



Doctoral Thesis

Mitigating Variability Issues for Feature-based indoor positioning

Author(s):

Zhou, Caifa

Publication Date:

2019

Permanent Link:

<https://doi.org/10.3929/ethz-b-000371393> →

Rights / License:

[In Copyright - Non-Commercial Use Permitted](#) →

This page was generated automatically upon download from the [ETH Zurich Research Collection](#). For more information please consult the [Terms of use](#).

DISS. ETH No. 26185

MITIGATING VARIABILITY ISSUES FOR
FEATURE-BASED INDOOR POSITIONING

A dissertation submitted to attain the degree of
DOCTOR OF SCIENCES OF ETH ZURICH
(DR. SC. ETH ZURICH)

presented by

CAIFA ZHOU

M.SC. HARBIN INSTITUTE OF TECHNOLOGY

BORN ON 11.07.1990

CITIZEN OF CHINA

accepted on the recommendation of

Prof. Dr. A. Wieser, examiner

Prof. Dr. J. Blankenbach, co-examiner

Prof. Dr. R. Chen, co-examiner

2019

Abstract

As a promising indoor positioning solution, feature-based indoor positioning systems (FIPSs) utilizing signals of opportunity as features are capable of providing indoor location-based services to pedestrians. The extended use of mobile devices with abundant built-in sensors has provided opportunities and challenges to FIPSs. On the one hand, these built-in sensors enable to acquire available location-relevant features for improving the positioning accuracy. On the other hand, the limited power of mobile devices creates the need of an FIPS capable of providing positioning services with affordable computational cost. Nevertheless, several challenges still limit the performance of FIPSs and hence prevent their widespread practical applicability. The positioning accuracy and computational burden have been so far particularly affected by: i) scale of the physical and feature spaces, ii) measurability variations of features, and iii) variability of feature values.

The objective of this thesis is to mitigate these variability issues and thereby enhance the practical applicability of FIPSs. Common approaches to tackle these problems include extending the positioning solution by fusing it with additional sensors or Bayesian filters. This work focuses on resolving these variability issues entirely within the context of FIPSs and thus improving its fundamental positioning performance. This research is divided into several modules aiming at: i) reducing the computational complexity to be quasi-invariant to the scale of the feature and physical spaces, ii) adapting the dissimilarity measure to be invariant to measurability variations, and iii) mitigating large errors in positioning by figuring in the variability of the feature values. The outcomes of the individual modules constitute the core contributions of this thesis.

The first part (covered in Chapter 3) deals with the computational complexity of the location estimation procedure. Compared to traditional FIPSs, additional information of two types is obtained from a reference fingerprint map (RFM). The first type of information are the measurable features within individual subregions obtained by dividing the entire region of interest into subregions. The second type of additional information are the selected features relevant for location estimation regarding individual subregions. The former is employed to select candidate subregions by set operations solely based on the measurability of the individual features. The subregion selection procedure is not affected by the measurability variations and it makes the location estimation quasi-invariant to the scale of the physical space. The latter is used to estimate the location within the selected subregions. Owing to the finite number of relevant features of each individual subregion, the refinement of the location estimation is independent from the scale of the physical and feature spaces.

The second part (covered in Chapter 4) handles the measurability variations. The compound dissimilarity measure (CDM) is proposed for adap-

tively quantifying differences in the feature space. The proposed CDM does not require imputing the missing features in order to obtain a full measurement vector, thereby avoiding introducing inductive biases to the dissimilarity measure. The CDM combines a distance metric with set operations. The set operations are used for adaptively weighting the contribution of individual features concerning their measurability.

The third part (covered in Chapter 5) focuses on mitigating large errors caused by the variability of the feature values. An iterative scheme capable of figuring in the variability of the feature values into the location estimation is proposed. The variability of individual features is quantified using an empirical standard deviation. The location-wise standard deviation is approximated based on the measurements associated to the neighborhood of a given reference location. The contribution of the individual features to the dissimilarity measure is regulated relative to their estimated variability values.

The experiments using a kinematically collected RFM have been employed to validate the presented solutions quantitatively and qualitatively. By tracking the movement of pedestrians with a high precision total station, the reference locations and their associated features are acquired simultaneously. The feature values are measured by the sensors built into a hand-held mobile device. In experiments with the real data, promising results have been observed. Benefiting from the subregion and feature selections, positioning time is reduced by up to 10 times while preserving the positioning accuracy. With the combination of a distance metric and set operations, the accuracy for detecting floors is increased from 81% to 94%. By introducing the location-wise variability into an iterative positioning process, the positioning accuracy defined as the percentage of locations whose radial positioning error is less than 2 m is improved from 65% to 86% when compared to the original k NN. In addition, the spatial distribution of the variability of the feature values qualitatively reveals the correlation between the high variances of the feature values and large errors.

In the course of this thesis some of the relevant challenges in FIPSs have been addressed. The solutions proposed herein can therefore enhance the practical applicability of FIPSs and contribute towards enabling final products for providing indoor location-based services to pedestrians.

Zusammenfassung

Als vielversprechende Indoor-Positionierungslösung können merkmalsbasierte Indoor-Positionierungssysteme (engl. feature-based indoor positioning system, FIPS) die Opportunitätssignale als Merkmale nutzen und Fussgängern standortbezogene Dienste in Innenräumen anbieten. Die weitverbreitete Nutzung mobiler Geräte mit den zahlreichen, integrierten Sensoren bietet dem FIPS Chancen und Herausforderungen. Einerseits ermöglichen diese eingebauten Sensoren die verfügbaren, ortsrelevanten Merkmale zur Verbesserung der Positionierungsgenauigkeit zu erfassen. Andererseits muss das Positionierungssystem den Ortungsdienst, wegen der begrenzten Leistung der mobilen Geräte, zu erschwinglichen Rechenkosten zur Verfügung stellen. Dennoch schränken verschiedene Herausforderungen die Leistung von FIPS ein und verhindern ihre weit verbreitete praktische Anwendbarkeit. Die Positionierungsgenauigkeit und die Rechenlast wurden bisher insbesondere beeinflusst durch: i) die Skalierung des Merkmals und der physikalischen Räume; ii) die Messbarkeitsvariationen von Merkmalen; und iii) die Variabilität der Merkmalswerte.

Das Ziel dieser Arbeit ist es, diese Variabilitätsprobleme zu mindern und damit die praktische Anwendbarkeit von FIPS zu verbessern. Traditionelle Ansätze zur Lösung dieser Probleme umfassen die Erweiterung der Positionierungslösung durch die Fusion mit zusätzlichen Sensoren oder Bayes'schen Filtern. Diese Arbeit konzentriert sich darauf, diese Probleme vollständig im Rahmen des FIPS zu lösen und damit die Leistungen der Positionierungslösung intrinsisch zu verbessern. Diese Forschung gliedert sich in mehrere Module, die darauf abzielen: i) die Komplexität der Berechnung so zu reduzieren, dass sie im Verhältnis zur Grösse des Merkmals und der physikalischen Räume quasi invariant ist; ii) das Unähnlichkeitsmasses anzupassen, um gegenüber Messbarkeitsvariationen unveränderlich zu sein; und iii) die Milderung grober Positionierungsfehler durch die Berechnung der Variabilität der Merkmalswerte. Die Ergebnisse der einzelnen Module bilden den Kernbeitrag dieser Arbeit.

Der erste Teil (in Kapitel 3) befasst sich mit der rechnerischen Komplexität des Verfahrens zur Standortschätzung. Im Vergleich zu herkömmlichen FIPS werden zwei zusätzliche Informationen aus einer Referenzfingerabdruckkarte (engl. reference footprint map, RFM) abgeleitet. Die erste Art von Informationen sind die messbaren Merkmale innerhalb einzelner Teilregionen, die durch die Segmentierung des betrachteten Gebietes (engl. region of interest, RoI) gewonnen werden. Die zweite Art sind die subregional ausgewählten Merkmale, die für die Standortschätzung relevant sind. Ersteres wird verwendet, um Kandidaten-Unterregionen durch Mengenoperationen auszuwählen, die ausschliesslich auf der Messbarkeit der einzelnen Merkmale basieren. Das Subregion-Auswahlverfahren wird von den Messbarkeitsschwankungen nicht beeinflusst, und die Standortschätzung unter der Rechenaufwand für die Positionsschätzung ist praktisch unabhängig von der

Grösse des physischen Raums. Letzteres wird verwendet, um den Standort innerhalb der ausgewählten Unterregionen zu schätzen. Aufgrund der endlichen Anzahl relevanter Merkmale jeder einzelnen Teilregion ist die Verfeinerung der Standortschätzung unabhängig von der Grösse des physikalischen Raums und des Merkmalsraums.

Der zweite Teil (in Kapitel 4) behandelt die Messbarkeitsvariationen. Ein nicht vektorbasiertes Unähnlichkeitsmass wird vorgeschlagen, um Unterschiede im Merkmalsraum adaptiv zu quantifizieren. Das vorgeschlagene, zusammengesetzte Unähnlichkeitsmass (engl. compound dissimilarity measure, CDM) erfordert kein Anrechnen der fehlenden Merkmalswerte, wodurch das Einführen induktiver Biases in das Unähnlichkeitsmass vermieden wird. Das CDM kombiniert eine Abstandsmetrik mit Mengenoperationen. Letztere dienen dazu, den Beitrag einzelner Merkmale hinsichtlich ihrer Messbarkeit adaptiv zu gewichten.

Der dritte Teil (in Kapitel 5) befasst sich mit der Minderung grober Fehler, die durch die Variabilität der Merkmalswerte verursacht werden. Ein iteratives Schema wird vorgeschlagen, welches in der Lage ist, die Variabilität der Merkmalswerte in die Ortsschätzung miteinzubeziehen. Die Variabilität einzelner Merkmale wird anhand einer empirischen Standardabweichung quantifiziert. Die ortsbezogene Standardabweichung wird basierend auf den Messungen angenähert, die der Nachbarschaft eines gegebenen Referenzorts zugeordnet sind. Der Beitrag jedes einzelnen Merkmals zum Unähnlichkeitsmass wird in Bezug auf seine geschätzten Variabilitätswerte geregelt.

Experimente mit einer kinematisch gesammelten RFM wurden verwendet, um die vorgestellten Lösungen quantitativ und qualitativ zu validieren. Durch die Verfolgung der Bewegung von Fussgängern mit einer Totalstation werden gleichzeitig der Standort und die zugehörigen Merkmale erfasst. Die Merkmalswerte werden von Sensoren, welche in einem tragbaren, mobilen Gerät eingebaut sind, gemessen. In Experimenten mit realen Daten wurden vielversprechende Ergebnisse beobachtet. Durch die Auswahl von Subregionen und Merkmalen wird die Positionierungszeit um das Zehnfache verkürzt, während die Positionierungsgenauigkeit erhalten bleibt. Durch die Kombination einer Abstandsmetrik und Mengenoperationen wird die Genauigkeit der Stockwerkserkennung von 81% auf 94% erhöht. Durch die Einführung der ortsbezogenen Variabilität in einen iterativen Positionierungsprozess wird die Positionierungsgenauigkeit, definiert als der Prozentsatz der Orte, deren radialer Positionierungsfehler weniger als 2 m beträgt, im Vergleich zum ursprünglichen k NN von 65% auf 86% verbessert. Darüber hinaus zeigt die räumliche Verteilung der Variabilität der Merkmalswerte qualitativ die Korrelation zwischen hohen Varianzen der Merkmalswerte und groben Fehlern.

Diese Arbeit befasst sich mit einigen der aktuell wichtigsten Herausforderungen in FIPS. Die hier vorgeschlagenen Lösungen können daher die praktische Anwendbarkeit von FIPS verbessern und dazu beitragen, Endprodukte für die Bereitstellung von standortbezogenen Diensten in Innenräumen zu ermöglichen.

Acknowledgment

I would like to express my gratitude to all who have contributed to the accomplishment of this dissertation. Since the list is long, it is unfortunately not possible to mention all of them here.

First and foremost, I would like to thank **Prof. Dr. Andreas Wieser** for granting me the opportunity to pursue my doctoral studies under his supervision at ETH Zürich. He has supported me through all the phases of my PhD. His valuable insight into the field of indoor positioning, his extraordinary reasoning skills, and thorough knowledge of geodesy are inspirational. The process of pursuing a PhD has helped me to learn a lot and to develop my research and teaching skills.

My gratitude also goes to my PhD committee members. I want to give a big thank you to both **Prof. Dr. Jörg Blankenbach** and **Prof. Dr. Ruizhi Chen** for taking their time to carefully review my thesis. It is an honor to have both of your names on my thesis.

Mr. Robert Presl, the technician of the Institute of Geodesy and Photogrammetry, deserves my acknowledgment for his great contribution to valuable discussions about the data collection campaign and for his professional mechanical support.

Ms. Alessandra Naldi Windler also deserves my special acknowledgment. She has helped me to apply the required documents being a foreign student at ETH Zürich. In addition, I want to thank her successor, **Ms. Kerstin Achermann**. Both of them have proven extremely helpful in solving administrative issues and managing problems.

Being able to work in the group of Geosensors and Engineering Geodesy at ETH Zürich is unforgettable. I give a great thank you to my friends and colleagues Zhan Gojic, Dr. David Salido-Monzú, Dr. Jemil Avers Butt, Andreas Bruno Graziano Baumann-Ouyang, Eugenio Serantoni, Dr. Geo Boffi, Ephraim Friedli, Sukant Chaudhry, Dr. Qiao Jing, Lorenz Schmid, Valens Frangez and Dr. Mariusz Frukacz. Our many interesting discussions and group activities have made this experience great. Your help for proof-reading have significantly improved the quality of the thesis. Dr. Gu Yang, a previous visiting scholar in our group, has worked together with me during his stay. This led to fruitful collaboration. Furthermore, I want to thank our student assistants, Evelyn Weiss, Nicholas Meyer, Isabelle Bai, and Guillem Bonet Filella, for their assistance in collecting the data for my PhD project.

My acknowledgments are extended to my family and all my friends for their support. My parents Fengying Li and Yingshen Zhou, and my sisters Caizhen and Caimin have always encouraged me in every moment through these years.

Finally, I want to thank China Scholarship Council for their financial support during my studies at ETH Zürich.

CONTENTS

| | |
|-------------------------------------------------------|-----------|
| Table of Contents | vi |
| Contents | vi |
| List of Figures | ix |
| List of Tables | xi |
| 1 Introduction | 1 |
| 1.1 Motivation | 1 |
| 1.2 Background of FIPSS | 2 |
| 1.2.1 Characteristics of FIPSS | 3 |
| 1.2.2 Generation of the RFM | 4 |
| 1.2.3 Maintenance of the RFM | 6 |
| 1.2.4 Feature-based positioning algorithms | 7 |
| 1.3 Research goals | 8 |
| 1.4 Contributions and their relations | 9 |
| 1.4.1 Core contributions | 10 |
| 1.4.2 Relation of the chapters | 13 |
| 1.5 Relevance to science and society | 15 |
| 2 On generating reference fingerprint maps | 17 |
| 2.1 Overview of the data acquisition system | 17 |
| 2.2 Mobile-based RFM generation | 19 |
| 2.3 A world model of the RFM | 24 |
| 2.3.1 Spatial distribution of the raw RFM | 24 |
| 2.3.2 Continuous representation of the RFM | 25 |
| 2.3.3 Enhancement of the traditional KS | 28 |
| 2.4 Conclusion | 31 |

| | | |
|----------|----------------------------------------------------------------------|-----------|
| 3 | On reducing computational complexity | 33 |
| 3.1 | Introduction | 34 |
| 3.2 | Related work | 37 |
| 3.2.1 | Subregion selection | 37 |
| 3.2.2 | Selection of relevant features | 38 |
| 3.3 | The proposed approach | 40 |
| 3.3.1 | Problem formulation | 40 |
| 3.3.2 | Subregion selection using MJI | 41 |
| 3.3.3 | Feature selection using AFBGS | 43 |
| 3.3.4 | The combination of subregion and feature selections | 47 |
| 3.3.5 | Computational complexity of online positioning | 48 |
| 3.4 | Experimental results and discussion | 49 |
| 3.4.1 | Testbed | 50 |
| 3.4.2 | Analysis using WLAN signals in RoI 1 | 53 |
| 3.4.3 | Analysis using WLAN and BLE signals in RoI 2 | 57 |
| 3.5 | Conclusion | 59 |
| 4 | On adapting to measurability variations | 63 |
| 4.1 | Introduction | 64 |
| 4.2 | Related work of distance metrics | 66 |
| 4.3 | Compound dissimilarity measure | 66 |
| 4.4 | An application of CDMs to FbP | 69 |
| 4.4.1 | The baseline algorithm, performance criteria and data sets | 69 |
| 4.4.2 | Evaluation and comparison of different CDMs | 71 |
| 4.4.3 | Tuning the regularization value α | 72 |
| 4.4.4 | Comparison of positioning performance | 73 |
| 4.5 | Conclusion | 76 |
| 5 | On mitigating large errors in positioning | 79 |
| 5.1 | Introduction | 80 |
| 5.2 | Related work | 82 |
| 5.3 | Feature-based positioning | 83 |
| 5.3.1 | Fundamental concepts | 83 |
| 5.3.2 | Kinematically acquired RFM | 84 |
| 5.4 | Robust estimation of the feature variability | 85 |
| 5.5 | Iterative scheme for online positioning | 88 |
| 5.5.1 | Weighted dissimilarity measure | 88 |
| 5.5.2 | Iterative scheme | 89 |
| 5.6 | Experimental analysis | 92 |
| 5.6.1 | Results of the variability estimation | 92 |
| 5.6.2 | Results of iterative scheme for positioning | 93 |
| 5.7 | Conclusion | 98 |

| | | |
|----------|-----------------------------------------------------|------------|
| 6 | Conclusions and outlook | 100 |
| 6.1 | Summary | 100 |
| 6.2 | Discussion of contributions | 101 |
| 6.2.1 | Reduction of the computational complexity | 101 |
| 6.2.2 | Adaptation to measurability variations | 102 |
| 6.2.3 | Mitigation of large errors in positioning | 103 |
| 6.3 | Discussion of limitations | 103 |
| 6.4 | Outlook | 105 |
| | Bibliography | 106 |
| A | List of Acronyms | 120 |
| B | List of Publications | 122 |
| C | Curriculum Vitae | 125 |

LIST OF FIGURES

| | | |
|------|------------------------------------------------------------------------------------------------------------------------------------|----|
| 1.1 | The relation between the chapters | 14 |
| 2.1 | The schematic of the data acquisition system | 18 |
| 2.2 | The schematic of the data acquisition process | 19 |
| 2.3 | The configuration of the FIPS | 20 |
| 2.4 | The flowchart of the logging software implemented using Geo-COM | 21 |
| 2.5 | The schematic of the synchronization operation | 21 |
| 2.6 | Example of cross-correlation-based TDE | 22 |
| 2.7 | Examples of the raw RFM of two APs | 25 |
| 2.8 | Examples of the smoothed RFM taking the raw RFM as the input to the traditional KS | 27 |
| 2.9 | The influence of the value of r^{KS} on the residuals, processing time, and storage space | 28 |
| 2.10 | Examples of the smoothed RFM yielded from the modified KS using the raw RFM as the input | 29 |
| 2.11 | Examples of the spatially filtered RFM | 29 |
| 2.12 | Examples of the smoothed RFM using spatially filtered RFM as the input to the modified KS | 30 |
| 2.13 | The schematic of the modeling of the geometric constraints | 31 |
| 2.14 | Examples of the smoothed RFM using spatially filtered RFM as the input to the modified KS with the geometric constraints | 32 |
| 3.1 | The proposed framework | 41 |
| 3.2 | Qualitative examples of MJI values | 42 |
| 3.3 | Two RoIs within one floor of an office building | 50 |
| 3.4 | Examples of generated RFM | 52 |
| 3.5 | The number of the available features of each subregion | 53 |
| 3.6 | Analysis of MJI between the subregions of RoI 1 (training data). | 54 |

| | | |
|------|------------------------------------------------------------------------------------------------------------|----|
| 3.7 | The subregion selection loss with respect to the number of selected subregions | 55 |
| 3.8 | The MSE paths of two subregions | 56 |
| 3.9 | Examples resulting in large subregion selection error | 57 |
| 3.10 | Examples resulting in small subregion selection error | 59 |
| 4.1 | The scheme of calculating the dissimilarity measure into three parts | 67 |
| 4.2 | An example of RMSE of three CDMs with $\alpha = 1.0$ | 72 |
| 4.3 | CV boxplots of <i>HIL</i> ($k = 1$, RCDM) | 73 |
| 4.4 | CV boxplots of <i>UJIIndoorLoc</i> ($k = 1$, RCDM) | 74 |
| 4.5 | Comparison of RMSE ($k = 1$, RCDM) | 75 |
| 4.6 | Comparison of success rate and RMSE of <i>UJIIndoorLoc</i> ($k = 1$) | 76 |
| 4.7 | Comparison of ECDF between different distance metrics (<i>HIL</i> , $k = 1$) | 77 |
| 5.1 | Examples of the raw and spatially filtered RFM of two arbitrarily selected APs | 85 |
| 5.2 | Schematic representation of the spatial filtering | 87 |
| 5.3 | Examples of kernel smoothed RFM of two arbitrary selected APs | 93 |
| 5.4 | Examples of the absolute residuals and estimated STD for two APs | 94 |
| 5.5 | Examples of iteratively searched location with different initializations | 95 |
| 5.6 | Comparison of the percentage of locations terminated with different conditions for $k = 1$ and 3 | 96 |
| 5.7 | The maximum distance between the locations consisting of the loop state for $k = 1$ | 96 |
| 5.8 | Comparison of ECDF with respect to the radial positioning errors | 97 |
| 5.9 | Distribution of the locations yielding large errors (> 8 m) in positioning | 98 |

LIST OF TABLES

| | | |
|------|----------------------------------------------------------------------------|----|
| 2.1 | The summary of the raw RFM (WLAN and BLE) | 24 |
| 3.1 | Selected acronyms used herein | 37 |
| 3.2 | Pseudocode of the forward greedy search algorithm | 45 |
| 3.3 | Pseudocode of the backward greedy search algorithm | 45 |
| 3.4 | Pseudocode of the AFBGS algorithm | 46 |
| 3.5 | Pseudocode of the proposed online positioning approach | 48 |
| 3.6 | Summary of RFM characteristics | 53 |
| 3.7 | The selected features of subregion 17 | 56 |
| 3.8 | Comparison of positioning times and accuracies of RoI 1 | 58 |
| 3.9 | Comparison of positioning times and accuracies of RoI 2 (WLAN) | 60 |
| 3.10 | Comparison of positioning times and accuracies of RoI 2 (BLE) | 61 |
| 4.1 | Distance metrics used herein | 69 |
| 4.2 | The characteristics of the datasets | 71 |
| 4.3 | Positioning results of <i>UJIIndoorLoc</i> | 75 |
| 5.1 | Radial positioning errors (meters) | 97 |

CHAPTER 1

INTRODUCTION

1.1 Motivation

Location awareness has become increasingly important in the era after the Age of Discovery. Relying only on the astronomic phenomena and geomagnetism is inadequate to satisfy the increasing demand of accurate location information. Benefiting from the sophistication of the radio technology, people have been able to apply the radio waves to the positioning solutions regardless of the weather conditions (Bhatta 2010). This gave rise to radio positioning systems such as Decca, OMEGA, Gee, and Loran that have been used until recently (Dippy 1946; Powell 1958). Later on, with the advancement of electronic technologies and space engineering, global navigation satellite systems (GNSSs), pioneered by the global positioning system (GPS), have been developed (Zaidi and Suddle 2006). These satellite-based positioning systems are capable of providing accurate position, velocity, and timing services to both military and civil applications (Bhatta 2010).

Unfortunately, GNSSs do not work well in indoor environments. This is primarily caused by the lack of clear line-of-sight propagation between the satellites and the receiver, since the satellite signals are easily blocked by buildings. To be able to provide sufficient location-based services (LBSs) to pedestrians in indoor environments, where modern people spend more than 80% of their time, alternative positioning systems capable of working under the indoor conditions are needed (Wu et al. 2018a).

Emerged about two decades ago, indoor object/pedestrian tracking systems have been attracting the attention of both academia and industry. Es-

pecially in academia, the researchers have sought for indoor positioning systems, which are capable of yielding accurate and low-cost indoor LBSs. Numerous technologies using ranges, angles or features as the observations, have been explored and exploited for inferring the location (Mautz 2012). These technologies make use of: i) ultrasound (e. g. Active Bat (Addlesee et al. 2001), Dolphin (Minami et al. 2004), or Cricket (Kolodziej and Hjelm 2006)); ii) infrared signals (Active Badge (Want et al. 1992), (Wang et al. 2017)); iii) radio frequency (RF) signals (e. g. radio frequency identification (RFID) (Hightower et al. 2000; Xu et al. 2017), ultra-wideband (UWB) signals (Norrdine et al. 2013; Alarifi et al. 2016), wireless local area network (WLAN) channels (Padmanabhan et al. 2000; Youssef and Agrawala 2008; He and Chan 2016), Bluetooth low energy (BLE) communications (Zhuang et al. 2016), frequency modulation (FM) signals (Cong et al. 2018), cellular networks (Schloemann et al. 2016)); iv) machine vision (Guan et al. 2016); v) magnetic fields (Kasmi et al. 2015; Norrdine et al. 2016; He and Shin 2018); vi) inertial sensors (Elloumi et al. 2016); and vii) visible light (Vadeny et al. 2016).

Irrespective of the used technology, the solution for indoor positioning of pedestrians should be cost efficient (e. g. the infrastructure or deployment burden), have a sufficiently high accuracy (e. g. a few meters or better), and a large spatial coverage. Recent research reveals one type of promising indoor positioning system, referred to as a feature-based (i. e. fingerprinting-based) indoor positioning system (FIPS) owing to: i) the popularity of mobile devices with abundant built-in and off-the-shelf sensors (e. g. inertial measurement units (IMUs) or WiFi/BLE module); ii) the development of data processing and interpretation (e. g. crowd sourcing and machine learning); and iii) the availability of measurable location-relevant signals of opportunity (SoP) (He and Chan 2016; Pei et al. 2016), i. e. any available signals that can be used for positioning, but are not primarily deployed for this purpose.

The FIPs that use the SoP, which are measured by the built-in sensors of a mobile device, do not require any additional infrastructure and hardware. However, due to the several challenges that require further investigations, their practical applicability is still hindered. In the course of this thesis some of these relevant challenges, such as computational complexity of positioning, measurability variations of the features, and variability of the feature values, are addressed.

1.2 Background of FIPs

An FIPS consists of two phases: an offline referencing phase and an online positioning phase. During the referencing phase, a world model, representing

the relationship between the measurable features, e. g. received signal strength from WLAN access points (APs) and their corresponding locations, is established. This world model is traditionally denoted as the reference fingerprint map (RFM), which is a more general terminology as the radio map. From the perspective of continuous distribution of the measurable features, the RFM is a model of the continuous field in the region of interest (RoI) (called physical space). During the online positioning phase, the user's location is inferred by matching the online measured features to the RFM according to either the chosen similarity/dissimilarity measure (e. g. Euclidean distance) or the selected probabilistic model (e. g. Gaussian process) with an optimization scheme (e. g. maximum a posteriori (MAP)).

1.2.1 Characteristics of FIPSSs

In order to achieve sufficiently accurate location estimation using the SoP as features, the values of the SoP have to be location-dependent and have a sufficient spatial gradient. The application of the location-relevant features leads to the intrinsic characteristics of FIPSSs. One characteristic is the measurability variation, i. e. a specific feature is observable only in a part of the physical space. The dimensionality of the feature space (i. e. the number of the measurable features) varies over the physical space. The widely adopted positioning algorithms, as described in Section 1.2.4, are only capable of taking vector-represented features as the input. This generates a large overhead and deteriorates the performance in the online positioning phase, because the values of the non-measurable features have to be imputed in order to fill the missing elements of the feature vector. Another characteristic is the number of the measurable features needed to ensure a sufficient spatial coverage throughout the whole RoI. Indeed, the minimum number of the measurable features increases with the increase of the physical space. This leads to the scalability problem related to physical and feature spaces, which has a large impact on the computational complexity when inferring the user's location. It becomes especially critical when estimating the pedestrian's location using a mobile device, which has limited computational resources and requires a low latency.

The employment of SoP and mobile devices contributes to another, possibly the most important, characteristics of FIPSSs. In the context of SoP, the quality of the measurable features is difficult to guarantee. On the one hand, the availability of the SoP is priorly unknown. Especially the total number of measurable features within the entire RoI is difficult to determine without a site survey covering the whole RoI. On the other hand, unexpected changes of SoP can occur, since their configuration and management are under the control of other parties. Considering the mobile devices, the diversity of measuring devices introduces further variations into the measurable features. On

the one hand, the limited sensitivity of the sensor built into the devices also leads to measurability variations and introduces measurement noise. On the other hand, diverse specifications of the devices makes it difficult to: i) control the quality when generating the RFM; and ii) preserve an adequate positioning accuracy when using a different device. These characteristics make it challenging to employ FIPSS to a large variety of practical cases.

To sum up, the characteristics, originating from the intrinsic properties of FIPSS in combination with the characteristics of the SoP and mobile devices, have a strong impact on both phases of the FIPSS. The measurability variations and the scalability problem (i. e. the scale of feature and physical spaces) have little impact on the offline phase and they have to be handled specifically during the online positioning phase. The characteristics resulting from employing the SoP and mobile devices to FIPSS represent a tradeoff to not requiring the extra infrastructure. The difficulties of controlling the quality of the RFM require specific treatment during either both, or at least one of the phases of the FIPSS. During the offline phase, the process of collecting the RFM should be well-controlled and the generated RFM should be kept up-to-date in order to preserve the positioning performance. During the online positioning phase, the feature-based positioning approach should be robust enough to be capable of achieving adequate positioning accuracy even in the case of the increased variability of measurable features.

1.2.2 Generation of the RFM

The generation of the RFM is the starting step of deploying an FIPS. Various approaches, addressing the trade-off between the accuracy of the RFM and the cost of time/labor, have been proposed for its generation (He and Chan 2016; Zhou and Wieser 2019b). These approaches can be divided into the following groups:

- **Measurement-based:** This method is the most adopted method for the RFM generation (Torres-Sospedra et al. 2014; Youssef and Agrawala 2008; Padmanabhan et al. 2000). A dedicated site survey is carried out to collect the values of measurable features at given reference locations with known coordinates within the RoI. Often, the measurement-based RFM generation is conducted in the static way, i. e. the surveyor stands still at individual reference locations when collecting the data. The resulting RFMs can correctly reflect the relationship between the location and features at the time of acquiring the data, but do not consider the temporal changes of the SoP. Furthermore, such a measurement-based way is labor-intensive and time-consuming. In addition, the generated RFMs cannot reflect the influence of the motion status (e. g. orientation or moving speed) of the pedestrians.

- **Model-based:** Indoor propagation models (e. g. (El-Kafrawy et al. 2010; Bisio et al. 2014; Jung et al. 2011)) and ray tracing (Renaudin et al. 2018) can be employed to generate the RFMs, especially in the case of using radio frequency signal strengths as features. The model-based approaches are cost-effective. However, the accuracy of the generated RFMs depend on the validity of the model assumptions (e. g. structures of indoor space, location of signal sources or material properties of objects). Therefore, in the real world scenarios, the RFMs generated using the model-based methods are less accurate than the RFMs generated using the measurement-based approaches. In addition, it is difficult to incorporate a realistic motion model into the model-based approaches.
- **Crowd-sourcing-based:** The crowd-sourcing-based approaches benefit from the abundance of the built in sensors of the mobile devices. The data collection can be done by an application running in the background on the mobile device under the consent of the user for contributing the data. In fact, crowd-sourcing-based approaches can be understood as a special case of the measurement-based ones. In the context of crowd-sourcing, the reference locations are either manually indicated by the user (Wu et al. 2013; Li et al. 2013) or determined using pedestrian dead reckoning (PDR) according to the readings of IMUs (Georgiou et al. 2015; Zeinalipour-Yazti and Laoudias 2017; Wu et al. 2018b). Crowd-sourcing-based approaches represent a low cost solution of collecting a large amount of data using diverse types of mobile devices. The generated RFMs include a variety of mobile devices as well as the motion characteristics of pedestrians. However, the quality of the RFMs is difficult to control. On the one hand, the reference locations—either reported by the user or determined using IMUs—are inaccurate. On the other hand, it is difficult to fuse the data collected by multi-users using various devices. Each individual dataset is referred to its own coordinate frame and has a different quality. These two factors require a registration of reference locations and quality control when merging different datasets to generate the RFMs (Gu et al. 2017, 2018).
- **Regression-based:** The regression-based approaches are widely used for refining and not for generating the RFMs, with the goal of enhancing their quality (e. g. densify the reference measurements or denoising the generated RFMs). It can be done by interpolation/extrapolation (Sorour et al. 2015; Talvitie et al. 2015), kernel smoothing (Huang and Manh 2016), or neural networks-based reconstruction (Zhou and Wieser 2016; Zhou and Gu 2017). However, the quality of the RFMs, refined using this type of methods, strongly depends on the quality of the initially collected ones. In addition, regression-based methods have high computational burden and the regressed models are difficult to in-

terpret, especially when deep neural networks are employed (Deng and Yu 2014; Hatcher and Yu 2018).

1.2.3 Maintenance of the RFM

In the FIPSs it is crucial that the RFM is kept up-to-date in order to control its quality and thereby preserving the positioning performance of FIPSs for a long-term. Most publications regarding this topic were published in past four years but several keywords, such as maintenance, adaptation, update, and renovation, are regularly used. The first three words are more frequently used than renovation and have been utilized interchangeably in the literature.

Taniuchi and Maekawa (2015) proposed a method for automatically updating the RFM based on PDR using the data provided by the IMU of the mobile devices. They proposed a change detection step for determining when the feature values in a sequence of measurements have been critically changed using Bayesian information criterion (BIC). Tang et al. (2015) introduced a fast fingerprint collection platform using unmanned ground vehicles (UGVs) with LiDAR and other sensors for simultaneous localization and mapping (SLAM). Both works do not consider the maintenance scheme and directly replace the RFM with the newly measured feature values. In He et al. (2017) the authors proposed a system for detecting the altered signals based on affinity propagation clustering algorithm (Frey and Dueck 2007). Consequently, the altered features are updated with new values using Gaussian process regression (GPR). The Log-distance path loss model (LDPLM) was employed for estimating the parameters of the Gaussian process when applying GPR. This model-based way has largely constrained the applicability of their approach.

Various RFM update schemes have been introduced in 2018. Jung and Han (2018) have proposed a method based on the evolutionary algorithm for updating the available RFM using the crowd-sourced data, which has no ground truth. Their updating process relies on LDPLM in order to obtain the positions of those unlabeled data. Sun et al. (2018) have proposed the RFM update method using crowd-sourced data in which the updates are performed Voronoi-wisely using neural network-based regression. Mendoza-Silva et al. (2018b) and Montoliu et al. (2018) have approached the RFM update task from a different perspective. They focus on minimizing the workload of re-collecting the data for the update, by optimally selecting the locations where the data collection should be performed using the evolutionary algorithm. Additionally the influence of missing features is mitigated using the imputation schemes. Kernelized support vector regression (SVR) has been applied for adapting the newly measured features to the available RFM. The work of Peng et al. (2018) extends the proposed approach of Tang et al. (2015) in ap-

plying a Kriging approach for updating the RFM by interpolating over the UGV measured data. All above mentioned works neglect the changes that might have occurred between the acquisitions and directly update the RFM with the newly collected data using a selected regression-based approach.

Another group of works utilizes the PDR for the RFM update, e. g. (Wu et al. 2018b; Yang et al. 2018; Tao and Zhao 2018; Zou et al. 2017). Based on the measurements acquired together with PDR, different approaches (e. g. partial least square regression (PLSR) (Wu et al. 2018b), GPR with extended Kalman filter (EKF) (Yang et al. 2018), fireworks algorithm (Tao and Zhao 2018), or polynomial surface fitting mean (PSFM) method (Zou et al. 2017)) have been used for adapting the RFM. However, the performance of these approaches is largely limited by the accuracy of the PDR.

1.2.4 Feature-based positioning algorithms

Inferring the user's location from the online measured features can be understood as a regression problem. In the context of feature-based positioning, this regression problem is resolved in a data-driven way, i. e. a model for estimating the user's location is directly derived (or learned) from the RFM without specifically assumed priors. Starting from the pioneering works of Padmanabhan et al. (2000) and Youssef and Agrawala (2008), the feature-based positioning algorithms can be divided into two categories: deterministic and probabilistic approaches.

Deterministic approaches. The deterministic methods employ a similarity/dissimilarity measure (e.g., Euclidean distance, cosine similarity or Tanimoto similarity) to match the online measured features to the RFM (Minaev et al. 2017). The user's location is estimated from the reference locations, which are similar in the feature space, i.e. whose corresponding feature values obtained from the RFM are similar to the online measured ones. Often, the k -nearest-neighbors (k NN) algorithm is used to calculate the user position in this framework (Padmanabhan et al. 2000). Other advanced deterministic algorithms, for instance support vector machine (SVM) (Wu et al. 2004), linear discriminant analysis (LDA) (Nuño-Barrau and Páez-Borrillo 2006) and neural networks-based regression (Sun et al. 2018; Zhou and Wieser 2016), show better positioning accuracy but have a higher computational cost than k NN. The advantage of deterministic algorithms is their low computational complexity. However, they cannot predict the accuracy of location estimation and the accuracy assessment can only be performed with field experiments.

In the context of the deterministic positioning approaches, the measurability variations have to be handled by the dissimilarity measure. Typically, the measured features are formulated as a vector, which has the same dimension

as the total number of the measured features contained in the RFM. The features, which are not observable at a specific location, are imputed by a value (e. g. a missing indicator) when representing them as a vector (Montoliu et al. 2018; Little 2015; Lee et al. 2012). However, the imputation of missing features can introduce inductive bias to the dissimilarity measure. Furthermore, the computational burden of imputing the features increases with the increase of the feature and physical spaces.

Probabilistic approaches. The probabilistic approaches use the statistical inference based on the RFM and the online measured feature values. Using the RFM, these approaches estimate the user’s position with maximum likelihood estimation (MLE) or MAP. For example, Youssef and Agrawala (2008) first approximate the probabilistic distribution of the RFM (e.g., multivariate Gaussian distribution) at each reference location and then estimate the user’s location with an MAP algorithm. The accuracy of the estimated positions using this probabilistic algorithm depends on the accuracy of the estimated probability distribution of the feature values at each reference location. Other probabilistic methods used for feature-based positioning are based on expectation-maximization (Ouyang et al. 2012), Kullback-Leibler divergence (Mirowski et al. 2012), or mixture of Gaussian processes (Ferris et al. 2007).

The probabilistic algorithms usually require assumptions regarding the stochastic properties of both feature values and RFM, such as the example that each individual feature being identically and independently distributed and the noise of the measured feature values having a Gaussian distribution. Furthermore, a higher computational effort and more datasets are required to train a probabilistic model than a deterministic one. Specifically, several measurements of all measurable features at each reference location are needed to estimate the distribution of the feature values for all locations, without applying interpolation (e.g., Kriging) (Seitz et al. 2010). The probabilistic algorithms allow predicting the positioning accuracy without field tests, using the assumptions.

1.3 Research goals

As discussed in the previous sections, several challenges still need to be resolved in order to ensure an adequate quality of indoor LBSs completely relying on an FIPS, which employs SoP as features. In this thesis, several of them, regarding the scalability problem, measurability variations, and variability of the feature values are addressed. The investigation is carried out in various aspects by using methodological solutions and software techniques.

