circular array is $\left[e^{-j2\pi \cos(2i\pi N^{-1} - \theta) R \lambda^{-1}} \right]_{i=0,1,\dots,N-1}^T$ with R being the circle radius, derived based on Figure 5.8.

Figure 5.8: Circular antenna array. The blue points denoting the antennas are uniformly placed on a circle with radius R . The signal AoA is θ .

One of the important metrics for an antenna array is its resolution: the minimum discernable angle difference between two neighboring AoAs. According to the statistics shown in Table 5.1, linear has the best resolution, but T-shaped is very close to it. This is obviously due

Table 5.1: Angle Resolution of Different Antenna Arrays.

	Linear	Circular	Cross	T-Shaped
Min Angle Diff (degree)	18	24	34	21

to the fact that linear uses one more antenna to measure phase differences, whereas T-shaped

has to partially sacrifice it for telling the sign of an AoA¹. We also illustrate in Figure 5.9 the same three AoAs recognized by different antenna arrays.

Figure 5.9: Angle spectra of different arrays.

Another related metric is the accuracy of measuring angles. We plot the CDF of angle errors result from different arrays in Figure 5.10. It is clear that the T-shaped array performs the best; this seems to be related to the rather regular spectrum produced by it compared with others, as shown in Figure 5.9. As the linear array is a 1D pattern, it has a border effect: AoAs close to zero degree may not be measured accurately. Other 2D patterns avoid such negative effect, but offers lower resolution. Therefore, our results advocate T-shaped array: it retains the good properties of linear array while avoiding its drawback. When we use the linear antenna array in later experiments, we take two measurements at one spot, with antenna directions perpendicular to each other. This allows the linear array to obtain much better accuracy than others (as it yields more readings at one spot), but at the cost of a higher system latency.

¹According to the discussion in Section 5.3.1, a 1D array cannot distinguish AoAs in $[-\pi/2, \pi/2]$ from those in $(-\pi, -\pi/2) \cup (\pi/2, \pi]$.

Figure 5.10: Accuracy of angle measurement by different arrays.

5.4 ALMod – AoA Reasoning

Given AoAs measured by ACModule, we need ALMod to extract information from these measurements, and also to determine whether further information is needed. According to the axis-aligned model discussed in Section 5.2.2.2, this reasoning proceed boils down two specific cases: rectangular area and L-shaped area. In the following, we first discuss the basic rules that ALMod applies to reason AoAs in a rectangular area, then we introduce the algorithm to perform localization and scanning in a complicated indoor space.

5.4.1 The Case of A Rectangular Area

As discussed in Section 5.2.2.2, indoor layouts are often axis-aligned from a geometric point of view. In particular, the most elementary unit is a rectangle area whose sides are aligned with the axes of the coordination system. Therefore, we first study this basic case where both iLocScan and the signal source are placed inside such a rectangular area with all four sides being able to reflect the signals. We also assume the axes of iLocScan are aligned with those of the area.

Our experiments in such a cell show that, except for direct path signal, only single-reflection signals are strong enough to be detected by ACMod, whereas those multiple-reflection ones usually have far less strength and thus may not be sensed. Under such a circumstance, ACMod should be able to detect up to five AoAs, and observing the distribution of the detected AoAs in the four quadrants would allow ALMod to figure out which one is the direct path. The following are the reasoning rules:

- **5-AoA:** We refer back to Fig. 5.1(b) for an illustration of such a case. Basically, there is always one quadrant containing three AoAs, so the one in the middle corresponds to the direct path: *middle rule* hereafter.
- **4-AoA:** These are degenerated cases where exactly one of the following three conditions holds:

C1: iLocScan and the source are co-linear along one axis: Figure 5.11(a).

C2: The source is close to one side: Figure 5.11(b).

C3: iLocScan is close to one side: Figure 5.11(c).

The consequence is that three AoAs are separated from the fourth by one axis, so the middle rule applies.

- **3-AoA:** These are degenerated cases where two or more of the C1–C3 hold, as shown by Figure 5.11(d)–(h). Note that this includes the case where the same condition holds twice, i.e., C2 is satisfied twice infers that the source is at a corner: Figure 5.11(d). For the cases shown by Figure 5.11(d)–(e), the AoA in the middle still corresponds to the direct path. However, we cannot draw a right conclusion due to the ambiguity introduced by Figure 5.11(f)–(g). Fortunately, we may break C3 by moving iLocScan away from the side, which would allow a right conclusion to be drawn. As for Figure 5.11(h), a *side rule* applies: the AoA aligned with one axis and on the majority side corresponds to the direct path.
- **2-AoA:** This is a very special case where both C2 and C3 hold and one of them holds twice, as shown by Figure 5.11(i). For this case, ALMod would again suggest breaking C3. The case of 1-AoA shown in Figure 5.11(j) is similar to the 2-AoA case.

In summary, ALMod should be able to identify the direct path using the middle rule under both 5-AoA and 4-AoA cases. If less AoAs are detected, ALMod check if C3 holds (which requires user confirmation). If false, the middle rule still applies; otherwise ALMod alerts the user to break C3.

Figure 5.11: AoA patterns. We denote iLocScan and the signal source by blue dot and red star, respectively. We also mark the coordinate system of the antenna array by red arrows, and the detected AoAs by green arrows.

5.4.2 A Rectangular Area with Open Side(s)

This case is similar to the closed area case, if we add a “virtual wall” on the open side and deem the object (iLocScan or the source) closer to the open side as being “on the wall”. This implies that either C2 or C3 always holds. Under this circumstance, 5-AoA does not exist, but all 4-AoA cases still allow the middle rule to be applied. However, a user cannot break C3 anymore, leaving the ambiguity between Figure 5.11(f) and (g) unsolvable. Nevertheless, the later LSMod should be able to find contradiction among one of the possibilities. For example, if the middle rule is applied to Figure 5.11(g), the area computed by LSMod will not be axis-aligned. As for the 2-AoA and 1-AoA cases, the strategy is to meet C3 by moving iLocScan, hence converting all such cases to Figure 5.11(h).

5.4.3 The L-Shaped Area and Beyond

Given a general axis-aligned polygon (see Figure 5.5(b)), there is yet another element differing from the rectangular area: the L-shaped area. Obviously, if ALMod handles both situations, it

works for any general axis-aligned polygon spaces, as they can always be reduced to a combination of multiple L-shaped areas. Therefore, we hereby discuss how ALMod copes with an arbitrary L-shaped area. When iLocScan and the source are not co-located in the same branch of the L-shaped area, none of the above cases apply. The strategy taken by ALMod is to move iLocScan such that we may get back to the rectangular area case. As shown in Fig. 5.12, when

Figure 5.12: L-shaped indoor structure.

all AoAs point towards the positive direction of the x-axis which also implies the location of the signal source in some extent. We then move our antenna array along the positive x-axis towards the hatched area until our antenna array reaches the same rectangular area as the signal source; this brings the situation back to what has been discussed in Section 5.4.2. Essentially, ALMod applies a *coordinate-wise searching* method, and always moves iLocScan along one axis directed by the AoA measurements.

