

An indoor pedestrian localisation system with self-calibration capability

A THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE
DEGREE OF DOCTOR OF PHILOSOPHY

IN THE

DEPARTMENT OF ELECTRONIC AND ELECTRICAL ENGINEERING

THE UNIVERSITY OF SHEFFIELD

By

Yang Liu

-2016-

ABSTRACT

The Global Positioning System (GPS), a space-based system, employs dozens of satellites to provide location determination and navigation services around the world. However, due to the constraints to the power consuming and long-distance transmission, the strength of the GPS signal received on the mobile device is weak. Errors of the detection of the line-of-sight (LOS) propagated components of the signals are expected to be high if the users are in urban areas or in buildings, since obstacles in the surrounding environments could attenuate the LOS propagated components of the GPS signals significantly, but might upfade the multi-path components (constructive multi-path effect). Therefore, GPS should be replaced by other techniques for providing localisation services in urban areas or, especially, in indoor environments.

Among all the candidates, received signal strength (RSS) location fingerprint based positioning systems attract great attentions from both the academia and industry. Usually, a time-consuming and labour-intensive site survey to collect dozens of training samples of RSS from access points (APs) in range on every reference position (RP) in the area of interest is required to build the radio map (RM), before the localisation services could be provided to users. The purpose of the thesis is to reduce the workload involved in the site survey while providing accurate localisation service from two aspects, as shown as follows.

Firstly, the quantity of the training samples collected on each RP is reduced, by taking advantage of the on-line RSS measurements collected by users to calibrate the RM. The on-line RSS measurements are geo-tagged probabilistically by an implementation of particle filter to track the trajectories of the users. The employed particles in estimation of the users' states are initialised by a supervised clustering algorithm, propagated according to the analysis of the data sourcing from inertial measurement units (IMUs), e.g., walking detection, orientation estimation, step and stepping moments detection, step length detection, etc., and corrected by the wall constraints. Furthermore, the importance weights of the particles are adjusted to reduce the negative influence of the multi-clustered distribution of the particles to the on-line localisation accuracy, by applying the on-line RSS-based localisation results when significant users' body turnings are detected. The final results confirm that the accuracy of the localisation service with the RM calibrated by the method proposed in this thesis is higher than the one proposed in the [1].

Secondly, a semi-automatic site-survey method which takes advantage of a route-planning algorithm and a walking detection module to recognise automatically the

index of the RP for the current site-survey task, inform the system automatically of the start/end of the process of the task on the current RP and switch automatically to the following RPs on the planned route for the following tasks. In this way, human beings' intervention to the site-survey process is greatly reduced. As a result, the errors made in the site-survey tasks, such as incorrect recognition of the index of the RP for the current task which is highly likely to occur when the technicians get absent-minded in the work, misoperations to start/end of the task for collecting RSS samples on the current RP at wrong time moments, forgetting to notify the system of the fact that the technician has moved on to the next RP, etc., are avoided. The technicians no longer feel bored or anxious in the process of fulfilment of site-survey tasks, and the working efficiency and robustness of the RM could be also improved.

Acknowledgements

Four-year study in Sheffield has given me very happy memories. During the period, not only I have become an researcher with independent research capability, but also I have found my wife and married her.

First and foremost, I want to take this opportunity to express my great thanks to my supervisor professor Jie Zhang for his great supports and encouragements throughout my four-year study.

Also, I would like to thank all academics and colleagues in the Communication group in the University of Sheffield and Bell laboratories at Dublin. Special thanks are given to Dr. Marzieh Dashti, Dr. Hamed Mahmoudi and Dr. Holger Claussen, for their precious guidance, suggestions and generous financial support for my five-month research in Dublin.

During the final thesis writing-up stage, Dr. Qing Lu and Dr. David Abson helped me a lot in proofreading the draft of the thesis. Please allow me to take this opportunity to express my extreme thanks to both of you. Especially, Dr. David Abson's old-school fashion of writing the comments and marking the incorrectness on printed copies of the thesis impresses me a lot. I will always keep them as a piece of precious memory in my life. Also, I will never forget that Dr. David Abson had been to a post office to deliver the reviewed draft to me chapter-by-chapter for several times, until all contents have been fully reviewed.

In addition, sincere appreciation is given to the examiners of the thesis, Dr. Xiaoli Chu and Dr. Andrew H. Kemp, for the most valuable suggestions and corrections to the work from them, which helps me a lot to improve the quality of the thesis.

Last but not least, please also allow me to express my deepest gratitude to my family, including my wife Xiaoyi Liu, my parents and my parents-in-law. Their warming encourages and supports with no hesitation at all have moved and inspired me a lot.

List of Publications

Liu, Y., Dashti, M., Claussen, H., & Zhang, J. (2015, March). Semi-automatic site survey for indoor localization. In Proceedings of *Positioning Navigation and Communication (WPNC) 2015 12th Workshop on*.

Liu, Y., Dashti, M., Rahman, A., Amiruddin, M., & Zhang, J. (2014). Indoor localization using smartphone inertial sensors. In Proceedings of *Positioning, Navigation and Communication (WPNC), 2014 11th Workshop on*.

Liu, Y., Dashti, M., & Zhang, J. (2013). Indoor localization on mobile phone platforms using embedded inertial sensors. In Proceedings of *Positioning Navigation and Communication (WPNC) 2013 10th Workshop on*.

Contents

| | |
|---|-------------|
| ABSTRACT | i |
| Acknowledgements | iii |
| List of Publications | iv |
| Contents | v |
| List of Figures | ix |
| List of Tables | xii |
| Abbreviations | xiii |
| 1 Introduction | 1 |
| 1.1 Motivation | 1 |
| 1.2 Overview of the RSS-based indoor localisation methods | 5 |
| 1.3 Overview of the IMUs | 6 |
| 1.4 Scope and contribution | 8 |
| 2 Literature Review | 12 |
| 2.1 Localisation technologies | 13 |
| 2.1.1 Infra-red | 13 |
| 2.1.2 Ultra-wideband | 14 |
| 2.1.3 Ultrasound | 14 |
| 2.1.4 Wi-Fi | 15 |
| 2.1.5 LED light | 15 |
| 2.2 Mathematical methods for localisation | 16 |
| 2.2.1 Triangulation methods | 16 |
| 2.2.1.1 Multi-lateration | 16 |
| 2.2.1.2 Multi-angulation | 18 |
| 2.2.2 RSS-based methods | 18 |
| 2.2.2.1 Empirical propagation model | 19 |
| 2.2.2.2 Location fingerprinting method | 19 |
| 2.3 Comparison methods for LF technique | 21 |
| 2.3.1 Discretisation | 23 |
| 2.3.1.1 Discretisation bias and variance | 24 |

| | | |
|----------|---|-----------|
| 2.3.2 | Kernel method | 26 |
| 2.4 | The applied machine learning algorithms | 27 |
| 2.4.1 | k -means and k -means++ | 28 |
| 2.4.2 | Affinity propagation clustering algorithm | 29 |
| 2.4.3 | SVM classifiers | 31 |
| 2.4.3.1 | Binary SVM | 31 |
| 2.4.3.2 | Multi-class SVMs | 32 |
| 2.4.4 | Fisher kernel | 33 |
| 2.4.5 | Generative approaches vs. discriminative approaches | 34 |
| 2.5 | DR methods based on the IMUs | 35 |
| 2.5.1 | Frequently-calibrated PDR | 37 |
| 2.5.2 | Step-based PDR | 38 |
| 2.6 | Route planning and application | 39 |
| 2.7 | Recursive Bayesian estimation | 41 |
| 2.7.1 | Kalman filter | 43 |
| 2.7.2 | Particle filter | 44 |
| 2.8 | Chapter Summary | 45 |
| 3 | Inertial navigation and pedestrian trajectory recognition | 47 |
| 3.1 | Introduction | 47 |
| 3.2 | Inertial sensors of smartphones | 48 |
| 3.2.1 | Gyroscope | 48 |
| 3.2.1.1 | Gyroscope measurement error | 49 |
| 3.2.2 | Accelerometer | 50 |
| 3.2.2.1 | Accelerometer measurement error | 50 |
| 3.2.3 | Magnetometer | 50 |
| 3.2.3.1 | Magnetometer measurement error | 51 |
| 3.3 | Conventional inertial navigation tracking method | 52 |
| 3.3.1 | Distance updating | 52 |
| 3.3.2 | Attitude estimation | 53 |
| 3.3.2.1 | Estimation of instantaneous attitude and tracking of the change via inertial sensors | 54 |
| 3.3.2.2 | Implementation of a complementary filter for attitude estimation | 57 |
| 3.3.3 | Evaluation results of the DR method | 59 |
| 3.3.3.1 | Evaluation results of attitude estimation | 59 |
| 3.3.3.2 | Evaluation results of travelling distance updated by taking integration of acceleration | 60 |
| 3.4 | Step-based PDR and trajectory recognition | 61 |
| 3.4.1 | Walking detection | 62 |
| 3.4.2 | Estimation of angle difference between user's walking direction and phone's orientation | 64 |
| 3.4.3 | Detection of steps and stepping moments | 70 |
| 3.4.4 | Step length estimation | 73 |
| 3.4.5 | Recognition of the PDR moving trajectory | 75 |
| 3.5 | Conclusion | 78 |

| | | |
|----------|---|------------|
| 4 | Wi-Fi indoor localisation | 80 |
| 4.1 | Introduction | 80 |
| 4.2 | Experimental Setup | 81 |
| 4.2.1 | Indoor area for experiments | 81 |
| 4.2.2 | Wi-Fi fingerprints collection | 81 |
| 4.3 | Rough localisation | 83 |
| 4.4 | Fine localisation | 89 |
| 4.4.1 | Analysis of the effect of varying bin width | 89 |
| 4.4.2 | Analysis of the effect of varying the number of training samples collected on each RP | 91 |
| 4.5 | Chapter Summary | 93 |
| 5 | Applying a particle filter for indoor pedestrian localisation and tracking | 94 |
| 5.1 | Introduction | 94 |
| 5.2 | Mathematical representation of environment constraints | 95 |
| 5.2.1 | Walls representation | 95 |
| 5.2.2 | Calculation of inter-RP distance | 96 |
| 5.2.3 | Automatic arrangement of cells to RPs | 97 |
| 5.3 | Implementation of particle filter | 98 |
| 5.3.1 | Particle propagation | 100 |
| 5.3.2 | Particle correction | 102 |
| 5.3.3 | Particle Re-sampling | 104 |
| 5.3.4 | Initialisation of particles for the filter | 106 |
| 5.4 | Experimental evaluation | 108 |
| 5.4.1 | Analysis of the effect of different numbers of particles | 108 |
| 5.4.2 | Analysis of the effect of applying Wi-Fi RSS clustering algorithm | 110 |
| 5.4.3 | Analysis the effect of applying Wi-Fi RSS fine localisation algorithm | 113 |
| 5.4.4 | Analysis of the effect of adjustment of importance weights of particles by applying the on-line W-Fi localisation results after significant user's turns are detected | 118 |
| 5.4.4.1 | The first path | 119 |
| 5.4.4.2 | The second path | 119 |
| 5.4.4.3 | The third path | 122 |
| 5.4.4.4 | The fourth path | 123 |
| 5.5 | Chapter summary | 124 |
| 6 | Using users' on-line measurements to calibrate the RM | 127 |
| 6.1 | Implementation of the EM algorithm | 129 |
| 6.2 | Calibrating the RM by taking advantage of the profiles of final surviving particles | 132 |
| 6.2.1 | Recording the particle evolving information in real-time | 132 |
| 6.2.2 | Back-tracking the final surviving particles, and calibrating the RM | 135 |

| | | |
|----------|--|------------|
| 6.3 | The semi-automatic site survey method | 137 |
| 6.4 | Experimental evaluation | 140 |
| 6.4.1 | The evaluation of the back-tracking method in the off-line phase | 141 |
| 6.4.2 | Comparison of the two RM calibration methods | 142 |
| 6.4.3 | Performance evaluation of the semi-automatic site survey approach | 147 |
| 6.5 | Chapter summary | 149 |
| 7 | Conclusion | 151 |
| 7.1 | Future work | 153 |

List of Figures

| | | |
|------|---|----|
| 1.1 | screen-shots of two GPS mobile applications: Easytrails GPS and iCartel. | 3 |
| 1.2 | The frequency ranges of all channels of Wi-Fi at 2.4 GHz. | 7 |
| 2.1 | Positioning with TOA method, reproduced from [2] | 16 |
| 2.2 | Positioning with the TDOA method, reproduced from [2] | 17 |
| 2.3 | Bias and variance in the distribution of arrows landing on a target. Bias implies that the archer misses in the same direction, systematically. Variance implies that the archers are scattered ([3]). | 26 |
| 2.4 | Stable-platform IMU, from[4]. | 36 |
| 2.5 | An example of graph model | 40 |
| 3.1 | Smartphone Reference Frame, from [5] | 49 |
| 3.2 | Global Reference Frame | 52 |
| 3.3 | Strapdown DR method | 54 |
| 3.4 | The estimation of yaw attitude of the smartphone held by a technician | 60 |
| 3.5 | Showcase of the inferred acceleration and velocity in the global reference frame. | 61 |
| 3.6 | Raw acceleration magnitude and movement hinter | 64 |
| 3.7 | Both sides of the smartphone are held by the palm of a user. | 65 |
| 3.8 | The variation trend of linear acceleration over the moving direction. Four gaits are shown particularly. The bottom and peak of the forward acceleration emerges in the duration of transforming the gait from the first one to the second one, and from the third one to the fourth one, respectively. | 67 |
| 3.9 | Showcase of the accelerations recorded along the x and y axis in the phone's location reference frame, when the user is walking. In this case, the moving direction aligns with the y axis | 68 |
| 3.10 | Showcase of the magnitude spectrum of the accelerations along the x and y axis in the phone's location reference frame, when the user is walking. | 68 |
| 3.11 | Vertical view of the accelerations \mathbf{a}_{moving} and \mathbf{a}_{side} on the x and y axis in the phone's local reference frame when a user's foot hits on the ground. In the two cases, a phone is held by the user with opposite angles to the moving direction. Note that the direction of \mathbf{a}_{side} could be opposite to the one indicated in the figure. | 69 |

| | | |
|------|---|-----|
| 3.12 | Illustration of the histogram of the difference between the estimate and the true value of (a): $\Upsilon = +15^\circ$ and (b): $\Upsilon = -15^\circ$; the red solid line highlights the fitted bimodal Gaussian distribution function. | 70 |
| 3.13 | Acceleration magnitude for the four continuous steps with smartphone in hand. | 72 |
| 3.14 | The step-counting error for different values of ρ and δ with smartphone in hand. | 73 |
| 3.15 | Linear relation between step length and walking frequency is learned | 75 |
| 3.16 | Overall trajectory recognition scheme. | 77 |
| 3.17 | A user is turning his body with an angle of θ at the time t . Position changes are calculated using triangulation. Here, the global reference frame defined in Figure 3.2 is applied. | 78 |
| 3.18 | An example of estimation of the tester's moving trajectory. | 78 |
| 4.1 | Map of the indoor area for experiments. Reference points are marked as black solid circles | 82 |
| 4.2 | The number of occurrences of different RSS levels in a RM with hundreds of RSS samples collected on every RP, indexed from low to high. | 82 |
| 4.3 | The averaged probability of missing a RSS VS. mean RSS level in every Wi-Fi scan result. | 83 |
| 4.4 | Clustering accuracy VS. number of clusters | 85 |
| 4.5 | Distribution of member RPs of clusters | 86 |
| 4.6 | Clustering accuracy VS. number of training samples | 88 |
| 4.7 | An example of probability distribution of 10 RSS training samples after discretization with different bin width | 90 |
| 4.8 | Analysis of effect of varying bin width | 91 |
| 4.9 | RMSE VS. bin width, when the number of training samples/RP is set at 40. | 92 |
| 5.1 | An example of walls representation | 96 |
| 5.2 | An example of arranging cell boundaries for a RP. | 98 |
| 5.3 | The flow chart of recognition boundaries of cells of all distributed RPs. | 99 |
| 5.4 | Arranged cells for the RPs distributed in the floorplan. | 100 |
| 5.5 | A collision is detected in the second scenario. | 105 |
| 5.6 | An example of a multi-clustered phenomenon | 105 |
| 5.7 | Four paths are evaluated specifically in the tests | 109 |
| 5.8 | The failure rate of tracking the user on the entire path vs. number of employed particles | 110 |
| 5.9 | An example of particles being incorrectly clustered for the starting position of the first path | 113 |
| 5.10 | Comparison of positioning error of two scenarios: particles being correctly/incorrectly clustered. | 114 |
| 5.11 | An example of particles being correctly clustered for the starting position of the first path | 115 |

| | | |
|------|---|-----|
| 5.12 | Performance of the localisation system with different number of RSS training samples being employed | 116 |
| 5.13 | An example of biased distributed particles being filtered out after first turning point for the fourth path. | 118 |
| 5.14 | Evaluation results of on-line importance weights adjustment for the first path. | 120 |
| 5.15 | Evaluation results of on-line importance weights adjustment for the second path. | 122 |
| 5.16 | Distribution of the particles propagated along the second path. | 122 |
| 5.17 | Evaluation results of on-line importance weights adjustment for the third path. | 123 |
| 5.18 | Evaluation results of on-line importance weights adjustment for the fourth path. | 124 |
| 6.1 | Flow chart of RM calibration with particle filter implementation | 133 |
| 6.2 | An example of particle indexing within the reflection table | 134 |
| 6.3 | An example of the sequential reflection table | 136 |
| 6.4 | The diagram of semi-automatic site survey | 139 |
| 6.5 | Showcase of the three paths utilised in the experiments | 141 |
| 6.6 | Three paths are evaluated specifically in the experiments for RM calibration (one direction for each path). | 143 |
| 6.7 | Comparison of the CDF of localisation errors when probabilistic NB approach, particle filter and back-tracking finally surviving particles are applied, respectively. | 143 |
| 6.8 | RMSE with respect to the bin width | 145 |
| 6.9 | RMSE with respect to the proportion of historical on-line RSS measurements being applied. | 148 |