Specifically, the following research questions are explored:

- **How can the computational complexity of the indoor positioning be constrained in order to make it (quasi-)independent of the scale of the physical and feature spaces?** Solving this question will enable to compute the user's location on a mobile device in the case of the increase of the size of the physical space as well as the dimension of the feature space. In addition, the low computational solution should be able to preserve a sufficiently high positioning accuracy.
- **Can the measurability variations of SoP be handled in an adaptive scheme by exploiting their intrinsic properties?** An ideal solution for resolving this problem aims not only at mitigating the measurability variation problem, but also at benefiting from it in order to improve the performance of an FIPS. The goal of developing an adaptive scenario for handling the measurability variations is to improve the positioning accuracy while reducing the memory complexity of an FIPS.
- **How can the large errors in positioning, caused by the spatial/temporal variability of the feature values, be reduced?** The goal is to explore and exploit an approach, which is able to reduce large errors in positioning completely within the framework of the FIPS. This implies that the solution does not require additional information (e. g. indoor maps) or extra infrastructural support (e. g. inertial navigation systems).

The solutions developed in the course of this thesis will enable mitigating the impact of the variability issues on the practical applicability of FIPSS as well as on their positioning performance. This in turn will lead to the reduction of the computational complexity of positioning, to the adaptation to measurability variations of SoP, as well as to the mitigation of large errors in FIPS. By pursuing solutions completely within the context of FIPSSs, several benefits are foreseen. On the one hand, the enhanced FIPSSs will be suitable for applications with the limited computational and power resources (e. g. the Internet of Things). On the other hand, it will be possible to further improve their performance by combining them with other techniques (e. g. PDR or Bayesian filtering).

1.4 Contributions and their relations

This thesis is written as a *cumulative dissertation*, in accordance with the ETH Zürich, Department of Civil Environment and Geomatics Engineering, Doctorate Ordinance of 2013. The core chapters (Chapter 3, 4, and 5) consist

of the publications in their original versions, only with minor typographical corrections¹. In addition, they are complemented with an introduction (the current chapter), a detailed description of the data acquisition used in the core publications (Chapter 2) and finally with conclusions and outlook (Chapter 6).

1.4.1 Core contributions

This thesis contributes to the advancement of FIPS in the context of employing SoP, which are measured by the built-in sensors of a mobile device, to provide adequate positioning services to pedestrians. Its main contributions are: i) a method for kinematic generation of the RFM; ii) reduction of the computational complexity; iii) adaptation to measurability variations; and iv) mitigation of large errors.

Generation of the RFM (Chapter 2). The generation of the RFM used for validating the proposed solutions of the research questions is discussed in Chapter 2. This RFM has the following characteristics:

1. The measured SoP, which are available in a typical office building are collected using a mobile device. This configuration fulfills the requirements stated in the research questions. On the one hand, these signals are deployed for the purpose of providing the Internet access and are managed by a third party. On the other hand, they are measured using the built-in sensors of the mobile device. The former reflects the characteristics of SoP, whose detailed information (e. g. the configuration of emission power or the number of signal sources) are priorly unknown. The latter can justify the case of providing positioning services to pedestrians using the mobile device and its built-in sensors.
2. The data collection is performed by a pedestrian who is continuously tracked by a high precision position referencing system. The continuous tracking is achieved by utilizing the automatic target recognition (ATR) mode of a total station. The trajectory of the pedestrian is observed by periodically logging the highly accurate angle and distance measurements by the software developed using the GeoCOM programming interface of the total station. The precisely tracked trajectory enables to accurately retrieve the coordinates of locations, where the fingerprints are obtained. The simultaneously acquired feature values and their corresponding precise reference locations enable the generated RFM to reflect the mobility of pedestrians while at the same time containing the trustworthy reference locations.

¹The references are listed in the end of the thesis to reduce repetitions.

The aforementioned characteristics make the presented approach to kinematically collect RFM a potentially very useful contribution to the field of indoor positioning. The dataset collected for the experimental research parts within this thesis has been made publicly available.

Reduction of the computational complexity (Chapter 3). In Chapter 3, an approach for reducing the data requirements and computational complexity during the online positioning phase, is proposed. The proposed approach has two important aspects: i) it does not require to know the total number of measurable features throughout the RoI when coarsely estimating the user's location; and ii) it is able to reduce the computational complexity of online positioning to the level, which is less dependent on the increasing size of the RoI and on the number of the measurable features. The proposed approach benefits from the fact that each individual measurable feature has limited spatial coverage and its feature value is location-dependent. The core contributions of Chapter 3 are as follows:

1. The primary step is to segment an RoI into subregions. For each individual subregion, a set of measurable features is summarized and a subset of preselected relevant features is identified by analyzing the collected RFM. These two analyses are carried out during the offline phase. The yielded data are used for coarsely determining as well as for refining the user's location estimation.
2. Several subregions are identified as the coarse estimation of the user's location using a modified Jaccard index (MJJ), which quantifies the similarity between the features observed by the user and those available within each individual subregion. The computation of the MJJ only takes the measurability of the features into consideration and does not require the information of the total number of the measurable features throughout the RoI.
3. An adaptive scheme is proposed for determining an adequate number of subregions with the goal of making the generalization of the MJJ-based subregion selection approach to different indoor regions and different features effortless. The scheme also enables adjusting the size of the searching space when refining the user's location.
4. The feature selection is achieved using an adaptive forward-backward greedy search (AFBGS), which based on a given feature based positioning method (e.g. k NN) determines a subset of relevant features for each subregion. Due to the fact that only a finite number of relevant features of each subregion are chosen by the AFBGS, the computational complexity of searching the refinement of the user's location within those selected subregions is independent of the total number of measurable features and of the size of the entire RoI.

5. In an empirical study using the kinematically collected RFM the proposed subregion and feature selection reduce the processing time during the online-stage by a factor of about 10, while the positioning accuracy does not deteriorate significantly. In fact, in one of the two study cases the 90th percentile of the circular error increased by 7.5% while in the other study case we even found a reduction of the corresponding circular error by 30%.

Adaptation to measurability variations (Chapter 4). An innovative dissimilarity measure is presented in Chapter 4. The dissimilarity measure quantifies the difference between the observed features and the ones stored in the RFM, with adaptive handling of the variations of the measurability of the features, by combining vector-based distance metrics with set operations. The proposed compound dissimilarity measure (CDM) is able to quantify similarity of collected attribute/feature pairs where not all features are presented in all measurements without imputing the value of the missing features. The CDM overcomes a typical challenge in the context of feature-based positioning algorithms, which can only take the vector representation of the features as the input. The core contributions of Chapter 4 are as follows:

1. The initial proposal of the CDM and different weighting schemes are introduced. Compared to the vector-based distance metrics (e. g. Minkowski), the merits of the proposed CDM are i) the data do not need to be converted to vectors of equal dimension; ii) shared and unshared attributes can be weighted differently; and iii) additional degrees of freedom within the measure allow to adapt its properties to application needs in a data-driven way.
2. The parameters of the CDM are determined using grid search in combination with the cross-validation. This makes the proposed CDM applicable to different FIPSSs, which are built in different indoor environments and using various SoP as features.
3. The performance of the proposed CDM metric is evaluated on the kinematically collected RFM as well as on the three publicly available data sets. On the publicly available data sets, CDM improves the accuracy of identifying buildings and floors by about 5 percent points, when compared to the conventional distance metrics. Furthermore, the 2d positioning errors in terms of root mean squared error (RMSE) are reduced by a factor of two, and the percentage of position solutions with less than 2 m error improves by over 10 percent points.

Mitigation of large errors (Chapter 5). An iterative scheme for feature-based positioning using a new weighted dissimilarity measure with the goal

of reducing the impact of large errors among the measured or modeled features is proposed in Chapter 5. The estimation of the variability of individual features during the offline referencing phase is carried out for understanding the sources of large errors. The estimated variability of features is employed to enhance the robustness of the location estimation. This is achieved by weighting the contribution of each individual feature to the dissimilarity measure according to its STD value when inferring the user's location. The core contributions of Chapter 5 are as follows:

1. An empirical analysis of the spatial distribution of the raw measurements contained in the kinematically collected RFM is carried out. The spatial filtering and kernel smoothing (KS) are employed for improving the spatial consistency of the raw measurements and for providing the continuous representation of the RFM. These two techniques enable to estimate the location-wise standard deviation (STD) of each individual feature using the median absolute deviation (MAD) of the feature values associated to the neighborhood of a given location. The variability estimation is based on the assumption that the expect values of features can be approximated using the measurements associated to the neighborhood of a given location.
2. The iterative scheme adapts the contribution of each feature to the dissimilarity measure according to its variability when searching the candidate estimation of the user's position during the online positioning phase. These searched locations are subsequently used for refining the estimation of the user's location.
3. Presented results of the estimated STD value show that each feature has different variability throughout the RoI, i. e. the STD value is dependent both on the feature as well as on the location. The distribution of large errors (e. g. > 8 m) is similar to the pattern where high variance of the feature values occur.
4. The experimental results show that the percentage of locations whose positioning error is larger than 5 m is reduced from 10.2% to 2.6% (almost reduced by about three-fold), when compared to the performance of k NN without the weighted dissimilarity measure. The maximum positioning error is reduced by about 40% as compared to that of k NN with CDM.

1.4.2 Relation of the chapters

Although the major technical contributions (Chapter 3, 4, and 5) were developed over a period of 4 years, the publications used in this cumulative dissertation were published or submitted shortly before this thesis was presented.

Therefore, given the recent review of the literature included in each chapter, it was not considered necessary to include a separate literature review at this point. The reader is referred to the related work presented in the individual core chapters. The technical relation between the chapters is briefly sketched out in Figure 1.1 and summarized as follows:

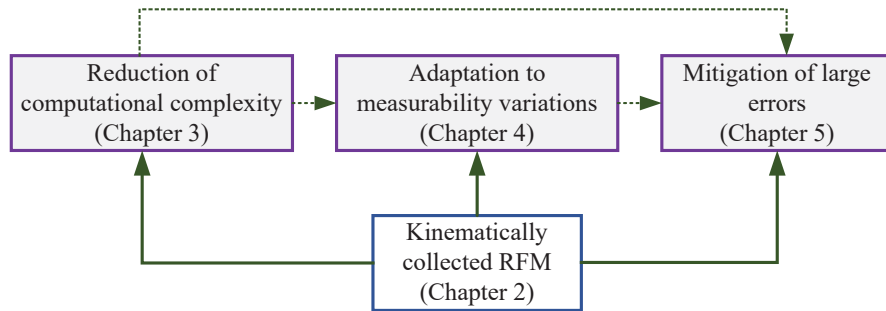


Figure 1.1: The relation between the chapters

- **Foundation:** Chapter 2 plays the role of providing the validation testbed to the major publications. The kinematically collected RFM and its continuous representation are discussed in this chapter.
- **Reduction of the computational complexity:** The main contribution of Chapter 3 is an approach for reducing the computational complexity during the online positioning phase. The proposed solution utilizes an MJI, which is a similarity measure defined using only the pairwise measurable features. The proposed approach has been employed in Chapter 4 and 5 for computing the weights of the CDM and for refining the user's location estimation according to the iteratively searched locations, respectively. Furthermore, the applicability of the kinematically collected RFM is verified in Chapter 3.
- **Adaptation to measurability variations:** The motivation of Chapter 4 is to adaptively deal with the measurability variations without inputting the values of arbitrarily missed signals and without requiring the prior knowledge of the total number of the measurable features. The exploitation of the MJI, used in Chapter 3, has shed light on combining vector-based distance metrics with set operations. The proposed CDM can take a variant number of relevant features as input for estimating the user's location. The non-vector way of computing the dissimilarity measure has been further employed in Chapter 5.
- **Mitigation of large errors:** The approach proposed in Chapter 5 builds on the relevant findings from Chapter 3 and 4. Based on the location-wise robustly estimated STD of each individual measurable SoP, it provides the potential of distinguishing large error cases occurred in

FIPS and of identifying out-dated signals largely deviated from the ones stored in the kinematically collected RFM. The latter could be used for adaptively updating the RFM.

1.5 Relevance to science and society

In a broad context, this PhD project has been conducted in an interdisciplinary way. It is an application of data-driven approaches to indoor positioning with the help of technologies commonly found in the field of engineering geodesy. The solutions have been developed by using solely the SoP for inferring the user's location and have aimed at mitigating the impact of variability issues, which jeopardize the practical applicability of FIPSs. This refers to three specific research goals: i) reducing the computational burden during the online positioning; ii) adaptively handling the measurability variations; and iii) mitigating large errors occurred in indoor positioning. The relevance of this thesis is discussed in two aspects:

Scientific contributions.

1. The kinematically collected RFM has been made publicly available². This dataset is referenced by a high precision positioning system while at the same time measuring the features using a mobile device. The kinematically collected RFM can be used for benchmarking comparison of different positioning algorithms.
2. The explorations made during the early phase of this PhD project have turned out valuable. On the one hand, four relevant papers, which focus on the topic of applying deep neural networks and graphSLAM to location estimation and RFM generation/reconstruction, have been published and have shown the potential of these techniques. On the other hand, the focus of the PhD project and technical path to take have been determined depending on the findings during the exploratory phase.
3. Each of the core chapters has presented a specific solution to each individual research question. These solutions can improve the practical applicability of the FIPS, especially when using SoP as the features in a large RoI. In addition, the proposed solutions are not only able to solve the specific research question that this thesis is looking into, but are also potentially applicable, either to other FIPSs, which employ different signals as the features or other cases, which encounter the difficulty of e. g. measurability variations.

²The dataset is accessible at <https://zenodo.org/record/2647508#.XNBr80gzZaQ>.

Promising practical applications. The solutions developed in this thesis will facilitate the actual implementation of FIPSs in large buildings (e.g., throughout a mall or covering an entire airport) without the need for special infrastructure or equipment, apart from the computers providing the service. This will open up commercial opportunities but also facilitate better user guidance and support. For example, positioning and trajectory tracking of the customers in big malls will (i) support the owners of the shops by providing user and location specific advertisement and information (e.g., make customers aware of special offers in other areas of the mall or dynamically place advertisement at highly frequented parts of the mall); (ii) assist the management of the mall in monitoring and to a certain degree controlling customer movement within the mall; and (iii) help the users to find desired items easier or to exit the mall quicker in an emergency situation.

CHAPTER 2

ON GENERATING REFERENCE FINGERPRINT

MAPS

The offline referencing phase, one of the core phases of a feature-based indoor positioning system (FIPS), is the key to deploying the positioning system. The reference fingerprint map (RFM) is acquired in order to establish the relationship between the location-relevant features and their corresponding locations during the offline stage. The generated RFM is later used for inferring the user's location during the online stage. In this chapter, an RFM kinematically collected using a mobile device with the help of a high precision tracking system is described in detail. Firstly, an overview on the composition of the data acquisition system is briefly introduced. Secondly, the details of the mobile-based RFM generation are presented in Section 2.2. Included in this subsection are the description of the position referencing system, of the acquisition of signals of opportunity (SoP) using the off-the-shelf sensors integrated in mobile devices, and of the synchronization operation. Thirdly, an analysis of the spatial distribution of the collected RFM and its continuous representation are discussed in Section 2.3.

2.1 Overview of the data acquisition system

The data acquisition system consists of three blocks, which are shown in the flow chart of (Figure 2.1) and defined as follows:

- **Position referencing system:** It is capable of kinematically tracking the

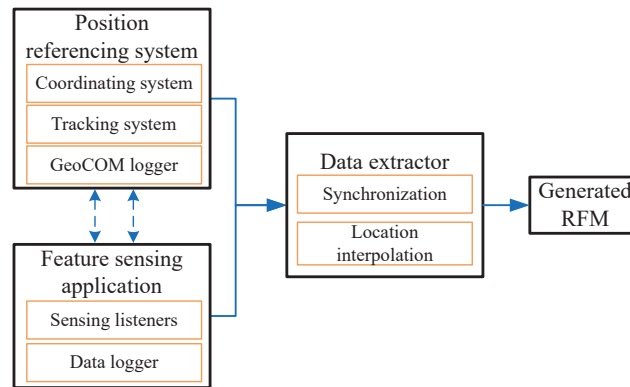


Figure 2.1: The schematic of the data acquisition system

surveyor (by targeting at the reflector mounted on top of a mobile device) with a high sampling rate (about 20 Hz) with the help of automatic target recognition (ATR) function of a total station. The total station is controlled by a software implemented in MATLAB using the application programming interface GeoCOM provided by Leica Geosystems. This software is able to automatically control the total station and log the location data (e. g. time stamp, coordinates, distance and angular values).

- Feature sensing application:** It is an application developed for the mobile device for the purpose of acquiring the feature values using the built-in sensors (e. g. WiFi/Bluetooth modules). To fulfill the requirement of the operating system (Android) installed on the mobile device, the application is implemented in Java and the data (e. g. signal values received from wireless local area network (WLAN) access points (APs)) acquisition has to be triggering-based, i. e. the value of features is passively measured when the signals (e. g. from WLAN APs or Bluetooth low energy (BLE) beacons) are detected by the built-sensors of the mobile device. This triggering-based data acquisition scheme leads to measurements of the features at arbitrary locations, since the signals arrive at arbitrary time.
- Data extractor:** The features and the locations (called reference locations), where the features are measured, are obtained using two different devices. The merging of the data is therefore necessary. An extractor is designed and implemented in MATLAB. The data extractor is used herein for synchronizing the time between the total station and the mobile device and for determining the reference location. The former is achieved by a cross-correlation-based time delay estimation (TDE). The latter is the benefit provided by the high sampling rate of the position tracking system. The detailed description of the tasks carried out using the data extractor is given in Section 2.2. The measured feature

values and their corresponding locations are collected as the raw RFM.

Compared to widely employed RFM collection schemes, such as statically measuring at each individual location or collecting via crowd-sourcing (Mendoza-Silva et al. 2018a; Huang and Manh 2016; Wu et al. 2018b), the implemented kinematic way of generating the RFM has two main advantages: i) the data is obtained at arbitrary locations with precisely referenced coordinates (cm-level); and ii) the kinematically collected features are more similar to the real-case scenario with moving pedestrians. These two advantages make the kinematically collected RFM more realistic as compared to the RFMs used in other of feature-based indoor positioning studies.

2.2 Mobile-based RFM generation

In this section, we describe the procedures of creating the RFM by tracking the mobile device using a total station. This procedure is able to simultaneously obtain the location-relevant features with the mobile application and the ground truth of the trajectory using the ATR function of the total station, which is controlled by the software via GeoCOM interface (Figure 2.2). The data collection consists of four steps: i) setup of the total station; ii) tracking of the mobile device with the ATR mode; iii) synchronization operation; and iv) acquisition of the fingerprinting data.

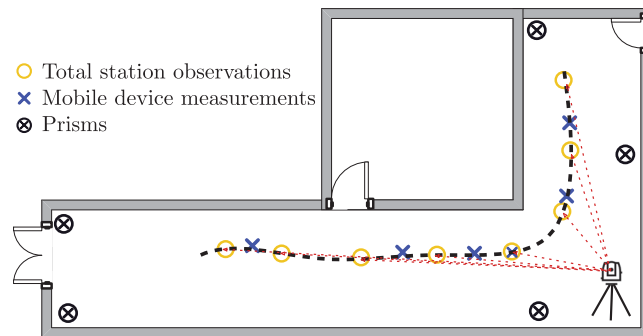


Figure 2.2: The schematic of the data acquisition process with the tracking of the total station

Setup of the total station. A total station measures angles and distances, which are then mapped on a two-dimensional planar surface. The required local coordinate system was established by arbitrarily selecting its origin and its orientation of the coordinate system within the region of interest (RoI). A set of 15 prisms (Leica GPR1 round prism) has been mounted on the ceiling and is distributed along the floor of the RoI (Figure 2.3). The total station was used

to measure the prism locations and establish the reference network, which is employed to determine the location of the total station via free-stationing.

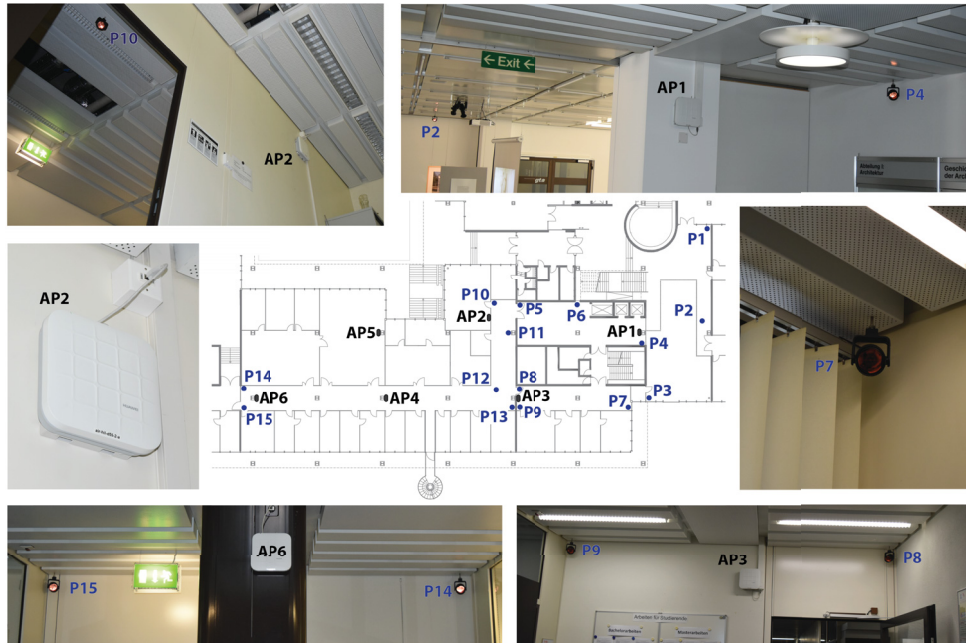


Figure 2.3: The configuration of the FIPS

Figure 2.3 depicts the the configuration of the FIPS. In the center of the figure is shown, a floor plan of the RoI, including the location of the reference prisms and several measurable WLAN APs. These WLAN APs are deployed by the IT department for providing Internet access and are opportunistically measurable. Each individual AP transmits the signals at the band of 2.4 and 5.0 GHz and has a built-in BLE beacon. For a visual impression, a photo documentation around the floor plan shows the system setup.

Tracking of the mobile device with ATR mode. A 360° mini-prism (Leica GRZ101) is mounted on top of the mobile device to be tracked with a total station. This tracking procedure is automatically performed via the GeoCOM interface and the software implemented in MATLAB¹. This GeoCOM logger can i) set and search the target mini-prism; ii) lock on the target and activate ATR mode; and iii) continuously determines the coordinates of the target (Figure 2.4). Instead of only logging the coordinates, a list of full measurements, including the distance, zenith angle, horizontal angle, and time stamp (from the internal clock of the total station) is obtained. These measurements are used both for computing the coordinates and for synchronizing the time

¹An implementation of GeoCOM-based real time tracking in Python is available at <https://github.com/rvermeiren/Leica-GeoCOM-for-drone-tracking/tree/master/scripts/src>.

between the total station and the mobile device. Herein the tracking function is carried out by a Leica MS50, which is able to acquire approximate 20 measurements per second (i. e. 20 Hz).

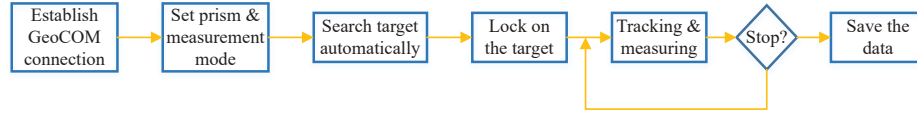


Figure 2.4: The flowchart of the logging software implemented using GeoCOM

Synchronization operation. When measuring the signal strength values of the WLAN signals with the mobile phone, it is simple to obtain the coordinated universal time (UTC) of acquisition via online web services as well. On the other hand, it is more difficult to obtain the UTC time stamp for the measurements performed by the total station. The time stamp returned from the total station via GeoCOM is a relative time stamp reporting the elapsed time in milliseconds after the power on of the total station. The UTC time stamp of the total station could be obtained using a dedicated global navigation satellite system (GNSS) receiver. However, such a solution is not applicable in indoor environments. In order to combine the observations from the total station and the mobile device, a common time frame has to be established. This synchronization operation is achieved by cross-correlation-based initial TDE before acquiring the fingerprinting data. Herein the time drifts of the internal clocks of both devices have not been taken into account because the duration of each round (i. e. one continuously tracked trajectory) of the data collection is less than 20 minutes. The internal clock drift of a typical Leica total station (e. g. MS50) within one hour is about 200 ms (Gojcic et al. 2017). The time drifts of the internal clocks in such a short time thus have little impact on the common time frame.

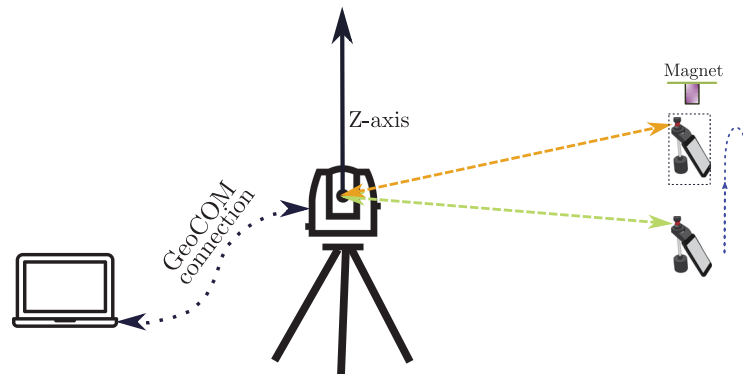
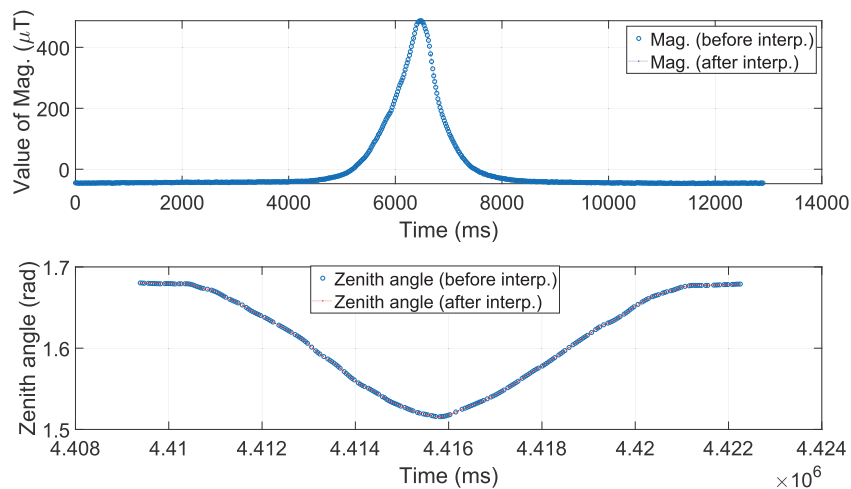


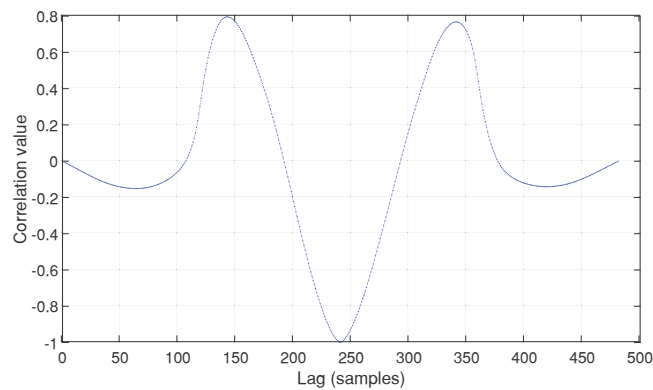
Figure 2.5: The schematic of the synchronization operation

Motivated by Gojcic et al. (2017), the TDE is determined by minimizing

the cross-correlation function between two simultaneously measured time series: i) the zenith angle measurement obtained using the total station; and ii) the value of magnetic field measured by the mobile device. As depicted in Figure 2.5, an artificial signal is generated by manually moving the mobile device up and down while being tracked by the total station. When moving the mobile device up (closer to the magnet), the zenith angle reduces while the value of the magnetic field (denoted short as Mag.) acquired by the built-in magnetometer of the mobile phone increases (Figure 2.6a). The changes of the zenith angle and magnetic field values are vice versa when moving the mobile device down.



(a) The measurements obtained by the magnetometer (top one) and the total station (bottom one) used for synchronization.



(b) Cross correlation

Figure 2.6: Example of cross-correlation-based TDE

Figure 2.6 shows an example of the TDE obtained as the lag, which minimizes the cross-correlation function of the data from the magnetometer built into the mobile device and the zenith angle obtained by the total station. Note that the highest achievable resolution of the cross-correlation-based TDE is equal to the sampling interval (the longer one of the two devices).

Therefore, the cross-correlation function is subsample interpolated using the parabola-based method for approximating the lags, at which the minimum cross-correlation value is achieved (Figure 2.6b). Considering the sampling rate of the magnetometer (50 Hz) and the measuring rate of the Leica MS50 (20 Hz), the highest achievable time resolution is about 25 ms. This time resolution is adequate for tracking the pedestrian with low mobility (e. g. walking about 1.5 m/s) in indoor environments.

Acquisition of the fingerprinting data. The fingerprinting application is developed in Java using the provided programming interfaces by Android, especially the ones for network management, sensors and location providers². The application can be used to collect the received signal strength (RSS) from WLAN APs/BLE beacons and the values from magnetometer and IMUs (e. g. accelerometer, gyroscope) using the built-in sensors of mobile devices. In Android, sampling of the features is triggered by a specific event (e. g. one new signal has been detected). It means that the measured data of a given sensor is logged/sampled by actively or passively triggering the corresponding event. One relevant tip for fingerprinting using mobile device is the passive triggering mode of WiFi/Bluetooth modules. This in turn leads to that the measuring interval of these two modules is arbitrarily determined by the sampling event, which is triggered by either the detection of new signals sources or the large change of the signal values (Schulz et al. 2018). The typical sampling rate of WiFi/Bluetooth modules built-in the Nexus 6P is about 0.67 Hz. Owing to the low sampling rate, the reference location can be determined by interpolating the coordinates, which can be obtained by the total station with a high data acquisition frequency. The detailed description of the design of the fingerprinting application can be found in e. g. Duewell (2019).

Data extraction. All pairs of measured data both from the total station and the mobile device are used to generate the raw RFM by interpolating the locations where the features are measured according to the tracked trajectory after the time synchronization. The signals from WLAN APs and BLE beacons have been measured and extracted for generating the raw RFM. The data collection has been carried out twice in the same office building. The time interval between two rounds of data collection is about two months. The characteristics of the collected RFM are summarized in Table 2.1. For the RSS from WLAN APs, we have filtered the mobile WiFi beacons according to the service set identifier (i. e. the name) of the APs. All BLE beacons are stored in the raw RFM since the information for filtering the mobile ones are not available. This also explains that a large number of newly measured BLE beacons have been observed in dataset *HIL-R2* as compared to dataset *HIL-*

²Details about the programming interfaces of Android is available on <https://developer.android.com/reference/packages>.

R1. The collected dataset has been made publically available such that other users can perform benchmarking comparison with different algorithms (Zhou 2019). In the analysis of this chapter, we focus on analyzing the raw RFM, which contains the RSS from WLAN APs.

Table 2.1: The summary of the raw RFM (WLAN and BLE). The dataset *HIL-Merged* is obtained by fusing dataset *HIL-R1* and *HIL-R2*.

| | WLAN | | BLE | |
|-------------------|--------------|---------------|--------------|---------------|
| | # of samples | # of features | # of samples | # of features |
| <i>HIL-R1</i> | 3911 | 490 | 3380 | 278 |
| <i>HIL-R2</i> | 3714 | 434 | 2898 | 408 |
| <i>HIL-Merged</i> | 7625 | 537 | 6278 | 653 |

2.3 A world model of the RFM

In this section, the spatial distribution of the measured features contained in the raw RFM is analyzed. The visualizations of them clearly suggests that a filtering procedure is required because the raw measurements are quite noisy (see Figure 2.7). In addition, a modified version of kernel smoothing (KS) is introduced both for improving the spatial consistency and for providing the continuous presentation of the RFM in the subsequent sections. The modified version of the KS is capable of reducing the computational complexity and mitigating over-smoothing problems of the traditional KS.

2.3.1 Spatial distribution of the raw RFM

Figure 2.7 shows the measured feature values of two APs. The characteristics of the raw RFM are: i) the raw measurements are obtained at arbitrary locations and their density varies over space. The density variations are caused by the inaccessibility (e. g. areas blocked by furniture or other objects) of the RoI; and ii) the measured values in the neighborhood of a fixed location have large variations and are quite noisy. These two characteristics are the motivation of proposing a world model of the RFM, which is able to continuously represent the RFM.

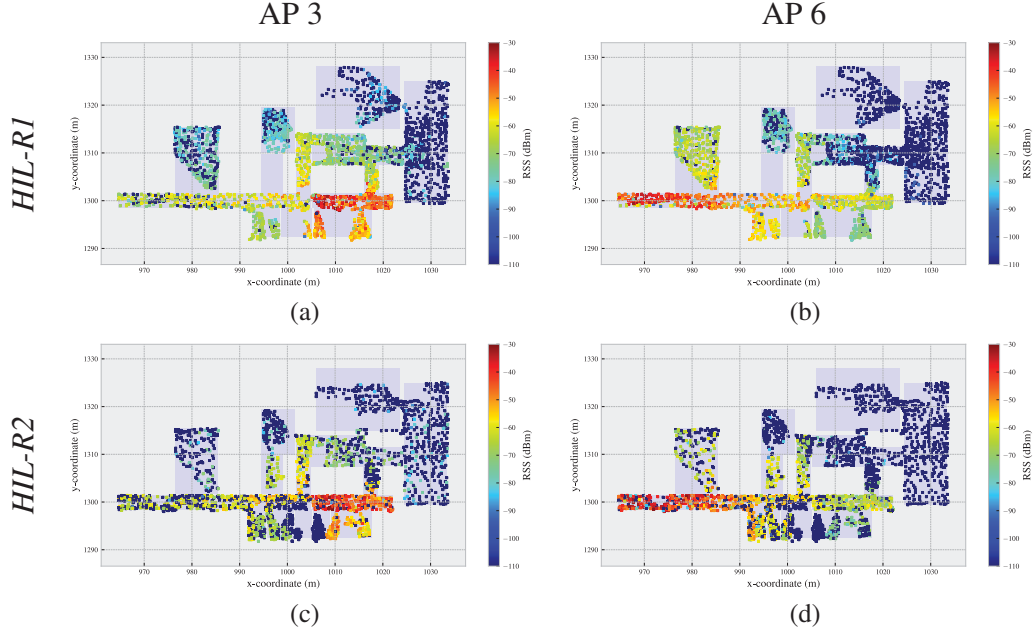


Figure 2.7: Examples of the raw RFM of two APs

2.3.2 Continuous representation of the RFM

Albeit any of interpolation approaches (e. g. splines) can be used to represent the RFM, KS is selected as the world model of the RFM for two reasons: i) its capability to filter out the noise; and ii) its flexibility in utilizing diverse spatial kernels (Berlinet and Thomas-Agnan 2011). More specifically, a world model \mathcal{F} of the RFM is used to retrieve the measurement \mathbb{O} at any location \mathbf{l} in the RoI³, i. e. $\mathbb{O} := \mathcal{F}(\mathbf{l})$.

Fundamentals of KS. KS can be formulated as an optimization problem in reproducing kernel Hilbert space (RKHS) \mathcal{H}^K :

$$\mathcal{F}^* = \operatorname{argmin}_{\mathcal{F} \in \mathcal{H}^K} \sum_{i=1}^{|\mathbb{F}|} d(\mathcal{F}(\mathbf{l}_i), \mathbb{O}_i) + \sigma^2 \|\mathcal{F}\|_{\mathcal{H}^K}^2 \quad (2.1)$$

where d is a dissimilarity metric, i. e. $d : \mathbb{O} \times \mathbb{O} \mapsto \mathbb{R}$ and σ^2 is the regularization parameter, which are used for quantifying the difference between each pair of measurements in the feature space and for stabilizing the optimization against noise, respectively (Murphy 2012). \mathbb{F} is the raw RFM collected at discrete locations throughout the RoI and consists of reference coordinates \mathbf{l}_i and its associated measurement \mathbb{O}_i . The solution of the above problem (treated as

³A measurement is formulated as pairs of the unique identifiable attribute and its associated values. A detailed definition and description can be found in the Section 3.3 in Chapter 3.

the world model of the RFM) is:

$$\mathcal{F}^*(\mathbf{l}) = \mathbf{z}_1'(\mathbf{G}'\mathbf{G} + \sigma^2\mathbf{I}_{|\mathbb{F}|})^{-1}\mathbf{o} \quad (2.2)$$

where $\mathbf{I}_{|\mathbb{F}|}$ is the identity matrix of size $|\mathbb{F}|$ and $\mathbf{o} \in \mathbb{R}^{|\mathbb{F}|}$ ($\mathbf{o}_i = \mathbb{O}_i, i \in \{1, 2, \dots, |\mathbb{F}|\}$) is the column-wise representation of the measurements stored in the discrete RFM \mathbb{F} . $\mathbf{z}_1 \in \mathbb{R}^{|\mathbb{F}|}$ is a column vector, which denotes the correlation between the query point \mathbf{l} and the reference locations stored in the RFM \mathbb{F} :

$$\mathbf{z}_1 = [\mathcal{K}(\mathbf{l}, \mathbf{l}_1); \mathcal{K}(\mathbf{l}, \mathbf{l}_2); \dots; \mathcal{K}(\mathbf{l}, \mathbf{l}_{|\mathbb{F}|})] \quad (2.3)$$

where $\mathcal{K}(\cdot, \cdot)$ is the kernel function (e. g. Gaussian or Matérn kernels). \mathbf{G} is the Gram-matrix of the selected kernel $\mathcal{K}(\cdot, \cdot)$. The kernel function is employed to approximate the variance-covariance matrix between the reference locations when used for interpolating the feature values. More details about KS can be found in e. g. Berlinet and Thomas-Agnan (2011).

Figure 2.8 shows the results of the smoothed RFM using the traditional KS⁴. Herein the Matérn kernel is applied as the spatial kernel and its parameters: i) the length scale; and ii) the differential factor are set to 1.0 according to the density of the reference locations contained in the raw RFM and 1.5 for reducing the computational complexity, respectively (Pedregosa et al. 2011; Murphy 2012). Compared to the raw RFM visualized in Figure 2.7, the traditional KS can filter out the noise to a certain degree and thus improving the spatial consistency of the feature values. However, two flaws of the traditional KS are exposed in Figure 2.8. One is the limited capability of filtering out the noise. The KS cannot smoothen out the high intensity noises contained in the raw RFM. This limitation leads to the low spatial consistency of feature values in the regions where the associated raw measurements have high noise (see Figure 2.8c and 2.8d). Another flaw is that the KS may smoothen the feature values over the geometric boundaries (e. g. walls or corners) of indoor regions (called over-smoothing problem). The former indicates that the spatial consistency can be improved further. The latter degrades the discontinuity of the RFM yielded by the indoor environments themselves. Indeed, the discontinuity related to indoor environment is relevant for feature-based positioning (Bong and Kim 2012). These flaws are addressed by introducing spatial filtering and geometric constraints in Section 2.3.3.

Computational complexity of KS. As the solution shown in Equation (2.2), the computational complexity of the KS (mainly from the matrix inversion) is $\mathcal{O}(|\mathbb{F}|^3)$ ($\mathcal{O}(|\mathbb{F}|^{2.373})$ using optimized algorithms) (Cormen et al. 2009) and $\mathcal{O}(|\mathbb{F}|^2)$ with respect to the time and storage, respectively. The time cost

⁴Though KS can continuously represent the RFM, only the smoothed measurements at the locations contained in the raw RFM are visualized for easy comparison.