5.5 LSMod – Autonomous Scenario Modeling and Problem Solving

With the information extracted by ALMod (probably after a few movements of iLocScan), it is now ready for LSMod to simultaneously perform source localization and sketching the indoor structure. The general principle behind LSMod is to form and then solve a least squares problem so that we fit the variables to be estimated to the measured AoAs. While forming and solving such a problem is rather standard for a human user, we would like the computing system to automatically complete the whole procedure.

Without loss of generality, LSMod takes the initial spot of iLocScan to be the origin of its global coordinate system. Remember iLocScan also has a local coordinate system, used by its antenna array, for measuring AoAs. The axes of both coordinate systems are aligned, but the local one always has the origin at the center of the antenna array. Also note that the user of iLocScan has to (visually) guarantee that the axes of iLocScan are aligned with those of the targeted indoor space. In Fig. 5.13, we use X - Y to denote the global coordinate system, and x - y to denote the local coordinate system. In the figure, we take a closed rectangular area as an example, whose four sides, under X - Y , can be expressed by $\Omega = \{\Omega_1 : x = w_1, \Omega_2 : y = l_1, \Omega_3 : x = -w_2, \Omega_4 : y = -l_2\}$ where $w_1, w_2, l_1, l_2 > 0$. Also, the unknown location of the signal source is $p = (x_p, y_p)$. The functionality of the LSMod is to determine these variables.

Figure 5.13: Illustrating automatic problem formulation.

As iLocScan may need to visit more than one spot in order to acquire sufficient AoA information, we denote by $\hat{\theta}_i$ the direct path AoA and by $\mathcal{A}_i = \{\theta_{i,1}, \dots, \theta_{i,k}\}_{1 \leq k \leq 4}$ the set of the reflection path AoAs, in the i -th measurements at spot $q_i = (\tilde{x}_i, \tilde{y}_i)$. For any $\theta_{i,k} \in \mathcal{A}_i$, its reflection wall is denoted by $\Omega(\theta_{i,k}) \in \Omega$. As iLocScan is using dead-reckoning to estimate the location of later spots relative to the initial one, $q_i = (\tilde{x}_i, \tilde{y}_i)$ is the input to LSMod. Now LSMod can express the AoA-side relationship as follows:

$$\cos \theta_{i,k} = f_{i,k}(p, w_1, w_2, l_1, l_2 | q_i) = \frac{(\tilde{p}_{i,k} - q_i) \cdot e_y}{\|\tilde{p}_{i,k} - q_i\|_2} \quad (5.6)$$

where e_y is the unit vector along the positive direction of Y -axis, $\|\cdot\|_2$ denotes the Euclidean norm, and $\tilde{p}_{i,k}$ indicates the mirror source of the targeted signal source with respect to $\Omega(\theta_{i,k})$; it can be represented in terms of p and coordinate of $\Omega(\theta_{i,k})$. Moreover, LSMod introduces the following equation for the direct path AoA $\hat{\theta}_i$:

$$\cos \hat{\theta}_i = f_i(p|q_i) = \frac{(p - q_i) \cdot e_y}{\|p - q_i\|_2} \quad (5.7)$$

Based on sufficient AoA observations, LSMod can now formulate a least squares problem:

$$\begin{aligned} & \underset{x_p, y_p, w_1, w_2, l_1, l_2}{\text{minimize}} && \sum_i \sum_k (f_{i,k} - \cos \theta_{i,k})^2 + \sum_i (f_i - \cos \hat{\theta}_i)^2 \\ & \text{subject to} && -w_2 \leq x \leq w_1 \\ & && -l_2 \leq y \leq l_1 \\ & && z, w_1, w_2, l_1, l_2 \geq 0 \end{aligned} \quad (5.8)$$

where $\{x_p, y_p, w_1, w_2, l_1, l_2\}$ are the variables. We apply *Trust-Region-Reflective* algorithm [73] to solve this optimization problem with bound constraints. As the dimension of the problem depends on the number of walls in an indoor space and the number of signal sources to be located, the problem cannot be of very large scale, as we normally aim at finding a few sources in a space with tens of walls. Therefore, solving this optimization problem does not lead to a significant overhead in computing.

5.6 Implementation Details

In this section, we present the technical details on the construction of our iLocScan prototype. In a nutshell, our hardware platform is based on USRP N210 (hereafter USRP2), while the software part, including ACMod, ALMod and LSMod, is implemented using Python and C++ under GNU Radio on a host computer.

5.6.1 A Multi-Input Radio System

The physical layer of our iLocScan prototype is a multi-input radio system shown in Figure 5.14(a), which mainly consists of seven USRP2 units. This corresponds to the module at the top of the iLocScan architecture shown in Figure 5.3. Each USRP2 unit is equipped with a RF frond end including an SBX daughter board and an omnidirectional antenna for receiving the signals transmitted by the targeted source. These RX USRP2 units are controlled by a host

computer through a Gigabit Ethernet switch, through which the signal samples taken by the RX USRP2 units are also fed back to the host computer, serving as the input to the software modules running on the host computer. Additionally, the RX USRP2 units share a common reference of 10 MHz and 1 PPS generated by an external clock for the purpose of synchronizing their native clocks, such that they can sample the incoming signals exactly at the same moment.

The physical construction of iLocScan is shown in Figure 5.14(b). We use a double-deck trolley to hold the system such that it may move freely. The USRP2-based antenna array is put at the upper deck, along with the external clock and the reference signal source (see Section 5.6.2 for details). The lower deck holds the Ethernet switch. The major disadvantage of this construction is that it has to be powered; this has limited our choice of testing sites to a few research labs (rather than going to the real-life indoor spaces such as a shopping mall).

5.6.2 Phase Calibration

Only synchronizing the native clocks of the RX USRP2 units is not sufficient for our application, due to the random phase shifts caused by the radio's *Phase Locked Loop* (PLL) during the *Digital Down Conversion* (DDC) at each RF front end. These unknown phase shifts are added to the signal phases and thus may cause large estimation errors in ACMod's AoA detection procedure. To calibrate the RF front ends, we employ a reference USRP2 unit to transmit calibration signal (e.g., a 2.4 GHz sinusoidal carrier) to the RX USRP2 units through a SMA splitter. Because all the RX USRP2s are connected to the SMA splitter via cables of equal length, the incoming calibration signal at each USRP2 device has the same phase. Let us denote by φ_{ref} and $\tilde{\varphi}_i$ the phase of the incoming calibration signal and the phase of the signal sample at the i -th RX USRP2 unit, respectively. Then the phase shift caused by DDC at the i -th RX USRP2 unit is $\tilde{\varphi}_i - \varphi_{ref}$. As the AoA measuring procedure run by ACMod is concerned with only the relative phase offsets between the RX USRP2 units, we simply need to align the phases of the RX USRP2 units to one of them (e.g., the first RX USRP2 unit). In particular, we can calibrate the i -th RX USRP2 units by subtracting $\tilde{\varphi}_i - \tilde{\varphi}_1$ from its signal sample, where $i = 1 \dots 7$.