List of Tables

| | | |
|-----|---|-----|
| 2.1 | Performance vs. Price of gyroscope | 37 |
| 3.1 | The accuracy of the estimation of the polarity of Υ , when the smart- phone is held with an angle bias to the the tester's orientation. During the first five pieces of walks, $ \Upsilon $ is approximately 30° ; $ \Upsilon $ is about 15° for the last five pieces of walks. Numerators count the number of one-second periods during which the polarity has been estimated correctly in each walk; denominators count the to- tal number of one-second periods in each walk. | 70 |
| 4.1 | Experimental results of the best positing accuracy and optimal bin width VS. number of training samples/RP. | 92 |
| 5.1 | The parameter values for ϖ_1 | 101 |
| 5.2 | The parameter values for ϖ_2 | 101 |
| 5.3 | Path summary | 108 |
| 5.4 | The experiment results related to correctly clustered RSS samples, for the first path. | 111 |
| 5.5 | The experiment results related to correctly clustered RSS samples, for the second path. | 111 |
| 5.6 | The experiment results related to correctly clustered RSS samples, for the third path. | 111 |
| 5.7 | The experiment results related to correctly clustered RSS samples, for the fourth path. | 111 |
| 6.1 | The optimal bin index VS. Proportion of all historically-collected RSS measurements | 147 |
| 6.2 | The ratio of overhead time to the total consumed time in site survey VS. the number of RSS collections on each RP | 148 |

Abbreviations

| | |
|-------|---|
| AIR | Active Infra-red |
| AOA | Angle-of-arrival |
| AP | Access Point |
| ARW | Angle Random Walk |
| BNS | BeiDou Navigation Satellite System |
| BS | Base Station |
| CDF | Cumulative Distribution Function |
| CIR | Channel Impulse Response |
| DGNSS | Differential Global Navigation Satellite System |
| DR | Dead-reckoning |
| EM | Expectation Maximisation |
| 4G | 4-th generation |
| GNSS | Global Navigation Satellite Systems |
| GP | Gaussian Process |
| GPS | Global Positioning System |
| IMU | Inertial Measurement Unit |
| IR | Infra-red |
| KNN | K-nearest Neighbour |
| LBS | Location-based Service |
| LED | Light Emitting Diode |
| LF | Location Fingerprinting |
| LOS | Line-of-sight |
| LTE | Long Term Evolution |
| MEMS | Micro-electromechanical System |
| ML | Maximum Likelihood |

| | |
|------|--|
| MMSE | Minimising the Mean of the Squared Error |
| NB | Naive Bayesian |
| 1-D | One Dimensional |
| OVA | One-vs-all |
| OVO | One-vs-one |
| PDF | Probability Distribution Function |
| PDR | Pedestrian Dead-reckoning |
| PIR | Passive infra-red |
| RF | Radio Frequency |
| RFID | Radio Frequency Identification |
| RM | Radio Map |
| RP | Reference Position |
| RSS | Received Signal Strength |
| STFT | Short-term Fourier Transform |
| SVM | Support Vector Machine |
| 3-D | Three-dimensional |
| 2-D | Two-dimensional |
| TDOA | Time-difference-of-arrival |
| TOA | Time-of-arrival |
| UT | User Terminal |
| UWB | Ultra-wideband |
| VLC | Visible Light Communication |
| WKNN | weighted K-nearest Neighbour |
| ZUPT | Zero-velocity Update |

Chapter 1

Introduction

1.1 Motivation

This thesis is concerned with the problem of estimating and tracking the locations of normal users with mobile smartphones on hand in indoor scenarios.

Nowadays, with the widespread adoption of smartphones, mobile applications developed dedicatedly on such platforms have brought tremendous conveniences to humans' daily life. Specifically, location information of the users has become an extremely important resource for the location-based mobile services (LBS). In particular, LBS is defined by [6] as the services which fuse the position information with the information obtained from other sources for the purpose of providing added values to the subscribers. More accurate location information could be leveraged for smarter commercial advertising, attaching more specific location to the users' message published on social networking applications, more reliable pedestrian navigation, etc.

Also, obtaining location information of users is helpful to improve the performance of the mobile networks, such as improving the data capacity, better radio resource management, etc. Two such examples are given as follows.

Firstly, according to Qualcomm Inc, as the need, mobile capacity may increase up by 1000 times someday in a near future (also known as 'the 1000x data challenge'). One of the ways to meet the demand of such tremendous data exploding could be

deploying small cells by allowing the frequency spectrum to be reused many times over in an area of interests. However, due to the limited coverage of small cells (50-100 m), of which partial purpose is to reduce the interference to the macro-cell occupying the same or neighbouring frequency band, locations for deploying small cells should be selected carefully in order to accommodate more subscribers and maximising the return on the investment also. Imaging that moving trajectories of all subscribers in an area of interests could be estimated accurately, ‘hotspots’, the locations where subscribers gather around or pass through often, could be thus identified, and then small cells could be deployed by the mobile service provider on them.

Secondly, handover is usually categorised into two classes: soft handover and hard handover. With the soft handover deployed mainly by the code-division multiple access (CDMA) and wideband CDMA (W-CDMA) mobile communication system [7], which is also known as connect-before-break, the communication channel between the user terminal (UT) and the base station (BS) of the source cell is still maintained for a duration while establishing the channel between the UT and the destination cell. In this way, the disruption to the on-going speech communications could be eliminated. On the other hand, when the hard handover is implemented, the radio-link connection between the UT and the BS of the destination cell is suspended to be established before the link between the UT and the BS of the source cell is terminated. As a result, the mobile services are expected to be disrupted for a while and the failure rate of the services are also higher, compared with the cases in which the soft handover is applied. Suppose the UT is at the boundary of two cells, the hard handover could be implemented for several times in case that the RSS of the links between the UT and two BSs beats each other alternatively, which, of course, leads to a situation that the service is disrupted for several times. However, the soft handover is eliminated in the 4-th generation (4G) mobile communication standard: long-term evolution (LTE), since the UT is required to resynchronise to a different set of sub-carriers occupying different frequency band when the handover is implemented. As a result, with more accurate geometric information of the UTs, the hard handover could be implemented at proper time moments when the distance between the UT and the BS of the

destination cell is clearly found to be shorter than the distance between the UT and the BS of the source cell.

Generally, global positioning technology (GPS) could provide accurate and reliable localisation service in outdoor environments. For example, Easytrails GPS [8] collects and records the raw location data from users' smartphones in sequence over time, so that users could track their activities in real-time, which is useful to know the amount of their excises. Another example is iCartel [9] which analyses the crowd-sourced data from the users' phones to predict traffic delays expected to be experienced by the user on each road segment. Then, 'traffic-aware' routes expected to afford users the best on-road experiences are proposed according to the predications. The screen-shots of the two examples are indicated in Figure 1.1.

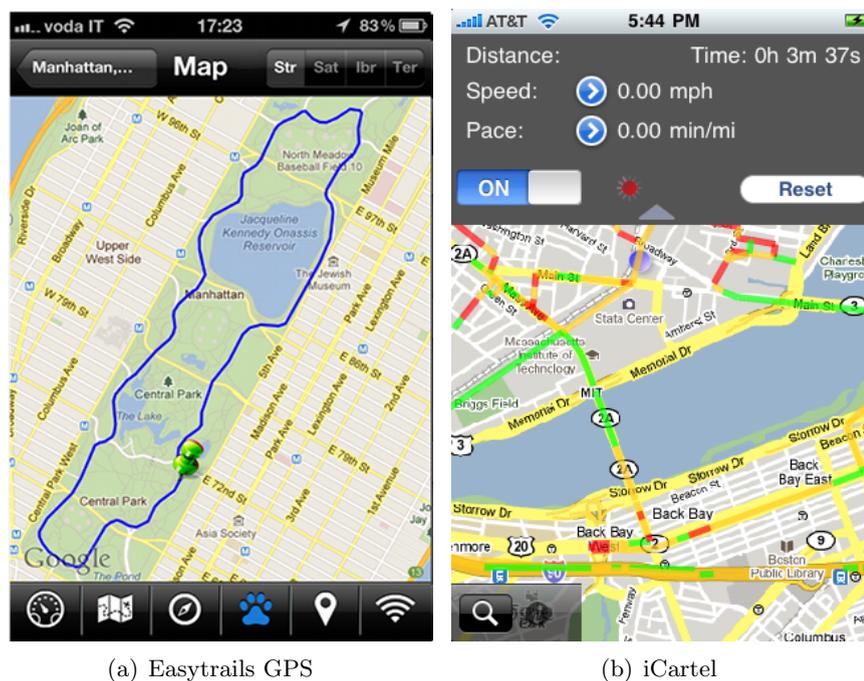


FIGURE 1.1: screen-shots of two GPS mobile applications: Easytrails GPS and iCartel.

However, although GPS could provide accurate and reliable localisation service in outdoor environments, adoptions of the technology in indoor scenarios are still infeasible, due to the poor penetration profile of GPS wireless signals, i.e., the signal is commonly too weak to be used for localisation purpose. Instead, assisted-GPS (A-GPS) was proposed to afford indoor localisation services to GPS-capable mobile phones [2], wherein location servers with another reference GPS receiver

are applied to help the partial GPS receiver on the mobile phones look for the weak GPS signals. The measurements from both the GPS constellation and the wireless mobile network are collected and combined by the location server for localisation [2]. However, only an average of 5-50 m accuracy could be provided by the approach in most cases, which may not satisfy the requirements of the LBS mobile applications.

Other wireless communication technologies and sensors such as cellular networks[10], infra-red (IR) sensing [11], ultrasonic signal [12], radio frequency identification (RFID) [13], radio frequency (RF) signal [14–16], light emitting diode (LED) devices [17–19] are also proposed for more accurate indoor localisation services. However, additional dedicated infrastructures or sensors are required to be installed for the systems, which prevents them from being widely utilised.

In recent years, the popular trend of Wi-Fi(IEEE 802.11 b/n/g) network infrastructures being widely distributed in buildings stimulates the interest in developing effective Wi-Fi positioning and tracking systems. Generally, the systems could be categorised based on the employed techniques as: time-of-arrival (TOA) [20], time-difference-of-arrival (TDOA) [21], angle-of-arrival(AOA) [22] and the RSS [23–26]. However, for applying the TOA and TDOA techniques, instalment positions of the APs should be carefully chosen for higher probability of building LOS wireless transmission between mobile smartphones and APs. Also, dedicated hardware such as antenna array is required upon mobile phones for implementing AOA localisation system. Therefore, RSS-based localisation method becomes the only one convenient choice for which neither the knowledge of instalment positions of APs nor dedicated hardware is required.

In this thesis, an accurate indoor localisation system based on Wi-Fi and inertial measurement units (IMUs) is proposed. Shorter time of the site-survey period and more robust self-calibration capability than the traditional wireless localisation systems are promised.

1.2 Overview of the RSS-based indoor localisation methods

Typically, current mobile smartphones on the market are armed with Wi-Fi network adapters by which both wireless local network access and measurement of the RSS could be achieved. Also, the RSS is known as the level of the power of the received wireless signal and the measurement results are recorded in units of dBm. Generally, the values of the RSS of the signal received from a particular AP differ in different locations. Theoretically, the RSS differences come from many factors such as the different distances between the AP and the locations, different numbers of obstacles existing in the direct paths, different materials of the obstacles, etc. In the case that the RSS values of a AP on two positions are comparable, the positions could be distinguished between each other by taking advantage of other APs from which the RSS values of the signal received are different. Therefore, the vector consisting of RSS measurements of multiple APs made on a individual location is regarded as the unique fingerprint identification of the location.

There exist two methods to derive the location of the collected on-line RSS measurements vectors: RSS-TOA interpretation and location fingerprinting (LF). In the first method, the distances between the mobile smartphone and APs are estimated according to various empirical propagation models. With distances between at least three APs and the targeted smartphone estimated, the approach as indicated in Section 2.2.1.1 could be applied for localisation. In particular, empirical propagation prediction models such as ECC-33 model [27], COST-231 Hata model [28], Motley-Keenan model [29], etc, could be applied for the purpose. However, due to the complicated indoor environments, most of the prediction results are not accurate. Therefore, inaccurate indoor localisation results are expected.

Another method traditionally consists of two phases. In the first phase, RSS measurements from detected APs are collected consecutively on all RPs, and then a database called ‘RM’ is built according to them. Also, the RSS measurements obtained in this phase are recognised as training data. Then, the location fingerprint of each RP is recognised by exploring the set of RSS training data collected on the RP. Then, in the second phase, user’s on-line RSS measurements are applied by

pattern recognition algorithms for similarity comparison between location fingerprints of all RPs in the RM and themselves or derivatives of themselves. If higher similarity is found between a user's on-line RSS measurement and the recorded RSS fingerprint of one particular RP, it is reasonable to estimate that the on-line location of the user is in the close proximity to the RP. However, the accuracy of this approach is distorted by the time-varying nature of the RSS of the wireless signal. The reasons for the time-varying nature of the RSS are found to be complicated. The reasons are explained as follows.

Firstly, because of the existences of obstacles, wireless signal experiences severe multi-path propagation in the indoor environment. Thus, even minor changes of the locations of the users' smartphones or obstacles in the propagation environments may lead to variations of the RSS.

Secondly, Doppler effect always occurs when the users' smartphones are being moved, which leads to the change of the frequency of the received wireless signal. Mathematically, this is equivalent to the temporal variation of the envelope of the electromagnetic wave occupying the frequency of the transmitted signal, and also the variation of the RSS measurement results in time domain.

Finally, it is common for smartphones to experience interference in Wi-Fi networks. Specifically, the occupied frequency ranges of different channels are partially overlapped with each other in Wi-Fi networks operated at 2.4 GHz, as indicated in Figure 1.2. An extreme example could be an AP occupying the neighbouring channel presents in proximity to a user's smartphone which is, however, in communication with another AP at a longer distance. Under this situation, interference is significant. If the interference sourcing AP switches often in two states (silent and busy), the RSS measurement results could be varied over time.