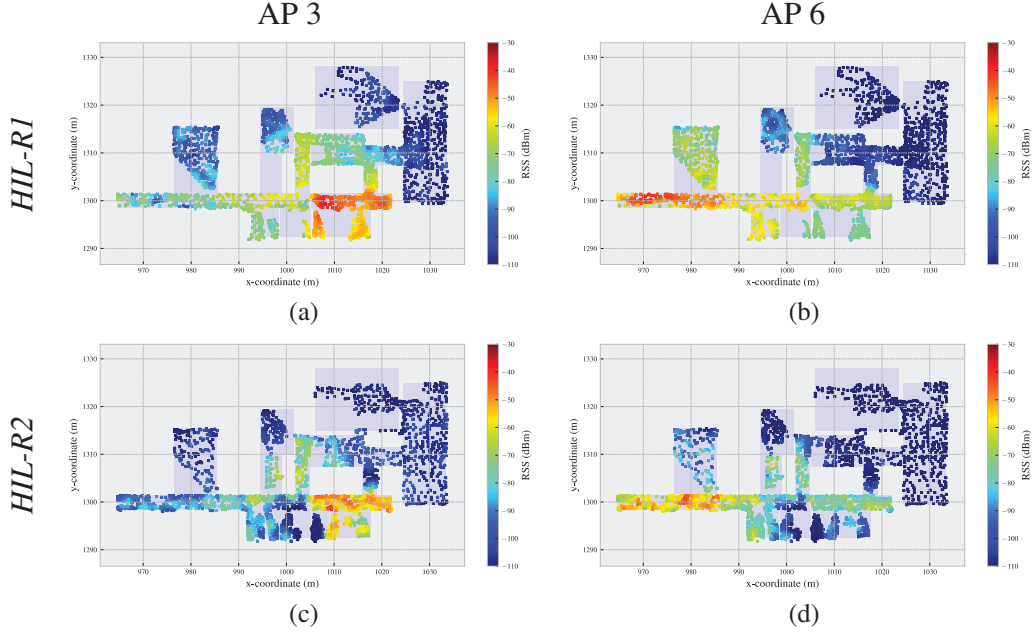


Figure 2.8: Examples of the kernel smoothed RFM taking the raw RFM as the input to the traditional KS

increases almost cubically regarding the number of measurements contained in the RFM. In fact, the online execution of KS for a big RFM (i. e. the area of the RoI) can be very computationally expensive.

KS using the neighborhood support set. In order to reduce the computational complexity of KS to a fixed and an adaptable level, especially for the online applications, we apply a variant to KS. Instead of using all measurements stored in the RFM for solving the KS problem, only a subset of the RFM in the neighborhood of a given location \mathbf{l} is used for smoothing and interpolating the measurements at that location. The neighborhood support set includes only the observed measurements associated to the locations that are within the circle with the radius of r^{KS} centered at the location. We evaluate the impact of the querying radius r^{KS} on the computational complexity with respect to the processing time and storage space, as well as the interpolation performance (defined as mean absolute error (MAE) of the residuals) by calculating the residuals between the estimated values only using queried RFM and the ones using the whole RFM.

As can be seen in Figure 2.9, the MAE of the interpolation value using KS decreases by about 400 times as the value of k^{KS} increases from 0.5 m to 8 m. Meanwhile, the processing time and storage cost increase by about 1000 times. In order to balance the value of residual and the computational complexity, we find out that the residual has no apparent reduction and the overall residual is low (Figure 2.9a) when r^{KS} is larger than 5 m. At that radius value, the process time of each KS query is around tens of milliseconds.

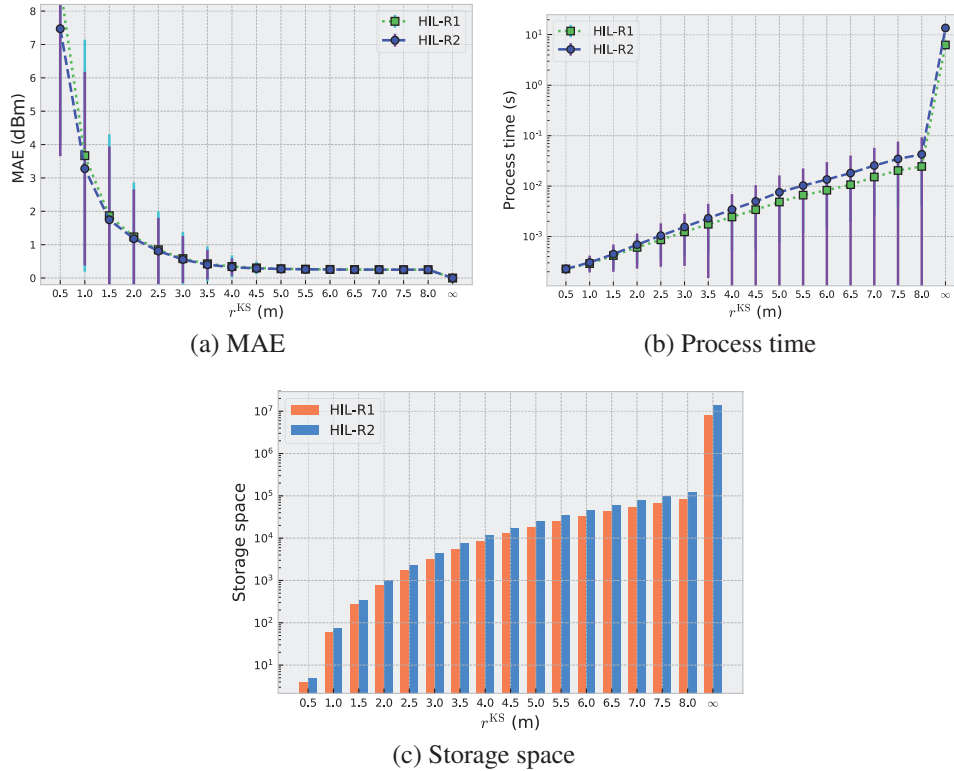


Figure 2.9: The influence of the value of r^{KS} on processing time, storage space and residuals. The value of $k = \infty$ means that using all the training examples for KS.

The smoothed RFM yielded from the modified KS is similar to that of the traditional KS by comparing Figure 2.8 and 2.10. The modified KS is more sensitive to the large variations occurred in the RFM, because it takes only the neighborhood of a given locations for computing the interpolated features values. In addition, the KS using the neighborhood support set can partially alleviate the over-smoothing problem.

2.3.3 Enhancement of the traditional KS

Spatial filtering. The spatial filtering is employed to the raw measurements in order to mitigate the impact of potential outliers on the KS. It in turn improves the robustness of the KS. A spatial median filter is applied to approximate the expected values of features according to the measurement, which are associated to the neighborhood of the given location (Zhou and Wieser 2019a). Following Zhou and Wieser (2019a), the support set is defined as the measurements collected at the up to 20 locations closest to a given location that at the same time lie within a 2 m radius about the given location. The spatially filtered RFM visualized in Figure 2.11 has much less variations

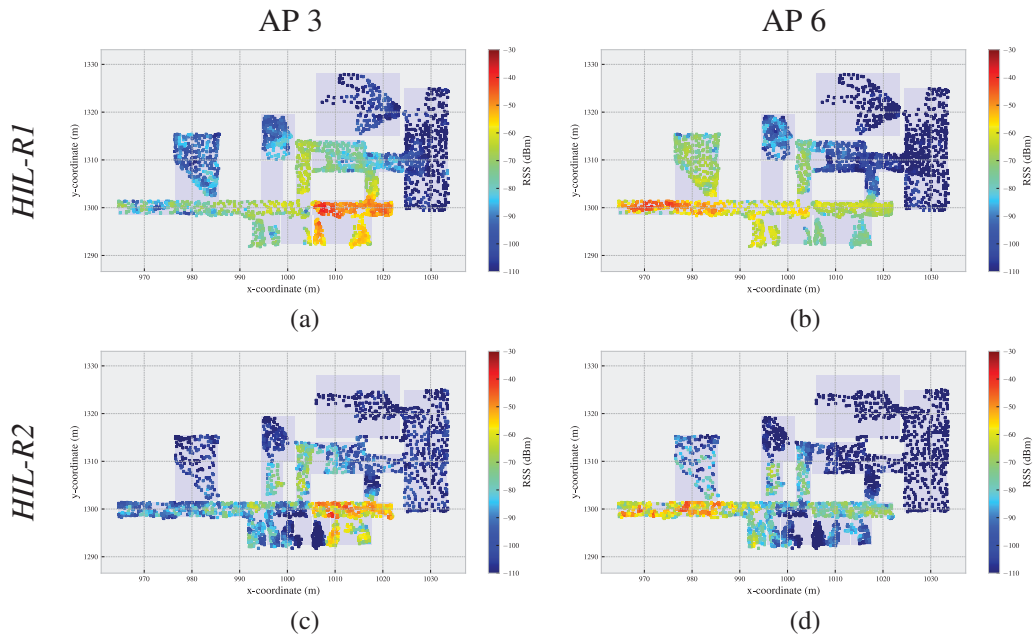


Figure 2.10: Examples of the smoothed RFM yielded by the modified KS using the raw RFM as the input

as compared to the raw measurements depicted in Figure 2.7. Owing to the higher spatial consistency of the spatially filtered RFM, the output from the modified KS also maintains high spatial consistency (see Figure 2.12).

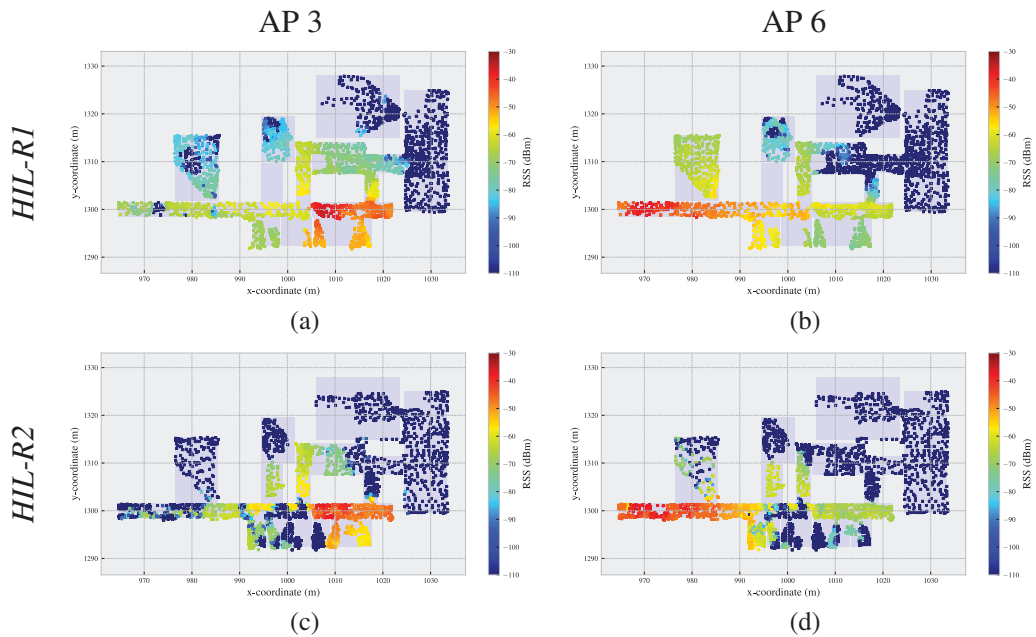


Figure 2.11: Examples of the spatially filtered RFM

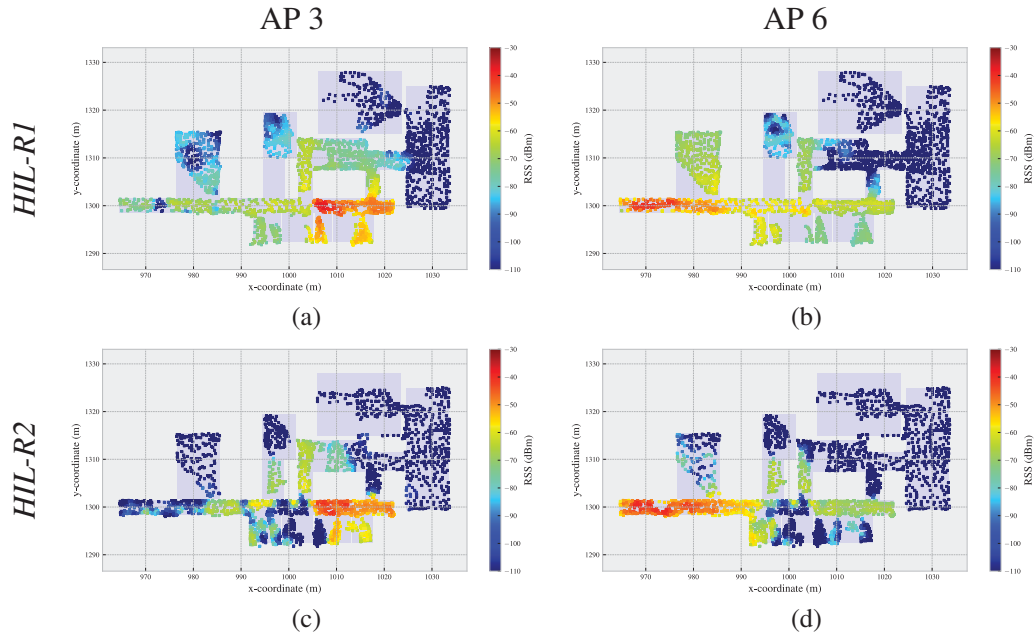


Figure 2.12: Examples of the smoothed RFM using spatially filtered RFM as the input to the modified KS

Geometric constraints. The over-smoothing, described in Section 2.3.2, is caused by the modeling problem of the traditional KS. In the application of modeling the RFM, the geometric constraints (such as walls) should be taken into account while computing the spatial contribution of each individual measurement. In order to introduce the geometric information into the KS, the vector map that abstractly represents the geographical features of the RoI into vectors (e. g. polygons) is required. Herein the geometric information of the RoI is roughly represented as a set of polygons according to the floor plan of the building. The impact of the geometric constraints on the KS is modeled as a part of the distance between each pair of locations. In this way, the spatial variance-covariance matrices approximated using a kernel function are affected by the geometric constraints. The physical distance between any pair of reference locations is exponentially scaled up according to the intersection between the line linking the two locations and the boundaries of the indoor regions. The scaled distance reduces the correlations between the locations, which are separated by the indoor structures when approximating the variance-covariance matrix using the kernel function.

More specifically, the support set $\mathcal{S}_q^{\text{KS}}$ at the query point \mathbf{l}_q consists of the locations, which lie in the circle with the radius of r^{KS} centered at that location (see Figure 2.13). Let d_{ij} and c_{ij} denote the physical distance between the location \mathbf{l}_i and \mathbf{l}_j and the number of intersections between the boundaries and the line linking these two locations, respectively. The exponentially scaled

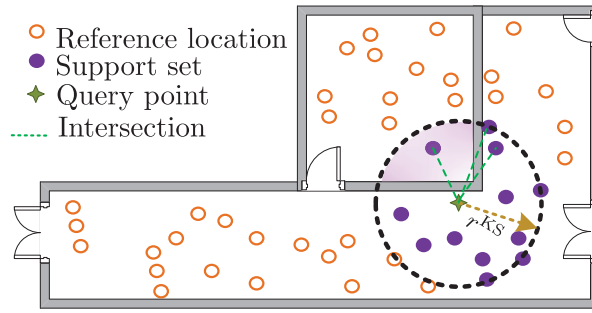


Figure 2.13: The schematic of the modeling of the geometric constraints

distance d_{ij}^{exp} is computed as:

$$d_{ij}^{\text{exp}} = (1 + d_{ij})^{\exp(\lambda c_{ij})} - 1 \quad (2.4)$$

where $\lambda > 0$ is the scale factor for controlling the enlarge extent of the distance and is set to 1 herein. The exponential way of scaling the distance is inspired by the fact that the intensity of radio frequency (i. e. electromagnetic waves) exponentially decays when propagating through space. In addition, Equation (2.4) ensures that the scaled distance is equal to the physical distance when c_{ij} is 0, i. e. no obstacle exists between the locations. Figure 2.14 shows two examples of the smoothed RFM using the modified KS with the combination of geometric constraints. The over-smoothing pattern becomes much less apparent with the help of the geometric constraints as compared to Figure 2.12, especially when encountering the boundaries between different rooms.

2.4 Conclusion

A data acquisition system with the combination of high precision tracking of the total station and off-the-shelf mobile devices with built-in sensors has been employed to generate the RFM. The configuration of the referencing system, the time synchronization, and the sensing scheme of the mobile device have been described in detail. The reference locations associated to the measurements stored RFM are in the cm-level accuracy while their associated measured features are observed kinematically. These two characteristics make this RFM a unique contribution to the indoor positioning research community. In addition, the spatial distribution of the collected RFM has been analyzed and its world model have been proposed by applying spatial filtering and the modified KS. The modified KS is capable of providing a continuous representation of the RFM while reducing the computational complexity and

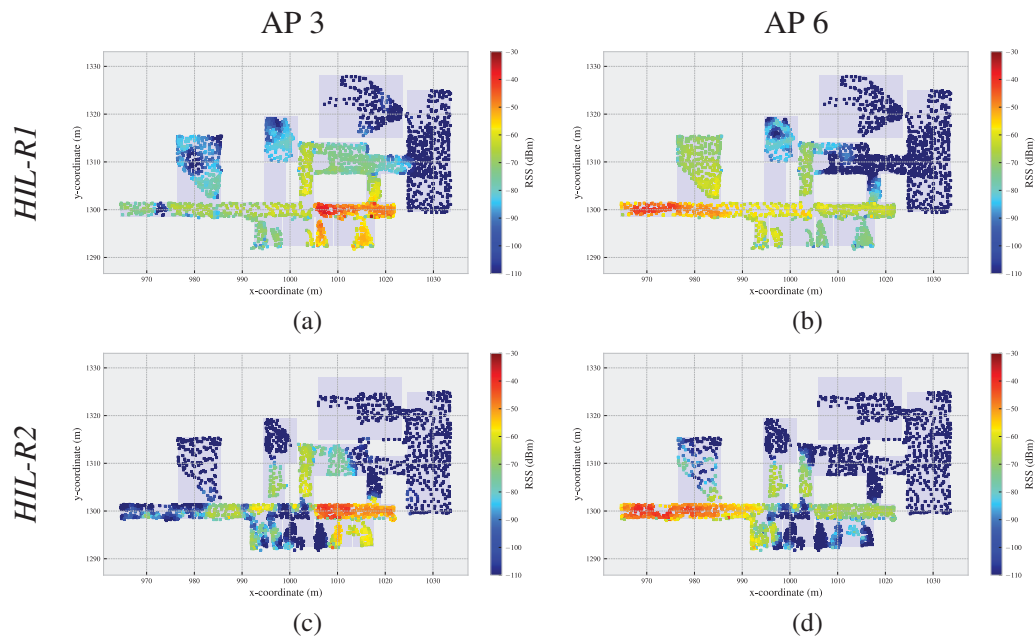


Figure 2.14: Examples of the smoothed RFM using spatially filtered RFM as the input to the modified KS with the geometric constraints

mitigating the over-smoothing problems of the traditional KS.

CHAPTER 3

ON REDUCING COMPUTATIONAL COMPLEXITY

This chapter corresponds to:

Zhou, Caifa, and Andreas Wieser. 2019. Modified Jaccard Index Analysis and Adaptive Feature Selection for Location Fingerprinting with Limited Computational Complexity. Journal of Location Based Services 13 (2). Taylor & Francis:128–157. <https://doi.org/10.1080/17489725.2019.1577505>.

except for formatting and the relocation of the bibliography to the end of the thesis. This is the Author Version of the paper. For the Typeset Version please refer to the original journal paper.

Modified Jaccard Index Analysis and Adaptive Feature Selection for Location Fingerprinting with Limited Computational Complexity

Caifa Zhou, Andreas Wieser

Abstract

We propose an approach for fingerprinting-based positioning which reduces the data requirements and computational complexity of the online positioning stage. It is based on a segmentation of the entire region of interest into subregions, identification of candidate subregions during the online-stage, and position estimation using a preselected subset of relevant features. The subregion selection uses a modified Jaccard index which quantifies the similarity between the features observed by the user and those available within the reference fingerprint map. The adaptive feature selection is achieved using an adaptive forward-backward greedy search which determines a subset of features for each subregion, relevant with respect to a given fingerprinting-based positioning method. In an empirical study using signals of opportunity for fingerprinting the proposed subregion and feature selection reduce the processing time during the online-stage by a factor of about 10 while the positioning accuracy does not deteriorate significantly. In fact, in one of the two study cases the 90th percentile of the circular error increased by 7.5% while in the other study case we even found a reduction of the corresponding circular error by 30%.

3.1 Introduction

Fingerprinting-based indoor positioning systems (FIPSs) are attractive for providing the location of users or mobile assets because they can exploit signals of opportunity (SoP) and infrastructure already existing for other purposes (He and Chan 2016). They require no or little extra hardware, (He and Chan 2016), and differ in that respect from many other approaches to indoor positioning like e.g., the ones using infrared beacons (Lee et al. 2004), ultrasonic signals (Hazas and Hopper 2006), Bluetooth low energy (BLE) bea-

cons (Kalbandhe and Patil 2016), radio frequency identification (RFID) tags (Bekkali et al. 2007), ultra-wideband (UWB) signals (Ingram et al. 2004), or foot-mounted inertial measurement units (IMUs) (Gu et al. 2017). FIPS benefit from the spatial variability of a wide variety of observable features or signals like received signal strength (RSS) from wireless local area network (WLAN) access points (APs) (Padmanabhan et al. 2000; Youssef and Agrawala 2008; Torres-Sospedra et al. 2014; Jun et al. 2017; Mendoza-Silva et al. 2018a), magnetic field strengths (Saxena and Zawodniok 2014; Torres-Sospedra et al. 2015b; Xie et al. 2016), or RSS of cellular towers (Driusso et al. 2016). Such signals are location dependent features, many of which can easily be measured using a variety of mobile devices (e. g. smartphones or tablets). FIPSs are therefore also called feature-based indoor positioning systems (Kasprzak et al. 2013). The attainable quality of the position estimation using FIPS mainly depends on the spatial gradient of the features and on their stability or predictability over time (Niedermayr et al. 2014).

Key challenges of FIPS, especially the ones using RSS readings from WLAN APs are discussed e.g. in (Kushki et al. 2007) and more recently in (He and Chan 2016; Yassin et al. 2017). The former publication focuses on four challenges of FIPS utilizing vectors of RSS from WLAN AP as features. In particular, the paper addresses i) the generation of a fingerprint database to provide a reference fingerprint map (RFM) for positioning, ii) pre-processing of fingerprints for reducing computational complexity and enhancing accuracy, iii) selection of APs for positioning, and iv) estimation of the distance between a fingerprint measured by the user and the fingerprints represented within in the reference database. Extensions to large indoor regions and handling of variations of observable features caused by the changes of indoor environments or signal sources of the features (e.g., replacement of broken APs) are addressed in (He and Chan 2016).

Regarding generation of the RFM, various approaches have been proposed (He and Chan 2016). Initially, the features were mapped by dedicated surveying measurements (e. g. (Youssef and Agrawala 2008; Park et al. 2010)). The resulting RFMs were accurate at the time of acquisition. However, such measurements are labor-intensive and time-consuming. Approaches based on forward modeling, e.g., using indoor propagation models (Jung et al. 2011; El-Kafrawy et al. 2010; Bisio et al. 2014) or ray tracing (Renaudin et al. 2018), were proposed as a cost-effective alternative, especially in case of radio frequency signal strengths as features. However, the accuracy of the resulting RFM depends on the validity of the model assumptions including the wave propagation models, the geometry and material properties of the objects and structures in the indoor space, and the location and antenna gain patterns of the signal sources. For many real-world application scenarios it is thus typically lower than the accuracy of measurement-based RFM generation. More recently, approaches for RFM generation using the sensors built into the mobile user devices have been proposed. They can be differentiated according

to the degree of user participation. Data collection for RFM generation can be done by an application running in the background such that the user only needs to consent to contributing data but not actively participate otherwise. Other approaches require the user to manually indicate his/her location on a map or to signal to the application that the current location corresponds to a certain marked ground truth, see e.g. (Wu et al. 2013; Ledlie et al. 2011; Li et al. 2013). For refining the RFM various approaches have been proposed, e.g. interpolation/extrapolation (Talvitie et al. 2015; Sorour et al. 2015) or kernel-smoothing (Huang and Manh 2016). We are not focusing on reducing the workload for building the RFM. However, we validate the proposed approach using a kinematically collected RFM (see Section 3.4) which is similar to a standard survey but less time-consuming.

The contribution of this paper is to reduce the online positioning computation complexity by introducing a specific approach to subregion selection and feature selection into the online positioning process. The proposed approach includes machine learning algorithms, i.e. algorithms whose performance on the specific task improves with experience (Mitchell 1997) and which are thus suitable for fingerprinting-based positioning (Bishop 2006). It is applicable to FIPSS using opportunistically measured location-relevant features. In case of FIPSS using SoP, the difficulty lies in both the type and number of features varying across the region of interest. This introduces a critical limitation which prevents the applicability of the aforementioned FIPS solutions due to changes in the dimension of the feature space across time and across location coordinate space. For instance, the fingerprinting-based positioning methods including the typical ones, e.g. k -nearest neighbors (k NN) (Padmanabhan et al. 2000) and maximum a posteriori (MAP) (Youssef and Agrawala 2008), and the advanced ones, e.g. fingerprinting-based positioning using support vector machine (SVM) (Wu et al. 2004), linear discriminant analysis (LDA) (Nuño-Barrau and Páez-Borrillo 2006), Bayesian network (Nandakumar et al. 2012), and Gaussian process (Ouyang et al. 2012), cannot be applied without special precautions for handling the varying dimension in such cases. Few previous publications address handling this problem e.g. (Zhou and Wieser 2018a). Additionally, the computational complexity of these fingerprinting-based positioning methods is proportional to the number of reference locations in the RFM and the number of observable features. This makes these approaches computationally expensive in large RoIs with many reference locations and many available features unless introducing specific means for mitigation. The discrepancy of the feature dimension and the computational complexity problems are typically mitigated by introducing subregion selection and feature selection (Zhou and Wieser 2018b; Kushki et al. 2007; Feng et al. 2012; Ouyang et al. 2012; Khalajmehrabi et al. 2017).

In this paper we propose (i) subregion selection based on a modified Jaccard index (MJI), (ii) a forward-greedy search to find an appropriate number of subregions, and (iii) an adaptive forward-backward greedy search

(AFBGS) algorithm (Zhang 2011) for selecting the relevant features for each subregion. We demonstrate the application of the proposed algorithms to both MAP- and kNN-based position estimation. We finally validate the performance of the proposed approach by carrying out experiments in two RoIs of different size using two types of opportunistically measured signals (i.e., WLAN and BLE). The selected acronyms are listed in Table 3.1.

The structure of the paper is as follows: A short review of the previous publications addressing the approaches for subregion selection and feature selection is given in Section 3.2. In Section 3.3, the MJI-based subregion selection, AFBGS-based feature selection, and the modified online positioning process are presented along with the computational complexity. We illustrate and validate the performance of the proposed approach in Section 3.4 by applying it to an FIPS, which utilizes SoP as the location-relevant features, and we compare the results to those obtained using previously proposed methods.

Table 3.1: Selected acronyms used herein

| Acronym | Meaning |
|---------|-------------------------------------------------|
| FIPS | fingerprinting-based indoor positioning system |
| RSS | received signal strength |
| SoP | signals of opportunity |
| RFM | reference fingerprint map |
| k NN | k -nearest-neighbors |
| MAP | maximum a posteriori |
| MJI | modified Jaccard index |
| AFBGS | adaptive forward-backward greedy search |
| LASSO | least absolute shrinkage and selection operator |
| MSE | mean squared error |
| ECDF | empirical cumulative distribution function |

3.2 Related work

3.2.1 Subregion selection

The subregion selection process contributes to constraining the search space. The selected subregions are treated as coarse approximations of the user's location. The process of refining the coordinate estimates is then carried out only within these subregions, and the search space for the final estimate is thus independent of the size of the RoI. There are mainly two types of ap-

approaches for subregion selection¹: approaches based on clustering and approaches based on similarity metrics. (Feng et al. 2012; Karegar 2017), and (Chen et al. 2006; Ouyang et al. 2012) applied affinity propagation and k -means clustering to divide the RoI into a given number of subregions according to the features collected within the RoI. Both papers present clustering-based subregion selection, and require prior definition of the desired number of subregions and knowledge of all features observable within the entire RoI. These clustering-based approaches take the fingerprint measured by the user into account during the clustering process which thus has to be repeated with each new user fingerprint obtained.

Similarity metric-based subregion selection relies on the identification of the subregions whose fingerprints contained in the RFM are most similar to the fingerprint observed by the user. They differ depending on the chosen similarity metric. E.g., (Kushki et al. 2007) use the Hamming distance for this purpose, measuring only the difference in terms of observability of the features, not their actual values. Still, these approaches typically need prior information on all observable features within the entire RoI when associating a user observed features with a subregion. This may be a severe limitation in case of a large RoI or changes of availability of the features. MJI-based subregion selection as proposed in this paper belongs to similarity metric-based subregion selection. However, the approach proposed herein requires only the prior knowledge of the features observable within each subregion when quantifying the similarity metric between the observations in the RFM and in the user observed measurements.

3.2.2 Selection of relevant features

Approaches to the selection of features (or sparse representation) actually used for positioning differ w.r.t. several perspectives. We focus on three aspects in their review: i) whether they take the relationship between positioning accuracy and selected features into account, ii) whether they help to reduce the computational complexity of position estimation, and iii) whether they are applicable to a variety of features or only features of a certain type. The chosen features for positioning should be the ones allowing to achieve the best positioning accuracy using the specific fingerprinting-based positioning method or achieving a useful compromise between accuracy and reduced computational burden.

Previous publications focused on feature selection for FIPS using RSS from WLAN APs and consequently addressed the specific problem of AP selection rather than the more general feature selection. (Chen et al. 2006) and

¹In other publications, subregion selection is called spatial filtering (Kushki et al. 2007), location-clustering (Youssef et al. 2003), or coarse localization (Feng et al. 2012).

(Feng et al. 2012) proposed using the subsets of APs whose RSS readings are the strongest assuming that the strongest signals provide the highest probability of coverage over time and the highest accuracy. (Kushki et al. 2007) and (Chen et al. 2006) applied a divergence metric (Bhattacharyya distance and information gain, respectively) to minimize the redundancy and maximize the information gained from the selected APs. The limitations of these approaches are i) they are only applicable to the FIPS based on RSS from WLAN APs, and ii) they only take the values of the features into account as selection criteria instead of the actual positioning accuracy. (Kushki et al. 2010) proposed an AP selection strategy able to choose APs ensuring a certain positioning accuracy using a nonparametric information filter. However, this approach uses consecutively measured fingerprints to select the subset of APs maximizing the discriminative ability w.r.t. localization. This method therefore needs several online observations for estimating one current position.

In (Zhou and Wieser 2018b), a feature selection algorithm based on randomized LASSO (Tibshirani 1996), an L_1 -regularized linear regression model, for selecting the relevant features for positioning is proposed. Each feature within the subregion is associated with an estimated coefficient. If the coefficient is sufficiently different from zero the corresponding feature is identified as relevant. However, this approach only connects the feature selection with the positioning error indirectly. Furthermore, LASSO-based feature selection is equivalent to MAP if the likelihood is Gaussian and the prior distribution is Laplace (Park and Casella 2008). These two assumptions are not necessarily justified with fingerprinting-based positioning. It is difficult to find a proper value of the hyper-parameter of LASSO-based feature selection, which makes the feature selection unstable (Fastrich et al. 2015). Finally, this feature selection algorithm is prone to overfitting. Applying this approach to select the relevant features for each subregion requires the number of observations in each subregion to be much larger than the number of the dimension of the features (e.g., the number of observable WLAN APs or BLE beacons). Normally, this requirement is not met in case of fingerprinting-based positioning using SoPs as the features.

In this paper, we thus propose an approach based on AFBGS to choose the most relevant features for fingerprinting-based positioning. This method differs from the previously mentioned ones in three ways: i) it takes the positioning error into account directly, i.e. the feature selecting process is directly combined with the fingerprinting-based positioning methods that are used at the online stage, ii) wrongly selected features from the forward greedy search step can be adaptively corrected by a backward greedy search step (Zhang 2011), and iii) it is a data-driven algorithm adapting automatically to the number of observations of each subregion.

3.3 The proposed approach

In this section, we briefly summarize the fundamentals of fingerprinting-based positioning and present the main contributions of this paper to reduce the computational complexity independent of the size of the RoI. In particular we present i) candidate subregion selection according to MJI, ii) selection of relevant features using AFBGS, and iii) adaptations of MAP and k NN-based positioning with the combination of the previous two steps. Finally, we briefly discuss the computational complexity of the proposed method.

3.3.1 Problem formulation

Each measured feature has a unique identifier and a measured value, e. g. the measurement related to a specific Wi-Fi AP can be identified by the media access control (MAC) address and has a RSS. It is thus formulated as a pair (a, v) of attribute a and value v . A complete measurement (i. e. fingerprint) \mathbb{O}_i^u taken by user u at location/time i consists of a set of attribute-value pairs, i. e. $\mathbb{O}_i^u := \{(a_{ik}^u, v_{ik}^u) | a_{ik}^u \in \mathbb{A}; v_{ik}^u \in \mathbb{R}; k \in \{1, 2, \dots, N_i^u\}\}$, where \mathbb{A} is the complete set of the identifiers of all available features and N_i^u ($N_i^u = |\mathbb{O}_i^u|$) is the number of features observed by u at i . The set of keys of such a fingerprint is defined as $\mathbb{A}_i^u := \{a_{ik}^u | \exists (a_{ik}^u, v_{ik}^u) \in \mathbb{O}_i^u\}$. A discrete RFM $\mathbb{M} := \{(\mathbf{I}_j, \tilde{\mathbb{O}}_j) | \mathbf{I}_j \in \mathbb{G}, j \in \{1, 2, \dots, |\mathbb{M}|\}\}$ is given as a set of position-fingerprint pairs representing the relationship between the location \mathbf{I} and the features \mathbb{O} at different locations within the RoI \mathbb{G} .

Specifically for the subregion selection, \mathbb{G} is divided into M non-overlapping subregions of arbitrary shape, i. e. $\mathbb{G} = \bigcup_{i=1:M} \mathbb{g}_i, \mathbb{g}_i \cap \mathbb{g}_j = \emptyset | i \neq j$. This segmentation can take contextual information into account by defining the subregions such that one subregion lies only in e.g. one building, one floor, one room or one corridor. Thus, the concept is directly applicable to multi-building or multi-floor situations. Each of the measurements (elements) of \mathbb{M} is assigned to the corresponding subregion. The subset of \mathbb{M} corresponding to the g^{th} subregion can thus be defined as $\mathbb{M}_g := \{(\mathbf{I}_j, \tilde{\mathbb{O}}_j) | \mathbf{I}_j \in \mathbb{g}_g, j \in \{1, 2, \dots, |\mathbb{M}_g|\}\}, \mathbb{M}_g \subseteq \mathbb{M}$, and the set of observable features in the corresponding subregion is denoted as $\mathbb{A}_g := \{a | a \in \bigcup_{j=1:|\mathbb{M}_g|} \tilde{\mathbb{A}}_j, \tilde{\mathbb{A}}_j \text{ of } \tilde{\mathbb{O}}_j, \exists (\mathbf{I}_j, \tilde{\mathbb{O}}_j) \in \mathbb{M}_g\} (g \in \{1, 2, \dots, M\}), \mathbb{A}_g \subseteq \mathbb{A}$.

The positioning process consists of inferring the estimated user location $\hat{\mathbf{I}}_i^u = f(\mathbb{O}_i^u)$ as a function of the fingerprint and the RFM where f is a suitable mapping from fingerprint to location, i. e. $f : \mathbb{O}_i^u \mapsto \hat{\mathbf{I}}_i^u$. Herein we propose the following solutions for mitigating the computational load associated with the online stage:

- identifying (during the offline stage) the most relevant features within each subregion using the AFBGS such that the actual location calculation can be carried out during the online stage using only those instead of using all features;
- selecting the subregion as a coarse approximation of the actual user location based on MJJ during the online-stage.

Within this paper we combine the above two steps with a MAP and k NN-based positioning for performance analysis. We implement it in a way to keep the computational complexity of the online stage almost independent of the size of the RoI and of the total number of observable features within the RoI.

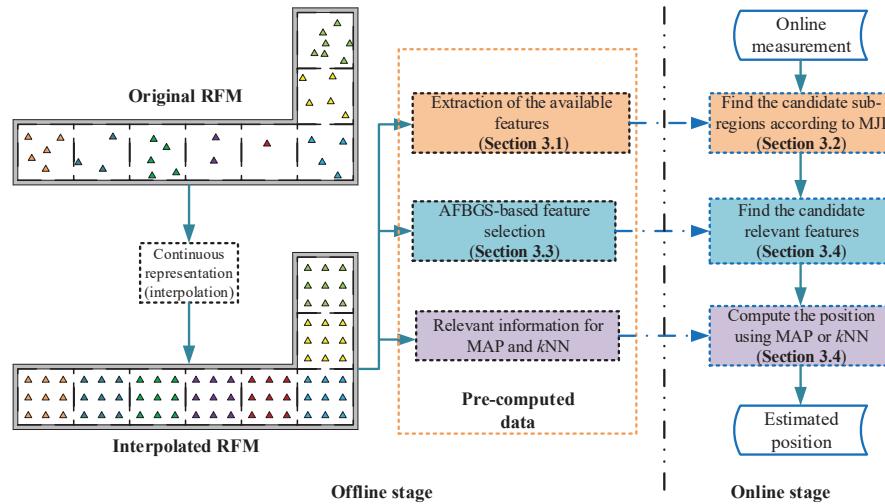


Figure 3.1: The proposed framework. In order to make the number of reference points within all subregions equal we interpolate the reference data in the original RFM using kernel smoothing (Berlinet and Thomas-Agnan 2011) to provide a denser regular grid of reference points. This interpolated RFM is used to calculate the pre-computed data for online positioning.

3.3.2 Subregion selection using MJJ

MJJ, an indicator of similarity between the keys of the measured fingerprints and the keys associated with the individual subregions in the RFM, is applied to identify a set of candidate subregions most probably containing the actual user location (Zhou and Wieser 2018b; Jani et al. 2015). We depict four qualitative examples in Figure 3.2. If the overlap between the features within a subregion (according to the RFM) and the features within a user's observation is large, the value of the MJJ is high, otherwise, it is low. For a more detailed discussion of MJJ, see (Zhou and Wieser 2018b).

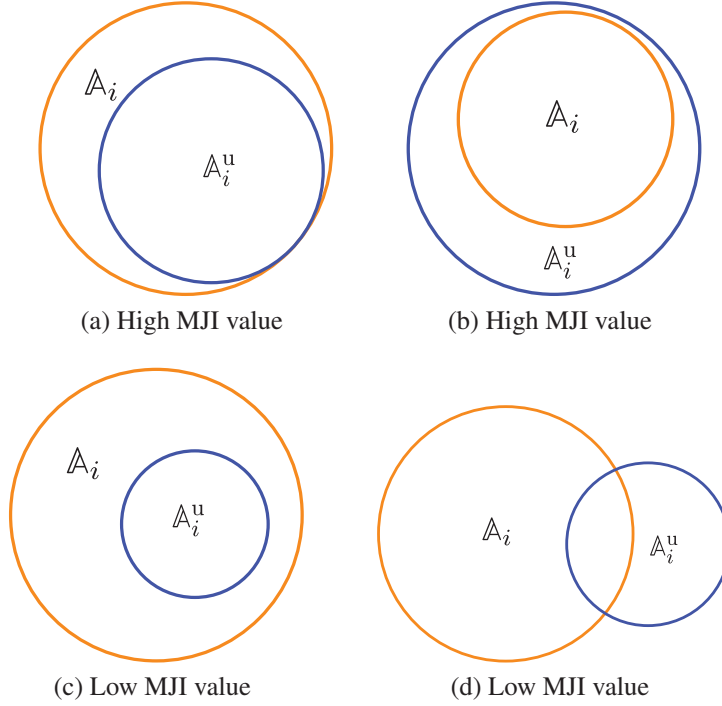


Figure 3.2: Qualitative examples of MJI values: (a) a few features within the current subset of the RFM are not measured, (b) there are some measured features which are not in the current subset of the RFM, (c) the measured features are only few of the ones in the current subset of the RFM, (d) there is little overlap between the measured features and the ones in the current subset of the RFM.