(a) Schematic for the antenna array.

(b) The outlook of iLocScan

Figure 5.14: System schematic and outlook of iLocScan.

5.6.3 Detecting WiFi Preamble

To acquire the bearing information of the targeted signal source, our ACMod needs to overhear the wireless communication originated from the source. As data packets are rather arbitrary and hence hard to control, we turn to the frame preamble. Each IEEE 802.11 frame starts with

a short preamble sequence consisting of ten identical short training symbols with duration $0.8 \mu\text{s}$ each. The short preamble is often fairly robust and stable, so it serves as a good source of input to ACMod. Besides, as the USRP2 unit has a maximum sampling frequency of up to 100 MS/s, this implies that it spends only 100 ns to take a sample from the incoming signal stream, sampling the short preamble sequence should be sufficient for the MUSIC algorithm as well as our fine-tuning version. We implement the preamble detection algorithm [74] in ACMod to extract the short preamble signals from the 802.11 frames. In particular, we set a buffer at each USRP2 unit's frond end. Once the short preamble sequences are detected in all of the buffers, the samples will be delivered to ACMod. In our implementation, we take 30 samples from each preamble for ACMod to perform AoA detection, which is shown to be adequate to suppress noise and to ensure the estimation accuracy.

5.7 Experimental Evaluations

We have conducted extensive experiments with our iLocScan prototype at multiple test sites to verify its efficacy and robustness. In this section, we first briefly discuss the experiment settings, then we report the results on evaluating iLocScan. As a byproduct, we also obtain a large amount of data on the reflection properties of various indoor structures, which deliver insights that can be useful for the future developments of indoor radio sensing systems.

5.7.1 Experiment Settings

We perform many tests in three research labs; the floor plan of one of them is shown in Figure 5.15 (top row). Taking the advantage of having the digitized floor plans of these test sites, we can accurately design the ground truth locations of the signal sources to be located, and we also have accurate measurements of the geometry of these sites. We use three WiFi APs to emulate the signal sources, and we fix their locations in each of the test sites. In order to distinguish among these APs, we implement a full WiFi receiver functionality on our iLocScan prototype such that the APs are identified according to their SSIDs. As our iLocScan is movable, we often perform the initial measurements close to the entrance, and then choose new spots (if necessary) following the method discussed in Section 5.4.3. At each spot, iLocScan may at most detect 5 AoAs for a given AP; this may not be sufficient to achieve accurate estimations. Therefore, we often take two observations at each spot, by moving iLocScan slightly off the spot for one meter.

5.7.2 A Concrete Example

Before diving into the statistical evaluations on the measurement accuracy of our iLocScan, we first use a concrete example to introduce how iLocScan prototype works in a real-life scenario and how the measurements have been obtained. As shown in Fig. 5.15, three targeted signal sources T1, T2 and T3 are placed arbitrarily in an S-shaped axis-aligned room, and they all operate on WiFi Channel 6. Recall that these WiFi APs are using CSMA mechanism to avoid interfering each other, so they do not transmit simultaneously. Consequently, the angle spectrum measured by iLocScan at given point in time corresponds only to one AP; this enables iLocScan to obtain three separated angle spectra shown in Fig. 5.15 (the lower three rows). The figures at the top row are illustrative, so measurement errors demonstrated in them do not represent exact values.

Figure 5.15: An illustration of iLocScan evaluation. This set of experiments is conducted in an 800 m² research lab. We fix three signal sources at location T1, T2, T3. Our iLocScan chooses three spots to perform AoA measures. Each column of the figures corresponds to one iLocScan spot. The top row marks the measurement spots and also shows the estimated source locations and the floor plans. Another three rows plot the AoA measures for individual signal sources.

As shown in the first column of Fig. 5.15, our iLocScan starts to measure AoAs right after

entering the space. It detects five AoAs from T1, but only one AoA from T2 and T3¹. The AoA information collected at this first spot is sufficient to locate T1 and to estimate the geometry of the area marked by the blue box (which is only a partial view of the whole floor plan with some virtual wall being introduced). We then move iLocScan forward along the direction suggested by the detect AoAs of T2 and T3, as shown by the second column of Fig. 5.15. In this second spot, iLocScan can detect five AoAs for both T1 and T2, but still observes only one AoA for T3. This allows iLocScan to estimate the location of T2, as well as the two areas marked in blue and green. Combining the estimated geometry from the first two spots, the left side of the floor plan has now been fully constructed. The further collected AoA information on T1 can be used to refine the localization results we have obtained. Thanks to the direction implied by the AoA of T3, we move the iLocScan to the third spot shown in the third column of Fig. 5.15. At this spot iLocScan finally detects five AoAs from T3; it is hence able to locate T3 and to construct the full floor plan.

5.7.3 Accuracy Evaluations

We first report the measurement accuracy by comparing our estimations with the ground truth. For localization accuracy, the metric is the commonly used square-root error. For the floor plan geometry, the error is the distance shift of an estimated wall. As our floor plan model and the ground truth are both axis-aligned polygons, the estimation errors are only in the form of distance shifts. As shown in Figure 5.16, the localization error is less than 4 meters and the geometry error is less than 5 meters for all antenna patterns. Such an estimation accuracy is satisfactory in practice. Linear array (maximum error 3 meters and median error 1.9 meters in localization) performs far better than others simply because we take two perpendicular observations right at each spot (see Section 5.3.3 for details). This shows that, with linear array, we can trade detection latency for higher accuracy. Within the remaining three patterns, T-shaped array appears to have slightly better performance than the others, for the reason that have been studied in Section 5.3.3.

As we have shown in Section 5.7.2, it is possible that iLocScan cannot locate the targeted signal sources and scan the floor plan fully with only a couple of spots. Using the large amount

¹As discussed in Sec. 5.7.1, we need two observations at a give spot to achieve better estimations. Due to the space limitation, we only show the angle spectrum results of the first observation. Note that we plot 38 snapshots to demonstrate the stability of observed angle spectra, although it takes only one snapshot for iLocScan to detect these AoAs.

Figure 5.16: Evaluation of measurement accuracy.

of data we have collected by randomly putting the WiFi APs in the three test sites, we show the chance of locating signal sources as an increasing function of the number of spots visited by iLocScan in Figure 5.17. Clearly, whereas one spot only allows less than 40% of the WiFi APs to be localized, almost all APs become localizable with up to 4 spots: the small fraction of non-localizable APs are at some corners, but the lab facilities prevent us from locating iLocScan properly (see Section 5.4.1 for details).