1.3 Overview of the IMUs

IMUs were initially introduced in areas of aircraft navigation in 1930s [30]. The usage of IMUs at that time was limited, due to its constraints in physical dimension, cost and power consumption. Recently, the introduction of micro-electromechanical

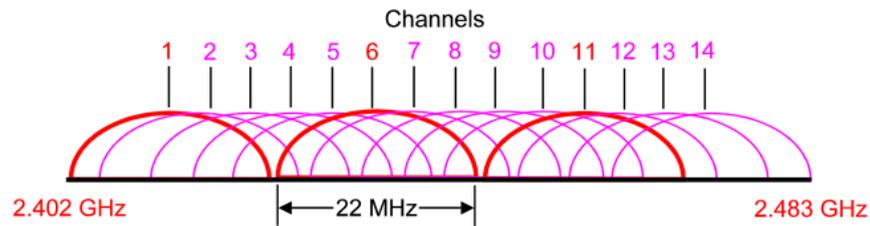


FIGURE 1.2: The frequency ranges of all channels of Wi-Fi at 2.4 GHz.

system (MEMS), with features such as small size, economy in use and power saving, has stimulated the wide spread of IMUs in consumer applications. Nowadays, almost all smartphones on market are equipped with MEMS IMUs.

Typically, each IMU consists of three components: accelerometer, gyroscope and magnetometer. They are used to detect the rates of accelerations, changes in rotation angles and magnetic field of the earth, respectively. All of the sensors have three degrees of freedom to take measurements from three orthogonal axes. The traditional dead-reckoning (DR) method [31] takes advantage of the measurements of them directly to derive the position and orientation of the objects. However, without periodic calibration given by GPS signal as what is implemented for outdoor applications, the derivation results would be biased at an increasing level due to the accumulation of errors in the measurements made by the IMUs.

In practice, pedestrian dead-reckoning (PDR) approaches are commonly applied, which has been proved to be a good candidate for the task. Especially, in the step-based PDR approaches, the unique pattern of the variation of accelerometer readings is recognised. Then, detection of the repetition of the pattern emerging during walk is utilised for displacement update. Note that this method is applied in this thesis. The task of indoor pedestrian orientation estimation is trickier. While measurements from a magnetometer are prone to be erroneous when it is surrounded by ferromagnetic objectives, estimation of the change of the angle obtained by taking advantage of measurements from gyroscopes are biased at an increasing level over time. Therefore, data fusion algorithm, such as a complementary filter, should be applied to provide the orientation estimation results. In this way, the respective disadvantages of them could be overcome.

1.4 Scope and contribution

In this thesis, an indoor localisation system based on a Wi-Fi location fingerprinting (LF) method and trajectory tracking with IMUs is proposed and evaluated. Such system will be shown to be capable of tackling the technical challenges mentioned in previous sections.

In Chapter 2, firstly, the existing wireless localisation techniques and approaches are reviewed. Especially, the LF methods are presented. Following that, the machine learning algorithms applied in the implementation or test of the thesis, such as k -means, k -means++, affinity propagation algorithm, support vector machine (SVM) classifiers, Fisher kernels, etc., are presented. Thirdly, pedestrian DR methods such as frequently calibrated pedestrian dead-reckoning (PDR) and step-based PDR are reviewed. Fourthly, the route planning algorithm used in this thesis is reviewed. Finally, the recursive Bayesian estimation method and the detailed implementation of it, such as Kalman filter and particle filter are reviewed.

In Chapter 3, the algorithms processing the data of the IMU modules to recognise the moving trajectory of pedestrians are presented. The modules such as motion status detection, orientation detection, step and stepping moment detection, step length detection, etc., are included. The advantage of applying complementary filter to fuse the information from a gyroscope and the information from the magnetometer to estimate the orientation of a smartphone is also presented. Several **contributions** are made in the chapter as shown below.

- The estimation algorithm of the orientation differences between users and their phones, including the absolute values and the polarities of the angle differences, is proposed.
- An accurate easy-to-implement threshold-based step detection algorithm is proposed. The employed thresholds are learnt for each individual user from a piece of trial walk of the user. Also, the fact that the consecutive steps of normal human beings are separated by at least a certain amount of time

is also leveraged to prevent noisy spikes from being detected as steps. Note that the algorithm has been published in [32] as the author's own work.

- A novel step length estimation algorithm is proposed, for which neither a general model built from the crowd-sourcing data nor a convergence process for personalisation of the parameters, as introduced in [33], are required.

Chapter 4 presents the indoor wireless localisation algorithm applied in the thesis. Overall, there are two phases involved in the algorithm: the rough localisation and the fine localisation. In particular, for the Wi-Fi RSS probabilistic naive Bayesian (NB) localisation algorithm used in the fine localisation phase, the importance of choosing the proper bin width to discretise the continuous RSS values is discussed. Also, the way of choosing the most appropriate bin width according to the scale of the RSS training data is also presented. Finally, the related experimental results are presented.

The **contribution** of Chapter 4 is shown as follow. The disadvantages of unsupervised clustering algorithms such as k -means and affinity propagation used in the rough localisation phase by [1] and [34], worse clustering accuracy than the supervised clustering algorithm and sparsely distributed cluster patterns, are discussed. On the basis of that, applying a supervised clustering algorithm with manually clustered reference locations for implementation of rough localisation is proposed in the chapter.

In Chapter 5, an indoor particle-filter-based localisation system fusing the information from the PDR tracking and the wireless localisation is presented. Firstly, it describes the method to determine automatically the boundaries of the cells of all distributed RPs. Secondly, detailed implementation of the particle filter, including particle propagation, particle correction and particle re-sampling, is presented. Note that similar implementation of particle filter applied in this thesis for localisation aim has been published in [35, 36] as the author's own work. Then main **contributions** of the chapter are made, as shown below.

- On-line adjustment of the importance weights of the employed particles opportunistically by taking advantage of Wi-Fi RSS fingerprints when a significant body turning of a user along a path is detected.
- Increasing the accuracy of the results of both the rough and fine localisation algorithms by applying at least two RSS fingerprints collected over time continuously to initialise the particles.

In Chapter 6, firstly, the necessity of RM calibration for indoor localisation systems is presented. Following that, main **contributions** of this chapter are made, as presented below.

- A novel RM calibration method based on back-tracking the positions of the final surviving particles along users' tracked trajectories is proposed.
- A semi-automatic site survey approach is proposed, with the background of the reduced requirement of the scale of the RSS training data for activating the indoor localisation system. Note that this approach has been published in [32] as the author's own work.

The RM calibration method based on the expectation maximisation (EM) algorithm introduced in [1] requires a complicated convergence progress for optimisation of the related parameters. In the RM calibration approach proposed in Chapter 6, such process is avoided, and also the robustness of the calibrated RM is expected to be improved further, as indicated in the evaluation results. Also, the experimental results show that it is important to re-adjust the bin width to discretise the continuous RSS values for providing more accurate indoor localisation service after bundles of on-line Wi-Fi RSS measurements have been accumulated. Furthermore, in the semi-automatic site-survey approach proposed in Chapter 6, the IMU reports of the technician and a path routing algorithm are applied for automatic update of the index of the RP where the current site-survey work is being implemented. In this way, the goal of improving working efficiency and guaranteeing the reliability of the built RM could be achieved..

In Chapter 7, the concluding remarks and directions for the future work are presented.

Chapter 2

Literature Review

Various types of technologies have been utilized to develop indoor wireless localisation systems, including IR sensing [11], ultrasonic [12], RFID [13], RF signal [14–16], LED [17–19], etc., which are reviewed in detail in Section 2.1.

In these wireless localisation systems, various mathematical methods are employed for location estimation with the information learned from these technologies. Specifically, they are usually categorised into two groups: triangulation methods and RSS-based methods. There are two derivations of the triangulation methods: multi-lateration and multi-angulation. In multi-lateration methods, distances from/to pre-installed APs are inferred and employed to estimate the user’s position. On the other hand, directions from/to the APs are used for localisation purposes. In RSS-based methods, vectors consisting of RSS measured from multiple APs of all positions are employed for the task of location estimation. Generally, RSS could be obtained in two different ways: an empirical propagation model and a site survey. Details are shown in Section 2.2. Following that, two different ways to process the location RSS fingerprints in the LF-based algorithms, discretisation and kernel method, are introduced. Also, the reasons of high discretisation bias and high discretisation variance are explained.

Recently, indoor localisation systems based on fusing information from different types of sensors have become popular, due to their immunity to a single point of failure. In other words, a serious error of instantaneous measurement made by

one particular sensor is unlikely to deviate the positioning result severely, given these localisation systems. For example, a hybrid positioning system based on ultra-wideband (UWB) transmission and inertial sensor is proposed by [37]. The recursive Bayesian estimation method, also known as Bayesian filter, is used as a sensor fusion algorithm for indoor localisation, where measurements from multiple types of sensors are utilised. In Sections 2.5 and 2.7, reviews of inertial navigation methods and sensor fusion algorithms are given.

2.1 Localisation technologies

2.1.1 Infra-red

The IR localisation technologies could be categorised into two kinds: active infra-red (AIR) [11] and passive infra-red (PIR) [38].

With AIR, the user is equipped with a dedicated device capable of transmitting the IR signals. If the signal is received by a particular IR receiver, the user's position could be estimated to be in close proximity to this receiver, due to the poor penetration nature of IR-frequency signals.

The detectors used in the PIR localisation method consist of low-resolution thermopile arrays. These thermal IR sensors measure thermal radiation emitted by any object, relative to ambient temperature, in its field of view. Since the temperature of human's skin is different from ambient temperature in an indoor environment, this method could be used to detect the presence of human beings. The obvious advantage of the PIR localisation method is that the user does not need to carry any transmitter or tag for the localisation service. However, human beings are not the only source of thermal radiation in indoor environments. Thus, the measurements of these detectors are also influenced by additional heat sources, which leads to biased localisation results.

2.1.2 Ultra-wideband

UWB refers to the radio technology pioneered by Robert A. Scholtz as in [39], with the following characteristics: short-duration signal pulse communication over short range with very low energy level. Benefiting from employed high bandwidth, UWB signals enable high temporal resolution and thus potentially high localisation accuracy in a LOS transmission situation. Localisation systems applying UWB could be found in [14, 40].

Multi-lateration and multi-angulation, as shown in Sections 2.2.1.1 and 2.2.1.2, are used for UWB localisation technologies. Thus, the accuracies of them may be degraded when LOS transmission condition requirement cannot be met. In addition, dedicated communication devices are required by these systems.

2.1.3 Ultrasound

Oscillating sound pressure waves, with a carrier frequency being higher than the upper bound of the human hearing frequency range, is termed ultrasound. For the ultrasonic localisation system such as the one proposed in [41], grid distributed receivers are mounted on the ceiling over the areas of interest. A small narrow-band ultrasound transmitter, known as Bat, is held by every user. Ultrasound pulses are emitted from the bat and received by more than three receivers. TOA information to the receivers is measured, and the multi-lateration method as shown in Section 2.2.1.1 is employed to localise the user carrying the bat. The system [41] is reported to achieve a accuracy within 3 cm during 95% of the time.

However, in the proposed system, receivers are requested to be densely distributed over the ceilings in the indoor environments, in order to let the user, at any time, be seen in LOS condition from at least three receivers, which is considered to be not economic in practice. Meanwhile, a dedicated device, such as bat, needs to be carried by user all the time.

2.1.4 Wi-Fi

Wi-Fi refers to the wireless local area network based on IEEE 802.11 standards. Due to the massive deployment of Wi-Fi APs in indoor environments, localisation systems applying Wi-Fi technologies have attracted dramatic attentions. Most of them could be categorised into two groups with two different localisation mathematical methods: empirical propagation model-based, and location-fingerprinting (LF) method-based, as presented in detail in Section 2.2.2.

2.1.5 LED light

In recent years, applying LEDs for landscape architecture or illumination has been highlighted due to their inherent advantage of high power efficiency. Moreover, as semiconductor light emitting devices, LEDs are capable of modulating informative signals onto a light-wave signal for applications such as visible light communication (VLC), as proposed in [42–44]. Another application of such modulation could be found in indoor localisation, as presented in [17–19]. The basic idea is applying the triangulation method, as shown in Section 2.2.1, using LEDs as anchors. All LED bulbs broadcast, via light-wave signal, their anchoring information, ie., geo-location co-ordinates of themselves and duty cycles. The light sensor on the user’s smartphone is employed for receiving the anchor’s information, measuring RSS, and then obtaining channel impulse response (CIR) for inferring distance, etc. The system proposed in [19] is reported to achieve up to 0.4m accuracy for 90% of all time.

However, LOS signal transmission is strictly requested by this method, which is often not achievable in reality. Imaging a user who holds a smartphone near his/her chest, LOS signal transmission to LEDs behind the user is highly likely blocked by the user’s body. Also, technicians deploying the localisation systems need to configure anchoring information and the arrangement of the communication channel, such as applying dedicated multiplexing protocols, to avoid collision of broadcasts, for every employed LED, which is time-costly.

2.2 Mathematical methods for localisation

2.2.1 Triangulation methods

Triangulation uses the geometric property of triangles to estimate users' location. There are two derivations: multi-lateration and multi-angulation. They are presented below, respectively.

2.2.1.1 Multi-lateration

In multi-lateration methods, measurements of distances between user and multiple APs are used to infer the user's position. The distance is usually calculated as the product of one-way wireless signal propagation time and velocity of signal (light). The diagram is shown in Figure 2.1. One of the most famous application of the TOA localisation methods is global navigation satellite systems (GNSSs) including the United States' GPS and the China's BeiDou Navigation Satellite System (BDS).

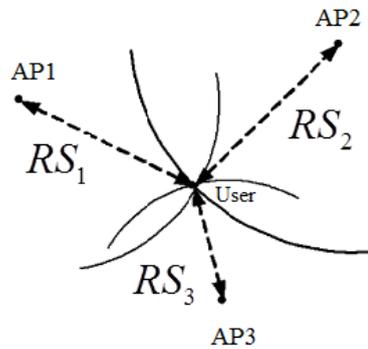


FIGURE 2.1: Positioning with TOA method, reproduced from [2]

Regarding the figure, a set of linear equations is built as:

$$\begin{cases} RS_1^2 = (P_{AP1_{x'}} - P_{user_{x'}})^2 + (P_{AP1_{y'}} - P_{user_{y'}})^2 + (P_{AP1_{z'}} - P_{user_{z'}})^2 \\ RS_2^2 = (P_{AP2_{x'}} - P_{user_{x'}})^2 + (P_{AP2_{y'}} - P_{user_{y'}})^2 + (P_{AP2_{z'}} - P_{user_{z'}})^2 \\ RS_3^2 = (P_{AP3_{x'}} - P_{user_{x'}})^2 + (P_{AP3_{y'}} - P_{user_{y'}})^2 + (P_{AP3_{z'}} - P_{user_{z'}})^2 \end{cases} \quad (2.1)$$

Here, $P_{\bullet_{x'}}$, $P_{\bullet_{y'}}$, and $P_{\bullet_{z'}}$ denotes the geometric co-ordinate of $\{\bullet\}$ along the x' , y' and z' axis in the global reference frame, as illustrated in Figure 3.2. Noting that Figure 2.1 and equation (2.1) are only a compact demonstration. In practice, TOA localisation system such as GPS requires TOA measurements from at least four satellites as APs to enable a three-dimensional (3-D) localisation service [45]. Also, the equation assumes the perfect timing synchronisation between the user's device and APs, which in most cases is hardly achieved. Given the knowledge of TOA measurements and coordinates of APs, the set of equations is applied to infer the user's position. There are a number of ways to solve the set of equations, including the Newton-Raphson method, the Taylor series method, etc [46].

Another variant of the multi-lateration method is the TDOA[21], as demonstrated compactly in Figure 2.2. Typical applications for outdoor scenarios could be found as differential global navigation satellite systems (DGNSS). The user lies on one of the intersection areas of two hyperboloids where locations exhibit the same difference of distances between APs and the user as measured. Finally, to choose one of two intersection areas as the user's position, TOA information of one of APs to the user is employed, illustrated as the TOA measurement of RS_1 in the figure.