The m subregions with the highest values of the MJI are selected as candidate subregions for the subsequent positioning. Their cell indices are collected in the vector $\mathbf{s}_m^u \in \mathbb{N}^m$ for further processing. If the subregions are non-overlapping, as introduced above, m needs to be large enough to accommodate situations where the actual user location is close to the border between subregions and small enough to reduce the computational burden of the subsequent user location estimation.

In order to determine a proper value of m , an indicator is introduced for denoting whether a validation example is contained in the selected top m subregions (indicator 1) or not (indicator 0). Throughout this paper we assume that the user is actually within the RoI and thus within one subregion when the subregion selection process is carried out. Given M^{val} validation examples $\{(\mathbb{O}_n^{\text{val}}, \mathbf{I}_n^{\text{val}})\}_{n=1}^{M^{\text{val}}}$, the indicator for the n -th example is thus defined as:

$$I(\mathbf{I}_n^{\text{val}}, m) = \begin{cases} 1, & \text{if } \mathbf{I}_n^{\text{val}} \in \bigcup_{j \in \mathbb{S}_m^{\text{val}}} \mathcal{G}_j \\ 0, & \text{otherwise} \end{cases} \quad (3.1)$$

where $\mathbb{S}_m^{\text{val}}$ contains the indices of the selected subregions for this particular validation example. The subregion selection loss is then defined as the fraction of the validation examples for which the true location is not within the selected subregions, i. e.

$$\mathcal{L}^{\text{MJ}}(m) = 1 - \frac{\sum_{n=1}^{M^{\text{val}}} I(\mathbf{I}_n^{\text{val}}, m)}{M^{\text{val}}}. \quad (3.2)$$

The subregion selection loss reduces as m increases. However, increasing the number of selected subregions contradicts to the idea of reducing the computational complexity of the online positioning. Balancing between subregion selection loss and computational complexity is a multi-objective optimization problem, and it has no single optimum. Herein we decided to select an appropriate number of subregions heuristically by analyzing plots of selection loss versus m and choosing a value from which on the curve is flat. I.e., at which further increasing m hardly reduces the loss.

3.3.3 Feature selection using AFBGS

In a real-world environment there may be a large number of features available for positioning, e.g., hundreds of APs may be visible to the mobile user device in certain locations. Not all of them will be necessary to estimate the user location. In fact, using only a well-selected subset of the available signals instead of all will reduce the computational burden and may provide a more accurate estimate because the measured signals are affected by noise and possibly interference. Furthermore, the number of observable features typically varies across the RoI, e.g. due to WLAN APs antenna gain patterns, structure and furniture of the building. However, it is preferable to use the same number of features throughout the candidate subregions for assessing the similarity between the measured fingerprint and the ones extracted from the RFM during the online phase.

We therefore recommend selecting a number h of features per candidate subregion for the final position estimation. To facilitate this selection during the online phase, the relevant features within each subregion are already identified beforehand once the RFM is available. We use the AFBGS (Zhang 2011) for this step. During the online phase h relevant features (possibly different for each subregion) are selected among the identified ones such that they are available both within the RFM and the user fingerprint.

The combination of forward and backward steps within the algorithm²

²The forward part of the algorithm is referred to as matching pursuit for least square regression in the signal processing community (Barron et al. 2008; Mallat and Zhang 1993). In

makes AFBGS capable of correcting intermediate erroneous selection of features. We first shortly review both forward and backward greedy algorithms, then present AFBGS.

Let $\mathbb{F}_{(h_i)}$ denote the set of indices of the selected relevant features of the i^{th} subregion. A fingerprinting-based positioning method f thus only uses these selected features to estimate the position, i.e. $f : \mathbb{O}|\mathbb{F}_{(h_i)} \rightarrow \mathbf{I}$. The forward greedy search algorithm adds features one at a time picking the one causing the biggest reduction of a given loss at each step. Herein we use the mean squared error (MSE), a widely applied loss function for regression problems, as the loss metric for the feature selection. For instance, with $\mathbb{F}_{(h_i)}$ being the set of h features selected for the i^{th} subregion, the loss is the MSE of the positions estimated for all validation examples using this particular set of subregions:

$$\mathcal{L}^{\text{FS}}(\mathbb{F}_{(h_i)}) = \frac{1}{M^{\text{val}}} \sum_{n=1}^{M^{\text{val}}} \|f(\mathbb{O}_n^{\text{val}}|\mathbb{F}_{(h_i)}) - \mathbf{I}_n^{\text{val}}\|_2^2 \quad (3.3)$$

An example of applying the forward greedy search to select features for the i^{th} subregion is described in Table 3.2. This algorithm works well in case the features are independent of each other. Otherwise the forward greedy search might wrongly select non-relevant features. Such wrong selections cannot be corrected by the forward greedy algorithm anymore. One remedy introduced to solve this problem is the backward greedy search (Table 3.3), which starts with all features and greedily removes them one at a time picking the one associated with the smallest increase of the loss at each step. However, also the backward greedy search has two disadvantages: i) it is computationally costly because it starts with all features; ii) it is prone to overfitting if the number of observations in each subregion is much lower than the number of observable features.

A practically useful alternative is the AFBGS, i.e., introducing a backward greedy search after each step of the forward greedy search. In this way the backward search starts with a set that does not overfit, and it can correct wrong selections made earlier. Features removed by the backward steps are treated again as candidates during subsequent forward steps.

Table 3.4 shows the pseudocode of the implementation of AFBGS. This algorithm adaptively identifies the relevant features for each subregion according to the chosen minimum reduction ε and minimum relative increment ν of the loss. The number of identified relevant features per subregion will therefore vary across the RoI. In (Zhang 2011), the author recommended to

machine learning it is known as boosting (Bühlmann 2006).

³Herein the initial loss is defined as the MSE by using the median of all RPs as the estimated position when no candidate feature is selected.

Table 3.2: Pseudocode of the forward greedy search algorithm

Algorithm: forward greedy search

- 1: Input: $\{\mathbb{O}_n^{\text{val}}, \mathbf{I}_n^{\text{val}}\}$, $n = 1, 2, \dots, M^{\text{val}}$;
 minimum reduction of the loss $\varepsilon > 0$;
 maximum number of relevant features $k \in \mathbb{N}$;
 all observable features \mathbb{F}_{all}
- 2: Output: relevant features $\mathbb{F}_{(h_i)}$ of i^{th} subregion
- 3: $\mathbb{F}_{(0)} = \emptyset$, initial MSE $\mathcal{L}^{\text{FS}}(\mathbb{F}_{(0)})$ ³
- 4: **for** $t = 1, 2, \dots, |\mathbb{A}_i|$:
- 5: $\mathbb{F}_{\text{avail}} = \mathbb{F}_{\text{all}} \setminus \mathbb{F}_{(t-1)}$
- 6: $\hat{\mathbb{F}} = \underset{\mathbb{F} = \mathbb{F}_{(t-1)} \cup \beta}{\text{arg min}} \mathcal{L}^{\text{FS}}(\mathbb{F}), \forall \beta \in \mathbb{F}_{\text{avail}}$
- 7: $\mathbb{F}_{(t)} = \hat{\mathbb{F}}$
- 8: $\Delta = \mathcal{L}^{\text{FS}}(\mathbb{F}_{(t-1)}) - \mathcal{L}^{\text{FS}}(\mathbb{F}_{(t)})$
- 9: **if** ($\Delta \leq \varepsilon$ or $|\mathbb{F}_{(t)}| \geq k$):
- 10: **break**
- 11: **end if**
- 12: **end for**
- 13: **return** $\mathbb{F}_{(h_i)} = \mathbb{F}_{(t)}$: set of selected features

Table 3.3: Pseudocode of the backward greedy search algorithm

Algorithm: backward greedy search

- 1: Input: $\{\mathbb{O}_n^{\text{val}}, \mathbf{I}_n^{\text{val}}\}$, $n = 1, 2, \dots, M^{\text{val}}$;
 maximum increment of the loss $\phi > 0$;
 minimum number of relevant features $k \in \mathbb{N}$;
 all observable features \mathbb{F}_{all}
- 2: Output: relevant features $\mathbb{F}_{(h_i)}$ of i^{th} subregion
- 3: $\mathbb{F}_{(|\mathbb{A}_i|)} = \mathbb{F}_{\text{all}}$
- 4: **for** $t = |\mathbb{A}_i| - 1, |\mathbb{A}_i| - 2, \dots, 1$:
- 5: $\hat{\mathbb{F}} = \underset{\mathbb{F} = \mathbb{F}_{(t+1)} \setminus \beta}{\text{arg min}} \mathcal{L}^{\text{FS}}(\mathbb{F}), \forall \beta \in \mathbb{F}_{(t+1)}$
- 6: $\mathbb{F}_{(t)} = \hat{\mathbb{F}}$
- 7: $\Delta = \mathcal{L}^{\text{FS}}(\mathbb{F}_{(t)}) - \mathcal{L}^{\text{FS}}(\mathbb{F}_{(t+1)})$
- 8: **if** ($\Delta \geq \phi$ or $|\mathbb{F}_{(t)}| \leq k$):
- 9: **break**
- 10: **end if**
- 11: **end for**
- 12: **return** $\mathbb{F}_{(h_i)} = \mathbb{F}_{(t+1)}$: set of selected features

set $v = 0.5$ which we do when applying the algorithm later on. According to (Zhang 2011), AFBGS will terminate in a finite number of steps, which is no more than $\lceil 1 + \frac{\mathcal{L}^{\text{FS}}(\mathbb{F}_{(0)})}{v\varepsilon} \rceil$, where $\mathcal{L}^{\text{FS}}(\mathbb{F}_{(0)})$ is the MSE by taking the median

of all RPs as the initially guessed position.

Table 3.4: Pseudocode of the adaptive forward-backward greedy search (AFBGS) algorithm

Algorithm: AFBGS

- 1: Input: $\{\mathbb{O}_n^{\text{val}}, \mathbb{I}_n^{\text{val}}\}, n = 1, 2, \dots, M^{\text{val}};$
 minimum reduction of the loss $\varepsilon > 0;$
 relative increment of the loss $\nu \in (0, 1);$
 all observable features \mathbb{F}_{all}
- 2: Output: relevant features $\mathbb{F}_{(h_i)}$ of i^{th} subregion
- 3: $t = 1, \mathbb{F}_{(0)} = \emptyset$, initial MSE $\mathcal{L}^{\text{FS}}(\mathbb{F}_{(0)})$
- 4: **while (True)**
- 5: $\mathbb{F}_{\text{avail}} = \mathbb{F}_{\text{all}} \setminus \mathbb{F}_{(t-1)}$
- 6: $\hat{\mathbb{F}}^{\text{forward}} = \underset{\mathbb{F} = \mathbb{F}_{(t-1)} \cup \beta}{\text{arg min}} \mathcal{L}^{\text{FS}}(\mathbb{F}), \forall \beta \in \mathbb{F}_{\text{avail}}$
- 7: $\mathbb{F}_{(t)} = \hat{\mathbb{F}}^{\text{forward}}$
- 8: $\Delta^{\text{forward}} = \mathcal{L}^{\text{FS}}(\mathbb{F}_{(t-1)}) - \mathcal{L}^{\text{FS}}(\mathbb{F}_{(t)})$
- 9: **if** ($\Delta^{\text{forward}} \leq \varepsilon$)
- 10: **break**
- 11: **end if**
- 12: $\mathbb{F}_{\text{backward}} = \mathbb{F}_{(t)}$
- 13: $c = |\mathbb{F}_{\text{backward}}|, \mathbb{F}_{(c)} = \mathbb{F}_{\text{backward}}$
- 14: **while (True)**
- 15: **if** ($|\mathbb{F}_{\text{backward}}| == 1$)
- 16: **break**
- 17: **end if**
- 18: $c = c - 1$
- 19: $\hat{\mathbb{F}}^{\text{backward}} = \underset{\mathbb{F} = \mathbb{F}_{(c+1)} \setminus \beta}{\text{arg min}} \mathcal{L}^{\text{FS}}(\mathbb{F}), \forall \beta \in \mathbb{F}_{(c+1)}$
- 20: $\mathbb{F}_{(c)} = \hat{\mathbb{F}}^{\text{backward}}$
- 21: $\Delta^{\text{backward}} = \mathcal{L}^{\text{FS}}(\mathbb{F}_{(c)}) - \mathcal{L}^{\text{FS}}(\mathbb{F}_{(c+1)})$
- 22: **if** ($\Delta^{\text{backward}} > \nu \Delta^{\text{forward}}$)
- 23: **break**
- 24: **end if**
- 25: **end while**
- 26: $\mathbb{F}_{(t)} = \mathbb{F}_{(c+1)}$
- 27: $t = t + 1$
- 28: **end while**
- 29: **return** $\mathbb{F}_{(h_i)} = \mathbb{F}_{(t)}$: set of selected features

3.3.4 The combination of subregion and feature selections with fingerprinting-based positioning

In this part, we present the way to combine the previously proposed subregion and feature selections with two widely used fingerprinting-based positioning methods, namely k NN and MAP, to estimate the user location from the fingerprint \mathbf{O}^u measured at the actual but unknown location \mathbf{I}^u . We assume that a set of candidate subregions \mathbb{S}_m^u has been selected using the MJI-based subregion selection. For each of the selected subregions $\mathbb{g}_i, i \in \mathbb{S}_m^u$, the set $\mathbb{F}_{(hi)}$ of relevant features has been chosen using AFBGS. While the cardinality of these sets $\mathbb{F}_{(hi)}$ will be different, both positioning approaches require the number of features taken into consideration to be the same for all candidate locations. So, we determine the set

$$\mathbb{F}_u^{\text{candidate}} = \mathbb{A}_i^u \cap \mathbb{F}_u \quad (3.4)$$

of candidate features, where $\mathbb{F}_u = \bigcup_{i \in \mathbb{S}_m^u} \mathbb{F}_{(hi)}$. The candidate features are all features actually observed by the user and available in at least one of the candidate subregions \mathbb{S}_m^u . We finally rank these features by the number of candidate subregions in which they are available. The h candidate features \mathbb{F}_h^u ranking highest is used for fingerprinting-based positioning.

MAP uses a variety of discrete candidate locations \mathbf{I} and applies Bayes' rule to compute for each of them the degree to which the assumption that the current location of the user is \mathbf{I} is supported by the available RFM and the currently observed fingerprint (Park et al. 2010; Madigan et al. 2005). Further details regarding the combination of MAP with subregion and feature selection are given in (Zhou and Wieser 2018b).

For k NN the k points \mathbf{I}_q within the RFM which are closest to the user's observation in feature space have to be identified. Assuming that their indices are collected in the set \mathbb{m}_k^u the estimated location $\hat{\mathbf{I}}_{k\text{NN}}^u$ of the user is obtained from:

$$\hat{\mathbf{I}}_{k\text{NN}}^u = \sum_q \omega_q \mathbf{I}_q, \forall q \in \mathbb{m}_k^u, \quad (3.5)$$

where the respective weights ω_q are defined as proportional to the inverse distance of the fingerprints in the feature space:

$$\omega_q = \frac{1/d(\mathbb{O}_i^u, \mathbb{O}_q)}{\sum_p 1/d(\mathbb{O}_i^u, \mathbb{O}_p)}, \forall p \in \mathbb{m}_k^u. \quad (3.6)$$

Herein, $d : \mathbb{O} \times \mathbb{O} \mapsto \mathbb{R}$ is a chosen distance metric, e.g., Euclidean distance

or Hamming distance, used to measure the dissimilarity between fingerprints. The adaptation proposed by us consists of the construction of these observations \mathbb{O} which contain exactly one row for each of the h candidate features selected previously and collected in \mathbb{F}_h^u . Generally these vectors are much smaller than for the standard k NN-approach, where they would need to have one entry for each feature observed anywhere within the RoI. The proposed online positioning approach is summarized in Table 3.5.

Table 3.5: Pseudocode of the proposed online positioning approach

Algorithm: Online positioning

- 1: Input: the user's observation \mathbb{O}_i^u ;
the selected relevant features $\{\mathbb{F}_{(h^1)}, \mathbb{F}_{(h^2)}, \dots, \mathbb{F}_{(h^M)}\}$ of M subregions;
the precomputed data for MAP and k NN⁴
- 2: Output: the estimated location $\hat{\mathbb{I}}_i^u$ of the user
- 3: $\mathbb{S}_m^u :=$ find the m candidate subregions according to the ranking of MJI
- 4: $\mathbb{F}_h^u :=$ find the candidate features according to the ranking of the frequency of the selected relevant features $\mathbb{F}_{\text{candidate}}^u$ w.r.t. $\mathbb{F}_{(h^i)}, \forall i \in \mathbb{S}_m^u$
- 5: $\hat{\mathbb{I}}_i^u :=$ using MAP or k NN according to \mathbb{F}_h^u and the precomputed data
- 6: **return** $\hat{\mathbb{I}}_i^u$: the estimated location

3.3.5 Computational complexity of online positioning

In this part, we analyze the computational complexity of the proposed approach and compare it to MAP and k NN-based positioning without subregion and feature selection.

The runtime computational complexity of estimating one position using MAP and k NN are $\mathcal{O}(\alpha M(|\mathbb{A}|^2 + 1))$ and $\mathcal{O}(k\alpha \cdot M \log(\alpha \cdot M)|\mathbb{A}|)$ ⁵, respectively, where α is the number of candidate locations in each of the M subregions and k is the number of nearest neighbors used for positioning. The computational complexity of the proposed method is instead approximately equal to $\mathcal{O}(\alpha|\mathbb{S}_m^u| \cdot |\mathbb{F}_h^u|^2)$ and $\mathcal{O}(\alpha k|\mathbb{S}_m^u| \log(\alpha \cdot |\mathbb{S}_m^u|) \cdot |\mathbb{F}_h^u|)$ for MAP and k NN, respectively. So, clearly the computational complexity of the proposed approach is significantly less than for the MAP- and k NN-based positioning approaches without subregion and feature selection. Furthermore, it is

⁴For MAP, the precomputed data are the likelihood of the selected features of each subregion and prior probability of each candidate location. For k NN, the precomputed data are the observations of the selected relevant features of each subregion.

⁵We implemented MAP as proposed in Youssef and Agrawala (2008). The implementation can be sped up using algebraic factorization (Bisio et al. 2016). For k NN, tree-based methods (e. g. kd-tree) are used in the scikit-learn implementation (Pedregosa et al. 2011).

independent of the size of the RoI and of the total number of available features within the RoI. The proposed approach is to constrain and limit the search to a set of candidate reference locations and selected features for the online positioning. Though we only give the analytical formula of the computational complexity of MAP and k NN, other fingerprinting-based location methods will also benefit from the proposed approach, because the computational complexity of fingerprinting-based positioning is proportional to the size of the search space.

Besides the RFM further data required during the online positioning stage can be precomputed already during the offline stage (Figure 3.1). This holds in particular for:

- the set \mathbb{A}_i of available feature keys of each subregion required for calculating the MJI at the online stage,
- the set $\mathbb{F}_{(h^i)}$ of relevant features of each subregion calculated using AFBGS,
- and the conditional distribution ($\text{Prob}(\mathbb{O}_j|\mathbb{I})$) of the selected relevant features within each subregion obtained from kernel density estimation.

At the online stage these pre-computed data are cached to the user device to achieve location estimation while realizing mobile positioning. Furthermore, only the observed values of the features that are selected as relevant ones in the RFM need to be loaded during the online positioning stage. The proposed preprocessing steps also reduce the required storage space for saving the cached pre-computed data because these data only need to cover the selected relevant features instead of all the features observable within the RoI.

3.4 Experimental results and discussion

In this section, we first describe the experimental configurations, including the RoIs, data collection and the two different types of features used, namely RSS from WLAN APs and BLE beacons. A detailed analysis of MJI-based subregion selection, a comparison of different feature selection methods (randomized LASSO, forward greedy, and AFBGS), as well as the positioning accuracy and cost of time for positioning are illustrated for one RoI using only WLAN RSSs. Finally, we apply the proposed approach to a larger RoI using both types of observables.

3.4.1 Testbed

In this section we analyze data obtained from real measurements collected using a Nexus 6P⁶ smartphone (for WLAN & BLE RSS) and a Leica MS50 total station (for position ground truth) within two RoIs in an office building (Figure 3.3) which is covered by a plethora of WLAN APs signals and BLE beacons⁷.

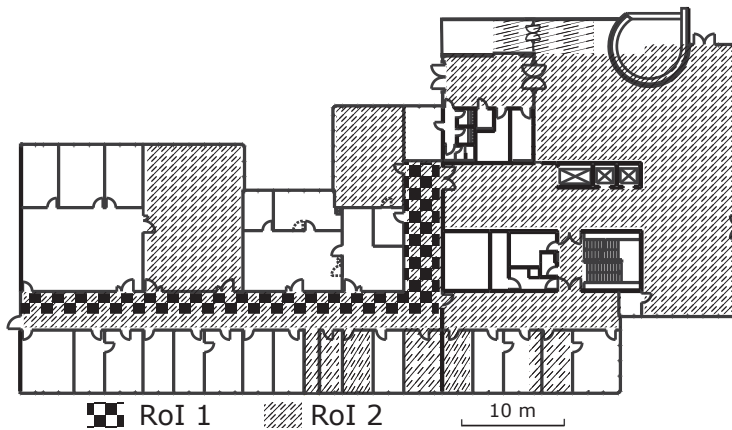


Figure 3.3: Two RoIs within one floor of an office building

We kinematically determined the RFM by recording data approximately every 1.5 seconds while a user walked through the RoI and a total station tracked a prism attached to the Nexus smartphone with an accuracy of about 5 mm. The recording rate is around 0.67 Hz. This is higher than typically reported update rates for WLAN RSS and was achievable on this device by changing the hardware settings of the scanning interval. At the online stage, the RSS collection can be further accelerated by incremental scanning as discussed in (Brouwers et al. 2014). This approach is a compromise between the high accuracy attainable by stop & go measurements at carefully selected reference positions and the low extra effort of crowd-sourced RFM data collection as outlined e.g. in (Radu and Marina 2013). For simplicity we do not apply further pre-processing, such as filtering out the APs yielding low RSS values, discarding rarely measured APs, or merging signals transmitted from the same signal source on multi-frequencies although such steps could further reduce the computational complexity and improve the results in a real application.

⁶Herein we carried out all experiments using only one device. It is to be expected that also the quality of the results obtained using our approach depends on the mobile device(s) used for data collection, see (Bisio et al. 2013). However, we leave a related investigation for future work.

⁷All APs and beacons had been deployed independently of this experiment and long before it for the purpose of internet access. Each AP supports two frequencies (2.4 GHz and 5 GHz) and has a built-in BLE beacon. There is no automatic power adjustment of the APs.

In order to evaluate the performance of the proposed approach independently an additional test dataset was collected. The coordinates of the test positions (TPs) as measured by the total station were later used as ground truth for calculating the positioning error in terms of the Euclidean distances between estimated and true coordinates. The use of the tracking total station for both RFM data collection and test point data collection meant that RSS data did not have to be collected at any specific points (e.g. marked ones) within the RoI or subregions, and it was not necessary to occupy the same points again. We report the 50th, 75th and 90th percentile of the horizontal positioning error (i. e. circular error (CE) 50, CE75, CE90) defined as the minimum radius for including 50%, 75%, and 90% of the positioning errors (Potort et al. 2018). Furthermore, as an indicator for outlying position estimates (in particular due to wrong subregion selection) we also report the percentage of positioning errors exceeding 10 m, and as an indicator of computational complexity the average time to calculate the position of the test point. Data processing according to the proposed algorithms was implemented in Python using the scikit-learn package (Pedregosa et al. 2011) as outlined in Figure 3.1.

The details for two RoIs are summarized in Table 3.6 and the numbers of available features (i. e. visible WLAN APs or BLE beacons) are illustrated in Figure 3.5. The number of available features changes throughout the RoI and is thus also different in the different subregions. . Abrupt changes of feature observability occur in some locations close to walls and close to support pillars or cable/pipe products (not contained in the available floorplan). This is caused by the uneven number of raw measurements assigned to each subregion. In RoI 1, we carried out 5-rounds of one-day data collections about one month apart. This allowed us to take both the spatial and temporal variations into account when building the RFM. For RoI 2, the collected RFM is used as an example for validating the performance of the proposed approach in case of a more extended RoI. Though the area of RoI 1 is a subset of RoI 2, the two RFMs are collected at different time and with different WLAN configurations⁸. Both RoIs are divided into subregions of size $2 \times 2m^2$ aligned with the floor plan of the RoI. While such an alignment may not be necessary, it is useful as it assures that individual subregions are not split by walls or other obstacles possibly causing discontinuities in the feature space. In many applications of FIPS a floor plan exists, and can thus be used for subregion definition because it is required for the FIPS and the associated location-based services anyway. In addition, some subregions contain no measurements (see Table 3.6). We have no need to treat the empty subregions specifically because they will be filtered out by subregion selection anyway without increasing the computational burden much.

⁸An upgrade of the WLAN (e. g. change APs) has been carried out after the data collection of RoI 1.

The originally observed RFM of each subregion is further densified to a regular grid of about 25 reference points per m^2 (i.e. spacing about $0.2 \times 0.2m^2$) by kernel smoothing interpolation (Figure 3.4) to mitigate the non-uniform point density of the original RFM (Figure 3.1). This process also ensures that data are available at the same location within one subregion. In this paper, we use a Matérn kernel (the length scale is 1) for smoothing the spatial distribution of the RSS and assume that the propagation channel introduces the additive Gaussian white noise to the RSS.⁹ The smoothly interpolated raw measurements are rounded to integers for reducing the storage requirements and for mitigating the impact of the quantization of RSS values on the positioning performance (Torres-Sospedra and Moreira 2017a) and the APs are treated as non-measurable if their RSS values are lower than -100 dBm. The resulting gridded RFM is used for all further processing steps.

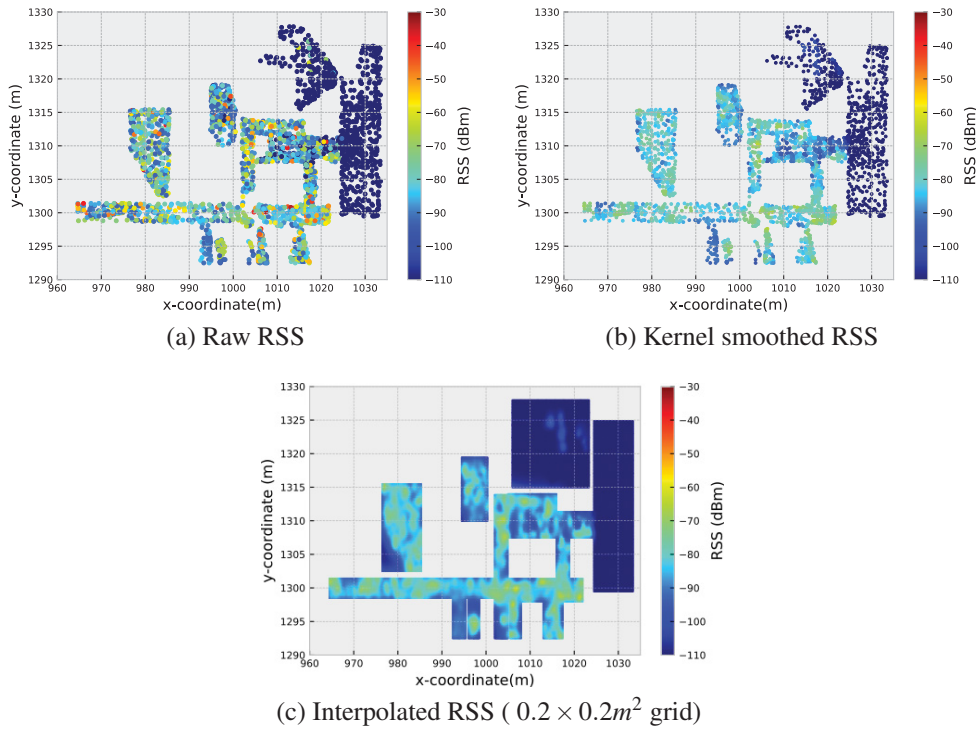


Figure 3.4: An example of the spatial distribution of RSS values as used for RFM generation

⁹Assessing the impact of different uniform or non-uniform subregion shapes and sizes as well as alternative interpolation strategies is beyond the scope of this paper and left for future work.

¹⁰Due to the constraints (e.g. furnitures and decorations) of accessibility of the RoI, several subregions have no observations. The numbers in parentheses denotes the numbers of subregions containing at least one observation.

¹¹The training data were obtained from the densified RFM obtained through kernel smoothing of raw RFM data, while the test data are separately collected raw data. By coincidence, the size of it is larger than that of the test data.

Table 3.6: Summary of RFM characteristics

| RoI | Area (m^2) | Number of subregions ¹⁰ | Number of features | | Training data ¹¹ | Test data |
|-----|----------------|------------------------------------|--------------------|-----|-----------------------------|-----------|
| | | | WLAN | BLE | | |
| 1 | 120 | 35 (34) | 399 | – | 1525 | 509 |
| 2 | 1100 | 326 (285) | 490 | 278 | 2855 | 476 |

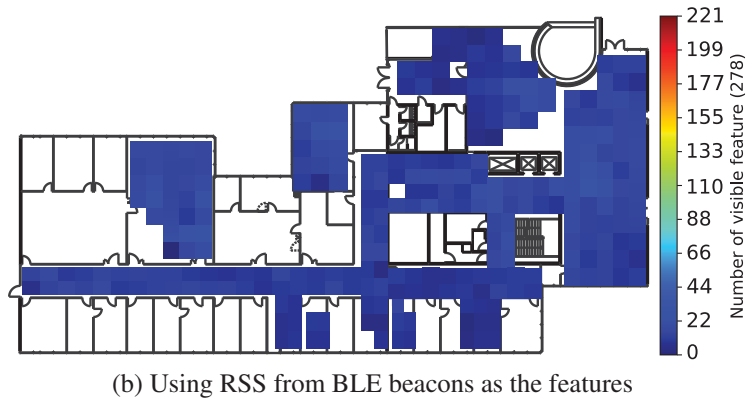
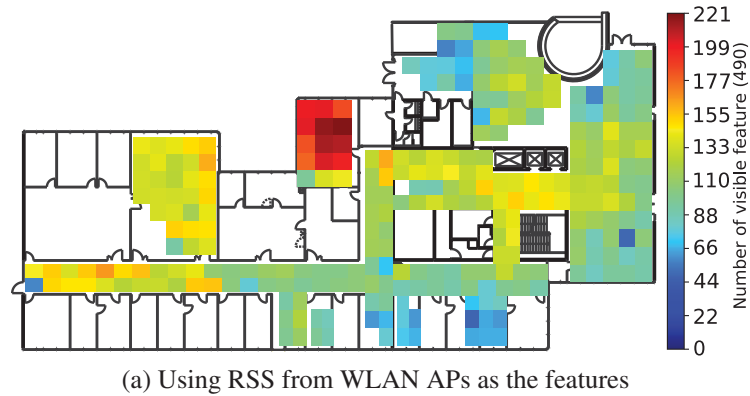


Figure 3.5: The number of the available features of each subregion

3.4.2 Analysis using WLAN signals in RoI 1

Validation MJI-based subregion selection. The RoI 1 consists of 34 subregions. Figure 3.6a shows the MJI for all pairs of subregions indicating that the index is related to the Euclidean distance of the training data. This corresponds to the expectation that the same APs are available in nearby subregions while different APs are observed in subregions far from each other.

The MJI value is used here as an indicator of the similarity between the features measured by the user and the ones available in the individual subregions according to the RFM. We now use the loss function (3.2) to determine a suitable number m of subregions for the MJI-based subregion selection. Figure 3.7a shows the loss as a function of m for RoI 1. The figure indicates that the loss is almost constant if m exceeds 16. In consecutive parts of this sub-

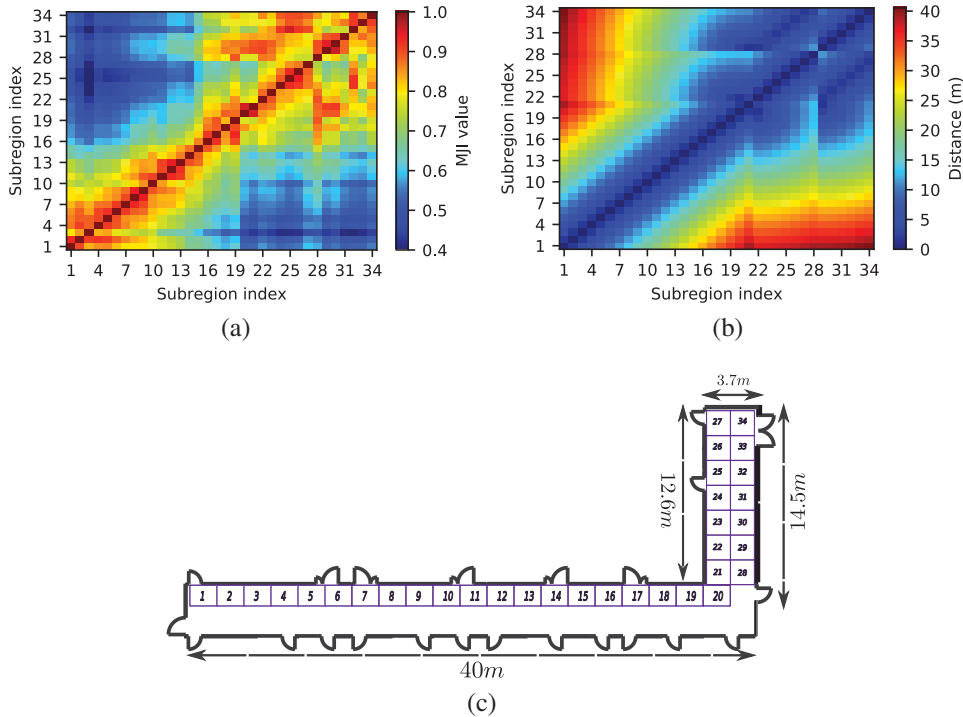


Figure 3.6: Analysis of MJI between the subregions of RoI 1 (training data). (a): Spatial distribution of the MJI. Apparent abrupt changes of MJI values appear in the upright part. This is related to the spatial distribution of the subregions. (b) Spatial distribution of the distance. (c) Schematic of subregion division. Each subregion is indexed by an integer for this analysis. The concrete number associated with a specific subregion is irrelevant.

section, we analyze the performance of feature selection and positioning accuracy w.r.t. a given value of $m \in \{11, 16, 21, 34\}$. In this range of m , we show that a small compromise of subregion selection accuracy does not harm the positioning accuracy too much, which is comparable to that obtained without subregion selection, i. e. for $m = 34$ in RoI 1.

Validation of AFBGS-based feature selection. In this part, we compare the feature selection performance using randomized LASSO as proposed in (Zhou and Wieser 2018b) to feature selection using the forward greedy algorithm and the AFBGS proposed herein. Since the latter two are directly related to fingerprinting-based positioning we evaluate the feature selection performance through the MSE of the position estimates calculated using the selected features. In Figure 3.8, we illustrate the MSE for two arbitrarily selected subregions using k NN and MAP. Regardless of the fingerprinting-based positioning method and subregion, all the curves in this figure have a similar pattern. We see that i) the MSE values achieved after applying forward search and AFBGS-based feature selection converge faster and more

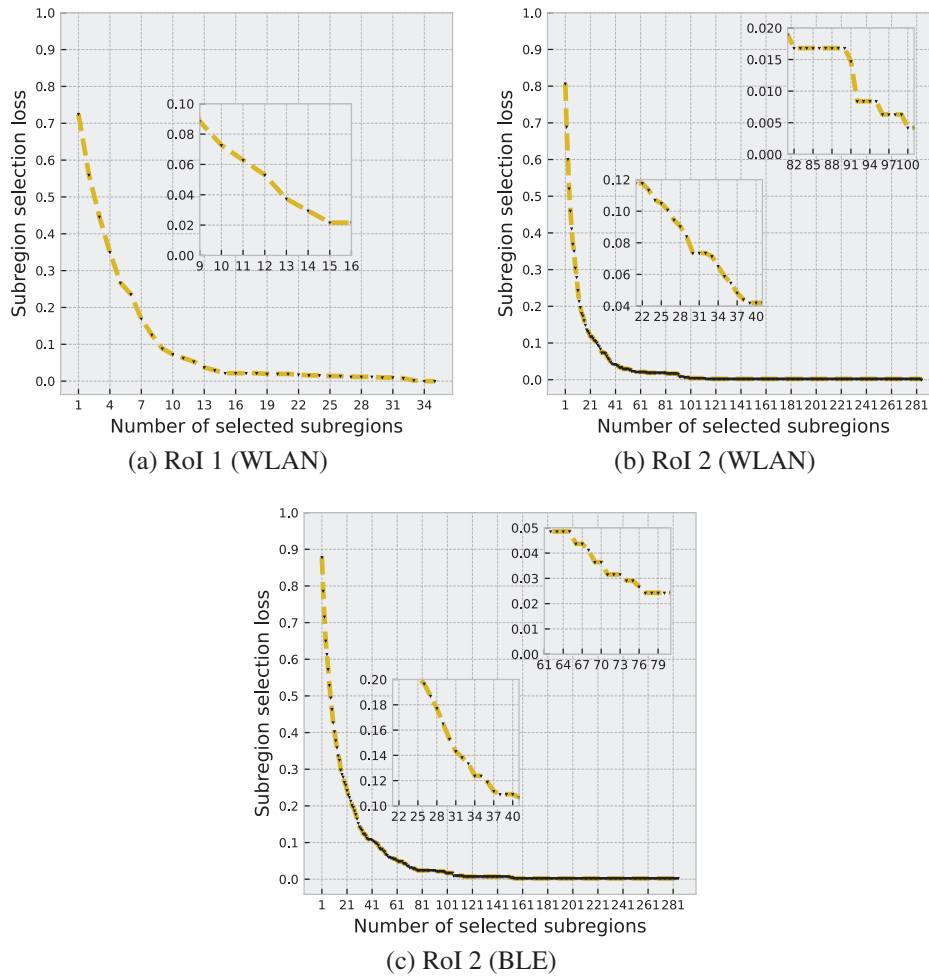


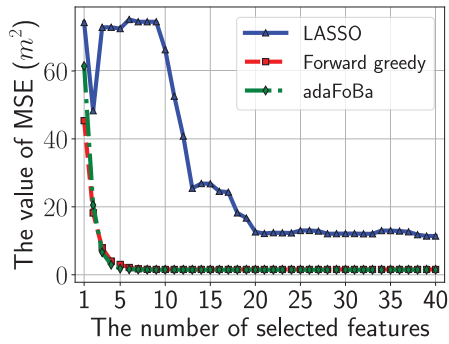
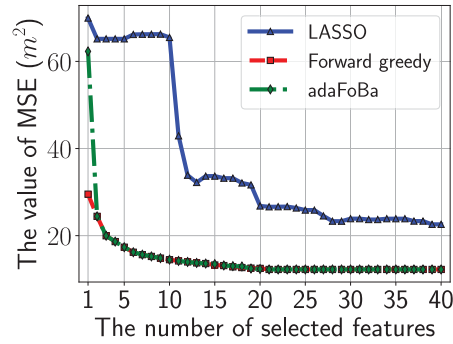
Figure 3.7: The subregion selection loss with respect to the number of selected subregions

consistently than after applying randomized LASSO-based feature selection, and ii) randomized LASSO-based selection performance stabilizes with only a large number of selected features (e. g. > 20). One explanation of this pattern is that randomized LASSO selects the features based on a regularized linear regression model instead of taking the fingerprinting-based positioning methods into account. However, if a higher number of features is used for positioning, the contribution of feature selection becomes less critical.

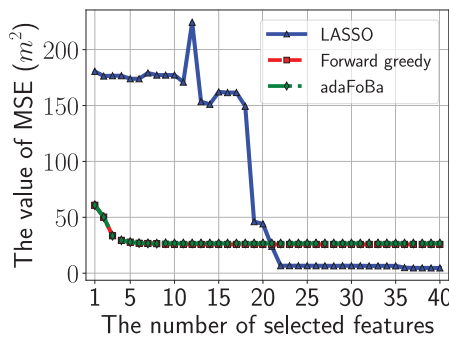
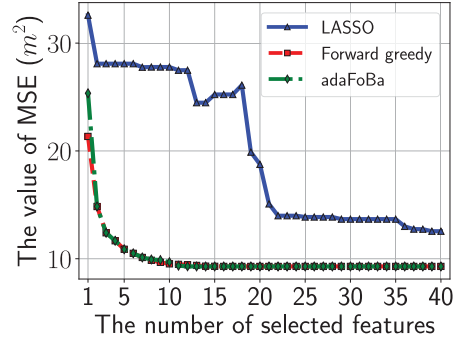
The difference of MSE between forward greedy search and AFBGS-based feature selection is very small (Figure 3.8), because the selected features are very similar in both cases, see Table 3.7. However, the AFBGS-based results are slightly better and we thus recommend it because of the increased flexibility.