Figure 5.17: More spots yield higher chance of locating a source.

Normally, we take two observations per spot; this is how we obtain all the aforementioned results. One may wonder if adding more observations (at the same spot but slightly shifted from each other) would lead to higher accuracy. We answer the question by showing more accuracy results in Figure 5.18. Apparently, the answer is yes, but at the cost of spending more time on the same spot.

Figure 5.18: More observations lead to higher accuracy.

5.7.4 AoA Detection under Varying TX Powers

We aim to design iLocScan to be compatible with a variety of wireless devices, which may have considerable heterogeneity, for example, in terms of tx power. Therefore, we now evaluate the robustness of iLocScan with respect to AoA detection in face of varying tx power. We set up one extra USRP2 unit to emulate a WiFi AP in our research lab and tune its output power from -80 to -40 dBm¹. We perform twenty AoA measures under each tx power setting at a fixed spot 5 meters away from the WiFi AP; the results are reported in Fig. 5.19. With extremely low tx powers (-80 to -70 dBm), the angle spectra are quite unstable so that it is rather difficult to estimate AoAs out of them. Further increasing the tx power to -60 dBm significantly stabilizes the angle spectra, but only with -40 dBm tx power iLocScan may detect all the three available AoAs. This set of tests suggest that increasing tx power affects the AoA detection in two ways: stabilizing the angle spectrum and improving the AoA resolution. As -40 dBm is still very low compared with normal WiFi tx power range, the ability of iLocScan to detect the reflection paths under this very low tx power has firmly demonstrated its applicability to real-life scenarios in locating off-the-shelf WiFi devices.

5.7.5 Reflections on Different Materials

Today's indoor space may be constructed or separated by various materials. Though it is well known that, given a signal with certain frequency, different materials exhibit diverse reflection ability. Although this is currently not the main issue concerning the design and evaluation of

¹A normal WiFi AP has a tunable tx power range around 10dBm, which is not low enough to test iLocScan in extreme cases.

Figure 5.19: Angle spectra under different tx powers of the signal source.

our iLocScan, the large amount of data we have gathered while testing our system do allow us to shed some light on it. During our extensive tests, we have come across quite a few different building materials. For example, internal walls are often made of concrete, but wall facing outside may well be made of glass. Moreover, metal boards can be used to separate a big hall into small rooms. Typical results are showed in Fig. 5.20. The reflection abilities of the three materials can be derived by comparing the magnitudes of the direct path signal with those of the reflection paths. We have observed that, among the three materials, metal has the strongest reflection ability as the reflection path signal may reach the same strength as the direct path signal, whereas the remaining two are comparable in terms of reflection ability. However, as glass is smoother than concrete on surface, the reflections tend to be slightly more stable.

In general, the reflection abilities of most indoor materials are sufficient for our iLocScan system to detect reflection path AoAs. However, knowing these properties may allow iLocScan to be better aware of the surrounding environments: it may not only estimate the geometry of an indoor space, but also figure out how it was constructed. We are planning to modify iLocScan into a pure scanning device such that, without a few already deployed WiFi APs in a building,

Figure 5.20: Reflections on concrete wall, glass window, and metal board.

we may build the floor plans automatically.

5.8 Summary

While the majority of indoor localization systems aim at locating the users themselves based on known floor plans, we have targeted to a slightly different problem in this chapter: locating a signal source in an unknown indoor space. To this end, we have innovatively exploited the power of multipath (which is often “antagonized” by wireless system researchers) and hence proposed a system called iLocScan; it is able to locate a signal source in an indoor space while constructing the floor map of the targeted space at the same time. Leveraging the ability of antenna arrays in detecting the Angle-of-Arrival (AoA) of a signal path, we have instrumented iLocScan to the point that it can simultaneously measure all AoAs induced by an indoor wireless transmission (due to its direct path and multiple reflection paths). This has involved fine-tuning a well known AoA detection algorithm and investigating the features of

various array patterns. We have also designed a logic module for iLocScan to judge which AoA corresponds to the direct path and whether the number of observed AoAs is a sufficient, as well as an autonomous problem formulation and solving module to fit the variables (source location and space geometry) to the AoAs. To demonstrate the viability of these ideas, we have implemented an iLocScan prototype using USRP2 units. Our extensive experiments with this prototype have strongly confirmed the efficacy of iLocScan and also delivered useful insights on indoor signal reflection and propagation.

Chapter 6

Conclusion

Since last decade, Indoor localization has become a popular research topic. The heat of indoor localization comes from real need of people’s daily life. Numerous approaches are proposed to attempt to solve the problem, while we believe a practical system that can both achieve high localization accuracy and consume minor deployment effort is still a vacancy. To this end, we propose three approaches of different kinds - GROPING (Chapter 3), MaWi (Chapter 4), iLocScan (Chapter 5) - as candidates to fill this vacancy in this thesis. We summarize our contribution in this three works in Section 6.1, and show some of our future research directions in Section 6.2.

6.1 Research Contribution

In chapter 3, we propose GROPING as a completely self-contained, lightweight, and practical prototype for indoor navigation. GROPING utilizes user contributed sensor data and semantic labels to build floor plan, and then performs localization based on the magnetic fingerprints collected, and finally it computes navigational routes using the early constructed map and the real-time location information. GROPING requires neither wireless infrastructure nor digitized floor maps. Our intensive experiments with GROPING demonstrate its usability and also show that it compares favorably with typical WiFi-based localization systems in supporting indoor navigation.

In chapter 4 we propose MaWi as a smart phone based indoor localization system using Magnetic field and Wi-Fi as fingerprints. Magnetic field and Wi-Fi fingerprints are used in a “duet” manner such that they complement each other. Through this smart combination, MaWi

achieves a scalable deployment due to its low demand on the fingerprint database, while getting very competitive localization accuracy compared to state-of-the-art systems. Furthermore, MaWi requires no dedicated devices or adaption of existing infrastructure. We deploy MaWi in an office building, a library, and a shopping mall with complicated floor plans. Our extensive experiments demonstrate a low deployment workload and high localization accuracy.

In chapter 5 we construct an antenna array system, iLocScan, using multiple USRP2 Software Defined Radios (SDRs). iLocScan is designed to locate a signal source and sketch the plan of the floor where the source is located at the same time. We engineer iLocScan to fully exploit the power of multipath rather than to simply avoid it; this enables us to utilize far more information embedded in the radio signals propagating indoors. Detailed experimental investigations are performed on the performance of various antenna arrays and the properties of indoor multipath propagations of radio signal; the results not only guide us in designing iLocScan but also have the potential to benefit future developments. We implement iLocScan using several USRP2 units, and we perform extensive experiments on it in various indoor spaces. The results strongly confirm the feasibility of exploiting multipath for assisting indoor localization, as well as the benefit of automatically constructing floor plans.