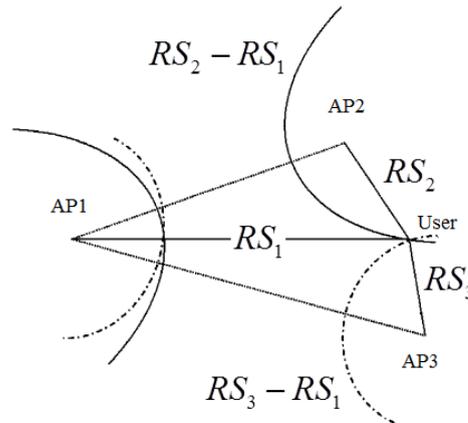


FIGURE 2.2: Positioning with the TDOA method, reproduced from [2]

The accuracies of the localisation systems applying multi-lateration methods are highly dependent on the LOS propagation condition. For example, in indoor scenarios where LOS propagation is hardly achieved, accuracies of such systems are

severely degraded. Also, in different outdoor scenarios, such as rural areas and urban areas, the performance of GNSS positioning system differs. For example, in urban areas, the LOS propagations are much likely to be blocked by surrounding high-rise buildings, which means that the number of ‘visible’ satellites is not sufficient. Meanwhile, additional errors are introduced by severe multi-path effects, brought from nearby obstructions.

2.2.1.2 Multi-angulation

AOA measurements are done by receivers equipped with antenna arrays where a number of individual antennae are grouped together systematically to estimate the directions of radio waves incident on them. In practice, the direction is inferred by measuring the TDOA at individual elements of the array. One of the typical applications of the multi-angulation localisation method is found as geo-location of cell phones by nearby BSs equipped with antenna arrays, complying with the regulations that require cell systems to report the locations of cell phones dialling emergency calls or to provide special localisation services to users [22]. Although the localisation methods only require at least two antenna arrays at two different positions to provide the services, inherent high structural complexities and costs of antenna arrays prevent such localisation systems from widely-spread utilisation.

2.2.2 RSS-based methods

The key to the RSS-based localisation system is to investigate the relationship between the RSS and the distance between the AP and the user device. As presented in the beginning of this chapter, RSS collections could be achieved in two different ways: the empirical propagation model [23, 24] and a site survey [25, 26]. They are presented in Section 2.2.2.1 and Section 2.2.2.2, respectively.

2.2.2.1 Empirical propagation model

The core of this method is solving equation (2.1) with the distances between the user and at least three APs being inferred from various of applied empirical propagation models.

Naturally, the accuracies of localisation systems applying this method is highly dependent on the accuracies of the estimated distances. In practice, reflections or fadings due to small surrounding obstacles may deviate the RSS severely, which is hardly taken into account by the propagation models.

Parameters such as path loss exponent are defined differently for different empirical propagation models. Also, every proposed propagation model may only fit a small proportion of all indoor environments. As a result, it is necessary to implement a comprehensive survey for every targeted indoor environment to choose the propagation model, as well as the values of parameters used in the propagation model, which is regarded as time-inefficient. Furthermore, as all triangulation localisation methods, the geometric distribution of all APs are required to be known in advance by the operators of the localisation systems, which, in reality, is hardly achieved.

2.2.2.2 Location fingerprinting method

Conventionally, location fingerprinting methods involve two phase: site survey training (off-line) and on-line phase. Site survey refers to the procedure of collecting RSS samples by RF-enabled devices such as smartphones and laptops at known RPs in the area of interest. In this way, RSS samples could be considered as the location fingerprints. The collection of RSS samples are then recorded in database called as radio map (RM). In online phase, user's mobile device monitors the real-time RSS vectors and compares them with the entries in the RM to estimate user's location. There are different comparison methods that will be shown in Section 2.3.

Ease of deployment is the most significant advantage of the location fingerprinting method. Firstly, dedicated communication devices are not required to be applied

in these location systems. APs could be common Wi-Fi routers that are deployed massively in indoor scenarios such as office buildings, apartments, etc. Secondly, unlike triangulation localisation methods, location knowledge of APs is not required to be known.

However, there are two shortcomings of LF methods. Firstly, accuracies of such localisation systems are highly dependent on the density of RPs and the number of training RSS samples at every RP. Therefore, in order to provide a localisation service with the required accuracy, time-consuming and labour-intensive site-survey effort is expected to be made by operators. Secondly, thanks to the time-varying nature of wireless signals due to changing indoor environments, the reliability of the RM will decrease over time.

Extensive efforts are made to overcome the first problem. Some papers such as [47, 48] use interpolation methods taking advantage of geometric correlation of neighbouring RPs in order to build a RM with less collection efforts. In [49], a machine learning algorithm such as Gaussian Process (GP) is applied to learn the RSS distribution over the area of interest with a few fixed pre-installed monitoring points. Others such as [50, 51] depend solely on crowd-sourcing methods to build a RM.

However, accuracies of RSS interpolation and estimation results of [47–49] are questionable when the scenarios considered are complicated indoor environments with many obstacles and walls. In [49], the authors propose to utilise human intervention to calibrate the estimation results to solve the problem, which consumes time and labour costs. For the works proposed by [50, 51], convergence time of RM building phase is highly dependent on the flow rate and distribution of visitors. Given a low flow rate and biased distribution of visitors, some areas may not be well surveyed in a long time, which may degrade the accuracies of these localisation systems. In the thesis, efforts are made to reduce the time used for off-line phase while building a robust RM.

The reliability of the RM, in the thesis, is maintained by geo-tagging and feeding user's on-line RSS measurements back to calibrate the RM database. Specifically,

expectation maximization (EM) algorithm, an iterative method for finding maximum *a posteriori* (MAP) estimates of parameters in a probabilistic model is found suitable for calibrating a RM with crowd-sourced data.

2.3 Comparison methods for LF technique

Generally, there are two kinds of approaches to estimate the user's location based on real-time RSS measurements and RM.

In deterministic approaches, the collected RSS samples from a site survey, on each location, are averaged, and then the results are stored in the RM as the signature of the position. Suppose a series of training RSS samples indexed from 1 to n_p are received from the i -th AP at the j -th location. They are denoted as $[\psi(o_{i,j}^1), \psi(o_{i,j}^2), \dots, \psi(o_{i,j}^{n_p})]^T$, the recorded averaged result is $\bar{\psi}_{i,j} = \frac{1}{n_p} \sum_{n'=1}^{n_p} \psi(o_{i,j}^{n'})$, where ψ denotes the RSS value. Therefore, the RM covering I APs and J locations is built as:

$$\begin{bmatrix} \bar{\psi}_1 & \bar{\psi}_2 & \cdots & \bar{\psi}_J \end{bmatrix} = \begin{bmatrix} \bar{\psi}_{1,1} & \bar{\psi}_{1,2} & \cdots & \bar{\psi}_{1,J} \\ \bar{\psi}_{2,1} & \bar{\psi}_{2,2} & \cdots & \bar{\psi}_{2,J} \\ \vdots & \vdots & \ddots & \vdots \\ \bar{\psi}_{I,1} & \bar{\psi}_{I,2} & \cdots & \bar{\psi}_{I,J} \end{bmatrix} \quad (2.2)$$

Suppose a particular RSS observation vector, $\boldsymbol{\psi}(\mathbf{o})$, is received in on-line phase at an unknown position, K-nearest neighbour (KNN) [25, 26], the most commonly applied deterministic method, calculates the Euclidean distance between $\boldsymbol{\psi}(\mathbf{o})$ and $\boldsymbol{\psi}_j, \forall j \in \{1, 2, \dots, J\}$ as:

$$ED_j = \|\bar{\boldsymbol{\psi}}_j - \boldsymbol{\psi}(\mathbf{o})\| \quad (2.3)$$

and then KNN chooses K ($K < J$) locations of which corresponding Euclidean distances are the minimum ones. Note that in practical implementation, the correspondences between the geometric coordinates and indices of the RPs are known to the system for the localisation purposes.

Finally, these K locations are averaged, and the result is regarded as the collection position of $\boldsymbol{\psi}(\mathbf{o})$. Besides KNN, distance-weighted KNN (WKNN) is also applied in other works [52] for the location determination task, where weights are inversely proportional to their Euclidean distances to the on-line RSS vector.

Besides deterministic approaches, localisation problems could be also solved with probabilistic approaches, as applied in [53–55]. In probabilistic approaches, for every RSS observation vector $\boldsymbol{\psi}(\mathbf{o})$, a *posteriori* distribution of locations over it will be derived provided that it is received in on-line phase. Supposing \mathbf{p}_j denotes the 2-D position vector of the j -th RP, then the *a posteriori* probability of \mathbf{p}_j over $\boldsymbol{\psi}(\mathbf{o})$ is obtained as:

$$Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o})) = \frac{Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j)Pr(\mathbf{p}_j)}{Pr(\boldsymbol{\psi}(\mathbf{o}))} = \frac{Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j)Pr(\mathbf{p}_j)}{\sum_{j'=1}^J Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_{j'})Pr(\mathbf{p}_{j'})} \quad (2.4)$$

Here, $Pr(\mathbf{p}_j)$ denotes the *a priori* probability of j -th location among all. When NB algorithm is implemented with the RM built by the training data only, the *a priori* probabilities of all locations are assumed the same. $Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j)$ denotes the conditional distribution obtained by analysing the RM database. Also, NB estimation algorithm holds strong independence assumptions between features. Therefore,

$$Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j) = \prod_{i=1}^I Pr(sl_i = \psi(o_i)|\mathbf{p}_j) \quad (2.5)$$

where sl_i denotes the signal level of the on-line RSS reception of the i -th AP.

Maximum a posteriori (MAP) could be used for location determination, as:

$$\hat{\mathbf{p}} = \arg \max_{\mathbf{p}_j} Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o})) = \arg \max_{\mathbf{p}_j} Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j), \quad \forall j \in \{1, 2, \dots, J\} \quad (2.6)$$

If involving a number of locations with the highest *a posteriori* estimations, location determination could be given as the weighted averaged result of them, where

weights are equal to the *a posteriori* probabilities of them. The localisation is solved as a regression problem.

When insufficient number of RSS samples are collected from the site-survey phase potentially, $Pr(sl_i = \psi(o_i)|\mathbf{p}_j)$ may equal to 0, which implies that such a value of $\psi(o_i)$ has not been observed in the i -th AP's training RSS data collected at j -th location. Two methods to overcome the problem, as presented in [54, 56], are shown as follows.

2.3.1 Discretisation

This method is closely related to discretisation of continuous values to discrete ones. Firstly let us denote ψ_{\min} and ψ_{\max} as the minimum and maximum RSS value that can be detected by the smartphone. A group of non-overlapping intervals (bins) covering the entire range of the variable from ψ_{\min} to ψ_{\max} is set in this method. The number of bins is denoted as N_{bin} , which is an adjustable value. The set of bins could be then expressed as $\{B_1, \dots, B_\xi, \dots, B_{N_{bin}}\}$, where B_ξ denotes the ξ -th bin. Within every bin, the estimated probability density is a constant value. Also, the width of every bin is assumed the same. Bin width is calculated as $BW = \lceil \frac{\psi_{\max} - \psi_{\min}}{N_{bin}} \rceil$. A RSS value ψ is categorised into B_ξ , according to:

$$\xi = \lceil \frac{\psi - (\psi_{\min} - 1)}{BW} \rceil \quad (2.7)$$

As a result, for a RSS observation $\boldsymbol{\psi}(\mathbf{o})$ received in on-line phase, $Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j) = \prod_{i=1}^I Pr(sl_i = \psi(o_i)|\mathbf{p}_j) = \prod_{i=1}^I Pr(sl_i \in B_\xi|\mathbf{p}_j)$, where $\psi(o_i)$ is categorised into ξ -th bin.

For example, if for a smartphone, ψ_{\min} and ψ_{\max} are detected as -100 dBm and -34 dBm, respectively. If N_{bin} is set to 5, BW is thus calculated as $\lceil \frac{-34 - (-100)}{5} \rceil = 14$. So, for example, $\psi = -67$ dBm is categorised into $\xi = \lceil \frac{-67 - (-100 - 1)}{14} \rceil = 3^{rd}$ bin.

Since $Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j) = \prod_{i=1}^I Pr(sl_i \in B_\xi|\mathbf{p}_j)$, $Pr(sl_i \in B_\xi|\mathbf{p}_j) = 0$, for any i , will lead to zero problematic probability of $Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j)$. To avoid occurrences of such cases, a Laplace smoothing solution is given as:

$$Pr(sl_i \in B_\xi | \mathbf{p}_j) = \frac{\sum_{n=1}^{N_{coll}} \delta_D(sl_i \in B_\xi | \boldsymbol{\psi}(\mathbf{o}_n)) Pr(\mathbf{p}_j | \boldsymbol{\psi}(\mathbf{o}_n)) + 1}{\sum_{\xi=1}^{N_{bin}} \sum_{n=1}^{N_{coll}} \delta_D(sl_i \in B_\xi | \boldsymbol{\psi}(\mathbf{o}_n)) Pr(\mathbf{p}_j | \boldsymbol{\psi}(\mathbf{o}_n)) + N_{bin}} \quad (2.8)$$

Here, N_{coll} denotes the number of all collected RSS samples in the site-survey phase; $\delta_D\{\bullet\}$ denotes the Dirac Delta function whose value equals to 1/0 when the statement in $\{\bullet\}$ is true/false; $Pr\{\mathbf{p}_j | \boldsymbol{\psi}(\mathbf{o}_n)\} = 1$ if \mathbf{o}_n , the n -th RSS training observation, is collected on \mathbf{p}_j , otherwise 0.

2.3.1.1 Discretisation bias and variance

Although the NB algorithm is used for regression in this thesis since the location estimation is implemented on a continuous 2-D space, it is used originally for classification. The performance of NB classifiers is evaluated by their classification error. The error could be factored into three parts, including a bias term, a variance term and an irreducible term. In particular, the bias term is used to measure the component of error resulting from the systematic error of the learning algorithm; the variance term is related to the random variation in the training data and random behaviour in the algorithm, and therefore measures the sensitivity of the algorithm to the changes in the training data [57]. The variance increases when the sensitivity is improved. In common, there exists trade-off between bias and variance [58]. Implementations on some aspects of the algorithm always result in opposite effects on bias and variance. The concepts of bias and variance are explained in [3] through an example of the distribution of the arrows landed on a target, as illustrated in Figure 2.3 (reproduced from [3]). The true value could be regarded as the bull's eye on the target; The set of training data are regarded as the arrows fired at bull's eye on the target. The employed algorithm learns from training data as arrows fired at the bull's eye. Bias and variance figure out what happens when the employed algorithm is used as a archer to fire many arrows at the target (the process of estimation). Of course, the optimal situation is that all fired arrows by the archer are distributed compactly in the center of the target, as illustrated in Figure 2.3(d). Bias measures the distance between the estimations from the ground-truth value, if the estimations (shot arrows) land continuously

on the target in a same direction. On the other hand, large variance implies that the landed arrows are widely scattered on the target; dissimilar results are given by the repeated shots. Since a small variance often occurs at the same time with a large bias, and a large variance often occurs at the same time with a small bias, trade-off between them should be made by the algorithm: the arrows fired should be distributed on the target with relatively low bias and relatively low variance.

Discretisation of the continuous RSS values with different bin width is expected to affect the classification bias and variance of the generated NB classifier. The effects are termed discretisation bias and variance. For example, in a one-feature-only case, suppose that the true decision boundary distinguishing one class from the other is included in a bin. If two values are distributed on the different sides of the decision boundary, they should be, optimistically, classified into two classes. However, since they are discretised into the same bin, they will be classified incorrectly into the same class. This often occurs when the employed bin width is large. The larger the bin width, the more likely that a bin contains decision boundary. Under this situation, more instances will be misclassified, and therefore high discretisation bias could be expected. This is also well known as under-fitting problem [59]. This concept applies to multi-classes cases also.