Table 3.7: The selected features of subregion 17. (The unique identification of AP is indexed from 0 to 398.)

| Feature selection method | Positioning method | Selected features (top 5) |
|--------------------------|--------------------|---------------------------|
| Randomized LASSO | – | 13, 107, 9, 80, 11 |
| Forward greedy search | k NN | 11, 270, 272, 144, 19 |
| | MAP | 10, 398, 19, 20, 25 |
| AFBGS | k NN | 71, 11, 270, 272, 99 |
| | MAP | 66, 71, 12, 140, 272 |

(a) k NN, subregion 11

(b) MAP, subregion 11

(c) k NN, subregion 21

(d) MAP, subregion 21

Figure 3.8: The MSE paths of two subregions. In the above test, the number of the selected subregion is fixed, i. e. $m = 21$.

The processing time and positioning accuracy. Table 3.8 presents the processing time¹² for positioning one TP of RoIs 1 using WLAN signals as the features. The positioning time of MAP based on the AFBGS selected features within 21 selected subregions is about 2.9 seconds, which is almost 10 times faster than that of using all features for positioning by searching over all the subregions. The positioning time is also lower than when using LASSO for feature selection. One explanation is that a lower number of features is selected as relevant by the proposed algorithms than by LASSO, as shown in

¹²We used Python to implement the proposed method and evaluate the processing time using the *time* package (<https://docs.python.org/3/library/time.html#module-time>).

Figure 3.8. As for the positioning performance, the 90th percentile of the circular error (CE90) increases by less than 0.7 m for both MAP and MAP based on the AFBGS selected features within 9 selected subregions as compared to that of using all features for positioning and searching over the whole RoI. In addition, the resampling and the subregion selection reduce the percentage of large errors. So the reduced processing effort comes at the price of a small loss in accuracy. If need be, the percentage of outlying position estimates could be further reduced by position filtering taking the user's motion or prior knowledge like floor plans into account during subregion selection and position estimation. This is beyond the scope of this paper and thus not further investigated here.

3.4.3 Analysis using WLAN and BLE signals in RoI 2

In the larger RoI 2 both WLAN and BLE signals are extracted as fingerprints. In this subsection, we present the results of applying the proposed algorithm to that RoI and both signal types. As shown in Figure 3.7b and Figure 3.7c, MJI-based subregion selection is applicable also in this case. The figure indicates that there is no need to search within all subregions, but actually a subset is sufficient. Appropriate numbers of the selected subregions are 32 and 38 for using WLAN and BLE signals, respectively. These search spaces are less than 12% of the area of the RoI.

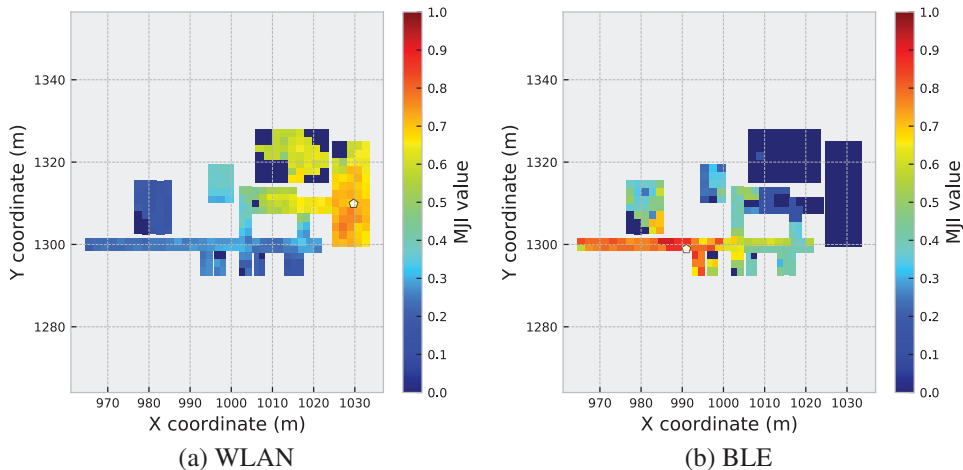


Figure 3.9: Examples resulting in large subregion selection error. In the figure, the white color filled pentagon indicates the ground truth of the TPs.

Figure 3.7 also shows that there is a small fraction of test points (e.g. about 7% for WLAN signals and RoI 2) for which the correct subregion is not among the selected ones even when choosing 30 subregions. This may seem astonishing, but an investigation of the related test cases (see e.g., Figure

Table 3.8: Comparison of positioning times and accuracies of RoI 1 ($h = -1$ denotes that the all the relevant features within the selected subregions are used for positioning. These features are selected adaptively by AFBGS (see Section 3.3.3) ; their number is thus different for different subregions.)

| Methods | Values of (m, h) | Mean positioning time (s) | CE50 (m) | CE75 (m) | CE90 (m) | Ratio of large errors (%) |
|--------------------------------------|-------------------------|------------------------------|----------|----------|----------|------------------------------|
| MAP (without interpolation) | all | 1.6 | 2.4 | 5.1 | 8.0 | 6.9 |
| k NN (without interpolation) | all | 2.1×10^{-2} | 1.8 | 3.8 | 7.0 | 4.5 |
| MAP (with interpolation) | all | 34.7 | 2.2 | 4.9 | 8.0 | 5.5 |
| k NN (with interpolation) | all | 0.1 | 1.8 | 4.0 | 6.8 | 3.7 |
| MAP (LASSO) | (11, -1) | 2.4 | 2.9 | 5.4 | 9.0 | 8.6 |
| | (16, -1) | 4.3 | 2.6 | 5.3 | 9.2 | 8.6 |
| | (21, -1) | 6.5 | 2.7 | 5.3 | 9.2 | 8.6 |
| | (34, -1) | 11.6 | 2.5 | 5.0 | 8.8 | 8.6 |
| | (34, 399) | 34.7 | 2.2 | 4.9 | 8.0 | 5.5 |
| k NN (LASSO) | (11, -1) | 0.3×10^{-2} | 2.2 | 4.5 | 7.5 | 4.1 |
| | (16, -1) | 0.5×10^{-2} | 2.3 | 4.3 | 7.4 | 4.7 |
| | (21, -1) | 0.6×10^{-2} | 2.0 | 4.2 | 7.3 | 4.9 |
| | (34, -1) | 1.0×10^{-2} | 2.2 | 4.0 | 7.1 | 4.7 |
| | (34, 399) | 1.0×10^{-2} | 1.8 | 3.9 | 6.9 | 4.5 |
| MAP (Forward greedy search) | (11, -1) | 1.4 | 2.7 | 5.3 | 8.6 | 7.1 |
| | (16, -1) | 2.5 | 2.7 | 5.6 | 9.2 | 8.4 |
| | (21, -1) | 3.9 | 2.7 | 5.6 | 9.4 | 8.8 |
| | (34, -1) | 6.9 | 2.7 | 5.5 | 9.3 | 8.3 |
| | (34, 399) | 34.9 | 2.4 | 5.2 | 8.2 | 6.3 |
| k NN (Forward greedy search) | (11, -1) | 0.2×10^{-2} | 2.5 | 4.8 | 7.5 | 4.7 |
| | (16, -1) | 0.3×10^{-2} | 2.8 | 5.1 | 8.2 | 5.5 |
| | (21, -1) | 0.3×10^{-2} | 2.5 | 4.7 | 7.7 | 5.7 |
| | (34, -1) | 0.5×10^{-2} | 2.3 | 4.7 | 8.4 | 5.3 |
| | (34, 399) | 1.0×10^{-2} | 1.9 | 4.0 | 7.1 | 4.1 |
| MAP (AFBGS) | (11, -1) | 1.3 | 2.6 | 5.0 | 8.6 | 6.9 |
| | (16, -1) | 2.4 | 2.7 | 5.5 | 9.2 | 8.1 |
| | (21, -1) | 3.7 | 2.8 | 5.5 | 9.1 | 8.4 |
| | (34, -1) | 6.6 | 2.7 | 5.7 | 9.2 | 8.3 |
| | (34, 399) | 35.0 | 2.3 | 5.2 | 8.0 | 5.9 |
| k NN (AFBGS) | (11, -1) | 0.2×10^{-2} | 2.9 | 5.0 | 7.4 | 4.5 |
| | (16, -1) | 0.3×10^{-2} | 3.2 | 5.5 | 8.2 | 4.5 |
| | (21, -1) | 0.3×10^{-2} | 2.7 | 5.3 | 8.1 | 5.1 |
| | (34, -1) | 0.4×10^{-2} | 2.8 | 5.0 | 8.7 | 7.1 |
| | (34, 399) | 1.0×10^{-2} | 1.8 | 4.0 | 6.8 | 3.7 |

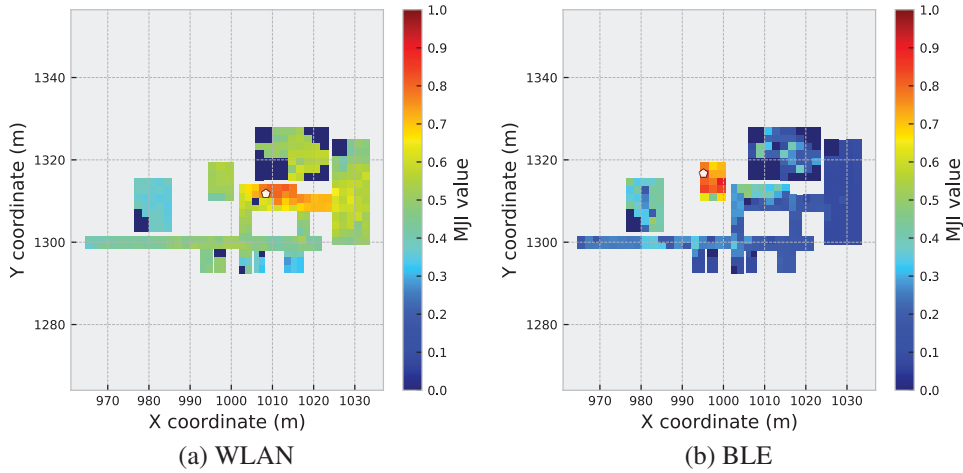


Figure 3.10: Examples resulting in small subregion selection error. In the figure, the white color filled pentagon indicates the ground truth of the TPs.

3.10) shows that this occurs mostly in cases where the gradient of the MJI within an extended neighborhood of the test point is small. In that case a few or individual additional or missing features can significantly influence the ranking of the subregions in terms of MJI. We observe and need to expect this case in larger, unobstructed areas like a hall without large furniture where there is little variability in the IDs of the features observed (e.g., the visible APs in case of WLAN RSS). This characteristic suggests that an augmented scheme of selecting the subregions is required for future work, e.g. selecting an adaptive number of subregions according to the spatial gradient of the MJI, dividing the subregions of varying size, or including the feature values for subregion selection in areas with little variability.

We present the positioning results of MAP and k NN using WLAN and BLE signals as the features in Table 3.9 and Table 3.10, respectively. We conclude from the results that the proposed subregion and feature selections are beneficial for the positioning with respect to constraining the online positioning by reducing the processing time and increasing the positioning accuracy. In fact, the circular errors CE50, CE75, and CE90 as defined above get smaller and the percentage of errors larger than 10 m reduces.

3.5 Conclusion

We proposed herein an approach to fingerprinting-based indoor positioning using opportunistically measured WLAN and BLE signals as the features for coordinate estimation. The main contributions are proposals to reduce data storage requirements and computational complexity in terms of processing

Table 3.9: Comparison of positioning times and accuracies of RoI 2 using WLAN signal as the features ($h = -1$ has the same meaning as in Table 3.8.)

| Methods | Values of (m, h) | Mean positioning time (s) | CE50 (m) | CE75 (m) | CE90 (m) | Ratio of large errors (%) |
|--------------------------------------|-------------------------|------------------------------|----------|----------|----------|------------------------------|
| MAP (without interpolation) | all | 17.9 | 3.3 | 6.5 | 9.7 | 9.3 |
| k NN (without interpolation) | all | 0.1 | 1.6 | 3.1 | 4.8 | 1.7 |
| MAP (with interpolation) | all | 337.5 | 2.6 | 5.6 | 8.6 | 7.4 |
| k NN (with interpolation) | all | 0.30 | 1.4 | 2.6 | 4.6 | 1.9 |
| MAP (LASSO) | (11, -1) | 3.8 | 3.1 | 5.6 | 8.7 | 7.4 |
| | (16, -1) | 7.0 | 3.3 | 5.9 | 8.7 | 6.5 |
| | (21, -1) | 10.0 | 3.4 | 6.4 | 9.2 | 8.0 |
| | (34, -1) | 20.3 | 3.2 | 6.4 | 9.4 | 8.6 |
| | (34, 490) | 43.7 | 2.6 | 5.1 | 8.1 | 6.1 |
| k NN (LASSO) | (11, -1) | 0.6×10^{-2} | 2.3 | 4.2 | 7.2 | 4.2 |
| | (16, -1) | 0.8×10^{-2} | 2.3 | 4.3 | 6.8 | 4.0 |
| | (21, -1) | 1.0×10^{-2} | 2.2 | 4.1 | 6.9 | 4.6 |
| | (34, -1) | 1.6×10^{-2} | 2.3 | 4.2 | 6.6 | 4.4 |
| | (34, 490) | 1.3×10^{-2} | 1.3 | 2.6 | 4.7 | 1.9 |
| MAP (Forward greedy search) | (11, -1) | 3.5 | 2.4 | 4.4 | 7.5 | 4.6 |
| | (16, -1) | 6.2 | 2.4 | 4.7 | 7.7 | 5.0 |
| | (21, -1) | 9.2 | 2.5 | 4.7 | 7.6 | 4.4 |
| | (34, -1) | 17.7 | 2.7 | 5.4 | 8.6 | 6.5 |
| | (34, 490) | 43.0 | 2.7 | 5.1 | 8.1 | 6.1 |
| k NN (Forward greedy search) | (11, -1) | 0.4×10^{-2} | 2.5 | 4.3 | 7.0 | 3.4 |
| | (16, -1) | 0.5×10^{-2} | 2.3 | 4.5 | 6.9 | 5.5 |
| | (21, -1) | 0.6×10^{-2} | 2.2 | 4.3 | 7.4 | 5.7 |
| | (34, -1) | 0.8×10^{-2} | 2.0 | 3.9 | 6.3 | 3.8 |
| | (34, 490) | 1.3×10^{-2} | 1.3 | 2.6 | 4.3 | 2.1 |
| MAP (AFBGS) | (11, -1) | 3.4 | 2.6 | 4.6 | 7.4 | 5.0 |
| | (16, -1) | 6.4 | 2.6 | 5.2 | 7.9 | 5.9 |
| | (21, -1) | 10.3 | 2.7 | 5.1 | 7.9 | 5.7 |
| | (34, -1) | 22.5 | 2.8 | 5.2 | 8.4 | 5.5 |
| | (34, 490) | 46.2 | 2.7 | 5.0 | 8.0 | 5.9 |
| k NN (AFBGS) | (11, -1) | 0.6×10^{-2} | 2.5 | 4.1 | 6.8 | 3.6 |
| | (16, -1) | 0.7×10^{-2} | 2.5 | 4.5 | 7.2 | 4.2 |
| | (21, -1) | 0.8×10^{-2} | 2.4 | 4.3 | 6.8 | 3.6 |
| | (34, -1) | 1.2×10^{-2} | 2.0 | 4.0 | 6.4 | 2.5 |
| | (34, 490) | 2.3×10^{-2} | 1.3 | 2.6 | 4.5 | 2.1 |

Table 3.10: Comparison of positioning times and accuracies of RoI 2 using BLE signal as the features ($h = -1$ has the same meaning as in Table 3.8.)

| Methods | Values of (m, h) | Mean positioning time (s) | CE50 (m) | CE75 (m) | CE90 (m) | Ratio of large errors (%) |
|--------------------------------------|-------------------------|------------------------------|----------|----------|----------|------------------------------|
| MAP (without interpolation) | all | 5.9 | 5.2 | 9.7 | 17.1 | 23.6 |
| k NN (without interpolation) | all | 2.6×10^{-2} | 3.4 | 6.5 | 9.8 | 9.5 |
| MAP (with interpolation) | all | 197.0 | 4.7 | 8.6 | 12.3 | 13.7 |
| k NN (with interpolation) | all | 0.1 | 3.4 | 6.9 | 11.3 | 12.2 |
| MAP (LASSO) | (11, -1) | 2.0 | 3.8 | 7.8 | 11.0 | 13.1 |
| | (16, -1) | 3.1 | 4.0 | 8.0 | 11.3 | 12.9 |
| | (21, -1) | 4.4 | 4.2 | 8.0 | 11.7 | 14.4 |
| | (34, -1) | 8.0 | 4.2 | 8.1 | 12.5 | 15.3 |
| | (34, 278) | 23.5 | 4.3 | 8.4 | 13.4 | 15.8 |
| k NN (LASSO) | (11, -1) | 0.4×10^{-2} | 3.3 | 6.0 | 10.2 | 10.9 |
| | (16, -1) | 0.5×10^{-2} | 3.2 | 6.6 | 10.3 | 11.2 |
| | (21, -1) | 0.6×10^{-2} | 3.3 | 6.3 | 10.4 | 11.2 |
| | (34, -1) | 0.9×10^{-2} | 3.3 | 6.5 | 10.5 | 11.4 |
| | (34, 278) | 0.9×10^{-2} | 3.4 | 6.6 | 10.4 | 11.7 |
| MAP (Forward greedy search) | (11, -1) | 0.9 | 3.6 | 7.5 | 10.5 | 12.2 |
| | (16, -1) | 1.4 | 3.9 | 7.8 | 11.2 | 12.7 |
| | (21, -1) | 2.1 | 4.1 | 8.1 | 11.5 | 12.9 |
| | (34, -1) | 3.9 | 4.0 | 8.2 | 12.6 | 15.1 |
| | (34, 278) | 23.5 | 3.9 | 8.3 | 12.6 | 15.1 |
| k NN (Forward greedy search) | (11, -1) | 0.2×10^{-2} | 4.1 | 7.4 | 11.3 | 12.9 |
| | (16, -1) | 0.3×10^{-2} | 4.4 | 7.6 | 12.4 | 15.6 |
| | (21, -1) | 0.3×10^{-2} | 4.4 | 8.1 | 12.6 | 18.0 |
| | (34, -1) | 0.4×10^{-2} | 4.7 | 8.8 | 13.5 | 20.9 |
| | (34, 278) | 0.9×10^{-2} | 3.3 | 6.4 | 10.7 | 11.2 |
| MAP (AFBGS) | (11, -1) | 1.0 | 3.6 | 7.4 | 10.9 | 13.1 |
| | (16, -1) | 1.7 | 3.8 | 7.8 | 11.7 | 13.6 |
| | (21, -1) | 2.5 | 3.9 | 8.0 | 11.9 | 14.1 |
| | (34, -1) | 5.1 | 4.0 | 8.3 | 12.8 | 15.6 |
| | (34, 278) | 23.5 | 4.3 | 8.3 | 12.9 | 15.1 |
| k NN (AFBGS) | (11, -1) | 0.2×10^{-2} | 4.1 | 7.3 | 11.6 | 14.4 |
| | (16, -1) | 0.3×10^{-2} | 4.3 | 7.9 | 12.2 | 17.0 |
| | (21, -1) | 0.3×10^{-2} | 4.7 | 8.7 | 12.5 | 19.2 |
| | (34, -1) | 0.4×10^{-2} | 4.8 | 8.6 | 12.6 | 18.5 |
| | (34, 278) | 0.9×10^{-2} | 3.3 | 6.6 | 10.2 | 10.9 |

time by segmentation of the entire region of interest (RoI) into subregions, identification of a few candidate subregions during the online positioning stage, and use of a selected subset of relevant features instead of all available features for position estimation. Subregion selection is based on a modified Jaccard index quantifying the similarity between the features obtained by the user and those available within the RFM. Feature selection is based on an adaptive forward-backward greedy search yielding a pre-computed set of relevant features for each subregion. The reduction of computational complexity is obtained both from the reduction of the number of candidate locations needed to analyze during online positioning and from the reduction of the number of features to be compared.

The experimental results corroborated the claim of reduced complexity while indicating that the positioning accuracy is hardly reduced by subregion and feature selection for the small RoI and even improves for the large RoI. For both investigated RoIs, the time required for the position estimation in the online stage was reduced by a factor of about 10 when using the selected relevant features within 11 selected subregions instead of using all features and searching over the entire RoI. For the small RoI (i. e. RoI 1), the increment of the 90th percentile errors (CE90) is 7.5% (i. e. 8.6 m instead of 8.0 m). In the large RoI (i. e. RoI 2), the positioning accuracy using MAP reduces from 9.8 m to 7.4 m for WLAN signals and from 12.3 m to 10.9 m for BLE. For k NN, the positioning accuracy does not change with subregion and feature selection. Given a fixed number of candidate subregions and a fixed, low number of features the computational burden of the entire algorithm is almost independent of the size of the entire RoI and of the number of available features across the RoI.

Future research will concentrate on investigating the role of subregion definition (shape, orientation, homogeneity) and possible benefit of optimization, on taking into account a user motion model during subregion selection, on handling temporal changes of the reference fingerprinting map, and on fully integrating different types of features for improving the positioning accuracy.

Acknowledgment

The China Scholarship Council (CSC) financially supports the first author's doctoral research. We thank to anonymous reviewers for their useful comments and questions which helped significantly improving the paper.

CHAPTER 4

ON ADAPTING TO MEASURABILITY VARIATIONS

This chapter corresponds to:

Zhou, Caifa, and Andreas Wieser. 2018. CDM: Compound Dissimilarity Measure and an Application to Fingerprinting-Based Positioning. In Indoor Positioning and Indoor Navigation (IPIN), 2018 International Conference On, 1–7. Nantes, France: IEEE.

except for formatting and the relocation of the bibliography to the end of the thesis. This is the Author Version of the paper. For the Typeset Version please refer to the original conference paper.

CDM: Compound Dissimilarity Measure and an Application to Fingerprinting-Based Positioning

Caifa Zhou, Andreas Wieser

Abstract

A non-vector-based dissimilarity measure is proposed by combining vector-based distance metrics and set operations. This proposed compound dissimilarity measure (CDM) is applicable to quantify similarity of collections of attribute/feature pairs where not all attributes are present in all collections. This is a typical challenge in the context of e. g. fingerprinting-based positioning (FbP). Compared to vector-based distance metrics (e. g. Minkowski), the merits of the proposed CDM are i) the data do not need to be converted to vectors of equal dimension, ii) shared and unshared attributes can be weighted differently within the assessment, and iii) additional degrees of freedom within the measure allow to adapt its properties to application needs in a data-driven way.

We indicate the validity of the proposed CDM by demonstrating the improvements of the positioning performance of fingerprinting-based WLAN indoor positioning using four different datasets, three of them publicly available. When processing these datasets using CDM instead of conventional distance metrics the accuracy of identifying buildings and floors improves by about 5% on average. The 2d positioning errors in terms of root mean squared error (RMSE) are reduced by a factor of two, and the percentage of position solutions with less than 2m error improves by over 10%.

4.1 Introduction

The core idea of this paper comes from analyzing a particular challenge occurring during the online step of a fingerprinting-based indoor positioning system (e. g. using the received signal strength (RSS) from WLAN access points (APs) as the features) based on the nearness in the fingerprint space as the principle for localization (e. g. k NN). Each measured fingerprint consists of a collection of actually observed attributes (e. g. identifications of APs and corresponding signal strengths). Fingerprints measured at different lo-

cations or at different times may contain different numbers of measurable APs e.g. due to changed availability of APs or changed signal reception conditions. In such cases it is not clear how the similarity/dissimilarity between such fingerprints should be assessed, and in particular, how the similarity/dissimilarity between a fingerprint measured online (i.e. when the user position is to be determined) and the fingerprints collected in the RFM¹ should be handled.

In a general context this belongs to the class of missing data problems (Efron 1994) mostly addressed in the fields of data analysis, data mining and machine learning (Little 2015). A comprehensive review of the missing data problem is out-of-scope of this paper. Instead we focus on a concrete proposal to handle this problem within the context of positioning. In previous publications the authors either formulated the online measurements into vectors of equal length (Padmanabhan et al. 2000; He and Chan 2016; He et al. 2017) or used only the measurability of the individual attributes as binary features (Machaj et al. 2011). The former scheme requires filling in values for missing attributes and ignoring newly measured ones. In this way it is easy to apply vector-based distance metrics for computing the dissimilarity but there are two disadvantages. One is the limited flexibility in dealing with missing or newly available data, the other one is time and computational resource cost: in most cases the number of all APs contained in the RFM is much larger than the number of APs in an individual measured fingerprint. Therefore the vectorized data of uniform dimension which need to be composed and handled typically have many more elements than the individual measurements. The approach mapping the measured APs into a set of binary features, instead, is efficient in the sense of computational burden for assessing dissimilarity but it does not take the actual similarity of the measured values into account and thus does not support exploiting the potential for accurate positioning.

After analyzing how this case is handled in previous publications, we explore the possibility of estimating the dissimilarity between the measurements which have the characteristics of partially missing observations of the attributes without formulating them into vectors of equal length. To this end we propose a non-vector-based dissimilarity measure (Section 4.3) which is a compound of a typical distance metric (e. g. Minkowski) and set operations. In addition, we exploit the applicability of the proposed compound dissimilarity measure (CDM) by applying it to four datasets used for FbP and the result proves the benefits of the proposed dissimilarity measure (Section 4.4).

¹A RFM is a collection of fingerprints with labeled ground locations for representing the functional relationship between the location and fingerprint.

4.2 Related work of distance metrics

The concept of distance metrics used for measuring the nearness between the online measured features and the ones stored in the RFM is a key for the realization of FbP algorithms. The Euclidean distance is one of the most prevalent metrics in different research fields and communities (Cha 2007). However, there is a variety of alternative distance metrics which may be more suitable for certain applications. In (Cha 2007), Cha reported over 40 distance metrics or measures and analyzed their capability of measuring the difference between probability density functions (PDFs). Minaev et. al. (Minaev et al. 2017) followed Cha’s research and applied them to an FbP algorithm k NN by using the synthetic RSS from WLAN APs as the fingerprints and found that Lorentzian distance performs best among them. In (Torres-Sospedra et al. 2015a), Torres-Sospedra et. al. surveyed and analyzed the performance of different distance metrics by applying them to a fingerprinting-based WLAN indoor positioning system which covers multi-buildings and multi-floors (i. e. *UJIIndoorLoc* dataset). In this paper, we propose the concept of CDM joining it with the 8 distance metrics performing best according to (Minaev et al. 2017), and apply the CDM fingerprinting-based WLAN indoor positioning to using four different datasets.

4.3 Compound dissimilarity measure

Suppose that there are several collections of measurements which need to be compared but differ with respect to the number and type of data included. For instance, the measurements in a WLAN-based indoor positioning system consist of the RSSs from all available APs at individual locations. However, only APs associated with RSS values exceeding the measuring sensitivity of the used WiFi device are observable at the individual locations, thus the APs measured at different points in space time will differ. Each measurement consists of an attribute e. g. the media access control (MAC) address of the respective APs, and an RSS value. The fingerprint at a particular location is the collection of measurements actually made at that location. As stated above, we propose an approach herein to estimate the similarity or dissimilarity between such fingerprints without reformulating all the measurements with different attributes into vectors of equal length as in several other publications (He and Chan 2016; Mautz and Tilch 2011; Zhou and Wieser 2018b; Padmanabhan et al. 2000).

Given n measurements denoted as $\{\mathbb{O}^i\}_{i=1}^n$, each measurement consisting of a set of paired attribute and measured value, i. e. $\mathbb{O}^i := \{(a, v_a^i) | a \in \mathbb{A}^i, v_a^i \in$

$\mathcal{R}\}, i = 1, \dots, n$, where $\mathbb{A}^i \subseteq \mathcal{M}$ is the set of the attributes of the i^{th} measurement. The initial idea of measuring the dissimilarity between \mathbb{O}^i and \mathbb{O}^j is by splitting them into three parts (Figure 4.1), namely computing and weighting the dissimilarity of the shared and unshared attributes differently:

$$d_{CDM}(\mathbb{O}^i, \mathbb{O}^j) = \sum_{a \in (\mathbb{A}^i \cap \mathbb{A}^j)} g(\mathbf{v}_a^i, \mathbf{v}_a^j) + \alpha \left(\sum_{a \in (\mathbb{A}^i \setminus \mathbb{A}^j)} g(\mathbf{v}_a^i, \gamma) + \sum_{a \in (\mathbb{A}^j \setminus \mathbb{A}^i)} g(\mathbf{v}_a^j, \gamma) \right), \quad (4.1)$$

where $g(\cdot, \cdot)$ is a chosen distance metric, γ is a predefined value indicating a missing measured attribute and the regularization values $\alpha \in [0, +\infty)$ are introduced to regulate or balance the contribution to the dissimilarity from those mutually unshared attributes². The CDM offers additional degrees of freedom owing to the contribution of the hyperparameters (i. e. γ and α). Their values can be determined in a data driven approach according to the specific application.

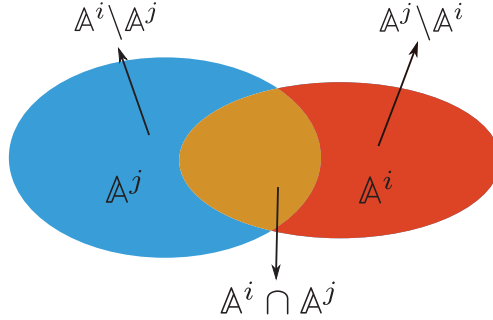


Figure 4.1: The scheme of calculating the dissimilarity measure into three parts

The basic CDM formulated in (4.1) weights the contribution from the shared and unshared attributes differently. However, we additionally propose two further variants of this measure which take also the actual numbers of these attributes into account. The application examples will later indicate that these are useful extensions. One is obtained by dividing the CDM in (4.1) by the total number of attributes thus yielding the average dissimilarity of the attributes, where $|\cdot|$ denotes the cardinality of a set, here:³

²Herein we regulate the contribution of unshared attributes equally due to there is no prior assumption that can be used to determine which of them should have more influence on the dissimilarity. In a specific application (e.g., feature-based positioning), it might be reasonable to weight this two terms differently.

³The same symbol is used in this contribution also to indicate the absolute value of a scalar.

$$d_{ACDM}(\mathbb{O}^i, \mathbb{O}^j) = d_{CDM}(\mathbb{O}^i, \mathbb{O}^j) / |\mathbb{A}_i \cup \mathbb{A}_j|. \quad (4.2)$$

We call this measure an averagely weighted compound dissimilarity measure (ACDM). The second extension is obtained by weighting the terms in (4.1) relatively according to the number of shared and unshared attributes, i. e.

$$d_{RCDM}(\mathbb{O}^i, \mathbb{O}^j) = \sum_{a \in (\mathbb{A}^i \cap \mathbb{A}^j)} g(\mathbf{v}_a^i, \mathbf{v}_a^j) + \alpha \left(\omega_{\mathbb{A}_i \setminus \mathbb{A}_j} \sum_{a \in (\mathbb{A}^i \setminus \mathbb{A}^j)} g(\mathbf{v}_a^i, \gamma) + \omega_{\mathbb{A}_j \setminus \mathbb{A}_i} \sum_{a \in (\mathbb{A}^j \setminus \mathbb{A}^i)} g(\mathbf{v}_a^j, \gamma) \right) \quad (4.3)$$

where $\omega_{\mathbb{A}_i \setminus \mathbb{A}_j}$ and $\omega_{\mathbb{A}_j \setminus \mathbb{A}_i}$ are calculated by

$$\begin{aligned} \omega_{\mathbb{A}_i \setminus \mathbb{A}_j} &= \frac{|\mathbb{A}_i \setminus \mathbb{A}_j|}{(|\mathbb{A}_i \cap \mathbb{A}_j| + \varepsilon)} \\ \omega_{\mathbb{A}_j \setminus \mathbb{A}_i} &= \frac{|\mathbb{A}_j \setminus \mathbb{A}_i|}{(|\mathbb{A}_i \cap \mathbb{A}_j| + \varepsilon)} \end{aligned} \quad (4.4)$$

In (4.4), $\mathcal{R} \ni \varepsilon > 0$ but $\varepsilon \ll 1$ is introduced for avoiding division by zero in case there are no shared attributes at all. The relatively weighted compound dissimilarity measure (RCDM) introduced in (4.3) yields a large dissimilarity value in such a case. Comparing to the widely used vector-based distance metrics, the CDMs have three advantages:

- The measurements do not have to be rearranged into vectors of equal length.
- CDMs can be used to balance the contributions to the dissimilarity from the shared and mutually unshared attributes.
- CDMs have hyperparameters and are capable of adapting to different data.

Subsequently, we compare the three proposed CDMs by applying them to FbP using different values of the hyperparameters and joining them with selected distance metrics (see Section 4.4).

⁴The formula and nomenclature are from (Cha 2007) except the formula of Jaccard distance. In all these equations \mathbf{x}_i and \mathbf{y}_i are the i -th element of the vectors \mathbf{x} and \mathbf{y} , respectively and d is the dimension of \mathbf{x} and \mathbf{y} .

⁵ $I(\cdot)$ is an indicator function and it yields 1 if and only if the condition is fulfilled.

⁶This formula is referred to (Minaev et al. 2017). γ is the indicator of a missing measured attribute. In case of CDM, Hamming and Jaccard distances are equivalent.

⁷In (Marques et al. 2012), City Block is named Manhattan distance.

Table 4.1: Distance metrics used herein

| Metric | Equation ⁴ |
|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Lorentzian | $g_{Lor}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^d \ln(1 + \mathbf{x}_i - \mathbf{y}_i)$ |
| Hamming ⁵ | $g_{Ham}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^d I(\mathbf{x}_i \neq \mathbf{y}_i) / d$ |
| Jaccard ⁶ | $g_{Jac}(\mathbf{x}, \mathbf{y}) = \frac{\sum_{i=1}^d I((\mathbf{x}_i \neq \mathbf{y}_i) \& [(\mathbf{x}_i \neq \gamma) (\mathbf{y}_i \neq \gamma)])}{\sum_{i=1}^d I((\mathbf{x}_i \neq \gamma) (\mathbf{y}_i \neq \gamma))}$ |
| Wave Hedges | $g_{WH}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^d (\mathbf{x}_i - \mathbf{y}_i / \max(\mathbf{x}_i , \mathbf{y}_i))$ |
| Canberra | $g_{Can}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^d (\mathbf{x}_i - \mathbf{y}_i / (\mathbf{x}_i + \mathbf{y}_i))$ |
| Clark | $g_{Cla}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^d (\mathbf{x}_i - \mathbf{y}_i / (\mathbf{x}_i + \mathbf{y}_i))^2}$ |
| City block ⁷ | $g_{CB}(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^d \mathbf{x}_i - \mathbf{y}_i $ |
| Minkowski | $g_{Min}(\mathbf{x}, \mathbf{y}, p) = \sqrt[p]{\sum_{i=1}^d \mathbf{x}_i - \mathbf{y}_i ^p}$ |

4.4 An application of CDMs to FbP

In this section, we first describe the fundamentals of FbP, the widely used positioning algorithm k NN, the chosen evaluation metrics, and four datasets used for practical application and assessment. We then evaluate the performance of the three CDM in terms of positioning results taking into account only very few different values α . Then the cross validation (CV) method is applied to search for particularly suitable values of α for a chosen distance metric and dataset. We compare the positioning performance of the approach using the CDM to that of k NN without CDM.

4.4.1 The baseline algorithm, performance criteria and data sets

Fingerprinting-based positioning. The measured fingerprints for the purpose of FbP have the characteristic of missing attributes because the coverage of an AP is restricted by the transmitting power, free space loss, signal attenuation and the sensitivity of the receiver. The coverage is higher for higher transmission power, higher sensitivity and lower attenuation. One benefit of using CDM instead of vector-based distance metrics is that it avoids the need for conversion of the measurements into vectors of equal length. Further experimental analysis in the consecutive sections shows that using CDM can

also improve the positioning accuracy and stability.

Generally an FbP (e. g. using the signal strength from WLAN APs as the fingerprints for indoor positioning) consists of two stages: offline fingerprinting stage and online positioning stage. During the offline stage, a RFM $\{(\mathbb{O}^i, \mathbf{I}^i)\}_{i=1}^N$ representing the relationship between the measurements (e. g. RSS) and locations is collected via carrying out a site survey (either by a professional surveyor or by crowd-sourcing). During online positioning stage a user measured observation \mathbb{O}^u is matched to the RFM using a FbP algorithm f for estimating the user's location \mathbf{I}^u (an estimated location is denoted as $\hat{\mathbf{I}}^u$), i. e. $f : \mathbb{O}^u \rightarrow \hat{\mathbf{I}}^u$.

Herein we use k NN, one of the most widely used FbP algorithms, as the baseline positioning method for evaluation and comparison. *UJIIndoor-Loc* (see Section 4.4.1) is a dataset including multiple buildings and multiple floors. We use the hierarchical k NN according to (Torres-Sospedra et al. 2014) for processing this dataset. For the other datasets we use k NN as follows:

- Computing the dissimilarity measure between the user's measurement and the ones stored in the RFM;
- Finding the k nearest reference points in the feature space, i. e. reference points with the k smallest dissimilarity;
- Taking the average coordinates of these k reference points as the user's location.

More details about FbP and k NN can be found, e. g. in (Zhou and Wieser 2018b; Padmanabhan et al. 2000).

Evaluation of positioning performance. The Euclidean distance between the estimated location $\hat{\mathbf{I}}$ and the ground truth location \mathbf{I} is used as the basic evaluation of the positioning error. In addition, we also use the statistical values (e. g. mean or standard deviation), root mean squared error (RMSE), and empirical cumulative distribution function (ECDF) with respect to the error distance as further performance evaluation.

The implementation of the proposed CDMs and their relevant functions are in Python and partially based on the scikit-learn package (Pedregosa et al. 2011)⁸.

Available positioning datasets. We use four different datasets (both the RFMs and validation datasets) from fingerprinting-based WLAN indoor positioning systems for evaluating and comparing the performance of the pro-

⁸<http://scikit-learn.org/dev/index.html>

posed CDMs. The datasets *Alcala2017*, *Tampere* and *UJIIndoorLoc* are available online⁹ (Sansano et al. 2016) and more details about them can be found in (Torres-Sospedra et al. 2015a, 2014). Another dataset *HIL* is described in (Zhou and Wieser 2018b). These four datasets represent different FbP scenarios with respect to area of indoor region, number of available APs, method of fingerprint collection, and device heterogeneity. The summarized characteristics of the datasets are given in Table 4.2.

Table 4.2: The characteristics of the datasets

| Dataset | # Buildings | # Floors | # APs | # Training samples | # Validation samples ¹⁰ |
|-----------------------------------|-------------|----------|-------|--------------------|------------------------------------|
| <i>Alcala2017</i> | 1 | 1 | 152 | 670 | 0 |
| <i>HIL</i> | 1 | 1 | 490 | 1525 | 509 |
| <i>Tampere</i> ¹¹ | 1 | 4 | 309 | 1478 | 0 |
| <i>UJIIndoorLoc</i> ¹² | 3 | 4-5 | 520 | 3818 | 1110 |

In case of applying the proposed CDMs to FbP, we use a fixed value of γ for indicating the missing attributes. In *HIL*, γ is set to -110 dBm and $\gamma = 100$ in other three datasets (Torres-Sospedra et al. 2014).