6.2 Future Directions

Although numerous indoor localization approaches have been proposed in last decade, we can still see plenty of vacancy need to be filled. With technology development, mobile device is evolving into new forms of wearable devices, such as glasses and wristbands. More and more nature or artificial information can be captured by new devices. For example, by capture vision information, a localization system can help people to locate and navigate themselves in a most natural way, without using any infrastructure or deployment effort. Such a system may not perform as well as a pre-defined system that use pre-surveyed fingerprint or pre-deployed anchor devices, but it would refine by the information it collect through human inference.

Comparing to normal people, disabled people could use more help from mobile indoor navigation system. Although some attempt approaches have been proposed in navigating disabled people recently, we still cannot see a mature and practical solution appears. Considering the disability of user, a navigation system should inform user about location in alternative ways, for example, system cannot show floor plan to eye disabled user, but can describe to the user

by language. Furthermore, in navigating disabled people, system should also select path which is suitable for users, e.g., blind road for eye disabled people, lift for leg disabled people, etc..

Our another future direction is Context Recognition in indoor space. From perspective of application, knowing location is barely enough. Some applications require to know the environment in which user is in as well. For example, user is less possible to enjoy a coupon when she is chatting with friend than when she is wondering around, whereas the two activities may happen at the same location. By combining localization with context recognition, we would be able to infer user's possible interest according to her location as well as her current activity. Numerous applications would be proliferated by such a localization/context recognition system. Most context recognition systems proposed recently utilize smart phone embedded camera/microphone to capture environment information, and then infer the context basing on it. The problem is that continuously recording environment information would not only drain the power of mobile devices, but also intrude user privacy. Although opportunistic, recording mechanism can reduce the risk of intruding privacy and lessen power consumption, it would result in lower context recognition accuracy inevitably. However, with the raise of mobile device revolution, we can see broad field lies in this research topic.

References

- [1] MOUSTAFA YOUSSEF AND ASHOK AGRAWALA. **The Horus WLAN Location Determination System**. In *Proc. of ACM MobiSys*, pages 205–218, 2005. 1, 5, 6, 10, 11, 33, 38, 48, 52, 64, 65, 66, 69, 76
- [2] SUNGWON YANG, P. DESSAI, M. VERMA, AND M. GERLA. **FreeLoc: Calibration-free Crowdsourced Indoor Localization**. In *IEEE INFOCOM*, pages 2481–2489, 2013. 1, 5, 6, 33, 37, 48, 52, 65
- [3] SUNGRO YOON, KYUNGHAN LEE, AND INJONG RHEE. **FM-based Indoor Localization via Automatic Fingerprint DB Construction and Matching**. In *ACM MobiSys*, pages 207–220, 2013. 2, 7, 48
- [4] KRISHNA CHINTALAPUDI, ANAND PADMANABHA IYER, AND VENKATA N. PADMANABHAN. **Indoor Localization Without the Pain**. In *ACM MobiCom*, pages 173–184, 2010. 2, 7, 48, 49
- [5] SOUVIK SEN, JEONGKEUN LEE, KYU-HAN KIM, AND PAUL CONGDON. **Avoiding Multipath to Revive Inbuilding WiFi Localization**. In *ACM MobiSys*, pages 249–262, 2013. 2, 7, 48
- [6] KAIKAI LIU, XINXIN LIU, AND XIAOLIN LI. **Guoguo: Enabling Fine-grained Indoor Localization via Dmartphone**. In *ACM MobiSys*, pages 235–248, 2013. 2, 7, 48
- [7] ANSHUL RAI, KRISHNA KANT CHINTALAPUDI, VENKATA N. PADMANABHAN, AND RIJUREKHA SEN. **Zee: Zero-effort Crowdsourcing for Indoor Localization**. In *ACM MobiCom*, pages 293–304, 2012. 2, 5, 6, 8, 10, 22, 38, 39, 40, 48, 65

-
- [8] HE WANG, SOUVIK SEN, AHMED ELGOHARY, MOUSTAFA FARID, MOUSTAFA YOUSSEF, AND ROMIT ROY CHOUDHURY. **No Need to War-drive: Unsupervised Indoor Localization.** In *ACM MobiSys*, pages 197–210, 2012. 2, 5, 8, 48
- [9] JUN-GEUN PARK, BEN CHARROW, DOROTHY CURTIS, JONATHAN BATTAT, EINAT MINKOV, JAMEY HICKS, SETH TELLER, AND JONATHAN LEDLIE. **Growing an Organic Indoor Location System.** In *ACM MobiSys*, pages 271–284, 2010. 2, 5, 6, 48, 65
- [10] SOUVIK SEN, BOŽIDAR RADUNOVIC, ROMIT ROY CHOUDHURY, AND TOM MINKA. **You Are Facing the Mona Lisa: Spot Localization Using PHY Layer Information.** In *Proc. of the 10th ACM MobiSys*, pages 183–196, 2012. 2, 73
- [11] X. JIE AND J. KYLE. **ArrayTrack: A Fine-grained Indoor Location System.** In *Proc. of the 10th USENIX NSDI*, pages 71–84, 2013. 2, 7, 73, 76, 77, 78, 79, 80, 82
- [12] S. SEN, J. LEE, K. KIM, AND P. CONGDON. **Avoiding Multipath to Revive Inbuilding WiFi Localization.** In *Proc. of the 11th ACM MobiSys*, pages 249–262, 2013. 2, 73, 76, 78, 79, 80
- [13] R.O. SCHMIDT. **Multiple Emitter Location and Signal Parameter Estimation.** *IEEE Trans. on Antennas and Propagation*, **34**(3):276–280, 1986. 4, 74, 79, 83, 84
- [14] ALEX VARSHAVSKY, ANTHONY LAMARCA, JEFFREY HIGHTOWER, AND EYAL DE LARA. **The SkyLoc Floor Localization System.** In *Proc. of IEEE PerCom*, pages 125–134, 2007. 5
- [15] STEPHEN P. TARZIA, PETER A. DINDA, ROBERT P. DICK, AND GOKHAN MEMIK. **Indoor Localization without Infrastructure using the Acoustic Background Spectrum.** In *ACM MobiSys*, pages 155–168, 2011. 5
- [16] CHENGWEN LUO, HANDE HONG, AND MUN CHOON CHAN. **PiLoc: a self-calibrating participatory indoor localization system.** In *Proceedings of the 13th international symposium on Information processing in sensor networks*, pages 143–154. IEEE Press, 2014. 5, 8