One possible solution is to identify the decision boundaries and set the interval boundaries close to them. However, the decision boundaries are not known by the system as a prior knowledge. An extreme solution is not applying discretisation at all. In this way, any existing decision boundary is not possible to be included in any bin. However, under the situation that only insufficient number of training data is available to the algorithm, the number of training instances per bin is less than required. Therefore, estimation of the *a posteriori* probability is highly likely to be unreliable and the truly most probable class cannot be identified by the system. Also, the less training instances per bin available to the algorithm, the more likely that the variance of the algorithm increases, because even a small change in the training data might lead to a totally different estimation result. Here, the concept of high discretisation variance is explained, which is also well known as over-fitting problem. [59].

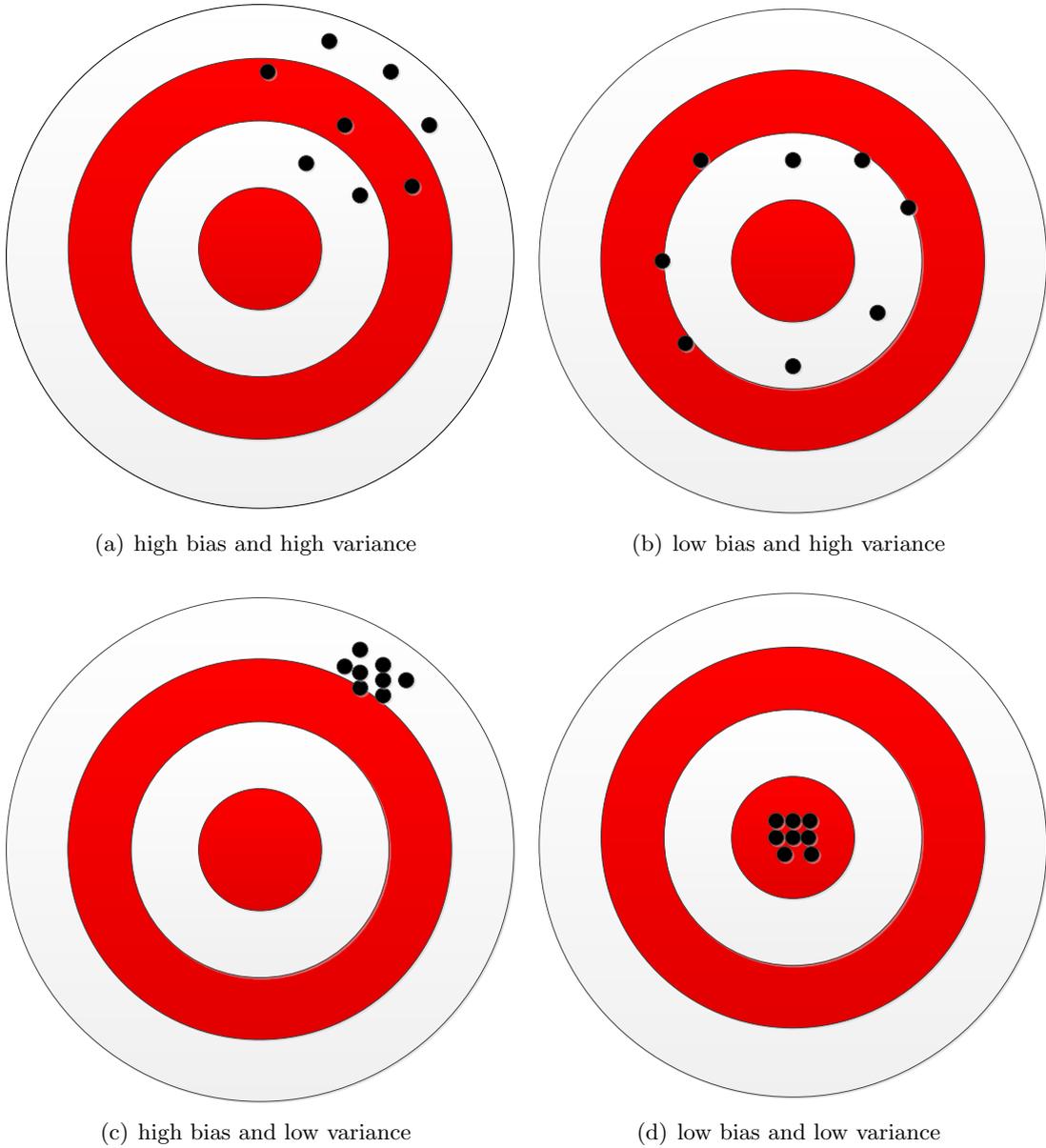


FIGURE 2.3: Bias and variance in the distribution of arrows landing on a target. Bias implies that the archer misses in the same direction, systematically. Variance implies that the archers are scattered ([3]).

2.3.2 Kernel method

In kernel method, the estimate of $Pr(\boldsymbol{\psi}(\boldsymbol{o})|\boldsymbol{p}_j)$ is obtained as:

$$Pr(\boldsymbol{\psi}(\boldsymbol{o})|\boldsymbol{p}_j) = \frac{1}{N_p} \sum_{n=1}^{N_p} K(\boldsymbol{\psi}(\boldsymbol{o}), \boldsymbol{\psi}(\boldsymbol{o}_j^n)) \quad (2.9)$$

Here, $K\{\bullet, \bullet\}$ denotes the used kernel function; $\boldsymbol{\psi}(\boldsymbol{o}_j^n)$ denotes the n -th RSS observation vector received at the j -th location from the site-survey phase. A widely-employed kernel function could be a Gaussian kernel, as:

$$K(\boldsymbol{\psi}(\boldsymbol{o}), \boldsymbol{\psi}(\boldsymbol{o}_j^n)) = \frac{1}{\sqrt{2\pi}\sigma_j^*} \exp\left(-\frac{\|\boldsymbol{\psi}(\boldsymbol{o}) - \boldsymbol{\psi}(\boldsymbol{o}_j^n)\|^2}{2(\sigma_j^*)^2}\right) \quad (2.10)$$

where σ_j^* denotes the optimal value of the adjustable specific kernel bandwidth σ_j for the j -th location. The smaller σ_j^* is, the smaller value is $K(\boldsymbol{\psi}(\boldsymbol{o}), \boldsymbol{\psi}(\boldsymbol{o}_j^n))$, and thereby the less correlation between $\boldsymbol{\psi}(\boldsymbol{o})$ and $\boldsymbol{\psi}(\boldsymbol{o}_j^n)$. To compute $Pr(\boldsymbol{\psi}(\boldsymbol{o})|\boldsymbol{p}_j)$ for a particular j -th location, all RSS observation vectors collected at it are requested to be involved in kernel computation. Moreover, to compute $Pr(\boldsymbol{\psi}(\boldsymbol{o})|\boldsymbol{p}_j)$, $\forall j \in \{1, \dots, J\}$, all training RSS observation vectors collected throughout the whole area of interest are requested to be involved for kernel computation, which consumes much more computational power than the previously-mentioned KNN deterministic method, where only the average of RSS samples from the site-survey phase are involved.

The accuracies of localisation systems applying such kernel method are highly dependent on the choice of kernel bandwidth σ_j^* [56]. In [56], the method of determining such a parameter is presented in detail.

The paper [54] presents comparable positioning accuracies for the localisation system applying discretisation and kernel methods. However, a high computational load is expected for the kernel method. As a result, this thesis focuses on the discretisation method.

2.4 The applied machine learning algorithms

In this section, the machine learning algorithms involved in the implementation of the thesis or used to be compared with the counterpart chosen in the implementation, such as the k -means, k -means++, the affinity propagation clustering algorithm, binary SVM, multi-classes SVMs, Fisher kernel, etc., are presented.

Finally, the concepts of generative approaches and discriminative approaches are explained. A comparison of them is also made.

2.4.1 k -means and k -means++

The k -means clustering algorithm is proposed originally by Stuart Lloyd in [60]. So, it is also well known as the Lloyd's algorithm. The purpose of the k -means algorithm is to distribute J data points into k clusters in which each data point belongs to the cluster with the closest Euclidean distance. Each of the data points functions as a member of the assigned cluster ($J > k$). Commonly, the clustering results are refined iteratively. Initially, k centroids are selected randomly with the implementation of the k -means clustering algorithm. Following that, two steps named as assignment step and update step are involved in every iteration. In particular, in the assignment step, each data point is arranged to a cluster in a way that the sum of squares of distances between membering data points and the corresponding centroid of the cluster is minimised. Following that, in the update step, the new centroid of each cluster is updated as the mean of the updated membering data points in the cluster. The k -means algorithm is guaranteed to find a local optimum point. However, since the centroids are initialised arbitrarily, global optimum point might be not found.

On the other hand, centroids are initialised with a deliberate selection scheme by the k -means++ clustering algorithm proposed in [61]. The selection scheme tries to make the initialised centroids be isolated from each other at a higher level than the standard k -means algorithm. The possibility of an data point being chosen as a centroid of a cluster is inverse proportional to the shortest distance from itself to the closest centroid which has been chosen already. Among k initialised centroids, only the first one is picked arbitrarily from the range of n_o observations. After all centroids of k clusters have been selected, the iteration for refinement of centroids and reassignments of cluster memberships to each data point is implemented as the standard k -means algorithm. Note that for both of the algorithms, the value of k should be designated manually.

The convergence criteria of k -means and k -means++ is that the cluster membership assignment of all the observations does not change any more from the previous iteration to the current one.

2.4.2 Affinity propagation clustering algorithm

In the affinity propagation clustering algorithm, clusters and corresponding exemplars are determined by recursively conveying real-valued messages between data samples. Such messages include similarity measurements between pairs of data samples. Specifically, similarity measurement $sm(i, j)$ implies the appropriateness level of employing an i -th data sample as an exemplar of a j -th data sample. Self-similarity measurement such as $sm(i, i)$, termed as preference, implies the prior appropriateness level of an i -th data sample being the exemplar of one of the cluster. Commonly, this value is set equal to every data sample, if the possibility of each data sample being an exemplar is assumed the same before practical implementation of the clustering algorithm. Furthermore, a higher value of preference set for data samples may result in more clusters generated. Reasonably, the preference of a particular data sample is set proportional to the median value of the similarity measurements between other data samples and itself.

Two classes of messages including a responsibility message and an availability message are transmitted between data samples. In detail, a responsibility message ($re(i, j)$) is conveyed from an i -th data sample to a j -th data sample, informing the appropriateness level of the j -th data sample being the exemplar of the i -th data sample; availability message ($av(i, j)$) is conveyed from the j -th data sample to the i -th data sample, informing the appropriateness level of the i -th data sample choosing the j -th data sample as its exemplar. Mathematically, they are denoted as:

$$re(i, j) = sm(i, j) - \max_{j' \neq j} \{av(i, j') + sm(i, j')\} \quad (2.11)$$

and

$$av(i, j) = \min \left\{ 0, re(j, j) + \sum_{i' \neq i} \max\{0, re(i', j)\} \right\} \quad (2.12)$$

Two other messages self-responsibility $re(i, i)$ and self-availability $av(i, i)$ are also required, as:

$$re(i, i) = sm(i, i) - \max_{j' \neq j} \{av(i, j') + sm(i, j')\} \quad (2.13)$$

and

$$av(i, i) = \sum_{j' \neq i} \max\{0, r(j', i)\} \quad (2.14)$$

Specifically, they reflect the accumulated evidence that the i -th data sample should serve as an exemplar.

To identify exemplars, for the i -th data sample, find a data sample indexed as j' , as:

$$j' = \arg \max_j \{av(i, j) + re(i, j)\} \quad (2.15)$$

Overall, there are two possible outputs. If $j' = i$, the i -th data sample is regarded as an exemplar. Otherwise, the j' -th data sample is regarded as exemplar of the i -th data sample. The messages are transmitted recursively between pairs of data samples by following the above updating rules, until a proper group of exemplars and related clusters are finally formed.

Similar to the work done in [34], for indoor localisation application, the similarity measurement between the averaged RSS vector on the i -th RP and the j -th RP from the training phase is calculated as

$$sm(i, j) = \|\bar{\psi}_i - \bar{\psi}_j\|^2, i \neq j \quad (2.16)$$

and self-similarity $sm(i, i)$ is calculated as

$$sm(i, i) = \text{median}\{sm(i, j)\}, \forall j \in [1, J], i \neq j \quad (2.17)$$

2.4.3 SVM classifiers

Firstly, the basic knowledges of binary SVM classifier are reviewed. Following that, two derivatives of it for solving the classification problem for multi-class samples are reviewed.

2.4.3.1 Binary SVM

SVM, a supervised algorithm applied for binary classification and regression analysis, is initially proposed in [62, 63]. In the training phase, a group of labelled training samples is employed to build the mathematical model for the SVM algorithm. With separable training data sets labelled with +1/-1, there exist many possible choices of building the mathematical model for the classifier. Particularly, the SVM looks for the decision hyperplane that represents the largest separation (margin) between the two classes. The distance between the defined decision hyperplane and the closest sample is termed the margin of the classifier. In other words, the hyperplane is chosen in a way that the distance from it to the nearest data point on each side is maximized. As could be found out, the decision hyperplane of the SVM is determined by a few data samples. As a result, these data samples are referred to as the support vectors.

For a non-linearly separable data sample set, some of the data samples (outliers) are inevitably misclassified by the decision hyperplane if no non-linear kernel function is used. Such misclassifications are required to be taken into account in a new optimisation model. In other words, the learned mathematical model should generalise the training data properly by preventing serious classification errors. To solve this problem, slack variables are used to build the new mathematical optimisation model. Details of the related mathematical models and solutions can be

found in [63–65]. In the thesis, a publicly-available powerful SVM implementation software library, named LIBSVM [66], is applied.

2.4.3.2 Multi-class SVMs

Multi-class SVMs could be achieved from three different implementations named as: one-vs-all (OVA) SVMs, one-vs-one (OVO) SVMs [67, 68], regularised multi-class least square SVMs (LS-SVMs). Training data samples with multi-class labels are required by each of them to train the mathematical classification model. In particular, performance of OVO-SVMs and OVA-SVMs is evaluated for rough localisation purpose while LS-SVMs [69] are applied as a discriminative approach to improve the accuracies of the estimation results from the NB algorithm, as implemented in [1].

For OVA-SVMs, totally N_c (number of different labels) SVMs are trained. Each of N_c hyperplane aims to separate a particular cluster from all RPs excluded from the cluster. An on-line RSS vector is substituted into every one of N_c trained hyperplanes to get a decision value. The sign of the value implies the decision: belonging to the cluster if it is positive (+) but not if otherwise. Finally, the on-line RSS vector will be assigned membership of the cluster whose decision function value is positively the highest.

For OVO-SVMs, totally $\binom{N_c}{2}$ SVMs are trained. Each of $\binom{N_c}{2}$ hyperplanes aims to separate each particular pair of two clusters. Similarly, an on-line RSS vector is substituted into each of $\binom{N_c}{2}$ hyperplanes, and for each substitution a decision value is returned. The sign of the value implies the decision: belonging to the cluster with the higher index if it is positive (+) or the one with the lower index if it is negative (-). Finally, the on-line RSS vector will be assigned membership of the cluster that is nominated with the highest frequency.

Equality constraints are applied to choose the parameters for minimisation of the objective function of the LS-SVMs [69]. Instead of a convex quadratic programming (QP) problem being required to be solved for finding the decision hyperplanes of classical SVMs, a set of linear equations is required to be solved for finding the

decision hyperplane of the LS-SVMs. Also, the LS-SVM is also known as a kernel-based learning method. The applied kernel could be in the form of linear kernel, polynomial kernel, radial basis function, etc. In [1], the Fisher kernel is applied to feed the probabilistic estimation result obtained from the NB algorithm to the LS-SVMs. Therefore, Fisher kernel will be particularly reviewed in the next section.