4.4.2 Evaluation and comparison of different CDMs

Herein we propose three versions to CDMs. However, we want to briefly investigate whether FbP is less sensitive with respect to dataset and distance metric included than the others. Since the results may also depend on α , we use a few fixed values i. e. $\alpha = \{0.5, 1.0, 1.25\}$ for the analysis and compare the results.

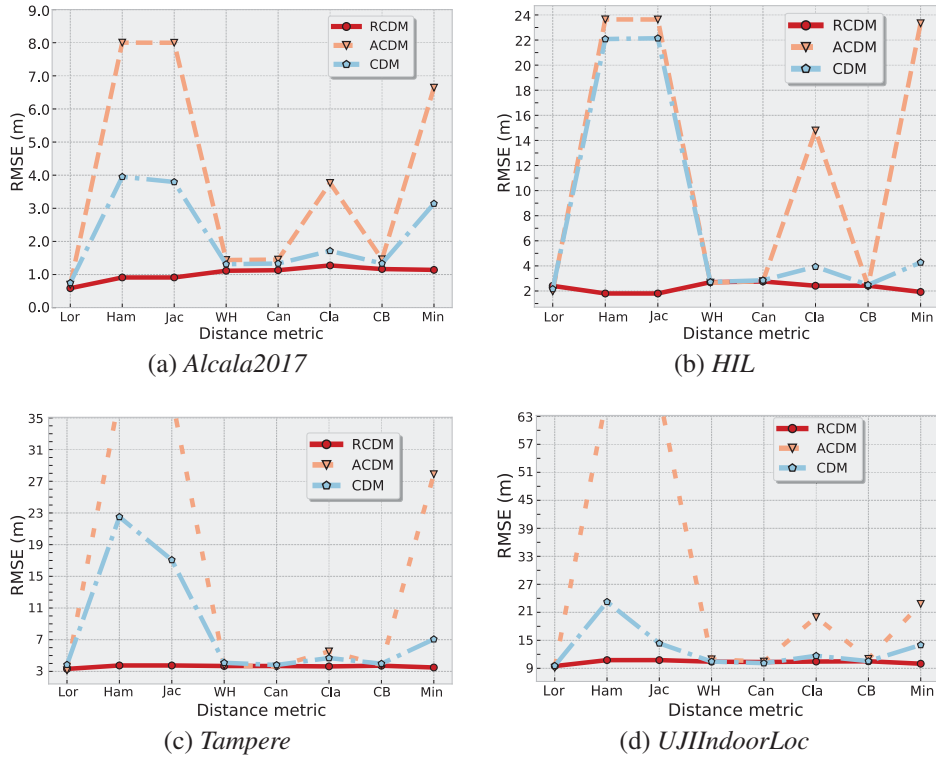
Figure 4.2 shows the RMSE of the positioning result using all three CDMs with $\alpha = 1.0$. Similar results are also obtained using the other two values of α . From Figure 4.2, we can conclude that the CDM relatively weighted by the shared and mutually unshared attributes performs more stable than that of other two CDMs in the sense that the RMSE of all four datasets of using RCDM compounding with all eight distance metrics has the smallest deviation. Therefore, we focus on the application of RCDM for the remainder of this paper.

⁹<http://indoorlocplatform.uji.es/>

¹⁰In case there is no provided validation samples, we randomly split the training samples into two datasets. 75% of them are used for training and the remaining ones are used for validation.

¹¹Herein we only use the data of one building from this dataset.

¹²The cleaning procedures are applied to this dataset, see .

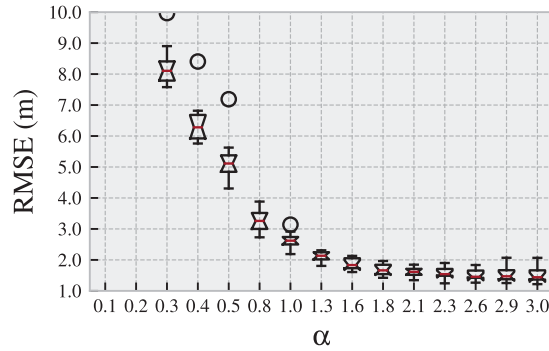
Figure 4.2: An example of RMSE of three CDMs with $\alpha = 1.0$

4.4.3 Tuning the regularization value α

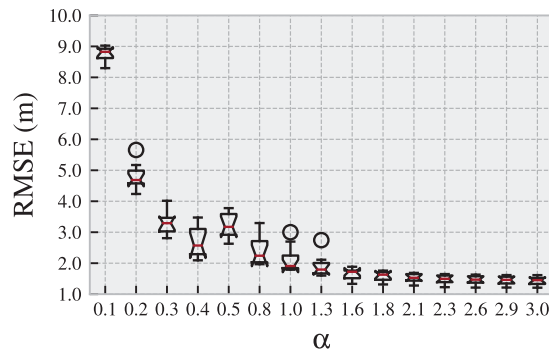
In order to find suitable values of α we carry out CV (Bishop 2006), a widely used method for model selection, for various distance metrics and the four datasets. CV can make full use of the training dataset by randomly splitting it into several folds and iteratively using one of them as the temporal test samples and the remaining ones as the temporal training dataset. Herein we use 10-fold CV and search for the suitable value of α in the range of $[0, 3]$ with the interval of 0.1¹³. In this paper, we only illustrate that a useful value of α can be found using CV. We leave a more thorough investigation of optimal values of α for future work. In the consecutive part of this section, we mainly show the results of *HIL* and *UJIIndoorLoc* using *kNN* ($k = 1$) as the FbP algorithm with RCDM (compounding with Lorentzian and Minkowski) for remaining the clarity. The results of other datasets and distance metrics are also similar to what presented herein.

As shown in Figure 4.3 and Figure 4.4, there is a value of α resulting in the minimum RMSE and maximum success rate for a chosen distance metric and dataset. Regarding dataset *Alcala2017*, *HIL*, and *Tampere*, we select

¹³In Figure 4.3 and Figure 4.4, we only plot part of the cross validation result for reserving the clarity.



(a) Lorentzian



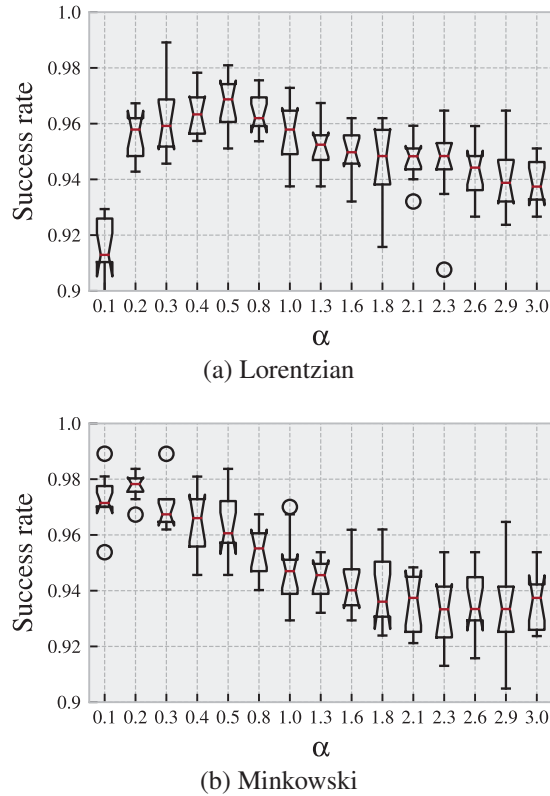
(b) Minkowski

Figure 4.3: CV boxplots of *HIL* ($k = 1$, RCDM)

the α which achieves minimum average value of RMSE of 10-fold CV as the suitable value for a chosen distance metric and dataset. For *UJIIndoorLoc*, we use the regularization value which obtains the maximum average success rate (Figure 4.4), defined as the percentage of correctly locating both the building and floor, as the proper value of α , since in case of applying FbP to multi-buildings and multi-floors, the success rate is a better indicator for the positioning performance than using RMSE (Marques et al. 2012). One reason using success rate instead of RMSE as the criterion is that the wrongly locating either the buildings or the floors introduces large positioning errors and it makes that the RMSE is no longer a good indicator of positioning performance. From the CV results shown in Figure 4.3 and Figure 4.4, the suitable values of α for *HIL* and *UJIIndoorLoc* are 2.7 and 3.0, and 0.5 and 0.2 in case of relatively compounding with both Lorentzian and Minkowski ($p = 2$) distances, respectively.

4.4.4 Comparison of positioning performance

We compare the RMSE of the positioning result obtained using RCDM (using the regularization value (α) found by CV) to the ones attained using vector-based distance metrics (Figure 4.5). As shown in Figure 4.5 and Figure 4.6b,

Figure 4.4: CV boxplots of *UJIIndoorLoc* ($k = 1$, RCDM)

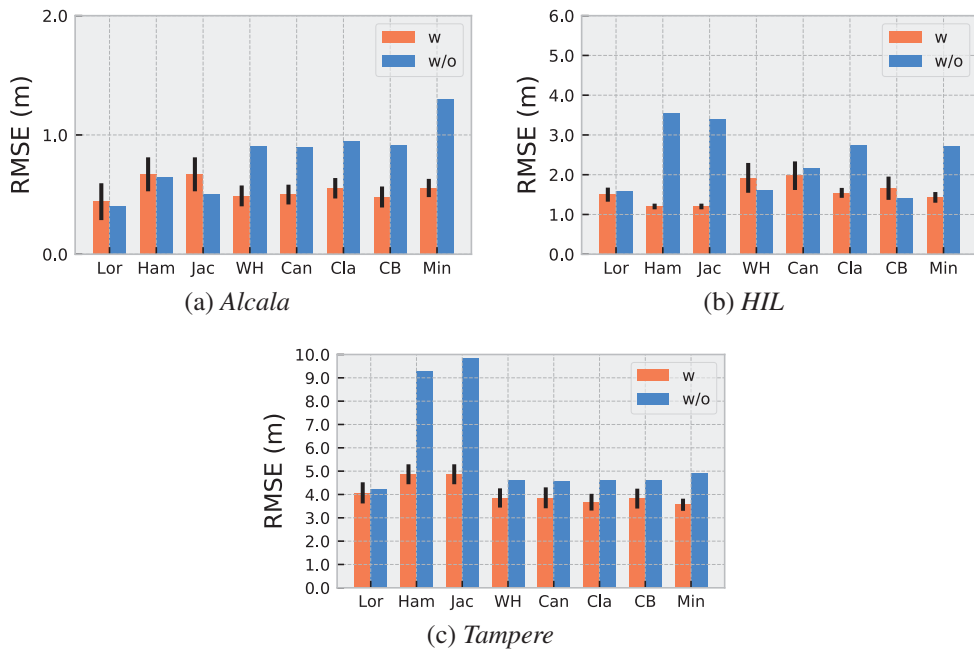
the proposed RCDM outperforms almost all eight original distance metrics on all four datasets (except in case of compounding with Hamming and Jaccard distances on *Alcala2017* (see Figure 4.5a) and Wave Hedges and city block distances on *HIL* (see Figure 4.5b)). In addition, the reduction of the RMSE is over two times comparing to that of without using RCDM and the deviation of the RMSE of compounding with all eight distance metrics is much smaller than that of the original ones. In Figure 4.6, the success rate of using RCDM is higher than that of using the original distance metrics and the improvement is up to 13% (Figure 4.6a).

According to the comparison of the ECDF of all eight original distance metrics and RCDM of dataset *HIL* (Figure 4.7), we can conclude that the cumulative positioning accuracy using RCDM is higher than that of using original distance metrics. In addition, the maximum positioning error distance using RCDM is much smaller than that without using RCDM. The FbP algorithms achieving a small maximum positioning error distance make it easier to constrain the upper bound of the positioning error.

In Table 4.3, we present the positioning results of applying the proposed approach to *UJIIndoorLoc*. From the building accuracy, defined as the percentage of correctly identifying the building, it seems that there are some validation samples which are not placed in the correct building by the positioning

Table 4.3: Positioning results of *UJIIndoorLoc*

| Metrics | | Lor | Ham | Jac | WH | Can | Cla | CB | Min |
|-------------------------------------|-----|-------|-------|-------|-------|-------|-------|-------|-------|
| Building accuracy (%) | w/o | 99.76 | 96.65 | 99.67 | 99.92 | 99.92 | 99.92 | 99.92 | 99.84 |
| | w | 99.92 | 99.92 | 99.92 | 99.92 | 99.92 | 99.92 | 99.92 | 99.92 |
| Success rate (%) | w/o | 94.12 | 80.98 | 85.55 | 92.9 | 93.47 | 92.73 | 92.57 | 91.1 |
| | w | 96.33 | 93.71 | 93.71 | 97.22 | 96.73 | 96.24 | 97.47 | 96.98 |
| Median of error distance (m) | w/o | 1.78 | 2.04 | 1.73 | 2.76 | 2.48 | 2.65 | 2.89 | 3.92 |
| | w | 1.44 | 2.07 | 2.07 | 1.48 | 1.53 | 1.64 | 1.38 | 1.47 |
| 80-percentile of error distance (m) | w/o | 9.35 | 14.72 | 14.76 | 11.06 | 10.65 | 10.65 | 11.19 | 13.76 |
| | w | 8.36 | 11.58 | 11.58 | 8.12 | 8.34 | 8.79 | 7.96 | 8.11 |

Figure 4.5: Comparison of RMSE ($k = 1$, RCDM). In the figure, w and w/o denotes the ones with RCDM and without RCDM, respectively.

algorithm, because the building accuracy of RCDM compounding with different distance metrics keeps the same, i. e. the building accuracy is saturated. This saturation might be caused by the cleaning of the dataset (see). Regarding the success rate, it improves about 5% on average using the RCDM and is over 97% of correctly identifying both the building and floor. In addition, the median and 80-percentile positioning error distances using RCDM reduce obviously except compounding with Hamming and Jaccard distances.

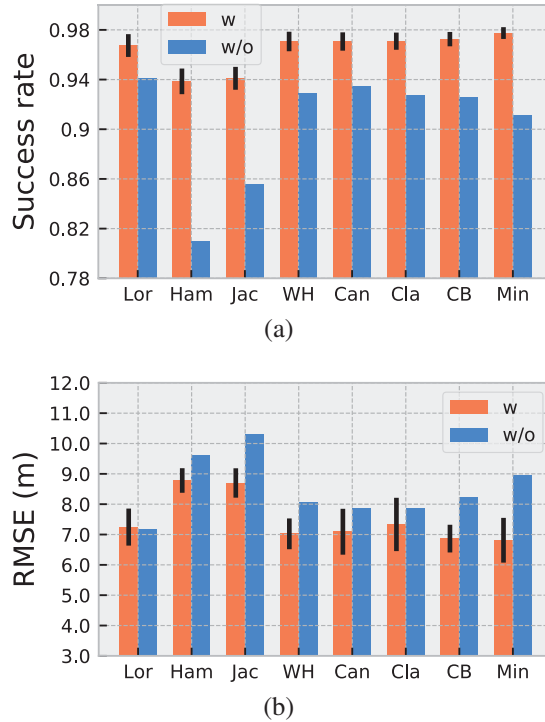


Figure 4.6: Comparison of success rate and RMSE of *UJIIndoorLoc* ($k = 1$)

4.5 Conclusion

We propose a non-vector-based dissimilarity measure, named compound dissimilarity measure (CDM), by combining a typical distance metric with set operations for the purpose of measuring the dissimilarity between measurements despite the possibility of missing attributes. The proposed CDM is flexible because it includes hyperparameters, which can be tuned according to the data and needs of the application. We apply the proposed CDM to four datasets collected in fingerprinting-based WLAN indoor positioning systems and the positioning performance verifies the validity of it. Both the accuracy of identifying buildings and floors, and the specific locations improve obviously, which are over 5% and 10%, respectively. Although the CDM is proposed herein starting from the idea of handling missing data in feature-based positioning (FbP), it is applicable to other missing data problems as well (e. g. searching the correspondences of point clouds according to sparsely described local features).

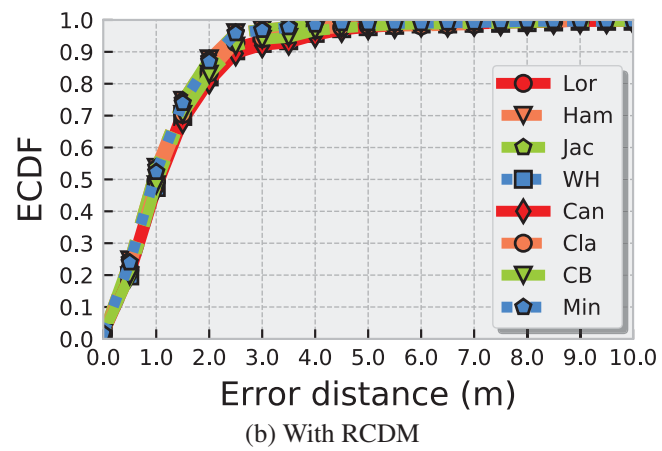
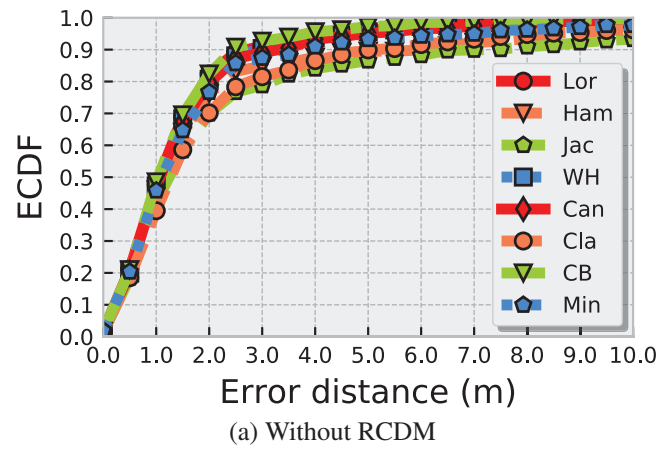


Figure 4.7: Comparison of ECDF between different distance metrics (*HIL*, $k = 1$)

Acknowledgment

The China Scholarship Council (CSC) financially supports the first author's doctoral research.

Appendix

We clean the *UJIIndoorLoc* dataset from two aspects: i) the invalid samples, and ii) the replicas in the training dataset. An invalid sample is a measurement that all of the APs are filled with missing values. A replica of the measurement is that at least two measurements are measured at the same location by the same user using the same device in a short time range (e. g. less than 5 minutes).

- Invalid samples: We find that 76 out of 20013 samples in the training dataset are invalid measurements by checking whether the RSS of all APs of a measurement is indicated by a missing value (e. g. 100 used in *UJIIndoorLoc*). We thus delete them from the training dataset.
- Replicas: These replicas are highly correlated and they might cause the failure of the cross validation using the training dataset because it is easy to get an over-optimistic results using a dataset containing replicas for cross validation (Bishop 2006). This makes the parameter found by cross validation not applicable to another test dataset. We find out there are a lot of replicas in the training dataset (only 3818 out of 19937 reference measurements do not have a replica). We thus randomly sample one of those replicas as the reference fingerprint for the training dataset in our experimental analysis ¹⁴.

¹⁴Herein we use only one of the replicas as the reference fingerprint, however, it is useful that grouping or averaging those replicas as one reference fingerprint.

CHAPTER 5

ON MITIGATING LARGE ERRORS IN POSITIONING

This chapter corresponds to:

Zhou, Caifa, and Andreas Wieser. 2019. An Iterative Scheme for Feature-based Positioning using a Weighted Dissimilarity Measure. Sensors xx (xx). Multidisciplinary Digital Publishing Institute:1–15. (Submitted)

except for formatting and the relocation of the bibliography to the end of the thesis. This is the Author Version of the paper. For the Typeset Version please refer to the original journal paper.

An Iterative Scheme for Feature-based Positioning using a Weighted Dissimilarity Measure

Caifa Zhou, Andreas Wieser

Abstract

We propose an iterative scheme for feature-based positioning using a new weighted dissimilarity measure with the goal of reducing the impact of large errors among the measured or modeled features. The weights are computed from the location-dependent standard deviations of the features and stored as part of the reference fingerprint map (RFM). Spatial filtering and kernel smoothing of the kinematically collected raw data allow efficiently estimating the standard deviations during RFM generation. In the positioning stage, the weights control the contribution of each feature to the dissimilarity measure, which in turn quantifies the difference between the set of online measured features and the fingerprints stored in the RFM. Features with little variability contribute more to the estimated position than features with high variability. Iterations are necessary because the variability depends on the location, and the location is initially unknown when estimating the position. Using real WiFi signal strength data from extended test measurements with ground truth in an office building, we show that the standard deviations of these features vary considerably within the region of interest and are neither simple functions of the signal strength nor of the distances from the corresponding access points. This is the motivation to include the empirical standard deviations in the RFM. We then analyze the deviations of the estimated positions with and without the location-dependent weighting. In the present example the maximum radial positioning error from ground truth are reduced by 40% comparing to k NN without the weighted dissimilarity measure.

5.1 Introduction

Feature-based (i. e. fingerprinting-based) indoor positioning systems (FIPSs), one of the promising indoor positioning solutions, have been proposed using various types of features (e. g. WLAN/BLE signal strengths (Padmanabhan et al. 2000; Youssef and Agrawala 2008; Zhuang et al. 2016), geomagnetic

field strengths (He and Shin 2018) or visible patterns (Guan et al. 2016)) for providing indoor location-based services (LBSs) to pedestrians (Brena et al. 2017; He and Chan 2016; Pei et al. 2016). The positioning accuracy of the state-of-the-art FIPSs using the received signal strength (RSS) of WLAN access points (APs) is in the range of a few meters (Mautz 2012). This is adequate for pedestrian indoor positioning and navigation in many cases. However, unexpected and unacceptably large errors (e. g. > 20 m in horizontal coordinates (Torres-Sospedra et al. 2017)) can be observed in real environments. They jeopardize the practical usability of FIPSs (Wu et al. 2017; Torres-Sospedra and Moreira 2017b). Such large errors may be caused by large deviations of the measured or stored feature values when performing the location estimation (Kaemarungsi and Krishnamurthy 2012).

In order to benefit from the attractive characteristics of FIPS while mitigating large errors, the trend is to combine the feature-based positioning with other techniques. Such hybrid approaches combine the feature-based information with e. g. pedestrian dead reckoning (PDR) (Li et al. 2016), map matching (Wang et al. 2015, 2012) or infrared ranging (Bitew et al. 2015). In addition, Bayes filtering methods, such as Kalman filters or particle filters are used to improve the estimated trajectory of pedestrians by combining the measurements with assumptions on the user’s motion (Li et al. 2016; Röbesaat et al. 2017). Merging different positioning solutions may help mitigating the impact of large errors of individual observations on the quality of a specific type of SBSs. However, such approaches requires either deploying additional infrastructure or providing extra information (e. g. the indoor map). It would be useful to detect or mitigate large errors in FIPS using only intrinsically available data. This has attracted little research attention in the past, see e. g. (Wu et al. 2017; Torres-Sospedra and Moreira 2017b; Lemic et al. 2019), and is the motivation for the present contribution.

We base our approach on the variability of the feature values at each individual location. Feature values measured during the positioning stage are snapshots affected by noise. Even if the expected value of the feature has not changed since the data collection for the generation of the reference fingerprint map (RFM), the measured value may be closer to the RFM value at a different position than to the one at the correct position because of this noise. It is therefore important to take the noise into account when assessing the similarity of measured and stored feature values. We facilitate this by storing the empirical standard deviations (STDs) in the RFM which is generated during the offline phase for representing the relationship between locations and their associated features. The estimation of the variability is carried out by empirically analyzing the spatial distribution of the raw data (e. g. RSS values) included in the RFM. It yields an extended representation of the RFM, which contains not only the spatially smoothed feature values, but also the location-wise estimated STD of each individual feature (see Section 5.4). These values can then be used to mitigate the impact of large errors in FIPS. To this end

we propose a weighted dissimilarity measure, which quantifies the difference between the online measured features and the features stored in the RFM, by adapting the contribution of the individual features to the dissimilarity measure relative to their estimated STD values (see Section 5.5.1). The positioning process is carried out in an iterative way because we need to assume the user's location, which is required for retrieving the STD of the online measured features (see Section 5.5.2). Beyond the use further discussed in this paper, the location-dependent standard deviations can also be employed for identifying (large) changes of features which may need an update of the RFM, see e.g. (He and Chan 2016; Tao and Zhao 2018; He et al. 2016).

The remaining of the paper is organized as follows: Section 5.2 summarizes the work related to reducing large errors in an FIPS. The fundamentals of the feature-based positioning are briefly described in Section 5.3. The robust estimation of the variability of the RFM and its application to positioning are presented in Section 5.4 and 5.5, respectively. Finally, the evaluation of the variability estimation as well as the positioning performance using the iterative scheme are presented in Section 5.6 for a real world dataset.

5.2 Related work

Herein we focus on publications that address the detection and reduction of large errors in an FIPS. We refer the interested readers to (Mautz 2012; He and Chan 2016; Brena et al. 2017) for more general information about indoor positioning. A comprehensive comparison of different feature-based indoor positioning algorithms using various similarity/dissimilarity metrics is available in (Retscher and Joksch 2016; Torres-Sospedra et al. 2015a; Minaev et al. 2017). A short review of the methods used for generation or creation of the RFM can be found in e. g. He and Chan (2016); Zhou and Wieser (2019b).

Torres-Sospedra and Moreira (2017b) provides a detailed analysis of the sources of large errors when employing deterministic feature-based positioning approaches (e. g. k NN). The analysis is based on simulations for different indoor scenarios. The authors consider the influence of several factors such as the quantization error of signal acquisition, the density of the reference measurements, and the selected dissimilarity metrics on the positioning error. The analysis shows that large observation errors mostly occur at locations where both the mean and the maximum value of the RSS are low. However, the authors do not report about a validation of their analysis in a really deployed FIPSs. On a related note, Kaemarungsi and Krishnamurthy (2012) proposes to simply disregard features with a large standard deviation for the estimation of the user's position.

There are only few works that focus on reducing or estimating the posi-

tioning errors based on the analysis of the RFM¹. Wu et al. (2017) introduces a weighted dissimilarity measure by computing the discriminative indicator for each feature according to the Log-distance path loss model. However, the variability of the online measured features which has an impact on the estimation of the discriminative factor is not taken into account. In (Lemic et al. 2019) and (Li et al. 2019), the authors propose different regression models (e. g. neural networks, random forest, or Gaussian processes) for estimating the positioning errors and uncertainties that can be used to improve the performance of tracking a pedestrian’s trajectory. Even if this is not the focus of these papers, the results suggest that the regression-based error prediction models cannot help to mitigate *large* errors because the predicted errors have a large uncertainty.

Compared to previous publications, we carry out the variability analysis of the RFM using a kinematically collected dataset, which includes not only the noise originating from the short term fluctuations of the features measured by a mobile device, but also the noise introduced by the motion status (e. g. moving speed and headings) of the mobile device. This setup is closer to the realistic situation of positioning and tracking pedestrians. The estimation of the variability is based solely on the raw RFM and is later used for reducing large errors by introducing an iterative scheme with the weighted dissimilarity measure in the online positioning phase.

5.3 Feature-based positioning

We start this section by introducing the fundamental concepts of feature-based positioning and then briefly describe the process of kinematically collecting the RFM.

5.3.1 Fundamental concepts

Each measured feature is uniquely identifiable and has a measured value. For example, the signal from an AP, can be identified by its media access control address and is associated with an RSS. Features are thus formulated as pairs of attribute a and value v , i. e. (a, v) . A measurement (i. e. fingerprint) \mathbb{O}_i^u taken by the user u at the location/time i consists of a set of measured features, i. e. $\mathbb{O}_i^u := \{(a_{ik}^u, v_{ik}^u) | a_{ik}^u \in \mathbb{A}; v_{ik}^u \in \mathbb{R}; k \in \{1, 2, \dots, N_i^u\}\}$, where \mathbb{A} is the complete set of the identifiers of all available features and N_i^u ($N_i^u = |\mathbb{O}_i^u|$)

¹Torres-Sospedra and Moreira (2017b) provides a complete discussion of the works focusing on reducing large positioning errors by support of other technologies (e. g. PDR, or Bayes filtering).

is the number of features observed by the user u at i . The set of attributes of \mathbb{O}_i^u is defined as $\mathbb{A}_i^u := \{a_{ik}^u | \exists (a_{ik}^u, v_{ik}) \in \mathbb{O}_i^u\}$ ($\mathbb{A}_i^u \subseteq \mathbb{A}$). The positioning process consists of inferring the estimated user location $\hat{\mathbf{I}}_i^u = f(\mathbb{O}_i^u)$ ($\hat{\mathbf{I}}_i^u \in \mathbb{R}^d$) as a function of the measurement and the RFM \mathcal{F} , where f is a suitable mapping algorithm from the measurement to location². \mathcal{F} represents the relationship between the location \mathbf{l} and the measurement \mathbb{O} , i. e. $\mathcal{F} : \mathbf{l} \mapsto \mathbb{O} | \mathbf{l} \in \mathbb{G}$ throughout the region of interest (RoI) \mathbb{G} . If the RFM is discretely represented, we denote it as $\mathbb{F} := \{(\mathbf{I}_j, \tilde{\mathbb{O}}_j) | \mathbf{I}_j \in \mathbb{G}, j \in \{1, 2, \dots, |\mathbb{F}|\}\}$ (where $\tilde{\mathbb{O}}_j = \mathcal{F}(\mathbf{I}_j)$). A discrete RFM can be obtained e. g. by collecting fingerprints at different known or independently measured locations within the RoI \mathbb{G} .

5.3.2 Kinematically acquired RFM

The kinematically obtained dataset used as the basis for the RFM herein has already been employed in (Zhou and Wieser 2019b). It was acquired using a mobile device (Nexus 6P) whose ground truth location was continuously measured with mm- to cm-level accuracy by a total station tracking a mini prism mounted on top of the mobile device. This procedure enables to simultaneously obtain accurate reference coordinates and the fingerprinting data collected by a pedestrian. The measurements were obtained at arbitrary locations lying on the trajectory of a pedestrian because the data acquisition on the mobile phone is passively triggered by the status of measurable features (e. g. the arrival of new features or the change of feature values) (Schulz et al. 2018). By carrying out a thorough site-survey, all the collected measurements and their tracked trajectories were merged and used to generate the raw RFM. Herein we use this dataset as the basis of our analysis. More details of its acquisition and processing can be found in (Zhou and Wieser 2019b).

Figure 5.1a and 5.1b show examples of the raw data collected for RFM generation, namely the RSS values from two WLAN APs. These are signals of opportunity as the APs had been installed for providing Internet access and the signals are their anyway, when using them for the purpose of indoor positioning. The raw measurements have been acquired at arbitrary locations throughout the RoI which consists of several rooms and corridors within an office building.

² The RFM \mathcal{F} is omitted from the positioning algorithm f for simplicity. d (e. g. $d = 2$) is the dimension of the coordinates.

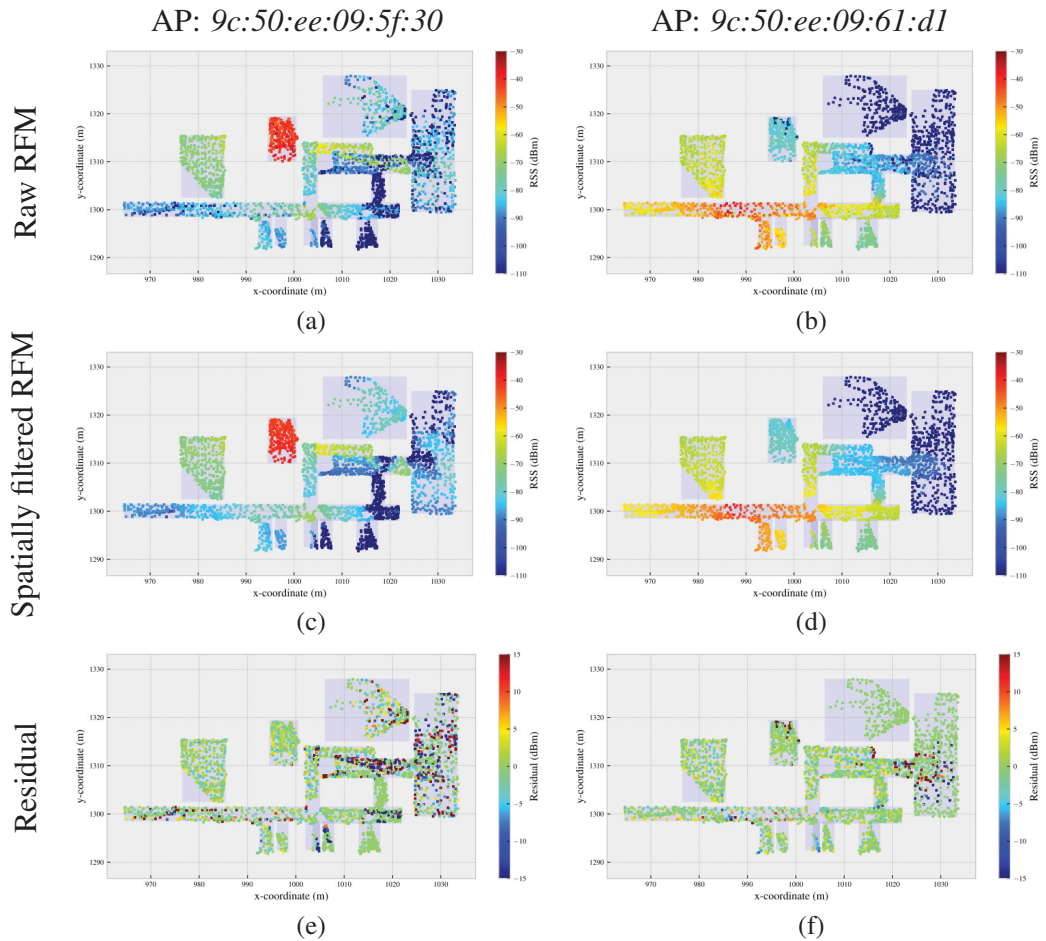


Figure 5.1: Examples of the raw and spatially filtered RFM of two arbitrarily selected APs. Third row shows residuals between the spatially filtered RFM and the raw RFM. The density of the reference locations over the RoI varies due to different accessibility (e. g. areas blocked by furniture or other facilities) and by different visiting frequency of the users.

5.4 Robust estimation of the feature variability

To estimate the noise of the measurable features at each location throughout the RoI, the features would have to be measured (ideally consecutively) multiple times at each location. However, even for a relatively sparse set of reference points throughout the RoI this would be prohibitively time-consuming and labor-intensive. We relax this requirement by assuming that the expected feature values change only little within a local, spatial neighborhood. Therefore, instead of estimating the standard deviation from the data collected only at a single location, we use all feature values obtained within a certain radius about a chosen reference location. The corresponding data are identified within the time series of data resulting while the user walked through the RoI.

We denote these fingerprints as *kinematically collected* ones. The estimation of the standard deviation is still possible if a sufficient number of measurements is obtained in the proximity of each reference location (see Figure 5.1). The measurements thus associated with an individual reference location contain data obtained consecutively within a short time at slightly different positions, but also data collected a certain time interval apart (e. g. half an hour) because the user passed most locations several times during the entire data collection process. The resulting standard deviations thus reflect also the temporal variability of the signals, and the impact of user motion during measurement, which will also apply during the positioning stage. We thus consider the kinematically collected RFM data suitable for the variability analysis.

Under the assumption that the expected value of each feature is locally obtainable, the location-wise STD of each feature can be approximated based on the measurements associated to the neighborhood of a given reference location. More formally, we estimate the STD σ_{jk} of k -th ($k = \{1, 2, \dots, |\tilde{\mathcal{O}}_j|\}$) feature at the reference location \mathbf{l}_j in the RFM \mathbb{F} . These estimated values of the STD are later included in the extended representation of the RFM, i. e. $\mathbb{F} := \{\mathbf{l}_j, \tilde{\mathbb{S}}_j\}$ with $\tilde{\mathbb{S}}_j = \{(a_{jk}, v_{jk}, \sigma_{jk}) | a_{jk} \in \tilde{\mathbb{A}}_j\}$. We start the estimation of the feature values for the RFM by applying a spatial median filter to the raw measurements in order to mitigate potential outliers. We proceed with the kernel smoothing (KS) that enables us to reduce the impact of noise and obtain a quasi-continuous representation of the RFM by interpolation. It allows us to approximate the expected value of the measurements at any location throughout the RoI. We perform spatial filtering and KS in two separate steps because KS is non-robust and the preceding filtering allows us to remove outliers before filtering noise and interpolating. The location-wise STD for each measurable feature is finally calculated as empirical standard deviation of the raw measurements (before filtering and kernel smoothing) within a neighborhood of the specific reference points. In the following, the individual steps of the algorithm are explained in more detail.

As can be seen in Figure 5.1a and 5.1b the measured feature values in the neighborhood of a given location may vary significantly. This is particularly visible around locations with very low signal strength values, i.e. values close to the sensitivity limit of the mobile devices. In order to mitigate the impact of these variations on the representation of the RFM, we apply the spatial filtering which replaces the originally measured feature value v_a of feature a at the given location \mathbf{l} by the median value of the values measured within the neighborhood of \mathbf{l} . We have chosen to define the neighborhood as the set of measurements collected at the up to m locations closest to \mathbf{l} that at the same time lie within the given radius r about \mathbf{l} (see the schematic in Figure 5.2).

In the second step, we estimate a continuous RFM using KS in order to be able to retrieve the expected measurements at any location within the RoI (Berlinet and Thomas-Agnan 2011). Albeit KS can reduce noise by implicit

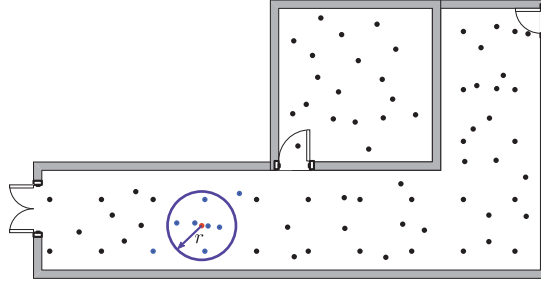


Figure 5.2: Schematic representation of the spatial filtering

filtering, it is not robust and the results could therefore be severely contaminated by outliers in the measured features (Figure 5.3a and 5.3b). Therefore, we apply KS to the media filtered data rather than to the original ones. Because the structure of the indoor region is not taken into account, KS tends to smoothen the RFM over discontinuities like large changes of feature values or change from feature presence to feature absence over short distances e.g. because of walls. This over-smoothing degrades the quality of the RFM for certain features at certain locations. This may be relevant for positioning (Bong and Kim 2012), especially when using radio frequency signals such as WLAN whose propagation is highly influenced by obstacles. Herein we employ a modified version of KS which uses only a subset of the data in the neighborhood of a given location for approximating the expected feature values (Berlinet and Thomas-Agnan 2011). This alleviates the impact of over-smoothing, while at the same time reducing the computational complexity (Berlinet and Thomas-Agnan 2011; Cormen et al. 2009).³

The distribution of the measured noise shown in Figure 5.4 clearly suggests that the variances are location-dependent, are different for different features, and cannot be represented as just a function of feature value or of geometric distance from a single point per feature (e.g. the AP location). So, we propose to model the STD as a location-dependent quantity, independently for each individual feature. To this end, we compute the absolute residuals of the raw data with respect to the spatially filtered and kernel smoothed RFM in order a robust estimate of the STD. At the reference location \mathbf{I}_j in the RFM \mathbb{F} , the STD σ_{jk} of k -th ($k = \{1, 2, \dots, |\tilde{\mathcal{O}}_j|\}$) feature contained in $\tilde{\mathcal{O}}_j$ is computed by the median absolute deviation (MAD) of the measured feature values associated to locations defined as the support set for spatial filtering. The extended representation of the RFM with the estimated STD at location \mathbf{I}_j is denoted as $\tilde{\mathbb{S}}_j = \{(a_{jk}, v_{jk}, \sigma_{jk}) | a_{jk} \in \tilde{\mathcal{A}}_j\}$ and is continuously represented using KS, i. e. $\tilde{\mathbb{S}}_j := \mathcal{F}(\mathbf{I}_j)$.

³A detailed analysis of the over-smoothing problem, the computational complexity of KS, and a discontinuity preserving approach to KS is beyond the scope of this paper and left for future work.