-
- [17] P. BAHL AND V. N. PADMANABHAN. **RADAR: An In-Building RF-based User Location and Tracking System**. In *Proc. of IEEE INFOCOM*, pages 775–784, 2000. 5, 6, 10, 33, 38, 65, 66, 68
- [18] JAEWOO CHUNG, MATT DONAHOE, CHRIS SCHMANDT, IG-JAE KIM, PEDRAM RAZAVAI, AND MICAELA WISEMAN. **Indoor Location Sensing Using Geomagnetism**. In *ACM MobiSys*, pages 141–154, 2011. 5, 49
- [19] KALYAN P. SUBBU, BRANDON GOZICK, AND RAM DANTU. **LocateMe: Magnetic-fields-based Indoor Localization using Smartphones**. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(4):27, 2013. 5, 6
- [20] YIFEI JIANG, XIN PAN, KUN LI, QIN LV, ROBERT P. DICK, MICHAEL HANNIGAN, AND LI SHANG. **ARIEL: Automatic Wi-Fi Based Room Fingerprinting for Indoor Localization**. In *ACM UbiComp*, pages 441–450, 2012. 5, 6
- [21] MARTIN AZIZYAN, IONUT CONSTANDACHE, AND ROMIT ROY CHOUDHURY. **SurroundSense: Mobile Phone Localization via Ambience Fingerprinting**. In *Proc. of ACM MobiCom*, pages 261–272, 2009. 5
- [22] PHILIPP BOLLIGER. **Redpin - Adaptive, Zero-configuration Indoor Localization through User Collaboration**. In *Proc. of ACM MELT*, pages 55–60, 2008. 6
- [23] HE WANG, SOUVIK SEN, AHMED ELGOHARY, MOUSTAFA FARID, MOUSTAFA YOUSSEF, AND ROMIT ROY CHOUDHURY. **No Need to War-Drive: Unsupervised Indoor Localization**. In *Proc. of ACM MobiSys*, pages 197–210, 2012. 6, 7, 10, 22, 27, 38
- [24] DONNIE H. KIM, KYUNGSIK HAN, AND DEBORAH ESTRIN. **Employing User Feedback for Semantic Location Services**. In *Proc. of ACM UbiComp*, pages 217–226, 2011. 6
- [25] YING ZHANG, CHUANJIANG LUO, AND JUAN LIU. **Walk&Sketch: Create Floor Plans with an RGB-D Camera**. In *Proc. of ACM UbiComp*, pages 461–470, 2012. 7
- [26] H. LIM, C.L. KUNG, J.C. HOU, AND H. LUO. **Zero Configuration Robust Indoor Localization: Theory and Experimentation**. In *Proc. of the 25th IEEE INFOCOM*, pages 1–12, 2006. 7

-
- [27] H. LIU, Y. GAN, J. YANG, S. SIDHOM, Y. WANG, Y. CHEN, AND F. YE. **Push the Limit of WiFi based Localization for Smartphones.** In *Proc. of ACM MobiCom*, pages 305–316, 2012. 7, 48, 49
- [28] XIAOFAN JIANG, CHIEH-JAN MIKE LIANG, KAIFEI CHEN, BEN ZHANG, JEFF HSU, JIE LIU, BIN CAO, AND FENG ZHAO. **Design and Evaluation of a Wireless Magnetic-based Proximity Detection Platform for Indoor Applications.** In *ACM IPSN*, pages 221–232, 2012. 7
- [29] KAMIN WHITEHOUSE, CHRIS KARLOF, AND DAVID CULLER. **A Practical Evaluation of Radio Signal Strength for Ranging-based Localization.** *ACM/SIGMOBILE Mob. Comput. Commun. Rev.*, **11**(1):41–52, 2007. 7
- [30] MOUSTAFA YOUSSEF, ADEL YOUSSEF, CHUCK RIEGER, UDAYA SHANKAR, AND ASHOK AGRAWALA. **PinPoint: An Asynchronous Time-Based Location Determination System.** In *Proc. of the 4th ACM MobiSys*, pages 165–176, 2006. 7
- [31] RAJALAKSHMI NANDAKUMAR, KRISHNA KANT CHINTALAPUDI, AND VENKATA N. PADMANABHAN. **Centaur: Locating Devices in an Office Environment.** In *Proc. of the 18th ACM Mobicom*, pages 281–292, 2012. 7
- [32] PATRICK LAZIK AND ANTHONY ROWE. **Indoor Pseudo-ranging of Mobile Devices using Ultrasonic Chirps.** In *Proc. of the 10th ACM SenSys*, pages 99–112, 2012. 7
- [33] J. LUO, H.V. SHUKLA, AND J.-P. HUBAUX. **Non-Interactive Location Surveying for Sensor Networks with Mobility-Differentiated ToA.** In *Proc. of the 25th IEEE INFOCOM*, pages 1241–1252, 2006. 7
- [34] JIE XIONG AND KYLE JAMIESON. **SecureAngle: Improving Wireless Security Using Angle-of-arrival Information.** In *Proc. of the 9th ACM HotNets*, pages 11:1–11:6, 2010. 7
- [35] 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 *Proc. of ACM UbiComp*, pages 421–430, 2012. 7, 11, 22, 38
- [36] E. FOXLIN. **Pedestrian Tracking with Shoe-mounted Inertial Sensors.** *Computer Graphics and Applications, IEEE*, pages 38–46, 2005. 7

-
- [37] PATRICK ROBERTSON, MICHAEL ANGERMANN, AND BERNHARD KRACH. **Simultaneous Localization and Mapping for Pedestrians using only Foot-mounted Inertial Sensors**. In *ACM UbiComp*, pages 93–96, 2009. 7
- [38] O. WOODMAN AND R. HARLE. **Pedestrian Localisation for Indoor Environments**. In *Proc. of the 10th UbiComp*, pages 114–123, 2008. 7, 81
- [39] 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 *ACM UbiComp*, pages 421–430, 2012. 7
- [40] OLIVER WOODMAN AND ROBERT HARLE. **Pedestrian Localisation for Indoor Environments**. In *Proc. of ACM UbiComp*, pages 114–123, 2008. 8, 22, 40
- [41] RUIPENG GAO, MINGMIN ZHAO, TAO YE, FAN YE, YIZHOU WANG, KAIGUI BIAN, TAO WANG, AND XIAOMING LI. **Jigsaw: Indoor floor plan reconstruction via mobile crowdsensing**. In *Proceedings of the 20th annual international conference on Mobile computing and networking*, pages 249–260. ACM, 2014. 8
- [42] YUANQING ZHENG, GUOBIN SHEN, LIQUN LI, CHUNSHUI ZHAO, MO LI, AND FENG ZHAO. **Travi-navi: Self-deployable indoor navigation system**. In *Proceedings of the 20th annual international conference on Mobile computing and networking*, pages 471–482. ACM, 2014. 8
- [43] FADEL ADIB AND DINA KATABI. **See Through Walls with WiFi!** In *ACM SIGCOMM*, pages 75–86, 2013. 8
- [44] CHENREN XU, BERNHARD FIRNER, ROBERT S. MOORE, YANYONG ZHANG, WADE TRAPPE, RICHARD HOWARD, FEIXIONG ZHANG, AND NING AN. **SCPL: Indoor Device-free Multi-subject Counting and Localization using Radio Signal Strength**. In *ACM IPSN*, pages 79–90, 2013. 8
- [45] YANG ZHAO, NEAL PATWARI, JEFF M. PHILLIPS, AND SURESH VENKATASUBRAMANIAN. **Radio Tomographic Imaging and Tracking of Stationary and Moving People via Kernel Distance**. In *ACM IPSN*, pages 229–240, 2013. 8
- [46] **EkaHau Positioning Engine**. <http://www.ekahau.com/>. 10