2.4.4 Fisher kernel

The Fisher kernel, proposed in [70], is derived to train a generative model on the available data to be used by discriminative approach. Probability distribution is taken advantage of to afford a similarity measurement on targeted objects. Given the parameter set denoted as $\boldsymbol{\varepsilon}$, according to [70], the Fisher score is defined as:

$$FS(\boldsymbol{\varepsilon}, \boldsymbol{\psi}(\boldsymbol{o})) = \nabla_{\boldsymbol{\varepsilon}} \ln Pr(\boldsymbol{\psi}(\boldsymbol{o}); \boldsymbol{\varepsilon}) \quad (2.18)$$

which is a vector of the gradient of the log-likelihood of $\boldsymbol{\psi}(\boldsymbol{o})$ with respect to the generative model parameters $\boldsymbol{\varepsilon}$. The Fisher kernel refers to as the inner product in the space defined by the Fisher score, and is defined for the m -th and n -th on-line RSS observations, $\boldsymbol{\psi}(\boldsymbol{o}_m)$ and $\boldsymbol{\psi}(\boldsymbol{o}_n)$, as:

$$K(\boldsymbol{\psi}(\boldsymbol{o}_m), \boldsymbol{\psi}(\boldsymbol{o}_n)) = FS(\boldsymbol{\varepsilon}, \boldsymbol{\psi}(\boldsymbol{o}_m))^T IM^{-1} FS(\boldsymbol{\varepsilon}, \boldsymbol{\psi}(\boldsymbol{o}_n)) \quad (2.19)$$

Here, IM denotes the Fisher information matrix. Practically, evaluation of the Fisher information matrix is always infeasible [1]. Therefore, the Fisher kernel is simplified as:

$$K(\boldsymbol{\psi}(\boldsymbol{o}_m), \boldsymbol{\psi}(\boldsymbol{o}_n)) = FS(\boldsymbol{\varepsilon}, \boldsymbol{\psi}(\boldsymbol{o}_m))^T FS(\boldsymbol{\varepsilon}, \boldsymbol{\psi}(\boldsymbol{o}_n)) \quad (2.20)$$

2.4.5 Generative approaches vs. discriminative approaches

Algorithms trying to learn the conditional probability of labels given the input data pattern directly (such as SVM classifiers), or algorithms that try to learn mappings directly from the space of inputs to the binary-class or multi-class labels are termed discriminative approaches [71].

On the other hand, algorithms that instead try to model conditional probability of input data pattern given labels (and prior probability of labels) are termed generative approaches [71]. On such example is the probabilistic NB algorithm. Before using the algorithm for estimation, the algorithm should learn the conditional probability of input data pattern on labels and labels' *a priori* probability in advance.

In [71], the performance of a generative approach, the probabilistic NB algorithm, and a discriminative approach, logistic regression [72], is compared with the scale of the database of the training samples being varied. Two conclusions are drawn from the experimental results. Firstly, The generative approaches possess higher (worse) asymptotic errors (as the number of training examples becomes large) than the discriminative approaches. However, The generative model may also approach its asymptotic error much faster than the discriminative approaches—probably with a number of training data that is only logarithmic, rather than linear, in the number of parameters required to be fit in the model. These two conclusions are very important to decide which category of algorithms (generative or discriminative) should be chosen to full-fill the task of solving classification problems when the scale of database of the training data is a prior knowledge available to the localisation system. This also affords an intuition of which of following algorithms, the KNN (a discriminative approach) or the probabilistic NB algorithm, should be chosen to be applied in fine localisation phase. Since the proposed localisation algorithm in this thesis requires a more efficient site survey (less training samples are collected on each RP) to be implemented before providing the on-line localisation service, the probabilistic NB algorithm is chosen.

However, the performance of the combination of a generative approach and a discriminative approach by taking advantage of the Fisher kernel is not evaluated

in [71]. Instead, [1] shows that such combination could calibrate the estimation results given by the generative approach when the generative model (conditional probability of input data pattern given labels) are calibrated with the unlabelled on-line data observations.

2.5 DR methods based on the IMUs

DR methods update a user's current position by taking into account moving speed, moving direction over the current time interval, based on the previously estimated position. Due to measurement errors, position updating results will be biased from the ground-truth ones more and more over time. The accumulation of error is widely known as drifts. The periodic resetting of estimated positions by taking advantage of absolute localisation technologies such as RFID tags or Wi-Fi LF is required by [50].

Conventionally, an IMU includes a 3-axis accelerometer and a 3-axis gyroscope. However, without knowledge of the absolute orientation, only orientation changes could be tracked. To estimate the instantaneous absolute orientation, a 3-axis magnetometer should be applied, which is also regarded as a member of IMU device. Inertial navigation algorithm updates the IMU's position and velocity vector in a desired reference frame, usually global reference frame, as shown in figure 3.2 included in the next chapter.

There are many different designs of IMU sensors. Generally, they could be divided into two groups: stabilised platform and strapdown platform.

Traditionally, stable-platform technologies, as shown in figure 2.4, are used for inertial navigation applications in objects such as spacecraft, missiles, ships, etc. The IMU sensors are mounted on a stable platform that is isolated mechanically from the rotational motion of the carrier. The isolation is achieved by rotating the gimbals by corresponding torque motors to counteract the rotational motion of the carrier. Carrier-experienced rotations are sensed by the gyroscopes, and resulting attitude tracking information is fed back, as a reference, to generate the gimbals' rotation control signal to keep the platform stable. In this way, the accelerometer

readings are already in a global reference frame, and could be directly used to update velocity and distance.

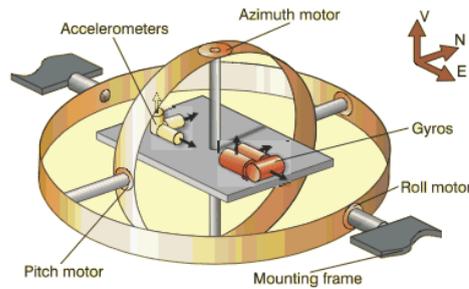


FIGURE 2.4: Stable-platform IMU, from[4].

While stable-platform IMUs are reliable and accurate, their complicated mechanical structure prevents their wide usage in modern carriers such as thin smartphones. On the other side, strapdown IMUs have removed the mechanical complexities of stable-platform IMUs by attaching sensors rigidly to carriers. The major disadvantage of the method is the substantial increase of computational loads, which is becoming a less serious problem, due to the rapid development of the modern mobile CPU technologies. Therefore, IMUs applying strapdown structure are becoming widely used in smartphones. In this method, the attitude of the IMUs is applied to convert measurements from accelerometers in a local reference frame to the desired reference frame. Acceleration due to gravity is then subtracted from the transformed readings to obtain accelerations enforced by external forces other than gravity. To track the attitude of IMUs, readings from the gyroscope and the magnetometer are also applied.

However, drift in position estimation from a conventional DR tracking method is expected to increase at a rate that is proportional to t^3 , where t denotes the device's traversed time [73], which is partly a result of deviated gyroscope measurements. Note that converting accelerations from the local reference frame to the desired reference frame requires knowledge of gyroscope readings. Given erroneous gyroscope reading, converted accelerations in the desired reference frame will also be deviated. According to the performance difference, gyroscope sensors could be divided into four groups: consumer-grade, automotive-grade, technical-grade and navigation grade, of which details are shown in table 2.1. From the table, it

could be figured out that performance of consumer-grade gyroscopes embedded in smartphone is the worst of all.

Besides gyroscope measurement errors, another contribution to position estimation drifts is from errors in accelerometer measurement themselves. Details of accelerometer measurement errors are shown in Section 3.2.

| | Consumer | Automobile | Technic | Navigation |
|-----------------------|------------|--------------|------------------|------------|
| Performance (deg/hr) | >200 | 10-200 | 0.1-10 | <0.01 |
| Price (\$) | 50-1,000 | 5,000-10,000 | 10,000-15,000 | >100,000 |
| Classical application | smartphone | automobile | guided munitions | spacecraft |

TABLE 2.1: Performance vs. Price of gyroscope

Two solutions have been given out to overcome the problem resulting from the poor performance of consumer-grade IMUs, which are frequently-calibrated PDR and step-based PDR.

2.5.1 Frequently-calibrated PDR

The conventional inertial navigation method, as shown in Figure 3.3, is applied to update the object's movement. However, the rapidly accumulated position estimation errors are calibrated frequently by taking advantage of the fact that zero velocity presents when a foot hits the ground in the stance phases. With this method, IMUs are requested to be placed on the shoe of that foot. Generally, The method is also called zero-velocity update (ZUPT).

Detection of such stance phases is very important. Many methods, such as accelerometer or gyroscope readings magnitude based, magnetometer based, neural network based, etc., have been proposed, as shown in [74]. Once upon the start and end moments of stance phases have been detected, any non-zero velocity measurement during the phases could be viewed as errors. Then, the averaged value of velocity errors during every stance phase could be propagated back to calibrate the velocity values during the previous movement period.

ZUPT does not make any assumption about the user's movement. As a result, it is also suitable for distance updating when users run, walks backward or even

steps sideways. However, the requirement of mounting IMU on a foot has to be met, which is not convenient for common smartphone users.

2.5.2 Step-based PDR

In contrast to ZUPT, a step-based PDR has no such device mounting position requirement, and is widely used in smartphone tracking applications and in this thesis also. Step-based PDR, commonly, is composed of three techniques: stepping moments recognition, step length estimation and step orientation estimation.

Firstly, each stepping moment need to be recognised. Commonly, smartphone is either put in user's pocket or held in user's hand. For this two smartphone poses, acceleration magnitudes of carrying IMUs, in the walking status, are found to be possessing repetitive pattern, which could be employed for recognition of step moments.

The most straightforward method of step length estimation is to assign each recognised step of every user a constant step length. However, step length, for a particular user, is likely to vary during a walking period with different walking frequency. Also, average step length varies across different users of different heights, of which the most evident events could be found as observing walks of teenagers and adults. As a result, a mathematical model taking in account all these factors is required. For example, In [33, 75], such model is applied, as:

$$l = k_1 \cdot t_{step} + k_2 \quad (2.21)$$

where l denotes step length; t_{step} denotes the reciprocal of walking frequency; k_1 and k_2 are two parameters that are required to be personalised. In [33], k_1 and k_2 , initially, are set as the same as the parameter values of a generic model learned from over 4000 manually labelled step samples from 23 testers with diverse physical characteristics. Then, k_1 and k_2 are finely tuned, according to the combination of turning detection and re-sampling methods applied in the on-going implemented customised particle filtering algorithm. Also, in [33], the evaluation results show

that the estimated step length error are reduced to be lower than 10% after 5 significant turns have been experienced by the testers on average. In the implementation of this thesis, efforts are made to reduce the complexity and time cost involved in learning k_1 and k_2 . Specifically, for accurate stepping moments recognition and step length estimation, every user is requested to take a trial to learn the parameters for the two tasks simultaneously before activating the localisation service, of which details are shown in Section 3.4.4.

Theoretically, a magnetometer-alone could be used to derive the absolute orientation of a user's every step. However, magnetometer measurement is highly likely to be corrupted by interference sourcing from surrounding magnetic material, which is common in an indoor scenario that is filled with easily magnetised electronic devices. To calibrate the orientation estimation, gyroscope measurement is utilised as another orientation information source such as in [76, 77]. Similar to their implementation, in this thesis, a complementary filter is applied to combine information from two respective sources, to estimate the user's orientation.

2.6 Route planning and application

In the graph theory, a graph contains a group of vertices where some pairs of them are connected with each other by links termed edges. An example is shown in Figure 2.5. As could be observed from the figure, there are five vertices and some of them are mutually connected to each other. An adjacency matrix, a method of describing which vertices are adjacent to which other vertices [78], is employed to equivalently express the graph mathematically. For example, the value of the i, j -th entry (1/0) of a adjacency matrix represents the connectivity of the i -th and j -th vertex (connect/disconnect) of the graph. Therefore, the adjacency matrix of the graph shown in Figure 2.5 is denoted as:

$$AJ = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 \end{bmatrix} \quad (2.22)$$

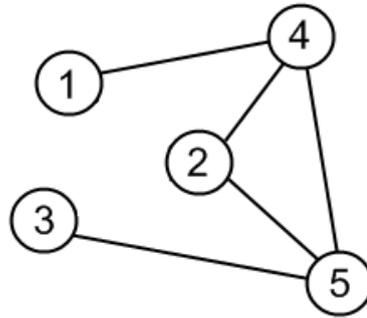


FIGURE 2.5: An example of graph model

With the knowledge of the adjacency matrix, the route between each pair of vertices could be found by taking advantage of a dedicated route planning algorithm. A route planning algorithm named Dijkstra's algorithm is hereby reviewed.

Dijkstra's algorithm is designed for finding the shortest path between any two vertices that are reachable from one to another; it was originally published in [79]. The algorithm works step by step as follows:

Step 1: Tentative distance values ($Dist$) are distributed to all vertices. For the source vertex, this value is set as 0. In contrast, the value is set as infinity to all other vertices.

Step 2: Mark the source vertex as the currently visited node. Create a set consisting of all previously unvisited vertices, which is termed the unvisited set.

Step 3: For the currently visited node, with the aid of an adjacency matrix, all its neighbouring vertices could be known. For the i -th neighbouring vertex, the tentative distance is calculated as $Dist(VX_c) + EG(VX_c \rightarrow VX_{n_i})$. Here VX_c denotes the currently visited node and VX_{n_i} denotes the i -th neighbouring vertex of VX_c ; EG denotes the value of the edge. Provided that the newly calculated tentative

distance for a neighbouring vertex is smaller than the previously stored one, the tentative distance is updated and its ancestor node is set as the currently visited node. Otherwise, both the distance and ancestor node are not changed. The calculation and comparison should be done for all neighbouring vertices included in the unvisited set.

Step 4: Exclude the currently visited node from the unvisited set.

Step 5: Set the unvisited vertex that holds the lowest tentative distance value as the new currently visited node, and the algorithm goes back to step 3 to start a new loop. The loop will end when the destination vertex is selected as the currently visited node.

Step 6: Provided that the destination vertex has been excluded from the unvisited set, the loop stops, and a route is generated by reversing the order of a series of ancestor nodes of the destination node to the source node.

2.7 Recursive Bayesian estimation

The inherent time-varying nature of wireless signal degrades the accuracy of the Wi-Fi LF localisation service, which activates the usage of other information sources, such as step-based PDR, to calibrate the localisation results. The recursive Bayesian estimation method, also known as Bayesian filtering, recursively over time, probabilistically estimates the state of the dynamic system which could only be observed indirectly via noisy measurements. For a pedestrian localisation task, the state of the dynamic system refers to as the vector containing all available information related to the movements of the user, such as the user's 2-D position, velocity, orientation, etc. In this section, Bayesian filters are introduced in their general form.

Suppose a series of time-evolved hidden state vectors are represented as $\{\mathbf{x}^0, \mathbf{x}^1, \dots, \mathbf{x}^T\}$; the corresponding measurement vectors are denoted as $\{\mathbf{m}^1, \mathbf{m}^2, \dots, \mathbf{m}^T\}$, the joint posterior probability of all states, given all measurement, is denoted as:

$$Pr(\mathbf{x}^{0:T}|\mathbf{m}^{1:T}) = \frac{Pr(\mathbf{m}^{1:T}|\mathbf{x}^{0:T})Pr(\mathbf{x}^{0:T})}{Pr(\mathbf{m}^{1:T})} = \frac{Pr(\mathbf{m}^{1:T}|\mathbf{x}^{0:T})Pr(\mathbf{x}^{0:T})}{\int Pr(\mathbf{m}^{1:T}|\mathbf{x}^{0:T})Pr(\mathbf{x}^{0:T})d\mathbf{x}^{0:T}} \quad (2.23)$$

In this framework, $Pr(\mathbf{x}^{0:T}|\mathbf{m}^{1:T})$ is required to be updated whenever one new measurement is obtained.

As could be found, the complexity of computing $Pr(\mathbf{x}^{0:T}|\mathbf{m}^{1:T})$ increases exponentially as T increases [80], and finally becomes intractable. As a result, the class of dynamic model has to be restricted to be a Markov process to avoid computational intractability.