5.5 Iterative scheme for online positioning

Inspired by the finding that the variability of the features has a large effect on the positioning error (Kaemarungsi and Krishnamurthy 2012), we employ the robustly estimated STD of the features to reduce the impact of uncertain feature values when calculating the position estimate. We construct a weighting scheme that reduces the weight of a feature with high STD relative to features with low STD. Therefore, a discrepancy between online measured and expected value of a feature with low STD has more impact on the dissimilarity measure—and thus on the estimated position—than the same discrepancy for a feature with a high STD. This dissimilarity measure is used to identify which subset of reference locations is taken into account when inferring the user’s location using deterministic feature-based positioning algorithms such as k NN.

5.5.1 Weighted dissimilarity measure

Given the online measured features \mathbb{O}_i^u at the location \mathbf{I}_i^u , the weighted dissimilarity measure d^w between \mathbb{O}_i^u and the j -th reference fingerprint $\tilde{\mathbb{O}}_j$ stored in the RFM is computed as:

$$d^w(\mathbb{O}_i^u, \tilde{\mathbb{O}}_j) = \sum_{a \in \mathbb{A}_i^u \cap \tilde{\mathbb{A}}_j} w_{ik}^u \mathcal{G}(v_{ik}^u, v_{jk}) + \alpha_1 \cdot \sum_{a \in \mathbb{A}_i^u \setminus \tilde{\mathbb{A}}_j} w_{ik}^u \mathcal{G}(v_{ik}^u, \gamma) + \alpha_2 \cdot \sum_{a \in \tilde{\mathbb{A}}_j \setminus \mathbb{A}_i^u} w_{ik}^u \mathcal{G}(\gamma, v_{jk}) \quad (5.1)$$

where \mathcal{G} is the selected dissimilarity measure (e. g. Minkowski distance) and γ is the missing value indicator (e. g. -110 dBm). This equation represents a compound dissimilarity measure (CDM) as defined in (Zhou and Wieser 2018a), and correspondingly α_1 and α_2 are hyperparameters regulating the contribution of mutually unshared features to the dissimilarity measure. However, the CDM herein uses a new distance metric, not covered in (Zhou and Wieser 2018a), by location-wise weighting of individual features instead of only weighting according to the respective observability. w_{ik}^u is the weight of the k -th feature at the location/time i and is computed by employing the variability derived from the estimated expected measurement $\tilde{\mathbb{S}}_i^u$ obtained at \mathbf{I}_i^u . In case that the k -th feature in $\tilde{\mathbb{A}}_{ij}$ ($\tilde{\mathbb{A}}_{ij} := \mathbb{A}_i^u \cup \tilde{\mathbb{A}}_j$) is not measurable at location \mathbf{I}_i^u , the weight of the corresponding feature is set to the minimum value of the weights of the measurable features thus reducing their impact on the estimation of the location.

We selected the softmax function (Murphy 2012; Gal and Ghahramani 2016)

$$w_{ik}^u = \frac{e^{-\beta\sigma_{ik}^{-2}}}{\sum_{l=1}^{|\mathbb{S}_i^u|} e^{-\beta\sigma_{lk}^{-2}}} \quad (5.2)$$

to calculate the weight of each feature using the estimated STD, where $\mathbb{S}_i^u = \mathcal{F}(\mathbf{I}_i^u)$ and $\beta > 0$ is the scale factor for adapting the concentration of the softmax function. The denominator normalizes the weights and makes the solution invariant to the scale of the weights. We have also tried to use a weight function corresponding to the one frequently employed for weighted least-squares (and actually motivated by maximum likelihood estimation with normally distributed observations), namely setting each weight proportional to the inverse of the respective variance. However, the accuracy of the solutions was worse than using the softmax function.

The weighted dissimilarity measure is used to identify the candidate locations, whose dissimilarity values are smallest among all reference fingerprints stored in the RFM. We estimate the user's location using k NN or weighted k NN by averaging or weighted averaging (e. g. inversely proportional to the value of dissimilarity measure) of the candidate locations. More details about k NN and weighted k NN can be found in e. g. (Padmanabhan et al. 2000; Zhou and Wieser 2019b).

5.5.2 Iterative scheme

The position estimation requires to calculate the weight of each feature. However, the weight depends on the standard deviation which in turn varies with location. The required value can only be extracted from the RFM once the location is known. We thus carry out the positioning in an iterative way by i) assuming a position (initialization); ii) retrieving the STDs from the RFM, calculating the weights and estimating the position (update step); and iii) repeating ii) until a termination condition of the iterative scheme is fulfilled. These steps are explained in more detail in the following subsections.

Determination of the initial location. The initial location $\hat{\mathbf{I}}_i^{(0)}$ of the user is used to derive the weights for the first iteration. One straightforward way of initializing is to choose the location estimated by the standard k NN without the weighted dissimilarity measure (i. e. the traditional k NN). When processing real world data we found out that the solution obtained at the termination of the iterative process is quite stable when initializing the location even randomly (see Section 5.6). This suggests that the positioning performance does not depend strongly on the choice of the initial location.

Update step. At the t -th iteration ($t \in \mathbb{N}^+$), the weights as well as the dissimilarities are computed according to the variability obtained at the location searched at the $(t-1)$ -th iteration. The weight $w_{ik}^{(t)}$ of the k -th feature at location/time i and the t -th iteration is defined as:

$$w_{ik}^{(t)} = \frac{e^{-\beta(\sigma_{ik}^{(t-1)})^{-2}}}{|\tilde{\mathbb{S}}_i^{(t-1)}| \sum_{l=1}^{|\tilde{\mathbb{S}}_i^{(t-1)}|} e^{-\beta(\sigma_{lk}^{(t-1)})^{-2}}} \quad (5.3)$$

where $\tilde{\mathbb{S}}_i^{(t-1)}$ is the estimated expected value of features with their STD at location $\hat{\mathbf{I}}^{(t-1)}$. This updated weights are used to compute the dissimilarity measure as defined in (5.1) and consequently to infer the estimated location $\hat{\mathbf{I}}_i^{(t)}$ at the t -th iteration using e. g. k NN algorithm.

Termination condition. Ideally the searching process should converge to a fixed location. This state is assumed to be reached when the distance between two consecutively obtained location estimated is lower than a given small threshold. We denote this subsequently as converging state and terminate the iterative process when

$$|\hat{\mathbf{I}}_i^{(t)} - \hat{\mathbf{I}}_i^{(t-1)}|_2 < d^{\min},$$

where d^{\min} is the threshold, which we set to 10^{-3} m in the experimental analysis later on. We found out that the iterative process proposed herein sometimes enters a loop in which a (small) subset of locations are repeatedly obtained as estimates in the same sequence. We denote this as the looping state and introduce a second termination condition which is met when this state is recognized. We implement it as a threshold on the distance between the location estimate obtained at the iteration t and the ones estimated at previous iterations except the estimated location at the $(t-1)$ -th iteration. More formally, the second condition is satisfied and the iteration is terminated when

$$\min_{m=1, \dots, t-2} \{|\hat{\mathbf{I}}_i^{(t)} - \hat{\mathbf{I}}_i^{(m)}|_2\} < d^{\min}.$$

Finally, the maximum number T of iterations is also limited (e. g. $T = 100$) in order to prevent long or endless search for a solution. If the search for an estimate is terminated due to this condition, we denote it as max. state.

Assuming that the iterations terminate after T' iterations we select or compute the final estimate of the position depending on the termination flag (TF) $\varepsilon_i^u \in \{0, 1, 2\}$, indicating the respective state, as follows:

- **Converging state:** The location estimated at the T' -th iteration is se-

lected as the final estimate of the user's location $\hat{\mathbf{I}}_i^u$, and ε_i^u is set to 0.

- **Looping state:** In this case the searched locations do not converge to a single point. If the number of locations exceeds a certain minimum (e. g. 4) and if the locations visited in the looping state are not farther apart than a chosen maximum (e. g. 0.01 m) (see Figure 5.5) we use the minimum covariance determinant (MCD) estimator⁴ for computing the estimated location $\hat{\mathbf{I}}_i^u$ of the user from the convex hull of the visited locations. If the number of points is too low or if they are too far apart from each other the situation is handled like the max. state. If the looping state termination condition is met and the MCD is reported as the final estimate, the TF ε_i^u is set to 1.
- **Max. state:** This case actually means that the position estimation using the weighted dissimilarity measure fails because no position can be found where the measured features and the predetermined standard deviations are compatible. In this case, we can either report a failure of the algorithm and not calculate a solution, or we can calculate an estimate ignoring the variability information. We have chosen the latter herein. In particular, we determine $\hat{\mathbf{I}}_i^u$ from all searched locations $\hat{\mathbb{L}}$ analyzing the similarities between the user measured fingerprint \mathbb{O}_i^u and the expected ones at the searched locations. Specifically, we employ the modified Jaccard index (MJJ), which has been used for identifying subregions according to the measurability of features (Zhou and Wieser 2019b), as the similarity metric. The MJJ value S^{MJJ} between \mathbb{O}_i^u and the expected one $\tilde{\mathbb{O}}_i^{(t)}$ ($\tilde{\mathbb{O}}_i^{(t)} := \mathcal{F}(\hat{\mathbf{I}}_i^{(t)})$) at the searched location $\hat{\mathbf{I}}_i^{(t)}$ of the t -th iteration is computed by:

$$S^{\text{MJJ}}(\mathbb{O}_i^u, \tilde{\mathbb{O}}_i^{(t)}) = \frac{1}{2} \left(\frac{|\mathbb{A}_i^u \cap \tilde{\mathbb{A}}_i^{(t)}|}{|\mathbb{A}_i^u \cup \tilde{\mathbb{A}}_i^{(t)}|} + \frac{|\mathbb{A}_i^u \cap \tilde{\mathbb{A}}_i^{(t)}|}{|\mathbb{A}_i^u|} \right) \quad (5.4)$$

where \mathbb{A}_i^u and $\tilde{\mathbb{A}}_i^{(t)}$ are the sets of the measured features contained in \mathbb{O}_i^u and $\tilde{\mathbb{O}}_i^{(t)}$, respectively. The estimated user's location $\hat{\mathbf{I}}_i^u$ is then the one that has the biggest MJJ value among all searched locations $\hat{\mathbb{L}}$, i. e. the one with the maximum number of common measurable features is selected as the final estimate of the user's location.

⁴MCD is a highly robust estimation of multivariate location and scatter. We use the implementation of MCD from scikit-learn (Pedregosa et al. 2011).

5.6 Analysis of the variability estimation and positioning performance

We start this section by presenting the results of the location-wise variability of each individual feature estimated using the kinematically collected RFM data, which is discussed in detail in (Zhou and Wieser 2019b). We conclude this section with an analysis concerning the characteristics of iteratively searched locations as well as the positioning performance of the proposed iterative scheme.

5.6.1 Results of the variability estimation

Herein we set $m = 20$ and $r = 2$ m to obtain the spatially filtered RFM, which visually has an adequate spatial consistency in the neighborhood of each location. Figure 5.1c and 5.1d clearly show that the spatial filtering can reduce the large variations contained in the raw RFM to a great extent. For further analysis, we compute the residuals between the raw and the spatially filtered RFM. The obtained residuals are close to zero-mean distributed and have a location-dependent magnitude as illustrated in Figure 5.1e and 5.1f. Large residuals occur either in regions close to the boundaries of the RoI (e. g. close to the walls or corners of rooms and corridors) or at locations where the RSS values are hardly measurable by the mobile phone. In both cases the features are very likely affected by obstacles which also cause locally large variations of the feature values.

Figure 5.4 shows the results of the estimated STD value using the MAD of the measured feature values associated to the neighborhood of a given location. As can be seen, each feature has a different variability throughout the RoI, i. e. the STD value is dependent both on the feature as well as on the location. This is the primary motivation that the variability is modeled location-wise for each individual feature instead of simply expressing the variability as a function of the measured feature value or as a constant value. The regions where the feature values have a higher STD are clearly correlated to the local variations of the measured feature value and the geometry of the building (Figure 5.3 and 5.4). The high variances occur in the case that a low number of measurements has been collected in the neighborhood region. These are caused by the violation of the assumption that the expected feature values are locally obtainable.

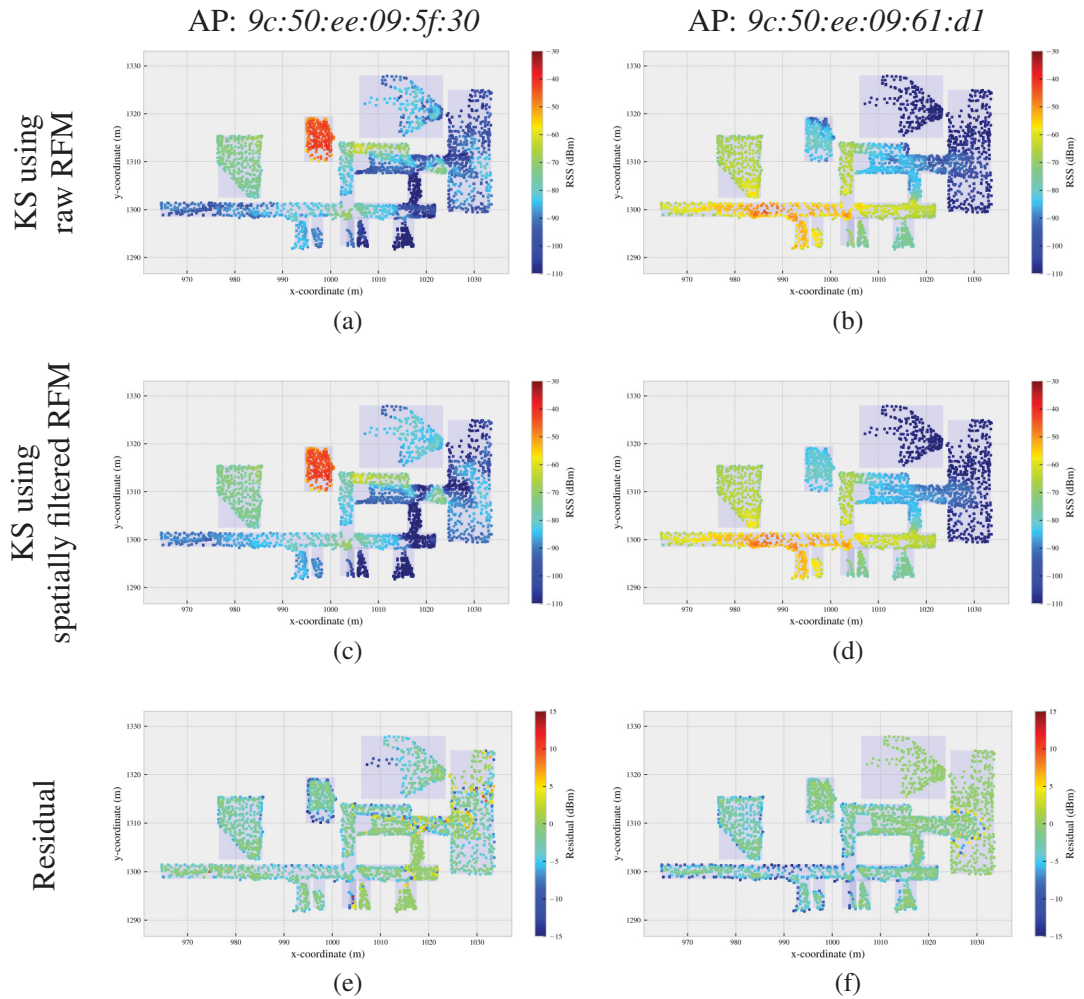


Figure 5.3: Examples of kernel smoothed RFM of two arbitrary selected APs. The results of the first two rows are yielded by taking the raw RFM and the spatially filtered one (i. e. the ones depicted in the first two rows of Figure 5.1) as the input to KS, respectively. Third row shows the residual between the kernel smoothed RFM using the spatially filtered one and the spatially filtered RFM. Though the KS provides the continuous representation of the RFM, we only visualize the smoothed features at the locations contained in the raw RFM for easy comparison.

5.6.2 Results of iterative scheme for positioning

The proposed iterative scheme for feature-based positioning is implemented using the application programming interface of scikit-learn package, a widely used machine learning package in Python (Pedregosa et al. 2011). Herein we present the results of the iterative positioning using k NN with the weighted Euclidean distance as the dissimilarity metric for measuring the distance in the feature space. The values of several hyperparameters have to be configured.

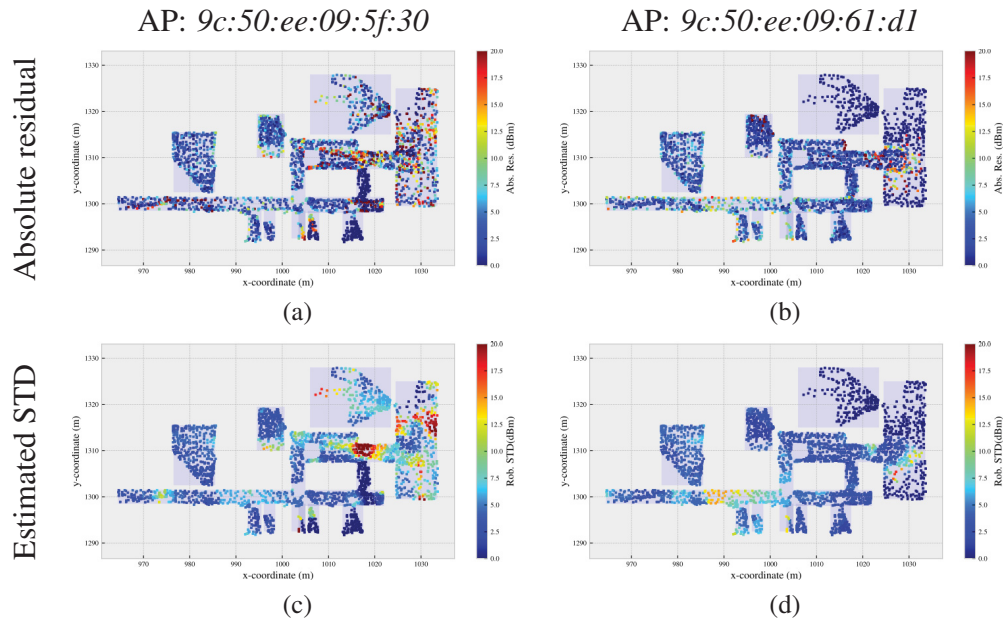


Figure 5.4: Examples of the absolute residuals and estimated STD for two APs

Regarding the weighted dissimilarity measure, as formulated in Section 5.5.1, α_1 and α_2 are set to 3.0 by following the results reported in (Zhou and Wieser 2018a). The number of nearest neighbors k in the k NN algorithm and the scale factor β of the softmax function are empirically set to 1 and 2.0, respectively. Optimization of the parameters (e. g. using grid/random search or Bayesian optimization (Wang and de Freitas 2014)) for achieving the best positioning performance is left for future work.

Figure 5.5 shows several examples of the searched locations of the iterative positioning with random as well as k NN initialization. Each individual subplot depicts the results of the iterative positioning at a fixed test location. The subplots depict the searched locations (red squares), the final estimation (blue triangle or coral diamond), the estimation using the traditional k NN algorithm, and the ground truth (black square)⁵. The initialization of the initial location has only a minor impact on the iterative searching process in this case because the RoI is relatively small. The initial location is determined by arbitrarily taking one of the reference locations stored in the RFM in case of random initialization. In addition, our schemes for determining the final estimation of the user’s location from these searched locations do not achieve the best potential positioning performance using the iterative scheme. Because there are locations in $\hat{\mathcal{L}}$ that are closer to the ground truth but they are not taken as the final estimation. This suggests that the iterative scheme for positioning has the potential to further improve the positioning performance if the proper technique is applied to retrieve the final estimation from these searched

⁵In order to improve the readability, the initial location has not been visualized.

locations. The optimal positioning accuracy (denoted as Ours (opt.) in Figure 5.8 and in Table 5.1) is defined by assuming that the technique for the final estimation is capable of retrieving the searched location that is the closest to the ground truth. As depicted in Figure 5.8, our scheme for retrieving the final estimation can achieve comparable performance to that of the optimal one when comparing the overall positioning accuracy. However, from Table 5.1, it also suggests that the maximum positioning error can be reduced to a large extent if the optimal positioning can be achieved.

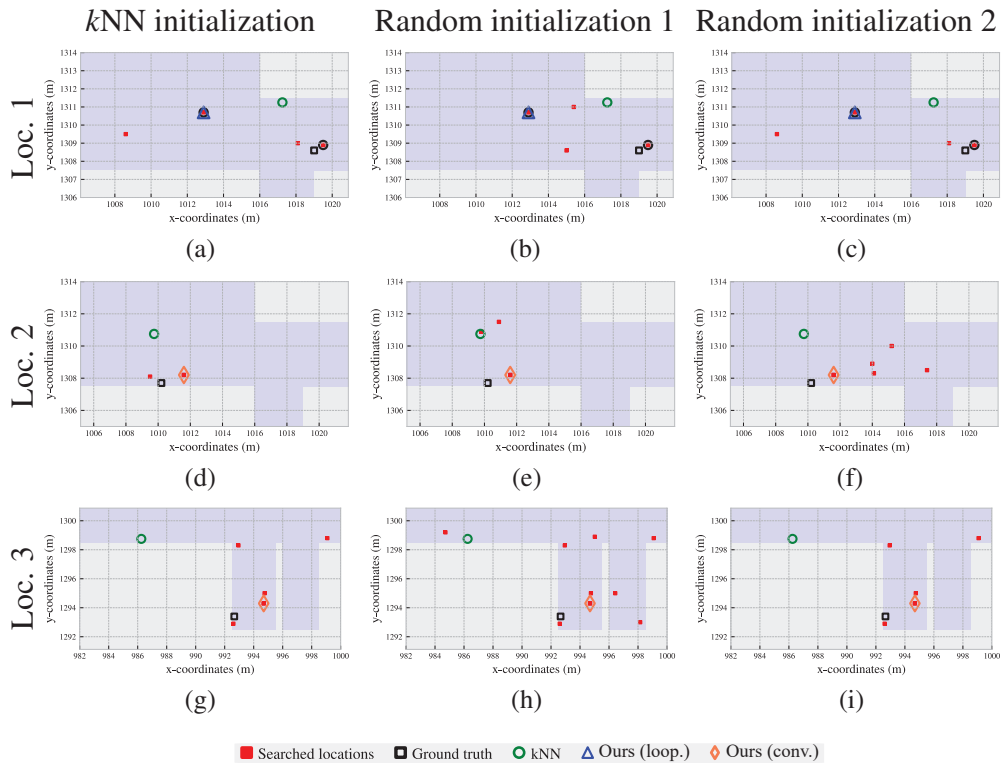


Figure 5.5: Examples of iteratively searched location with different initializations. The locations with black circles are repeatedly searched in the same sequence (i. e. the looping state) when using the iterative scheme.

Figure 5.6 shows the statistics of the TF, denoted by the percentage of locations terminated with different conditions. In case of $k = 1$ about 83% iterative search processes have terminated with the converging state. This is about 35 percent points higher than in case of $k = 3$. We therefore set the number of the nearest neighbors for k NN to 1. In addition, we have analyzed the searched locations within the looping state cases are distributed on space. Figure 5.7 shows the distribution of the maximum distance between points within the same loop. Most of the maximum distances are significantly less than 10 m, though, in some extreme cases we observed up to 60 meters. As shown in Figure 5.8, the schemes proposed for the loop state (MCD or MJI) are still capable of properly selecting position estimates close to the ground

truth also in most of these cases.

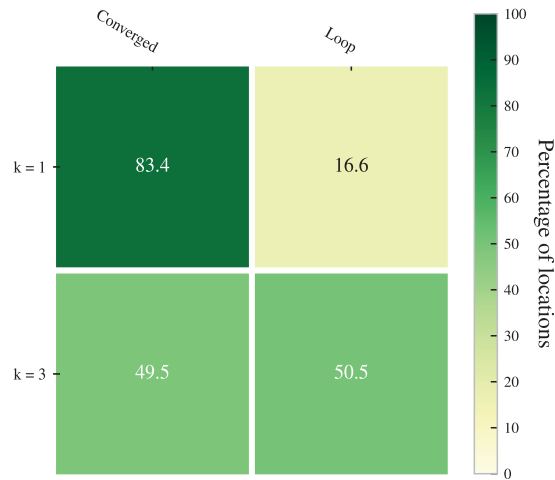


Figure 5.6: Comparison of the percentage of locations terminated with different conditions for $k = 1$ and 3. The max. state is not included in the figure because it has not happened in the experimental analysis.

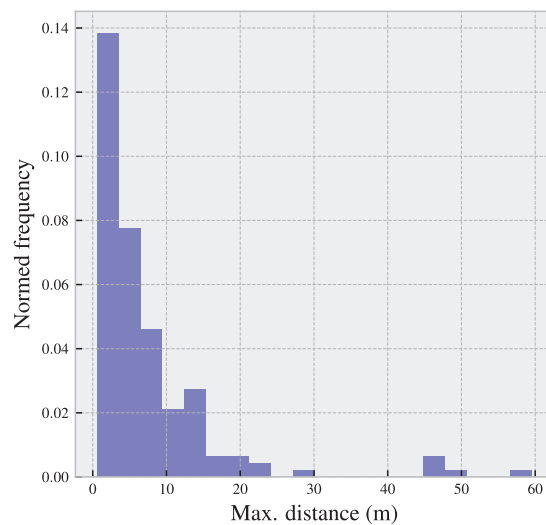


Figure 5.7: The maximum distance between the locations consisting of the loop state for $k = 1$

The empirical cumulative distribution function (ECDF) of the radial positioning errors is presented in Figure 5.8. The proposed approach can significantly improve the positioning performance as compared to the performance of the traditional k NN and k NN with CDM. Using the algorithm proposed herein, about 86% of the estimated locations have a positioning error smaller than 2 m and around 97% of estimated locations have an error of less than 4 m. Compared to k NN, this represent an improvement of up to 20 and 10 percent points, respectively. The improvement is also up to 10 and 6 percent

points as compared to k NN with CDM. The percentage of estimated locations whose error distance is larger than 5 m is reduced from about 10.2% and 6.7% to 2.6%, when compared to the traditional k NN, and k NN with CDM, respectively. We also report the circular error (CE) defined as the minimum radius for including a given percentage of positioning errors (e. g. CE 50 for the 50th percentile) in Table 5.1. The maximum positioning error is reduced by about 40%, from 37.0 m to 22.2 m when comparing k NN with CDM to our approach. The CE 50, CE 75, and CE 90 are reduced by one third when compared to the k NN without iterative positioning. Furthermore, in Figure 5.9 we illustrate and compare the distribution of the locations, at which the positioning error is larger than 8 m using the original k NN. Figure 5.9a shows that these locations yielding large errors are mostly located close to the accessible boundaries of the indoor regions, i. e. close to corners of corridors and rooms, or to the walls. This pattern is similar to the spatial distribution of high variance of the feature values contained in the raw RFM as shown in Figure 5.4. Our approach can significantly reduce the number of occurrences where the positioning errors are larger than 8 m.

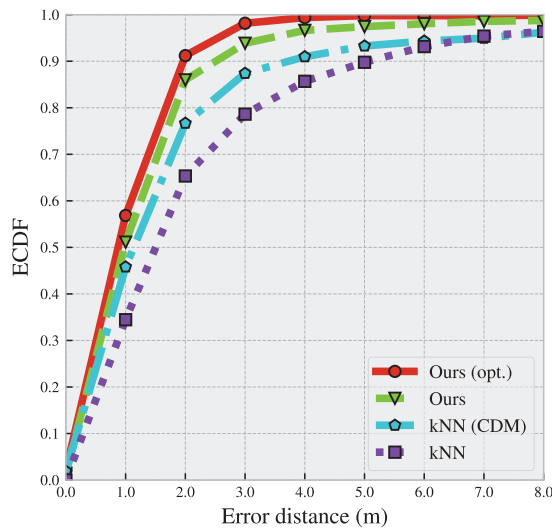


Figure 5.8: Comparison of ECDF with respect to the radial positioning errors.

Table 5.1: Statistics of positioning errors (meters)

| | CE 50 | CE 75 | CE 90 | Max. error |
|--------------|-------|-------|-------|------------|
| k NN | 1.4 | 2.6 | 5.2 | 29.6 |
| k NN (CDM) | 1.1 | 1.9 | 3.7 | 37.0 |
| Ours | 1.0 | 1.5 | 2.2 | 22.2 |
| Ours (opt.) | 0.9 | 1.4 | 1.9 | 9.2 |

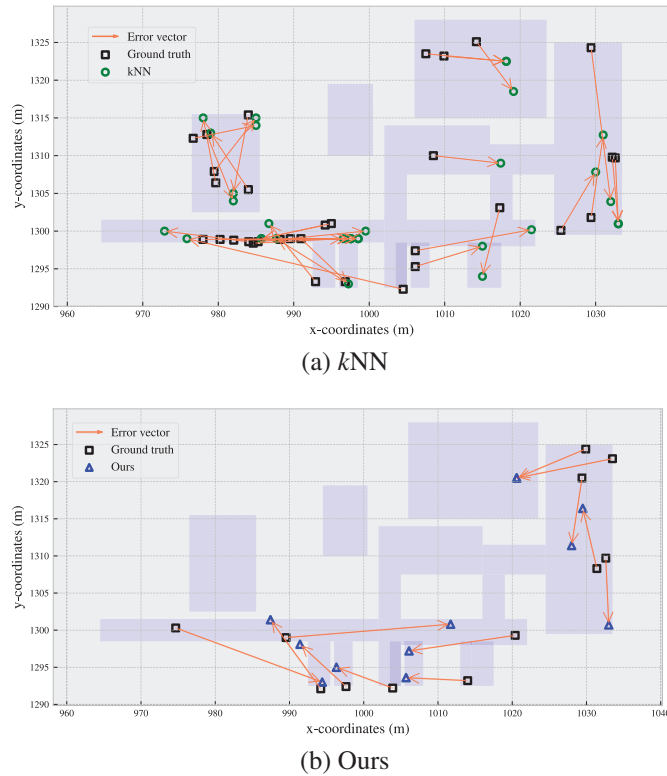


Figure 5.9: Distribution of the locations yielding large errors (> 8 m) in positioning

5.7 Conclusion

We have proposed an iterative scheme for feature-based positioning, which is based on the weighted dissimilarity measure, for reducing large errors occurring in FIPs. Appropriate weights for the individual feature can be obtained by analyzing the variability of the kinematically collected raw data underlying the RFM. The location-wise standard deviation of each feature is robustly computed using the MAD between the raw data and the spatially smoothed RFM. This variability information is stored as an additional layer of the RFM and used for weighting the contribution of each feature to the dissimilarity measure during the online positioning phase.

Using real WLAN RSS data collected along with location ground truth in an office building, we could show that the noise of the raw observations indeed depends on the location and on the feature. We have implemented the proposed algorithms in Python and have validated the performance of the proposed iterative scheme. Compared to *kNN* with CDM, the maximum positioning error is reduced by more than 40% and the iterative scheme can improve the overall positioning performance. The positioning accuracy defined as the percentage of the locations whose radial positioning error is less

than 2 m is improved from 65% to 86% when compared to traditional k NN.

In future work, we will further investigate the proposed algorithms using data from other environments. We will further investigate the loop state and the handling of remaining outliers. Finally, we will investigate how the standard deviations modeled within the RFM can help to identify the need for updates of the RFM.

Acknowledgment

The Chinese Scholarship Council has supported C. Zhou during his doctoral studies at ETH Zürich. The data used within the experimental investigation were collected by the students E. Weiss, I. Bai, N. Meyer, and G. Filella. The device holder for the ground truth measurements was designed by R. Presl.

CHAPTER 6

CONCLUSIONS AND OUTLOOK

6.1 Summary

The results and findings of this thesis are the outcome of a study on the mitigation of variability issues encountered in feature-based indoor positioning systems (FIPSs), which utilize signals of opportunity (SoP) as features. The milestones of the research are represented by the major technical chapters (i. e. Chapters 3, 4, and 5) of this dissertation. The main challenges jeopardizing the practical usability of FIPSs are the unknown prior information about the measurable SoP as well as the limited control of their quality. In order to overcome these difficulties, this thesis proposes solutions for tackling three relevant research problems, which are the key to enhancing the applicability of FIPSs. These problems and the proposed solutions are briefly summarized as follows:

1. The first problem deals with the scalability issue intrinsically originating from FIPSs in the context of reducing the computational complexity during the online positioning phase. The proposed solution is a hierarchical combination of a modified Jaccard index (MJJ)-based subregion selection and an adaptive forward-backward greedy search (AFBGS)-based feature selection approach as discussed in Chapter 3.
2. The second problem tackled in the dissertation is the measurability variations of those SoP in the context of adaptively quantifying the difference between pairs of the measurements in the feature space. In order to solve this, a non-vector-based dissimilarity measure, named compound

dissimilarity measure (CDM), has been presented in Chapter 4. The CDM combines a typical distance metric with set operations.

3. The third problem addressed by this dissertation relates to the mitigation of large errors in FIPs. The proposed approach, as presented in Chapter 5, is an iterative scheme for positioning with the weighted dissimilarity measure, which adapts the contribution of each individual feature to the dissimilarity measure according to the robustly estimated location-wise standard deviation of each individual feature.

The proposed solutions have been validated using the kinematically collected reference fingerprint map (RFM), which includes received signals from wireless local area network (WLAN) access points (APs) or Bluetooth low energy (BLE) beacons as features and has been presented in several relevant publications. The scientific relevance of this work is primarily two-fold. On the one hand, the proposed solutions can be extended to utilize any other kind of location-relevant features and incorporated into different indoor application cases, e. g. the Internet of Things. On the other hand, the individual solutions can be employed in other fields aside from indoor positioning. For instance, the proposed CDM is applicable to situations dealing with missing values when quantifying the difference between pairs of vectors. This work also has high potential impact to industry and therefore to society. The proposed solutions contribute towards enabling final indoor location-based services (LBSs) by improving the positioning accuracy while reducing the computational burden.

6.2 Discussion of contributions

As previously stated, the objective of this thesis is to mitigate variability issues affecting FIPs in the context of providing indoor LBSs to pedestrians using a mobile device and SoP measured by its built-in sensors. Throughout the study, the potential of the proposed solutions to enhance indoor positioning has been discussed in detail. In the following the most important points are highlighted. Each subsequent subsection presents the individual solutions concerning the aforementioned research questions.

6.2.1 Reduction of the computational complexity

The first contribution of this work relates to the reduction of data storage requirements and computational complexity in terms of processing time. This is achieved through segmentation of the entire region of interest (RoI) into

subregions, identification of a few candidate subregions during the online positioning stage, and use of a selected subset of relevant features instead of all the available features for the position estimation. The subregion selection is based on an MJI quantifying the similarity between the features obtained by the user and those available within the RFM. Feature selection is based on an AFBGS yielding a pre-computed set of relevant features for each subregion. The reduction of computational complexity is obtained from the reduction of both the number of candidate locations needed to analyze during online positioning and the number of features to be compared.

The positioning performance of the proposed approach has been evaluated by employing the kinematically collected RFM as validation dataset. The time required for the position estimation during the online stage was reduced by a factor of about 10 when using the selected relevant features within 11 selected subregions instead of using all the available features and searching over the entire RoI. In addition, the positioning accuracy using maximum a posteriori (MAP) is reduced from 9.8 m to 7.4 m when WLAN signals are used as features and from 12.3 m to 10.9 m when features are obtained from BLE beacons signals. For k -nearest-neighbors (k NN), the positioning accuracy does not change with subregion and feature selection. Given a fixed number of candidate subregions and a fixed, low number of features, the computational burden of the entire algorithm is almost independent of the size of the entire RoI and of the number of available features across the RoI.

6.2.2 Adaptation to measurability variations

A non-vector-based dissimilarity measure, named CDM, has been proposed by combining a typical distance metric with set operations aiming at measuring the dissimilarity between measurements despite possible missing features. The adaptability of the proposed CDM is achieved with the inclusion of hyperparameters, which can be tuned according to the data and needs of the application. The proposed CDM is validated using the kinematically collected RFM as well as other three publicly available datasets by analyzing the positioning performance. Both the accuracy for identifying buildings and floors, and for calculating specific locations improve significantly, reaching improvements over 5 and 10 percent points, respectively. Although the CDM is proposed herein aiming at handling missing data in feature-based positioning, it is applicable to other missing data problems as well (e. g. searching the correspondences of point clouds according to sparsely described local features).

6.2.3 Mitigation of large errors in positioning

An iterative scheme for feature-based positioning, which is based on the weighted dissimilarity measure, has been proposed for reducing large errors in FIPSSs. Appropriate weights for the individual features can be obtained by analyzing the variability of the kinematically collected raw data underlying the RFM. The location-wise standard deviation of each feature is robustly computed using the median absolute deviation between the raw data and the spatially smoothed RFM. This variability information is stored as an additional layer of the RFM and used for weighting the contribution of each feature to the dissimilarity measure during the online positioning phase.

Using real WLAN received signal strength data collected along with location ground truth in an office building, we could show that the noise of the raw observations indeed depends on the location and on the feature. We have implemented the proposed algorithms in Python and have validated the performance of the proposed iterative scheme. Compared to k NN with CDM, the maximum positioning error is reduced by more than 40% and the iterative scheme can improve the overall positioning performance. The positioning accuracy defined as the percentage of the locations whose radial positioning error is less than 2 m is improved from 65% to 86% when compared to traditional k NN.

6.3 Discussion of limitations

This dissertation, as discussed in the previous section, has contributed to relevant topics of FIPSSs. There is, however, potential room for further improvement, as the proposed approaches have a number of disadvantages when compared to alternative ones. Here the most important aspects of these limitations are discussed.

Segmentation of the RoI. The segmentation of the entire RoI according to the available indoor maps (or a floor plan of the RoI) has been applied to achieve the coarse estimation of the user's location and reduce the search space for refining the location estimation according to the selected candidate subregions (see Chapter 3). The role of the subregion definition (shape, orientation, homogeneity) with the adaptation of the characteristics of the RoI and possible benefit of optimization should be further investigated. Promising research directions worth considering are: i) a user motion model during subregion selection, ii) temporal changes of the reference fingerprinting map, and iii) integration of different types of features for improving the positioning accuracy.

Refinement of the iterative scheme. An iterative scheme has been proposed for reducing large errors in FIPS and the results presented in Chapter 5 have shown that this scheme can potentially improve the positioning accuracy. This potential has not been achieved by the refined scenarios used in the chapter and is worth looking further into from the perspective of retrieving the one closest to the ground truth as the final estimation from the iteratively searched locations. In addition, it would be beneficial to explore the capability of identifying large error cases from the distribution of the iteratively searched locations.

Computational complexity at the offline stage. The proposed solutions put the extra computational burden to the offline stage as the price to pay for reducing online positioning processing time and storage needs (Chapter 3), and to enable employing the weighted dissimilarity measure (Chapter 5). The available computation resources on a commonly used computer are able to handle the offline computation in a reasonable time in a middle level scalability (e. g. hundreds of measurable features in an RoI consisting of one floor of a typical office building), affording the increment of the computational burden in the situation of large scalability (e. g. huge RoI with millions of measurements) would become critical. Therefore, the improvement of the computational efficiency (e. g. parallelization) and utilization of new computational resources (e. g. GPUs) are of significance.

Optimization of hyperparameters. Although data-driven algorithms and machine learning approaches can be employed without requiring explicit prior knowledge of the intrinsic model built in the data, these methods have to be tuned regarding their hyperparameters, which are nested in the machine learning algorithms. In the major chapters a naive parameter optimization scheme (e. g. grid search) has been used for tuning the relevant parameters (e. g. the nested parameters of k NN) for different applications independently. In future work, it would be beneficial to employ an optimization framework that is able to search an optimal combination of parameters in the context of taking into account all the relevant parameters into one optimization procedure. Among the available parameter search and optimization frameworks, Bayesian optimization could be a promising candidate (Wang and de Freitas 2014; Shahriari et al. 2016).

Fusion of the proposed solutions. The proposed solutions for mitigating variability issues in FIPSs are performed in individual frameworks and can be potentially fused into a single pipeline. This integral solution would output an end-to-end solution while providing reduced positioning time, robustness against measurability variations, and mitigation of large errors according to the user measured features during the online positioning phase.