-
- [47] **Google Maps Indoor.** <http://maps.google.com/help/maps/indoormaps/>. 10, 11, 12
- [48] JEFFREY HIGHTOWER AND GAETANO BORRIELLO. **Particle Filters for Location Estimation in Ubiquitous Computing: A Case Study.** In *Proc. of ACM UbiComp*, pages 88–106, 2004. 10
- [49] JIE XIONG AND KYLE JAMIESON. **Towards Fine-Grained Radio-based Indoor Location.** In *Proc. of ACM HotMobile*, pages 13:1–13:6, 2012. 10
- [50] YIFEI JIANG, XIN PAN, KUN LI, QIN LV, ROBERT P. DICK, MICHAEL HANNIGAN, AND LI SHANG. **ARIEL: Automatic Wi-Fi based Room Fingerprinting for Indoor Localization.** In *Proc. of ACM UbiComp*, pages 441–450, 2012. 10, 11, 27
- [51] ERIK JONSSON. *Inner Navigation: Why We Get Lost and How We Find Our Way.* Scribner, 2002. 11
- [52] R.K. GANTI, F. YE, AND H. LEI. **Mobile Crowdsensing: Current State and Future Challenges.** *IEEE Communications Magazine*, **49**(11):32–39, 2011. 11
- [53] JUN-GEUN PARK, BEN CHARROW, DOROTHY CURTIS, JONATHAN BATTAT, EINAT MINKOV, JAMEY HICKS, SETH TELLER, AND JONATHAN LEDLIE. **Growing an Organic Indoor Location System.** In *Proc. of ACM MobiSys*, pages 271–284, 2010. 11, 22, 38
- [54] ZHENG YANG, CHENSHU WU, AND YUNHAO LIU. **Locating in Fingerprint Space: Wireless Indoor Localization with Little Human Intervention.** In *Proc. of ACM MobiCom*, pages 269–280, 2012. 22, 39, 40
- [55] H. SAKOE AND S. CHIBA. **Dynamic Programming Algorithm Optimization for Spoken Word Recognition.** *Acoustics, Speech and Signal Processing, IEEE Trans. on*, **26**(1):43 – 49, 1978. 24
- [56] A. DOUCET, N. DE FREITAS, AND N. GORDON. *Sequential Monte Carlo Methods in Practice.* Springer, New York, 2001. 25
- [57] SEBASTIAN THRUN, DIETER FOX, WOLFRAM BURGARD, AND FRANK DELLAERT. **Robust Monte Carlo Localization for Mobile Robots.** *Artificial Intelligence*, **128**:99–141, 2001. 27

-
- [58] BRENDAN J. FREY AND DELBERT DUECK. **Clustering by Passing Messages between Data Points.** *Science*, **315**:972–976, 2007. 30
- [59] R. TANAWONGSUWAN AND A. BOBICK. **Gait Recognition from Time-Normalized Joint-Angle Trajectories in the Walking Plane.** In *Proc. of IEEE CVPR*, pages 726–731, 2001. 31
- [60] KATRIN ARNING, MARTINA ZIEFLE, MING LI, AND LEIF KOBELT. **Insights Into User Experiences and Acceptance Of Mobile Indoor Navigation Devices.** In *Proc. of ACM MUM*, pages 1–10, 2012. 33
- [61] **Amazon Mechanical Turk.** <https://www.mturk.com/mturk/welcome>. 33, 34
- [62] SASANK REDDY, DEBORAH ESTRIN, MARK HANSEN, AND MANI SRIVASTAVA. **Examining Micro-Payments for Participatory Sensing Data Collections.** In *Proc. of ACM UbiComp*, pages 33–36, 2010. 34
- [63] MOHAMED MUSTHAG, ANDREW RAIJ, DEEPAK GANESAN, SANTOSH KUMAR, AND SAUL SHIFFMAN. **Exploring Micro-incentive Strategies for Participant Compensation in High-Burden Studies.** In *Proc. of ACM UbiComp*, pages 435–444, 2011. 34
- [64] **Google Indoor Map Uploading Manual.** <https://support.google.com/gmm/answer/1685896>. 36
- [65] GUOBIN SHEN, ZHUO CHEN, PEICHAO ZHANG, THOMAS MOSCIBRODA, AND YONGGUANG ZHANG. **Walkie-Markie: Indoor POathway Mapping Made Easy.** In *USENIX NSDI*, pages 85–98, 2013. 48, 65
- [66] W. STORMS, J. SHOCKLEY, AND J. RAQUET. **Magnetic Field Navigation in an Indoor Environment.** In *Ubi Pos. Indoor Nav. and Loc. Based Service (UPINLBS)*, pages 1–10, 2010. 52
- [67] K.P. SUBBU, B. GOZICK, AND R. DANTU. **Indoor Localization Through Dynamic Time Warping.** In *IEEE SMC*, pages 1639–1644, 2011. 52
- [68] DIETER FOX, WOLFRAM BURGARD, FRANK DELLAERT, AND SEBASTIAN THRUN. **Monte Carlo Localization: Efficient Position Estimation for Mobile Robots.** In *AAAI*, pages 343–349, 1999. 59

-
- [69] SOUVIK SEN, BOŽIDAR RADUNOVIC, ROMIT ROY CHOUDHURY, AND TOM MINKA. **You are Facing the Mona Lisa: Spot Localization Using PHY Layer Information.** In *ACM MobiSys*, pages 183–196, 2012. 69
- [70] HE WANG, SOUVIK SEN, AHMED ELGOHARY, MOUSTAFA FARID, MOUSTAFA YOUSSEF, AND ROMIT ROY CHOUDHURY. **No Need to War-drive: Unsupervised Indoor Localization.** In *Proc. of the 10th ACM MobiSys*, pages 197–210, 2012. 76
- [71] JÁNOS SALLAI, ÁKOS LÉDECZI, AND PÉTER VÖLGYESI. **Acoustic Shooter Localization with a Minimal Number of Single-channel Wireless Sensor Nodes.** In *Proc. of the 9th ACM SenSys*, pages 96–107, 2011. 77
- [72] 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 *Proc. of the 14th ACM UbiComp*, pages 421–430, 2012. 81
- [73] T.F. COLEMAN AND Y. LI. **An Interior, Trust Region Approach for Nonlinear Minimization Subject to Bounds.** *SIAM Journal on Optimization*, **6**:418–445, 1996. 93
- [74] T.M. SCHMIDL AND D.C. COX. **Robust Frequency and Timing Synchronization for OFDM.** *IEEE Trans. on Communications*, **45**(12):1613–1621, 1997. 96

Appendix A

Apply MUSIC to WiFi Signals

We have explained the principle of MUSIC algorithm from a theoretical viewpoint. But, applying MUSIC to the modern WiFi communication system is highly nontrivial, as WiFi system has a complicated modulation/demodulation mechanism. As the key of the MUSIC algorithm, are the phase offsets between the receiving antennas still available in a WiFi system? On the other hand, many up-to-date WiFi devices adopt OFDM (*Orthogonal Frequency-Division Multiplexing*) which transmitting data at multiple orthogonal subcarriers, another question is that, is it necessitated to identify the phase offsets at each subcarrier? To answer these questions, we first review how OFDM works.