In the Markov process, three models are usually involved: initial prior distribution of the state $Pr(\mathbf{x}^0)$, propagation model $Pr(\mathbf{x}^T|\mathbf{x}^{T-1})$ and measurement model $Pr(\mathbf{m}^T|\mathbf{x}^{T-1})$. For the first-order Markov process, two assumptions are made. Firstly, the current state only depends on the most recent previous state. Secondly, the current measurement vector only depends on the current state. Also, instead of $Pr(\mathbf{x}^{0:T}|\mathbf{m}^{1:T})$, the marginal distribution of the current state \mathbf{x}^T given the measurements made by now $\mathbf{m}^{1:T}$, $Bel(\mathbf{x}^T) = Pr(\mathbf{x}^T|\mathbf{m}^{1:T})$, is considered.

Under the Markov assumption, $Bel(\mathbf{x}^T)$ can be calculated by updating the distribution at the previous time moment $Bel(\mathbf{x}^{T-1})$. A Bayesian filter is applied to track the state of the dynamic system over time. Note that $Pr(\mathbf{x}^T|\mathbf{m}^{1:T-1})$ is denoted as $Bel^-(\mathbf{x}^T)$. The posterior belief of the most recent previous state, $Bel(\mathbf{x}^{T-1})$, is propagated forward to perform the update according to the propagation model of the system, as:

$$Bel^-(\mathbf{x}^T) = \int Pr(\mathbf{x}^T|\mathbf{x}^{T-1})Bel(\mathbf{x}^{T-1})d\mathbf{x}^{T-1} \quad (2.24)$$

$Bel^-(\mathbf{x}^T)$ is then corrected with the measurement \mathbf{m}^T to obtain the posterior distribution $Bel(\mathbf{x}^T)$, as:

$$Bel(\mathbf{x}^T) \propto Pr(\mathbf{m}^T|\mathbf{x}^T)Bel^-(\mathbf{x}^T) \quad (2.25)$$

Note that applying more than one measurement source in a single update of $Bel(\mathbf{x}^T)$ is also possible: the previously updated posterior is used as the prior in successive iterations of the correction step. In this way, different types of measurement could be incorporated with different measurement models to correct the prior distribution. Therefore, Bayesian filters are regarded as supporting sensor fusion scheme.

Here, we can know that a framework for estimating the state of a dynamic system over the time domain probabilistically could be provided by the Bayesian filters. In practice, different Bayesian filtering techniques are developed given different scenarios. The most widely used types of Bayesian filter are described below.

2.7.1 Kalman filter

A Kalman filter [81, 82], recursively, estimates the state of a dynamic system with the target of minimising the mean of the squared error (MMSE). In a Kalman filter, the system and measurements noises are assumed to be white Gaussian distributed. Here, ‘whiteness’ implies that the noise is time-uncorrelated. In particular, the propagation model and the measurement model are shown, respectively, as:

$$\mathbf{x}^t = \mathcal{A}\mathbf{x}^{t-\delta t} + \mathcal{B}\mathbf{u}^{t-\delta t} + \boldsymbol{\omega}^{t-\delta t} \quad (2.26)$$

and

$$\mathbf{m}^t = \mathcal{H}\mathbf{x}^t + \mathbf{q}^t \quad (2.27)$$

where \mathcal{A} , the state transition matrix, relates the state at the previous time step to the current step; \mathcal{B} , the control matrix, maps the optional control input, $\mathbf{u}_{t-\delta t}$, to the current state; \mathcal{H} , the measurement matrix, relates the measurement vector to the current state; $\boldsymbol{\omega}$ and \mathbf{q} denotes the system and measurement errors, both of which follow multivariate Gaussian distribution with different covariance matrices. Under these assumptions of noise distribution, the updated state \mathbf{x}^t could also be found to follow a Gaussian distribution, where the mean value, median value and mode value coincides with each other. The optimal estimation could be given by

either of them. However, the assumption of the Gaussian probability distribution of the noise is too strict, which makes it potentially not suitable for practical implementation.

2.7.2 Particle filter

On the other side, a particle filter [83] does not have any restrictions on the form of propagation model and measurement model, which is more suitable for practical implementation.

In a particle filter, estimation belief at time t is approximated by a group of weighted particles, as:

$$\Lambda^t = \{\mathbf{x}_i^t, \omega_i^t\}, i = 1, 2, \dots, N_s. \quad (2.28)$$

Here, ω_i^t denotes the weight of the i -th particle at time t . The weights are normalised so that $\sum_{i=1}^{N_s} \omega_i^t = 1$. Then, the posterior estimation of \mathbf{x}^t is given as:

$$Bel(\mathbf{x}^t) = \sum_{i=1}^{N_s} \mathbf{x}_i^t \omega_i^t \quad (2.29)$$

Every particle is evolved over time through a propagation model. Then, the weight of every particle is adjusted according to measurements included in the measurement model. When the corresponding weights are lower than a certain threshold, the particles are filtered out as well, since they are regarded as being biased from the true states. As a result, particles are required to be re-sampled in proportion to their weights, to keep the number of particles employed for every iteration approximately the same. Also, the weights of re-sampled particles are assigned equally.

In the implementation of this thesis, both map constraints and RSS LF indoor localisation results, as measurements, are included in forming measurement model. Firstly, once upon an evolved particle is found to cross any segment of walls shown on the map, it will be assigned zero weight and therefore filtered out. Then, when a

significant user's body orientation turning is detected, the weights of the surviving particles are adjusted according to the conditional probability of the user being on their position given the applied on-line RSS measurement. Finally, re-sampling is implemented. Details of the above implementation are included in Section 5.3.3.

2.8 Chapter Summary

This chapter consists of four major parts: an overview of localisation technologies, mathematical methods used to process the information sourced from the localisation technologies, DR methods based on IMUs, and the recursive Bayesian estimation.

In the first part, the advantages and disadvantages of the state-of-art localisation technologies are presented. Specifically, to apply some of them, dedicated devices are required to be taken by users, which prevents them from being widely utilised.

In the second part, firstly, triangulation methods for indoor localisation are introduced. Then, the RSS-based methods are focused on, due to the easy-to-employ nature of Wi-Fi localisation system. In particular, the RSS-based methods are categorised into two classes: empirical model-based and LF-based methods, according to their different ways to characterise the RSS-position dependency. Next, two different ways to process the location RSS fingerprints in the LF-based algorithms, discretisation and kernel method, are introduced. Especially, the reasons leading to high discretisation bias and high discretisation variance are explained. Following that, the machine learning algorithms applied in the implementation or tests of the thesis are reviewed.

In the third part, initially, two different designs of IMU sensors, stabilised platform and strapdown platform, are introduced. Specifically, low mechanical complexity enables the strapdown platform IMU to be a good candidate used on modern carriers such as smartphone. Next, conventional inertial navigation methods are shown to work not good for PDR, due to the performance constraints of the applied customer-grade IMU devices in smartphones. Instead, two alternative approaches, named frequently calibrated PDR and step-based PDR, are presented. Among

them, step-based PDR does not have any strict restriction of sensor mounting position, which makes it popular for practical implementation.

Since measurements made by the sensors are unavoidably noisy, recursive Bayesian estimation is necessary to be used for fusing information from different types of sensors to obtain estimation results with lower uncertainty. Detailed implementation of the recursive Bayesian estimation, such as the Kalman filter and the particle filter are explained. Whilst strict assumptions of the system and measurement noises are made by the Kalman filter, these assumptions could be relaxed for the particle filter. Usually, implementation of the particle filter consists of three parts to process particles in sequence: a propagation through propagation model, a calibration by measurement model and a re-sampling technique.

Chapter 3

Inertial navigation and pedestrian trajectory recognition

3.1 Introduction

Nowadays, almost all smartphones on the market are equipped with three types of inertial sensors: accelerometer, magnetometer and gyroscope. Among them, an accelerometer and a gyroscope are Micro Electronic Mechanical System (MEMS) devices. Magnetometer sensors in smartphones apply the modern solid state technology to create a miniature Hall-effect sensor detecting the earth's magnetic field. Optimistically, given a user's starting position, his/her moving trajectory could be inferred easily from the information sourced from magnetometer and accelerometer. Specifically, the moving distance is calculated as taking double integration of acceleration in the moving direction over travelling time; the heading change of users could be tracked by taking integration of horizontal angular velocity provided by the gyroscope; a user's initial heading could be inferred from readings of an accelerometer and a magnetometer. This conventional DR tracking method is widely applied in inertial navigation system of ships, submarines, spacecraft, missiles, robots, etc [31].

However, in practice, due to the budget and size constraints, the performance of IMUs is lower than the devices applied in military or maritime applications.

By taking integration of raw readings, measurement errors could be accumulated over time quickly. For long travelling time, even though errors of every piece of measurement is small, a highly biased estimation of the moving trajectory would be generated. As a result, rather than the DR method, some other methods shown in this chapter are applied to estimate a much more accurate trajectory from outputs of these sensors.

The rest of this chapter is organised as follows: Section 3.2 presents an introduction to IMU sensors on smartphones. Specifically, a physical interpretation of output readings from these sensors for AndroidTM mobile operating system will be covered also. In Section 3.3, an overview of the conventional inertial navigation method will be drawn. the performance of such a method will be also evaluated. Especially, serious drift resulting from distance update by such a method will be presented. Finally, in order to overcome the serious drift, in Section 3.4, the proposed step-based PDR method will be presented. A novel contribution such as a simplified step length estimation method is also included.

3.2 Inertial sensors of smartphones

3.2.1 Gyroscope

A gyroscope is used to measure angular rotation velocity in the unit of radian per second (rad/s).

Figure 3.1 shows the smartphone reference frame, the same as that applied by the AndroidTM mobile operating system. The three axes of installed gyroscope are exactly aligned with the smartphone's reference frame. The output of the units are angular velocities $\Omega_x, \Omega_y, \Omega_z$ around three axes. To determine the polarities of measured angular rotation velocities, the right-hand rule is applied by the AndroidTM mobile operating system. For example, if a phone stands vertically on table and the user rotates the phone anti-clockwise around the y axis, the polarity of the output signal of the y axis should be positive.

For the AndroidTM mobile operating system, the sampling frequency of the gyroscope could be defined by developers for different purposes of application. In the implementation of this thesis, the sampling frequency is set as 50Hz.

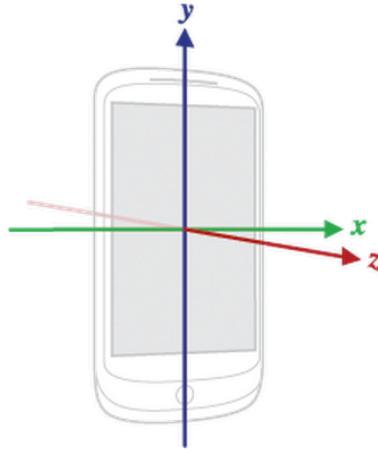


FIGURE 3.1: Smartphone Reference Frame, from [5]

3.2.1.1 Gyroscope measurement error

Measurement errors of MEMS gyroscopes are usually characterised into two groups: systematic bias and random noise.

Systematic bias around a certain axis could be measured as the mean value of the output reading of the axis, when the sensor is kept without any rotation. The measured systematic bias will result in an angular measurement error growing linearly with time. However, this could be easily done in a trial phase. Accordingly, an angular measurement around any axis, in practice, could be calibrated by subtracting the product of the observed systematic bias and the measurement time t from it.

The gyroscope output signal is also corrupted by random noise originating from thermo-mechanical events and electrical noise. Provided that system bias has been calibrated, the remaining errors could be modelled as the variable following a zero-mean Gaussian distribution. While tracking the attitude change, such errors are also integrated over time, which leads to an angle random walk (ARW) error following a zero-mean Gaussian distribution where the standard deviation increases

proportional to \sqrt{t} . ARW is commonly specified in terms of standard deviation [84]: error units per square root of the time unit. for example, deg/\sqrt{hr} .

3.2.2 Accelerometer

Accelerometers measure the acceleration induced by forces acting on smartphones along three orthogonal axes in space.

Similarly to a three-axis gyroscope and a magnetometer, the axes of the output signal from an accelerometer is also aligned with the smartphone reference frame shown in Figure 3.1. The output signal for three axes are defined as \mathbf{a}_x , \mathbf{a}_y , \mathbf{a}_z respectively. In particular, \mathbf{a}_x , \mathbf{a}_y are defined within the plane of the phone while \mathbf{a}_z is defined along the axis perpendicular to the plane of the phone. Moreover, polarities of output signals are defined as the directions of the arrows shown in Figure 3.1. For example, coordinates behind the screen possess negative value along the z axis.

For the AndroidTM mobile operating system, the sampling frequency of the accelerometer could be defined by developers for different purposes of applications. In the implementation of this thesis, the sampling frequency is set as 50Hz.

3.2.2.1 Accelerometer measurement error

Similarly to a gyroscope, accelerometers on smartphones are also MEMS devices; thus, majority categories of corrupting errors of accelerometers are the same as those mentioned in Section 3.2.1.1. However, since position tracking is achieved by taking integration of accelerometer readings twice over time, the positioning error, due to systematic bias and random noise, increases exponentially over time.

3.2.3 Magnetometer

A three-axis magnetometer is used to measure local magnetic field strength over three orthogonal directions. Similar to an accelerometer, the axes of a magnetometer are also aligned exactly with the smartphone reference frame shown in Figure

3.1. The output magnetic field strength signals for three directions are defined as m_x, m_y, m_z , respectively. The polarities of output signals of the magnetometer are defined in a similar manner to those of an accelerometer.

For the AndroidTM mobile operating system, the sampling frequency of the magnetometer could be defined by developers for different purposes of application. In the implementation of this thesis, the sampling frequency is set as 50Hz.

3.2.3.1 Magnetometer measurement error

Magnetometer measurements are prone to be corrupted by the surrounding magnetic interference in the vicinity of sensor. The type of interference could be categorised into two groups. The first group includes slowly time-varying magnetic interference caused by ferromagnetic structural materials in the proximity of the sensor. Field measurement errors due to such interference are referred to as hard iron bias [85].

The second group of interfering magnetic field originates from materials that are easily magnetised. In such cases, the materials generate their own magnetic field to resist the effects brought from the externally applied magnetic field. The generated field is influenced by both the magnitude and direction of the externally applied magnetic field. Such materials are termed soft iron, and the related errors are termed soft iron bias. The resulting interferences due to the surrounding soft iron materials are considered to be rapidly time-varying.

Overall, magnetometer measurement is prone to be corrupted in the indoor environment, where many objects, for example, running electronic computers, speakers, etc., could be regarded as potential interference sources.

3.3 Conventional inertial navigation tracking method

3.3.1 Distance updating

A smartphone reference frame is fixed rigidly to the phone. When the phone rotates, the reference frame also rotates in exactly the same way. Notice that the output signal from inertial sensors are also defined in the smartphone reference frame. To apply this information for tracking the user's position, the values need to be converted to the ones defined in a fixed reference frame. For this purpose, a fixed global reference frame is introduced and shown in Figure 3.2. To distinguish a global reference frame clearly from the smartphone reference frame, x' , y' , and z' are used to denote the three axes of the global reference frame, with directions of earth east, earth north and sky, respectively.

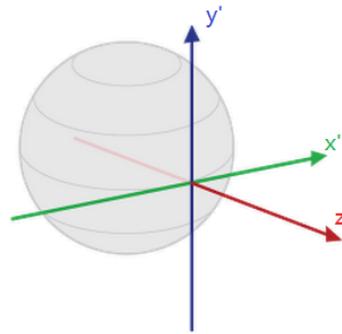


FIGURE 3.2: Global Reference Frame

The relationship between the position coordinates and accelerations in a defined global reference frame is well known. For example, for the x' direction,

$$\mathbf{v}_{x'}(t) = \mathbf{v}_{x'_0} + \int_0^t \mathbf{a}_{x'-l}(t) dt \quad (3.1)$$

and

$$\mathbf{p}_{x'}(t) = \mathbf{p}_{x'_0} + \int_0^t \mathbf{v}_{x'}(t) dt \quad (3.2)$$

In the equations above, $\mathbf{p}_{x'_0}$ and $\mathbf{v}_{x'_0}$ represent the initial x' coordination and initial x' -axis velocity of the user, respectively; $\mathbf{v}_{x'}(t)$ and $\mathbf{a}_{x'-l}(t)$ denote the instantaneous velocity and linear acceleration along the x' axis at time t . Similar rules and notations are also applied for the y' axis.