6.4 Outlook

The advancements presented in this thesis should be helpful to raise awareness of the progress in the field of indoor positioning, especially regarding the application of data-driven algorithms. Most of the solutions have been inspired by the state-of-the-art in the corresponding field as well as the knowledge and advances in the field of machine learning and deep learning. In the following, a few promising directions for future research are discussed, based on the methodological and practical findings of the current dissertation. These include open problems, proposals for future research and potential solutions to the limitations encountered with respect to new technologies and methodology.

New sensing techniques. In the state-of-the-art of FIPSs, the prevalent radio frequency signals, such as those from WLAN, BLE, and cellular networks, are the most widely studied ones. However, the sensing techniques used at present can be improved for obtaining new signals used as features for positioning. One promising signal is the radio frequency channel state information (CSI) (Halperin et al. 2011; Schulz et al. 2018). The CSI measurements can be used to model space, time, and frequency characteristics of the channel by observing the received signals (both the magnitude attenuation and phase shift) from multiple-input and multiple-output antennas. This WiFi sensing technique has been applied using specific hardware design and its extension to mobile devices is currently under development. The CSI measurements can be employed for positioning and can potentially improve the performance by modeling the signals according to the extended information, included in CSI, about the medium and environment that the signals have traveled through.

Exploitation of advanced deep learning models. The advancements of deep learning approaches have improved their interpretability and enabled their applicability by better handling the variations of measurability. This especially refers to the development of graph neural networks (Battaglia et al. 2018). Graph neural networks allow to embed arbitrary topological representations into vectors, e. g. embedding one signal into a vector representation according to the spatial-temporal topology, which is generated from the RFM. In addition, recent advances in the application of deep neural networks to time series analysis allow for the sequential deep learning approach to predict the trajectory as fed with the raw measurements from inertial sensors by breaking the cycle of error propagation with a sequence-based physical model. According to Changhao et al. (2018), this makes the proposed deep learning model robust against unbound drifts problem, which are challenging topics in traditional inertial navigation systems (e. g. pedestrian dead reckoning or strapdown inertial navigation).

BIBLIOGRAPHY

- Addlesee, M., Curwen, R., Hodges, S., Newman, J., Steggles, P., Ward, A., and Hopper, A. (2001). Implementing a sentient computing system. *Computer*, 34(8):50–56.
- Alarifi, A., Al-Salman, A., Alsaleh, M., Alnafessah, A., Al-Hadhrami, S., Al-Ammar, M., and Al-Khalifa, H. (2016). Ultra wideband indoor positioning technologies: Analysis and recent advances. *Sensors*, 16(5):707.
- Barron, A. R., Cohen, A., Dahmen, W., and DeVore, R. A. (2008). Approximation and learning by greedy algorithms. *The Annals of Statistics*, 36(1):64–94.
- Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., et al. (2018). Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*.
- Bekkali, A., Sanson, H., and Matsumoto, M. (2007). Rfid indoor positioning based on probabilistic rfid map and kalman filtering. In *Wireless and Mobile Computing, Networking and Communications, 2007. WiMOB 2007. Third IEEE International Conference on*, pages 21–21. IEEE.
- Berlinet, A. and Thomas-Agnan, C. (2011). *Reproducing Kernel Hilbert Spaces in Probability and Statistics*. Springer Science & Business Media, Berlin Heidelberg.
- Bhatta, B. (2010). *Global navigation satellite systems: insights into GPS, GLONASS, Galileo, Compass, and others*. BS Publications.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg.

- Bisio, I., Cerruti, M., Lavagetto, F., Marchese, M., Pastorino, M., Randazzo, A., and Sciarrone, A. (2014). A Trainingless WiFi Fingerprint Positioning Approach Over Mobile Devices. *IEEE Antennas and Wireless Propagation Letters*, 13:832–835.
- Bisio, I., Lavagetto, F., Marchese, M., and Sciarrone, A. (2013). Performance comparison of a probabilistic fingerprint-based indoor positioning system over different smartphones. In *2013 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS)*, pages 161–166.
- Bisio, I., Lavagetto, F., Marchese, M., and Sciarrone, A. (2016). Smart probabilistic fingerprinting for WiFi-based indoor positioning with mobile devices. *Pervasive and Mobile Computing*, 31:107–123.
- Bitew, M. A., Hsiao, R.-S., Lin, H.-P., and Lin, D.-B. (2015). Hybrid indoor human localization system for addressing the issue of rssi variation in fingerprinting. *International Journal of Distributed Sensor Networks*, 11(3):831423.
- Bong, W. and Kim, Y. C. (2012). Fingerprint wi-fi radio map interpolated by discontinuity preserving smoothing. In *Convergence and Hybrid Information Technology*, pages 138–145, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Brena, R. F., García-Vázquez, J. P., Galván-Tejada, C. E., Muñoz-Rodríguez, D., Vargas-Rosales, C., and Fangmeyer, J. (2017). Evolution of indoor positioning technologies: A survey. *Journal of Sensors*, 2017.
- Brouwers, N., Zuniga, M., and Langendoen, K. (2014). Incremental Wi-Fi scanning for energy-efficient localization. In *2014 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 156–162.
- Bühlmann, P. (2006). Boosting for high-dimensional linear models. *Ann. Statist.*, 34(2):559–583.
- Cha, S.-H. (2007). Comprehensive survey on distance/similarity measures between probability density functions. *International Journal of Mathematical Models and Methods in Applied Sciences*, 1(4):300–307.
- Changhao, C., Xiaoxuan, L., Andrew, M., and Niki, T. (2018). Ionet: Learning to cure the curse of drift in inertial odometry. In *The Thirty-Second AAAI Conference on Artificial Intelligence (AAAI-18)*, pages 6468 – 6476.
- Chen, Y., Yang, Q., Yin, J., and Chai, X. (2006). Power-efficient access-point selection for indoor location estimation. *IEEE Transactions on Knowledge and Data Engineering*, 18(7):877–888.

- Cong, L., Wang, H., Qin, H., and Liu, L. (2018). An environmentally-adaptive positioning method based on integration of gps/dtmb/fm. *Sensors*, 18(12):4292.
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to Algorithms, Third Edition*. The MIT Press, 3rd edition.
- Deng, L. and Yu, D. (2014). Deep learning: Methods and applications. *Foundations and Trends in Signal Processing*, 7(3–4):197–387.
- Dippy, R. J. (1946). Gee: a radio navigational aid. *Journal of the Institution of Electrical Engineers - Part IIIA: Radiolocation*, 93(2):468–480.
- Driusso, M., Marshall, C., Sabathy, M., Knutti, F., Mathis, H., and Babich, F. (2016). Indoor positioning using lte signals. In *2016 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8.
- Duewell, T. (2019). Fingerprinting-based indoor positioning using a mobile phone. Master’s thesis, Geosensors and Engineering Geodesy, D-BAUG, ETH Zürich, ETH Zürich.
- Efron, B. (1994). Missing data, imputation, and the bootstrap. *Journal of the American Statistical Association*, 89(426):463–475.
- El-Kafrawy, K., Youssef, M., El-Keyi, A., and Naguib, A. (2010). Propagation modeling for accurate indoor WLAN RSS-based localization. *IEEE Vehicular Technology Conference*, pages 1–5.
- Elloumi, W., Latoui, A., Canals, R., Chetouani, A., and Treuillet, S. (2016). Indoor pedestrian localization with a smartphone: A comparison of inertial and vision-based methods. *IEEE Sensors Journal*, 16(13):5376–5388.
- Fastrich, B., Paterlini, S., and Winker, P. (2015). Constructing optimal sparse portfolios using regularization methods. *Computational Management Science*, 12(3):417–434.
- Feng, C., Au, W. S. A., Valaee, S., and Tan, Z. (2012). Received-signal-strength-based indoor positioning using compressive sensing. *IEEE Transactions on Mobile Computing*, 11(12):1983–1993.
- Ferris, B., Fox, D., and Lawrence, N. D. (2007). Wifi-slam using gaussian process latent variable models. In *IJCAI*, volume 7, pages 2480–2485.
- Frey, B. J. and Dueck, D. (2007). Clustering by passing messages between data points. *science*, 315(5814):972–976.
- Gal, Y. and Ghahramani, Z. (2016). Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *international conference on machine learning*, pages 1050–1059.

- Georgiou, K., Constambeys, T., Laoudias, C., Petrou, L., Chatzimilioudis, G., and Zeinalipour-Yazti, D. (2015). Anyplace: A crowdsourced indoor information service. In *2015 16th IEEE International Conference on Mobile Data Management*, volume 1, pages 291–294. IEEE.
- Gojcic, Z., Kalenjuk, S., and Lienhart, W. (2017). Synchronization routine for real-time synchronization of robotic total stations. In *INGENEO 2017: Proceedings of the 7th International Conference on Engineering Surveying*.
- Gu, Y., Zhou, C., Wieser, A., and Zhou, Z. (2017). Wifi based trajectory alignment, calibration and crowdsourced site survey using smart phones and foot-mounted imus. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–6.
- Gu, Y., Zhou, C., Wieser, A., and Zhou, Z. (2018). Trajectory estimation and crowdsourced radio map establishment from foot-mounted imus, wi-fi fingerprints, and gps positions. *IEEE Sensors Journal*, 19(3):1104–1113.
- Guan, K., Ma, L., Tan, X., and Guo, S. (2016). Vision-based indoor localization approach based on surf and landmark. In *2016 International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 655–659. IEEE.
- Halperin, D., Hu, W., Sheth, A., and Wetherall, D. (2011). Tool release: Gathering 802.11n traces with channel state information. *SIGCOMM Comput. Commun. Rev.*, 41(1):53–53.
- Hatcher, W. G. and Yu, W. (2018). A survey of deep learning: Platforms, applications and emerging research trends. *IEEE Access*, 6:24411–24432.
- Hazas, M. and Hopper, A. (2006). Broadband ultrasonic location systems for improved indoor positioning. *IEEE Transactions on mobile Computing*, 5(5):536–547.
- He, S. and Chan, S.-H. G. (2016). Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys & Tutorials*, 18(1):466–490.
- He, S., Ji, B., and Chan, S. H. G. (2016). Chameleon: Survey-free updating of a fingerprint database for indoor localization. *IEEE Pervasive Computing*, 15(4):66–75.
- He, S., Lin, W., and Chan, S. H. G. (2017). Indoor localization and automatic fingerprint update with altered ap signals. *IEEE Transactions on Mobile Computing*, 16(7):1897–1910.
- He, S. and Shin, K. G. (2018). Geomagnetism for smartphone-based indoor localization: Challenges, advances, and comparisons. *ACM Computing Surveys (CSUR)*, 50(6):97.

- Hightower, J., Want, R., and Borriello, G. (2000). Spoton: An indoor 3d location sensing technology based on rf signal strength.
- Huang, C. C. and Manh, H. N. (2016). RSS-Based Indoor Positioning Based on Multi-Dimensional Kernel Modeling and Weighted Average Tracking. *IEEE Sensors Journal*, 16(9):3231–3245.
- Ingram, S., Harmer, D., and Quinlan, M. (2004). Ultrawideband indoor positioning systems and their use in emergencies. In *Position Location and Navigation Symposium (PLANS)*, pages 706–715. IEEE.
- Jani, S. S., Lamb, J. M., White, B. M., Dahlbom, M., Robinson, C. G., and Low, D. A. (2015). Assessing margin expansions of internal target volumes in 3d and 4d pet: a phantom study. *Annals of Nuclear Medicine*, 29(1):100–109.
- Jun, J., He, L., Gu, Y., Jiang, W., Kushwaha, G., A, V., Cheng, L., Liu, C., and Zhu, T. (2017). Low-Overhead WiFi Fingerprinting. *IEEE Transactions on Mobile Computing*, 1233(c):1–14.
- Jung, S. and Han, D. (2018). Automated construction and maintenance of wi-fi radio maps for crowdsourcing-based indoor positioning systems. *IEEE Access*, 6:1764–1777.
- Jung, S., oh Lee, C., and Han, D. (2011). Wi-fi fingerprint-based approaches following log-distance path loss model for indoor positioning. In *2011 IEEE MTT-S International Microwave Workshop Series on Intelligent Radio for Future Personal Terminals*, pages 1–2.
- Kaemarungsi, K. and Krishnamurthy, P. (2012). Analysis of wlan’s received signal strength indication for indoor location fingerprinting. *Pervasive and Mobile Computing*, 8(2):292 – 316. Special Issue: Wide-Scale Vehicular Sensor Networks and Mobile Sensing.
- Kalbandhe, A. A. and Patil, S. C. (2016). Indoor positioning system using bluetooth low energy. In *2016 International Conference on Computing, Analytics and Security Trends (CAST)*, pages 451–455.
- Karegar, P. A. (2017). Wireless fingerprinting indoor positioning using affinity propagation clustering methods. *Wireless Networks*.
- Kasmi, Z., Norrdine, A., and Blankenbach, J. (2015). Towards a decentralized magnetic indoor positioning system. *Sensors*, 15(12):30319–30339.
- Kasprzak, S., Komninos, A., and Barrie, P. (2013). Feature-based indoor navigation using augmented reality. In *2013 9th International Conference on Intelligent Environments*, pages 100–107.

- Khalajmehrabadi, A., Gatsis, N., and Akopian, D. (2017). Structured group sparsity: A novel indoor wlan localization, outlier detection, and radio map interpolation scheme. *IEEE Transactions on Vehicular Technology*, 66(7):6498–6510.
- Kolodziej, K. W. and Hjelm, J. (2006). *Local positioning systems: LBS applications and services*. CRC press.
- Kushki, A., Plataniotis, K. N., and Venetsanopoulos, A. N. (2007). Kernel-based positioning in wireless local area networks. *IEEE Transactions on Mobile Computing*, 6(6):689–705.
- Kushki, A., Plataniotis, K. N., and Venetsanopoulos, A. N. (2010). Intelligent dynamic radio tracking in indoor wireless local area networks. *IEEE Transactions on Mobile Computing*, 9(3):405–419.
- Ledlie, J., g. Park, J., Curtis, D., Cavalcante, A., Camara, L., Costa, A., and Vieira, R. (2011). Mole: A scalable, user-generated wifi positioning engine. In *2011 International Conference on Indoor Positioning and Indoor Navigation*, pages 1–10.
- Lee, C., Chang, Y., Park, G., Ryu, J., Jeong, S.-G., Park, S., Park, J. W., Lee, H. C., shik Hong, K., and Lee, M. H. (2004). Indoor positioning system based on incident angles of infrared emitters. In *30th Annual Conference of IEEE Industrial Electronics Society (IECON)*, volume 3, pages 2218–2222.
- Lee, S., Jung, S., and Han, D. (2012). Uncaught signal imputation for accuracy enhancement of wlan-based positioning systems. In *Proceedings of the First ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems*, pages 80–85. ACM.
- Lemic, F., Handziski, V., Aernouts, M., Janssen, T., Berkvens, R., Wolisz, A., and Famaey, J. (2019). Regression-based estimation of individual errors in fingerprinting localization. *IEEE Access*, 7:33652–33664.
- Li, L., Yang, W., and Wang, G. (2013). Hiwl: An unsupervised learning algorithm for indoor wireless localization. In *2013 12th IEEE International Conference on Trust, Security and Privacy in Computing and Communications*, pages 1747–1753.
- Li, X., Wang, J., Liu, C., Zhang, L., and Li, Z. (2016). Integrated wifi/pdr/smartphone using an adaptive system noise extended kalman filter algorithm for indoor localization. *ISPRS International Journal of Geo-Information*, 5(2):8.
- Li, Y., Gao, Z., He, Z., Zhuang, Y., Radi, A., Chen, R., and El-Sheimy, N. (2019). Wireless fingerprinting uncertainty prediction based on machine learning. *Sensors*, 19(2):324.

- Little, R. J. (2015). *Missing Data/Imputation*, pages 1–5. American Cancer Society.
- Machaj, J., Brida, P., and Pich, R. (2011). Rank based fingerprinting algorithm for indoor positioning. In *2011 International Conference on Indoor Positioning and Indoor Navigation*, pages 1–6.
- Madigan, D., Einahrawy, E., Martin, R. P., Ju, W. H., Krishnan, P., and Krishnakumar, A. S. (2005). Bayesian indoor positioning systems. In *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, volume 2, pages 1217–1227.
- Mallat, S. G. and Zhang, Z. (1993). Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415.
- Marques, N., Meneses, F., and Moreira, A. (2012). Combining similarity functions and majority rules for multi-building, multi-floor, wifi positioning. In *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–9.
- Mautz, R. (2012). Indoor positioning technologies. *Habilitation Thesis, Department of Civil, Environmental and Geomatic Engineering, ETH Zurich, Switzerland.*
- Mautz, R. and Tilch, S. (2011). Survey of optical indoor positioning systems. In *Indoor Positioning and Indoor Navigation (IPIN), 2011 International Conference on*, pages 1–7. IEEE.
- Mendoza-Silva, G., Richter, P., Torres-Sospedra, J., Lohan, E., and Huerta, J. (2018a). Long-Term WiFi Fingerprinting Dataset for Research on Robust Indoor Positioning. *Data*, 3(1):3.
- Mendoza-Silva, G. M., Torres-Sospedra, J., and Huerta, J. (2018b). Locations selection for periodic radio map update in wifi fingerprinting. In Kiefer, P., Huang, H., Van de Weghe, N., and Raubal, M., editors, *Progress in Location Based Services 2018*, pages 3–24, Cham. Springer International Publishing.
- Minaev, G., Visa, A., and Pich, R. (2017). Comprehensive survey of similarity measures for ranked based location fingerprinting algorithm. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–4.
- Minami, M., Fukuju, Y., Hirasawa, K., Yokoyama, S., Mizumachi, M., Morikawa, H., and Aoyama, T. (2004). Dolphin: A practical approach for implementing a fully distributed indoor ultrasonic positioning system. In *International Conference on Ubiquitous Computing*, pages 347–365. Springer.

- Mirowski, P., Whiting, P., Steck, H., Palaniappan, R., MacDonald, M., Hartmann, D., and Ho, T. K. (2012). Probability kernel regression for wifi localisation. *Journal of Location Based Services*, 6(2):81–100.
- Mitchell, T. M. (1997). *Machine Learning*. McGraw-Hill, Inc., New York, NY, USA, 1 edition.
- Montoliu, R., Sansano, E., Belmonte, O., and Torres-Sospedra, J. (2018). A new methodology for long-term maintenance of wifi fingerprinting radio maps. In *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–7.
- Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.
- Nandakumar, R., Chintalapudi, K. K., and Padmanabhan, V. N. (2012). Centaur: Locating devices in an office environment. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking, Mobicom '12*, pages 281–292, New York, NY, USA. ACM.
- Niedermayr, S., Wieser, A., and Neuner, H. (2014). Expressing location uncertainty in combined feature-based and geometric positioning. In *Proceedings European Navigation Conference 2014*, pages 154–166. EUGIN.
- Norrdine, A., Grimm, D., Blankenbach, J., and Wieser, A. (2013). An uwb based indoor compass for accurate heading estimation in buildings. In *4th International Conference on Indoor Positioning and Indoor Navigation (IPIN 2013)*, pages 116–119. IEEE.
- Norrdine, A., Kasmi, Z., and Blankenbach, J. (2016). Step detection for zupt-aided inertial pedestrian navigation system using foot-mounted permanent magnet. *IEEE Sensors Journal*, 16(17):6766–6773.
- Nuño-Barrau, G. and Páez-Borrillo, J. M. (2006). A new location estimation system for wireless networks based on linear discriminant functions and hidden markov models. *Eurasip Journal on Applied Signal Processing*, 2006:1–17.
- Ouyang, R. W., Wong, A. K. S., Lea, C. T., and Chiang, M. (2012). Indoor location estimation with reduced calibration exploiting unlabeled data via hybrid generative/discriminative learning. *IEEE Transactions on Mobile Computing*, 11(11):1613–1626.
- Padmanabhan, P. B., N., V., and N., V. (2000). RADAR: An in-building RF based user location and tracking system. *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No.00CH37064)*, 2(c):775–784.

- Park, J.-g., Charrow, B., Curtis, D., Battat, J., Minkov, E., Hicks, J., Teller, S., and Ledlie, J. (2010). Growing an organic indoor location system. In *Proceedings of the 8th international conference on Mobile systems, applications, and services*, pages 271–284. ACM.
- Park, T. and Casella, G. (2008). The bayesian lasso. *Journal of the American Statistical Association*, 103(482):681–686.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Pei, L., Zhang, M., Zou, D., Chen, R., and Chen, Y. (2016). A survey of crowd sensing opportunistic signals for indoor localization. *Mobile Information Systems*, 2016.
- Peng, Y., Niu, X., Tang, J., Mao, D., and Qian, C. (2018). Fast signals of opportunity fingerprint database maintenance with autonomous unmanned ground vehicle for indoor positioning. *Sensors (Switzerland)*, 18(10):1–18.
- Potort, F., Crivello, A., Barsocchi, P., and Palumbo, F. (2018). Evaluation of indoor localisation systems : comments on the ISO / IEC 18305 standard. (September):24–27.
- Powell, C. (1958). The decca navigator system for ship and aircraft use. *Proceedings of the IEE-Part B: Radio and Electronic Engineering*, 105(9S):225–234.
- Radu, V. and Marina, M. K. (2013). Himloc: Indoor smartphone localization via activity aware pedestrian dead reckoning with selective crowdsourced wifi fingerprinting. In *International Conference on Indoor Positioning and Indoor Navigation*, pages 1–10.
- Renaudin, O., Zemen, T., and Burgess, T. (2018). Ray-Tracing Based Fingerprinting for Indoor Localization. In *2018 IEEE 19th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5.
- Retscher, G. and Joksch, J. (2016). Comparison of different vector distance measure calculation variants for indoor location fingerprinting. In *Location-Based Services (LBSs), 2016 International Conference on*, pages 53–76. ICA Commission on LBSs.
- Röbesaat, J., Zhang, P., Abdelaal, M., and Theel, O. (2017). An improved ble indoor localization with kalman-based fusion: An experimental study. *Sensors*, 17(5):951.

- Sansano, E., Montoliu, R., Belmonte, O., and Torres-Sospedra, J. (2016). Uji indoor positioning and navigation repository.
- Saxena, A. and Zawodniok, M. (2014). Indoor positioning system using geomagnetic field. In *2014 IEEE International Instrumentation and Measurement Technology Conference (I2MTC) Proceedings*, pages 572–577.
- Schloemann, J., Dhillon, H. S., and Buehrer, R. M. (2016). Toward a tractable analysis of localization fundamentals in cellular networks. *IEEE Transactions on Wireless Communications*, 15(3):1768–1782.
- Schulz, M., Link, J., Gringoli, F., and Hollick, M. (2018). Shadow wi-fi: Teaching smartphones to transmit raw signals and to extract channel state information to implement practical covert channels over wi-fi. In *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys '18*, pages 256–268, New York, NY, USA. ACM.
- Seitz, J., Jahn, J., Boronat, J. G., Vaupel, T., Meyer, S., and Thielecke, J. (2010). A hidden markov model for urban navigation based on fingerprinting and pedestrian dead reckoning. In *2010 13th International Conference on Information Fusion*, pages 1–8. IEEE.
- Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., and De Freitas, N. (2016). Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175.
- Sorour, S., Lostanlen, Y., Valaee, S., and Majeed, K. (2015). Joint indoor localization and radio map construction with limited deployment load. *IEEE Transactions on Mobile Computing*, 14(5):1031–1043.
- Sun, Y., He, Y., Meng, W., and Zhang, X. (2018). Voronoi diagram and crowdsourcing-based radio map interpolation for GRNN fingerprinting localization using WLAN. *Sensors (Switzerland)*, 18(10):5–7.
- Talvitie, J., Renfors, M., and Lohan, E. S. (2015). Distance-based interpolation and extrapolation methods for rss-based localization with indoor wireless signals. *IEEE Transactions on Vehicular Technology*, 64(4):1340–1353.
- Tang, J., Chen, Y., Chen, L., Liu, J., Hyypäe, J., Kukko, A., Kaartinen, H., Hyypäe, H., and Chen, R. (2015). Fast fingerprint database maintenance for indoor positioning based on ugv slam. *Sensors*, 15(3):5311–5330.
- Taniuchi, D. and Maekawa, T. (2015). Automatic update of indoor location fingerprints with pedestrian dead reckoning. *ACM Trans. Embed. Comput. Syst.*, 14(2):27:1–27:23.

- Tao, Y. and Zhao, L. (2018). A novel system for wifi radio map automatic adaptation and indoor positioning. *IEEE Transactions on Vehicular Technology*, 67(11):10683–10692.
- Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288.
- Torres-Sospedra, J., Jiménez, A., Knauth, S., Moreira, A., Beer, Y., Fetzer, T., Ta, V.-C., Montoliu, R., Seco, F., Mendoza-Silva, G., et al. (2017). The smartphone-based offline indoor location competition at ipin 2016: Analysis and future work. *Sensors*, 17(3):557.
- Torres-Sospedra, J., Montoliu, R., Martnez-Us, A., Avariento, J. P., Arnau, T. J., Benedito-Bordonau, M., and Huerta, J. (2014). Ujiindoorloc: A new multi-building and multi-floor database for wlan fingerprint-based indoor localization problems. In *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 261–270.
- Torres-Sospedra, J., Montoliu, R., Trilles, S., Belmonte, O., and Huerta, J. (2015a). Comprehensive analysis of distance and similarity measures for wi-fi fingerprinting indoor positioning systems. *Expert Systems with Applications*, 42(23):9263 – 9278.
- Torres-Sospedra, J. and Moreira, A. (2017a). Analysis of Sources of Large Positioning Errors in Deterministic Fingerprinting. *Sensors (Switzerland)*, pages 1–48.
- Torres-Sospedra, J. and Moreira, A. (2017b). Analysis of sources of large positioning errors in deterministic fingerprinting. *Sensors*, 17(12):2736.
- Torres-Sospedra, J., Rambla, D., Montoliu, R., Belmonte, O., and Huerta, J. (2015b). Ujiindoorloc-mag: A new database for magnetic field-based localization problems. In *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–10.
- Vadeny, D., Chen, M., Huang, E., and Elgala, H. (2016). Vslam and vlc based localization. *Microsoft Indoor Localization Competition, Tech. Rep.*
- Wang, H., Sen, S., Elgohary, A., Farid, M., Youssef, M., and Choudhury, R. R. (2012). No need to war-drive: Unsupervised indoor localization. In *Proceedings of the 10th international conference on Mobile systems, applications, and services*, pages 197–210. ACM.
- Wang, J., Hu, A., Liu, C., and Li, X. (2015). A floor-map-aided wifi/pseudodometry integration algorithm for an indoor positioning system. *Sensors*, 15(4):7096–7124.

- Wang, K., Nirmalathas, A., Lim, C., Alameh, K., Li, H., and Skafidas, E. (2017). Indoor infrared optical wireless localization system with background light power estimation capability. *Optics Express*, 25(19):22923.
- Wang, Z. and de Freitas, N. (2014). Theoretical analysis of bayesian optimisation with unknown gaussian process hyper-parameters. *CoRR*, abs/1406.7758.
- Want, R., Hopper, A., Falcao, V., and Gibbons, J. (1992). The active badge location system. *ACM Transactions on Information Systems (TOIS)*, 10(1):91–102.
- Wu, C., Yang, Z., and Liu, Y. (2018a). *Wireless Indoor Localization: A Crowdsourcing Approach*. Springer.
- Wu, C., Yang, Z., Liu, Y., and Xi, W. (2013). Will: Wireless indoor localization without site survey. *IEEE Transactions on Parallel and Distributed Systems*, 24(4):839–848.
- Wu, C., Yang, Z., and Xiao, C. (2018b). Automatic radio map adaptation for indoor localization using smartphones. *IEEE Transactions on Mobile Computing*, 17(3):517–528.
- Wu, C., Yang, Z., Zhou, Z., Liu, Y., and Liu, M. (2017). Mitigating large errors in wifi-based indoor localization for smartphones. *IEEE Transactions on Vehicular Technology*, 66(7):6246–6257.
- Wu, C.-L., Fu, L.-C., and Lian, F.-L. (2004). Wlan location determination in e-home via support vector classification. In *IEEE International Conference on Networking, Sensing and Control, 2004*, volume 2, pages 1026–1031 Vol.2.
- Xie, H., Gu, T., Tao, X., Ye, H., and Lu, J. (2016). A Reliability-Augmented Particle Filter for Magnetic Fingerprinting Based Indoor Localization on Smartphone. *IEEE Transactions on Mobile Computing*, 15(8):1877–1892.
- Xu, H., Ding, Y., Li, P., Wang, R., and Li, Y. (2017). An rfid indoor positioning algorithm based on bayesian probability and k-nearest neighbor. *Sensors*, 17(8):1806.
- Yang, J., Li, Y., Cheng, W., Liu, Y., and Liu, C. (2018). EKF-gpr-based fingerprint renovation for subset-based indoor localization with adjusted cosine similarity. *Sensors*, 18(1):318.
- Yassin, A., Nasser, Y., Awad, M., Al-Dubai, A., Liu, R., Yuen, C., Raulefs, R., and Aboutanios, E. (2017). Recent Advances in Indoor Localization: A Survey on Theoretical Approaches and Applications. *IEEE Communications Surveys Tutorials*, 19(2):1327–1346.

- Youssef, M. and Agrawala, A. (2008). The Horus location determination system. *Wireless Networks*, 14(3):357–374.
- Youssef, M. A., Agrawala, A., and Shankar, A. U. (2003). Wlan location determination via clustering and probability distributions. In *Pervasive Computing and Communications, 2003.(PerCom 2003). Proceedings of the First IEEE International Conference on*, pages 143–150. IEEE.
- Zaidi, A. and Suddle, M. (2006). Global navigation satellite systems: a survey. In *2006 international conference on advances in space technologies*, pages 84–87. IEEE.
- Zeinalipour-Yazti, D. and Laoudias, C. (2017). The anatomy of the anyplace indoor navigation service. *SIGSPATIAL Special*, 9(2):3–10.
- Zhang, T. (2011). Adaptive forward-backward greedy algorithm for learning sparse representations. *IEEE transactions on information theory*, 57(7):4689–4708.
- Zhou, C. (2019). Kinematically collected reference fingerprint map (RFM) with the high precision tracking system for feature-based indoor positioning.
- Zhou, C. and Gu, Y. (2017). Joint positioning and radio map generation based on stochastic variational bayesian inference for fwips. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–10. IEEE.
- Zhou, C. and Wieser, A. (2016). Application of backpropagation neural networks to both stages of fingerprinting based wips. In *2016 Fourth International Conference on Ubiquitous Positioning, Indoor Navigation and Location Based Services (UPINLBS)*, pages 207–217. IEEE.
- Zhou, C. and Wieser, A. (2018a). Cdm: Compound dissimilarity measure and an application to fingerprinting-based positioning. In *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–7.
- Zhou, C. and Wieser, A. (2018b). *Jaccard Analysis and LASSO-Based Feature Selection for Location Fingerprinting with Limited Computational Complexity*, pages 71–87. Springer International Publishing, Cham.
- Zhou, C. and Wieser, A. (2019a). An iterative scheme for feature-based positioning using weighted dissimilarity measure. *Sensors*, xx(xx):1 – 15.
- Zhou, C. and Wieser, A. (2019b). Modified jaccard index analysis and adaptive feature selection for location fingerprinting with limited computational complexity. *Journal of Location Based Services*, 13(2):128–157.

Zhuang, Y., Yang, J., Li, Y., Qi, L., and El-Sheimy, N. (2016). Smartphone-based indoor localization with bluetooth low energy beacons. *Sensors*, 16(5):596.

Zou, H., Jin, M., Jiang, H., Xie, L., and Spanos, C. J. (2017). WinIPS: WiFi-based non-intrusive indoor positioning system with online radio map construction and adaptation. *IEEE Transactions on Wireless Communications*, 16(12):8118–8130.

APPENDIX A

LIST OF ACRONYMS

| | |
|--------|---------------------------------------------------|
| ACDM | averagely weighted compound dissimilarity measure |
| AFBGS | adaptive forward-backward greedy search |
| AP | access point |
| ATR | automatic target recognition |
| BIC | Bayesian information criterion |
| BLE | Bluetooth low energy |
| CDM | compound dissimilarity measure |
| CE | circular error |
| CSI | channel state information |
| CV | cross validation |
| ECDF | empirical cumulative distribution function |
| EKF | extended Kalman filter |
| FbP | fingerprinting-based positioning |
| FM | frequency modulation |
| FIPS | feature-based indoor positioning system |
| GNSS | global navigation satellite system |
| GP | Gaussian process |
| GPR | Gaussian process regression |
| GPS | global positioning system |
| IMU | inertial measurement unit |
| k NN | k -nearest-neighbors |
| KS | kernel smoothing |
| LASSO | least absolute shrinkage and selection operator |

| | |
|-------|----------------------------------------------------|
| LBS | location-based service |
| LDA | linear discriminant analysis |
| LDPLM | Log-distance path loss model |
| MAC | media access control |
| MAD | median absolute deviation |
| MAE | mean absolute error |
| MAP | maximum a posteriori |
| MCD | minimum covariance determinant |
| MJI | modified Jaccard index |
| MLE | maximum likelihood estimation |
| MSE | mean squared error |
| PDR | pedestrian dead reckoning |
| PLSR | partial least square regression |
| PSFM | polynomial surface fitting mean |
| RCDM | relatively weighted compound dissimilarity measure |
| RF | radio frequency |
| RFID | radio frequency identification |
| RFM | reference fingerprint map |
| RKHS | reproducing kernel Hilbert space |
| RMSE | root mean squared error |
| RoI | region of interest |
| RP | reference point |
| RSS | received signal strength |
| SLAM | simultaneous localization and mapping |
| SoP | signals of opportunity |
| STD | standard deviation |
| SVM | support vector machine |
| SVR | support vector regression |
| TDE | time delay estimation |
| TF | termination flag |
| TP | test position |
| UGV | unmanned ground vehicle |
| UTC | coordinated universal time |
| UWB | ultra-wideband |
| WLAN | wireless local area network |

APPENDIX B

LIST OF PUBLICATIONS

1. **Zhou, Caifa**, and Andreas Wieser. 2019. Modified Jaccard Index Analysis and Adaptive Feature Selection for Location Fingerprinting with Limited Computational Complexity. *Journal of Location Based Services* 13 (2). Taylor & Francis:128–157. <https://doi.org/10.1080/17489725.2019.1577505>.
2. Gojcic, Zan, **Caifa Zhou**, and Andreas Wieser. 2019. Feature-based approaches for geomonitoring using terrestrial laser scanning point clouds. 1st Swiss Workshop on Machine Learning for Environmental and Geosciences (MLEG 2019), Dübendorf, Switzerland, January 16-17, 2019.
3. Gojcic, Zan, **Caifa Zhou**, and Andreas Wieser. 2019. Robust Point Correspondences for Point Cloud Based Deformation Monitoring of Natural Structures. 4th Joint International Symposium on Deformation Monitoring (JISDM), 15-17 May 2019, Athens, Greece. (Awarded as *Best Oral Presentation*)
4. Gojcic, Zan, **Caifa Zhou**, Jan D. Wegner, and Andreas Wieser. 2019. The Perfect Match: 3D Point Cloud Matching with Smoothed Densities. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2019. <http://arxiv.org/abs/1811.06879>.
5. **Zhou, Caifa**, and Andreas Wieser. 2019. An Iterative Scheme for Feature-based Positioning using a Weighted Dissimilarity Measure. *Sensors xx (xx)*. Multidisciplinary Digital Publishing Institute:1–15. (Submitted)

6. **Zhou, Caifa**, and Andreas Wieser. 2018. Jaccard Analysis and LASSO-Based Feature Selection for Location Fingerprinting with Limited Computational Complexity. In *Progress in Location Based Services 2018*, edited by Peter Kiefer, Haosheng Huang, Nico de Weghe, and Martin Raubal, 71–87. Cham: Springer International Publishing.
7. Gojcic, Zan, **Caifa Zhou**, and Andreas Wieser. 2018. Learned Compact Local Feature Descriptor for TLS-based Geodetic Monitoring of Natural Outdoor Scenes. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 4 (2):113–120. <https://doi.org/10.5194/isprs-annals-IV-2-113-2018>.
8. **Zhou, Caifa**, and Andreas Wieser. 2018. CDM: Compound Dissimilarity Measure and an Application to Fingerprinting-Based Positioning. In *Indoor Positioning and Indoor Navigation (IPIN), 2018 International Conference On*, 1–7. Nantes, France: IEEE. (Awarded as *Best Student Paper*)
9. Gu, Yang, **Caifa Zhou**, Andreas Wieser, and Zhimin Zhou. 2018. Trajectory Estimation and Crowdsourced Radio Map Establishment from Foot-Mounted IMUs, Wifi Fingerprints and GPS Positions. *IEEE Sensors Journal*, 1. <https://doi.org/10.1109/JSEN.2018.2877804>.
10. **Zhou, Caifa**, and Yang Gu. 2017. Joint Positioning and Radio Map Generation Based on Stochastic Variational Bayesian Inference for FWIPS. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 1–10. <https://doi.org/10.1109/IPIN.2017.8115881>.
11. Gu, Yang, **Caifa Zhou**, Andreas Wieser, and Zhimin Zhou. 2017. Pedestrian Positioning Using WiFi Fingerprints and a Foot-Mounted Inertial Sensor. In *2017 European Navigation Conference (ENC)*, 91–99.
12. Gu, Yang, **Caifa Zhou**, Andreas Wieser, and Zhimin Zhou. 2017. WiFi Based Trajectory Alignment, Calibration and Crowdsourced Site Survey Using Smart Phones and Foot-Mounted IMUs. In *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 1–6.
13. **Zhou, Caifa**, and Andreas Wieser. 2016. Application of Backpropagation Neural Networks to Both Stages of Fingerprinting Based WIPS. In *2016 Fourth International Conference on Ubiquitous Positioning, Indoor Navigation and Location Based Services (UPINLBS)*, 207–217. Piscataway, NJ: IEEE.
14. Xia, Yin, Lin Ma, Zhongzhao Zhang, and **Caifa Zhou**. 2014. WLAN Indoor Positioning Algorithm Based on Semi-Supervised Manifold

Learning. *Journal of Systems Engineering and Electronics* 36 (7):1422–1428. <https://doi.org/10.3969/j.issn.1001-506X.2014.07.31.READS>.

15. **Zhou, Caifa**, Lin Ma, and Xuezhi Tan. 2014. Joint Semi-Supervised RSS Dimensionality Reduction and Fingerprint Based Algorithm for Indoor Localization. *Proceedings of the 27th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2014)*, Tampa, Florida, September 2014, 3201–3211.
16. Ma, Lin, **Caifa Zhou**, Danyang Qin, and Yubin Xu. 2014. Green Wireless Local Area Network Received Signal Strength Dimensionality Reduction and Indoor Localization Based on Fingerprint Algorithm. *International Journal of Communication Systems* 27 (5):4527–4542.
17. Ma, Lin, **Caifa Zhou**, Xi Liu, and Yubin Xu. 2013. Adaptive Neighboring Selection Algorithm Based on Curvature Prediction in Manifold Learning. *Journal of Harbin Institute of Technology* 20 (3):119–123.

APPENDIX C

CURRICULUM VITAE

Personal data

| | |
|---------------|--------------|
| Name | Caifa Zhou |
| Date of birth | Jul 11, 1990 |
| Nationality | China |

Education

| | |
|---------------------|------------------------------------------------------------------------------------------------|
| Sep 2015 – Jul 2019 | ETH Zurich, Switzerland PhD Student |
| Sep 2013 – Jun 2015 | Harbin Institute of Technology, China M.Sc., Information & Communication Engineering |
| Sep 2009 – Jun 2013 | Harbin Institute of Technology, China B.E., Communication Engineering |

Professional experience

| | |
|---------------------|-------------------------------------------------------------------|
| Sep 2015 – Jul 2019 | ETH Zurich, Switzerland Research and Teaching Assistant |
| Sep 2013 – Jul 2015 | Harbin Institute of Technology, China IT Assistant |