By and large, a OFDM transmitter first modulate the signals with multiple subcarriers at baseband, and then up-converts the baseband signal to *Radio Frequency* (RF) (i.e., 2.4 GHz ISM), while the receiver reverses the above process, first down-converts the RF signal to baseband and then carries out demodulation. It is obvious that, the phase offsets caused by the differences in propagation distances exists in RF signal. However, manipulating signals with such a high frequency cannot be supported by hardware circuit. Additionally, using SDR, the most primitive thing we can touch is the signal samples taken by ADCs at baseband. We hereby employ the following derivations to verify if we can observe phase offset at baseband. Suppose a 802.11 channel is divided into $\tilde{N}$ orthogonal subcarriers. The OFDM yields a symbol stream in time domain t at baseband, which can be defined by

$$S'(t) = \sum_{i=0}^{\tilde{N}} \chi_i e^{j2\pi \frac{i}{T} t} \quad (-T_g \leq t \leq T)$$

where χ_i is the QAM symbol at the i -th subcarrier at the baseband, while T and T_g is the OFDM symbol time and the guard interval. It is obvious that the subcarrier is spaced by $1/T$ at

---

the baseband. Recall that χ_i can be represented by $A_i e^{j\varphi_i}$ with A_i and φ_i being the amplitude and phase which carries the information we want to transmit. The based band signal is then up-converted to RF, i.e., $S'(t)e^{j2\pi f_c t}$ where f_c is the carrier frequency. In a realistic system, the signal transmitted by a frontend is

$$\mathcal{S}(t) = \text{Re} \left(\sum_{i=0}^{\tilde{N}} e^{j(2\pi(\frac{i}{T} + f_c)t + \varphi_i)} \right)$$

with $\text{Re}(\cdot)$ representing the real part of a complex number. In fact, we have

$$\mathcal{S}(t) = \sum_{i=0}^{\tilde{N}} A_i \cos(2\pi(f_i + f_c)t + \varphi_i).$$

where $f_i = i/T$ can be regarded as the frequency of the subcarriers at baseband. At a receiver whose distance to the transmitter is d , the incoming signal can be represented by

$$\mathcal{R}(t) = \sum_{i=0}^{\tilde{N}} A_i \cos(2\pi(f_i + f_c)t + \varphi_i + \delta_i).$$

where $\delta_i = \frac{2\pi(f_i + f_c)d}{c}$ is the phase offset induced by signal propagation. Since f_i are distributed at the narrow baseband at aa scale of hundreds of KHz, while f_c is at 2.4 GHz ISM, we have $f_i \ll f_c$. Therefore, $\delta_i \approx \frac{2\pi f_c d}{c} = \frac{2\pi d}{\lambda_c}$ where λ_c is the wave length of the RF carrier (i.e., 16 cm for 2.4 GHz RF band). It is said that the phase offsets MUSIC cares are mainly determined by the RF carrier frequency; we thus have $\delta_1 = \dots = \delta_{\tilde{N}} = \delta$ By down-converting $\mathcal{R}(t)$ to baseband, we obtain I-Q signals

$$\mathcal{R}'(t) = e^{j\delta} \sum_{i=0}^{\tilde{N}} A_i e^{j(2\pi f_i t + \varphi_i)}$$

It is clearly shown that, the above equation coincides with Equation (5.1). We thus can sample at multiple synchronized antennas, and use the signal samples as the input of the MUSIC algorithm.

Publications

JOURNAL ARTICLES

1. **Chi Zhang**, Kalyan P. Subbu, Jun Luo, and Jianxin Wu, “GROPING: Geomagnetism and cROwdsensing Powered Indoor NaviGation,” in *IEEE Transactions on Mobile Computing*.(to appear)
2. Kalyan P. Subbu, **Chi Zhang**, Jun Luo, and Athanasios V. Vasilakos, “Analysis and Status Quo of Smartphone-based Indoor Localization Systems,” in *IEEE Wireless Communications Magazine*, Vol. 21, No. 4, pp 106112, August 2014
3. **Chi Zhang**, Jun Luo, “TRack Others if You can: Localized Proximity Detection for Mobile Networks,” in *Springer Wireless Networks*, Vol. 20, No. 6, pp 14771494, August 2014
4. Feng Li, **Chi Zhang**, Jun Luo, Shi-Qing Xin, and Ying He, “LBDP: Localized Boundary Detection and Parametrization for 3D Sensor Networks,” in *IEEE/ACM Transactions on Networking*, Vol. 22, No. 2, pp 567579, April 2014

CONFERENCE PAPERS

5. **Chi Zhang**, Feng Li, Jun Luo, and Ying He, “iLocScan: Harnessing Multipath for Simultaneous Indoor Source Localization and Space Scanning,” *In Proceedings of the 12th ACM Conference on Embedded Networked Sensor Systems (SenSys)*, 2014.
6. Kai Han, **Chi Zhang**, and Jun Luo, “BLISS: Budget Limited robuSt crowdSensing through Online Learning,” in *Proceedings of the 11th IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, 2014.

-
7. **Chi Zhang**, Jun Luo, and Jianxin Wu, “A Dual-Sensor Enabled Indoor Localization System with Crowdsensing Spot Survey,” in *Proceedings of the 10th IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS)*, 2014.
  8. **Chi Zhang**, Jun Luo, and Jianxin Wu, “MaWi: A Hybrid Magnetic and Wi-Fi System for Scalable Indoor Localization,” in *Proceedings of the 13th ACM/IEEE Conference on Information Processing in Sensor Systems (IPSN Poster)*, 2014.
  9. **Chi Zhang**, Jun Luo, Liu Xiang, Feng Li, Juncong Lin, and Ying He, “Harmonic Quorum Systems: Data Management in 2D/3D Wireless Sensor Networks with Holes,” in *Proceedings of the 9th IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON)*, 2012.
  10. Feng Li, Jun Luo, **Chi Zhang**, Shiqing Xin, and Ying He, “UNFOLD: UNiform Fast On-Line boundary Detection for Dynamic 3D Wireless Sensor Networks,” in *Proceedings of the 12th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, 2011.