To obtain $\mathbf{a}_{x'-l}(t)$ and $\mathbf{a}_{y'-l}(t)$, the vector \mathbf{a}_{phone} including the 3-axis acceleration $\mathbf{a}_x, \mathbf{a}_y, \mathbf{a}_z$ obtained directly from inertial sensors, is first required to be transformed as follows:

$$\mathbf{a}_{gobal}(t) = H(t)\mathbf{a}_{phone}(t) \quad (3.3)$$

Here, $\mathbf{a}_{gobal}(t)$ expresses the vector containing $\mathbf{a}_{x'}(t), \mathbf{a}_{y'}(t), \mathbf{a}_{z'}(t)$, all of which are in global reference frame; $H(t)$ is well known as a direction cosine matrix [77, 86], and could be derived from phone's attitude information. Details of the derivation of $H(t)$ and the estimation of a phone's attitude are discussed in Section 3.3.2.

Since the values of the readings of an accelerometer are influenced by the force of gravity, a gravity vector $\mathbf{g} = [0, 0, g]^T$ is required to be subtracted from $\mathbf{a}_{gobal}(t)$ to get the vector $\mathbf{a}_{linear}(t)$ containing linear accelerations $\mathbf{a}_{x'-l}(t), \mathbf{a}_{y'-l}(t), \mathbf{a}_{z'-l}(t)$ influenced by external forces other than gravity in the global reference frame as follows:

$$\mathbf{a}_{linear}(t) = \mathbf{a}_{gobal}(t) - \mathbf{g} \quad (3.4)$$

Overall, the diagram of stardown DR method is shown in Figure 3.3.

3.3.2 Attitude estimation

The attitude of any object could be specified in different ways, such as Euler angle, direction cosine matrix, etc. For estimation of the instantaneous attitude using only accelerometer and magnetometer readings, an intuitive choice is the Euler angle. However, if the system fuses information from a gyroscope also, the direction cosine matrix is a more convenient choice.

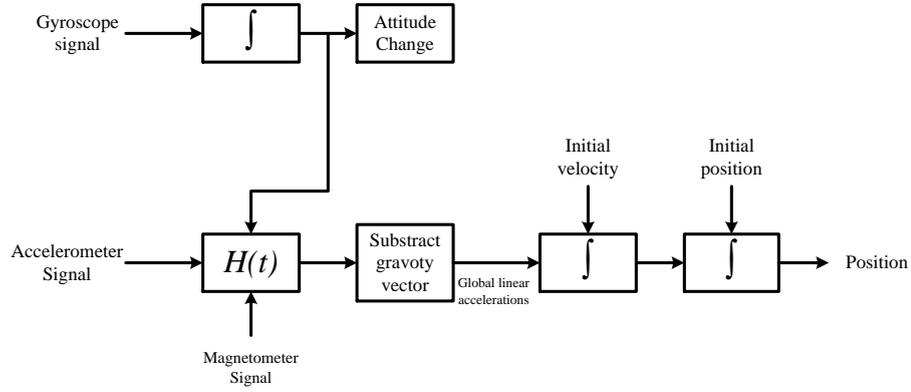


FIGURE 3.3: Strapdown DR method

3.3.2.1 Estimation of instantaneous attitude and tracking of the change via inertial sensors

To transform the acceleration vector from a smartphone reference frame to the global reference frame, it is important to know how to rotate the axes of the smartphone reference frame to let them align with the axes of the global reference frame. The Euler angle is applied in the implementation of this thesis for this purpose. In particular, the rotation angle around the z axis is defined as yaw (α); the rotation angle around the x axis is defined as pitch (β); the rotation angle around the y axis is defined as roll (γ). Also, valid ranges of three Euler angles have been defined: β varies from -90° to $+90^\circ$; α and γ vary in the range -180° to $+180^\circ$. In this way, every possible attitude of a smartphone could be described uniquely by a sequence of rotations around three axes, starting from the initial attitude: the origin of the global reference frame.

Derivations of pitch and roll angle of the phone from an accelerometer's readings is shown as follows [45]:

$$\beta(t) = \arctan\left(\frac{\mathbf{a}_y(t)}{\sqrt{\mathbf{a}_x(t)^2 + \mathbf{a}_z(t)^2}}\right) \quad (3.5)$$

and

$$\gamma(t) = \arctan\left(-\frac{\mathbf{a}_x(t)}{\mathbf{a}_z(t)}\right) \quad (3.6)$$

To derive yaw, readings $m_x(t)$, $m_y(t)$, $m_z(t)$ from a magnetometer are also required:

$$\alpha(t) = \arctan\left(\frac{m_x(t) \cos(\gamma(t)) + m_z(t) \sin(\gamma(t))}{m_x(t) \sin(\beta(t)) \sin(\gamma(t)) + m_y(t) \cos(\beta(t)) - m_z(t) \sin(\beta(t)) \cos(\gamma(t))}\right) \quad (3.7)$$

With the knowledge of the Euler angles, the basic rotation along each axis could be expressed mathematically as below [77]:

$$R_z(\alpha(t)) = \begin{bmatrix} \cos(\alpha(t)) & \sin(\alpha(t)) & 0 \\ -\sin(\alpha(t)) & \cos(\alpha(t)) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.8)$$

$$R_x(\beta(t)) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta(t)) & \sin(\beta(t)) \\ 0 & -\sin(\beta(t)) & \cos(\beta(t)) \end{bmatrix} \quad (3.9)$$

$$R_y(\gamma(t)) = \begin{bmatrix} \cos(\gamma(t)) & 0 & -\sin(\gamma(t)) \\ 0 & 1 & 0 \\ \sin(\gamma(t)) & 0 & \cos(\gamma(t)) \end{bmatrix} \quad (3.10)$$

Assuming that the acceleration vectors are transformed from the global reference frame to the smartphone reference frame by applying the following rotations in sequence:

- phone is rotated around the z axis for yaw $\alpha(t)$ rotation.
- phone is rotated around the x axis for pitch $\beta(t)$ rotation.
- phone is finally rotated around the y axis for roll $\gamma(t)$ rotation.

It is termed z - x - y 3-D rotation. Mathematically, it could be expressed as:

$$\mathbf{a}_{phone}(t) = R_y(\gamma(t))R_x(\beta(t))R_z(\alpha(t))\mathbf{a}_{global}(t) \quad (3.11)$$

In practice, however, the known information is the values of elements in $\mathbf{a}_{phone(t)}$, and the target is transforming them to the values in the global reference frame. To achieve that, $\mathbf{a}_{phone(t)}$ is required to be rotated, according to the negative values of the three known Euler angles in inverse sequence, as:

$$\mathbf{a}_{global}(t) = R_z(-\alpha(t))R_x(-\beta(t))R_y(-\gamma(t))\mathbf{a}_{phone}(t) \quad (3.12)$$

It should be noted that the Euler angles obtained from equations of (3.5) - (3.7) only apply to the implemented z - x - y rotations. For example, assuming a particular attitude change of the phone is obtained by rotating it for $\alpha = 180^\circ$, $\beta = 45^\circ$ and $\gamma = 180^\circ$ in sequence. Exactly the same change of attitude, also, could be achieved by rotating the phone in other sequences with different euler angles, e.g., $\alpha = 180^\circ$, $\gamma = 180^\circ$ and $\beta = -45^\circ$ in sequence. Serious attitude estimation error will result if the obtained euler angles are applied to wrongly ordered rotations.

Comparing equation of (3.12) and (3.3), it is easy to know that:

$$\begin{aligned} H &= R_z(-\alpha)R_x(-\beta)R_y(-\gamma) \\ &= \begin{bmatrix} \cos \alpha \cos \gamma - \sin \alpha \sin \beta \sin \gamma & -\sin \alpha \cos \beta & \cos \alpha \sin \gamma + \sin \alpha \sin \beta \cos \gamma \\ \sin \alpha \cos \gamma + \cos \alpha \sin \beta \sin \gamma & \cos \alpha \cos \beta & \sin \alpha \sin \gamma - \cos \alpha \sin \beta \cos \gamma \\ -\cos \beta \sin \gamma & \sin \beta & \cos \beta \cos \gamma \end{bmatrix} \end{aligned} \quad (3.13)$$

In the equation above, to save space and keep the presentation concise, t is neglected in the symbols. For example, $H(t)$ is abbreviated as H .

Although the attitude of a smartphone could be estimated solely via magnetometer and accelerometer information, as shown in equation of (3.5), (3.6) and (3.7), the accuracy of estimation, in most occasions, would be deviated due to the constrained performance of MEMS devices themselves and environmental factors such as the surrounding magnetic interference. In order to improve the accuracy of an attitude estimation, a gyroscope is also applied as another information source, as

mentioned before. In particular, a gyroscope is used for tracking the change of attitude of phones, as presented in [86, 87]:

$$H(t + dt) = H(t) \cdot \begin{bmatrix} 1 & -d\alpha & d\gamma \\ d\alpha & 1 & -d\beta \\ -d\gamma & d\beta & 1 \end{bmatrix} \quad (3.14)$$

where $d\alpha = \Omega_z(t)dt$, $d\beta = \Omega_x(t)dt$ and $d\gamma = \Omega_y(t)dt$. The equation above shows a piece of rotation increment at one time moment. The sum of all such rotation increments over time yields the absolute attitude of the phone, given the initial direction cosine matrix $H(0)$.

An advantage of a gyroscope is found to be its high sensitivity to the change of attitude of a smartphone. Also, unlike a magnetometer, the output signals from a gyroscope are not prone to be influenced by the surrounding environment. However, during the process of attitude updating by taking integration of gyroscope readings, measurement errors introduced in every iteration will be accumulated over time, and therefore result in gyroscope drift. To overcome the mentioned disadvantages of magnetometer/accelerometer and gyroscope, and combine the information from respective sources, an easy-to-deploy complementary filter is applied in the implementation of this thesis, and will be discussed in the next section.

Also, gyroscope drifts will gradually reduce the mutual orthogonality conditions held by rows or columns of a direction cosine matrix. Therefore, the updated direction cosine matrix is required to be calibrated. In the implementation of this thesis, the renormalisation technique, shown in [86], is applied.

3.3.2.2 Implementation of a complementary filter for attitude estimation

Due to the noisy disturbances enforced on an accelerometer and a magnetometer, high-frequency vibration could be found in their attitude estimation results. Also, as mentioned in the previous section, attitude estimation resulting from a gyroscope is prone to drift over long-term time, but accurate over short-term time.

To overcome the disadvantages of measurements from respective sensors, a complementary filter is applied in the implementation of this thesis, mathematically shown as follows:

$$H_{fused}(t) = \varsigma \cdot H_{gyro}(t) + (1 - \varsigma) \cdot H_{acc+mag}(t) \quad (3.15)$$

where

$$H_{gyro}(t) = \begin{cases} H_{fused}(t - dt) \begin{bmatrix} 1 & -d\alpha & d\gamma \\ d\alpha & 1 & -d\beta \\ -d\gamma & d\beta & 1 \end{bmatrix}, t > dt \\ H_{acc+mag}(0) \begin{bmatrix} 1 & -d\alpha & d\gamma \\ d\alpha & 1 & -d\beta \\ -d\gamma & d\beta & 1 \end{bmatrix}, t = dt \end{cases} \quad (3.16)$$

and

$$\varsigma = \frac{\tau}{\tau + dt} \quad (3.17)$$

In the equations above, ς and τ are termed filter efficient and time constant, respectively. Specifically, τ is the relative duration of signal it will acts on [76]. For a low/high-pass filter, τ describes from which point an informative signal stops/starts to be filtered [76]. For the case of a complementary filter of attitude estimation, it could be regarded as a time boundary to determine if passing estimation results from an accelerometer/magnetometer or gyroscope through [76]. In detail, in the implementation of thesis thesis, ς is set as 0.98. With dt being set as 0.02 second, τ is calculated to be 0.98 second.

$H_{acc+mag}(0)$ denotes the direction cosine matrix built only from initial accelerometer and magnetometer output signals at initial point; $H_{gyro}(t)$ denotes the direction cosine matrix updated by gyroscope measurements at time t .

3.3.3 Evaluation results of the DR method

In this section, firstly, the attitude estimation results obtained from IMU sensors are evaluated. In particular, the yaw estimation results are presented in Figure 3.4. Following that, the travelling distance updating approach, used in a conventional DR algorithm, is evaluated.

3.3.3.1 Evaluation results of attitude estimation

Obtaining Euler angles from the direction cosine matrix H is shown as follow:

$$\beta = \arcsin H^{(3,2)} \quad (3.18)$$

$$\gamma = \arctan\left(-\frac{H^{(3,1)}}{H^{(3,3)}}\right) \quad (3.19)$$

and

$$\alpha = \arctan\left(-\frac{H^{(1,2)}}{H^{(2,2)}}\right) \quad (3.20)$$

Here, $H^{(i,j)}$ denotes the entry on the i -th row and j -th column of H .

In the test, a technician who is equipped with a smartphone (in his hand) pointing perpendicular to the plane of his body walks along a indoor corridor. Figure 3.4 compares the yaw estimation results obtained from applying only the accelerometer and the magnetometer (marked as a red solid line) with the ones obtained from the complementary filter implementation (marked as a blue solid line). In the figure, extensive high-frequency vibrations in the estimation result when applying only the accelerometer and magnetometer are indicated. On the other hand, as could be clearly found in the yaw estimation result when the complementary filter is implemented, both the high-frequency and long-term variations are filtered significantly. Therefore, in the following implementation, Euler angle estimation resulting from the complementary filter are employed.

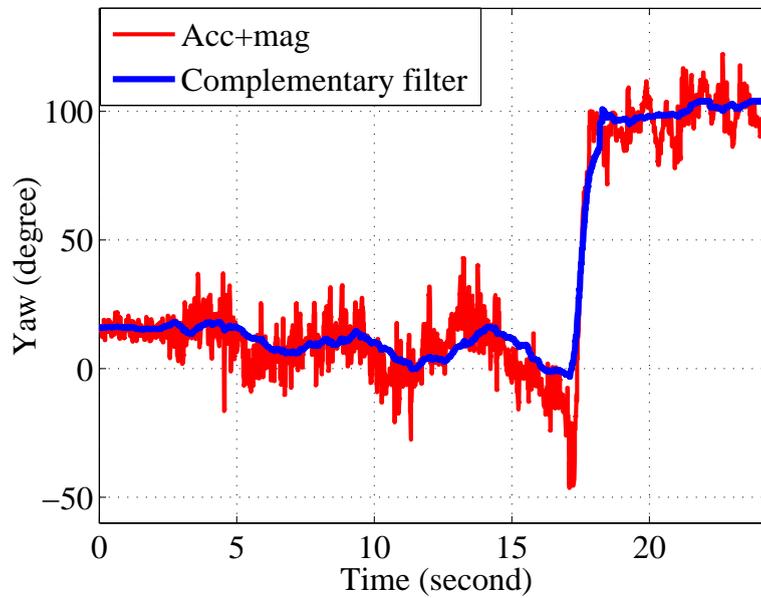


FIGURE 3.4: The estimation of yaw attitude of the smartphone held by a technician

3.3.3.2 Evaluation results of travelling distance updated by taking integration of acceleration

In this test, a tester walked straight along x' direction with a smartphone in his hand pointing to the same direction. Besides, in the test, he tried hard to keep a constant stepping frequency and travelling velocity. Meanwhile, the smartphone recorded accelerations along the x , y and z axes in its own reference frame. Then, they were converted to linear acceleration values along the x' , y' and z' axis in the global reference frame using the equation (3.3) and (3.4). The inferred smartphone acceleration data along the x' axis and the corresponding walking velocity inferred are shown in Figure 3.5. As could be found in Figure 3.5(b), the inferred velocity varies fiercely, and drifts upwards throughout the whole walk. Evidently, the inferred velocity could not be used for updating the pedestrian's travelling distance.

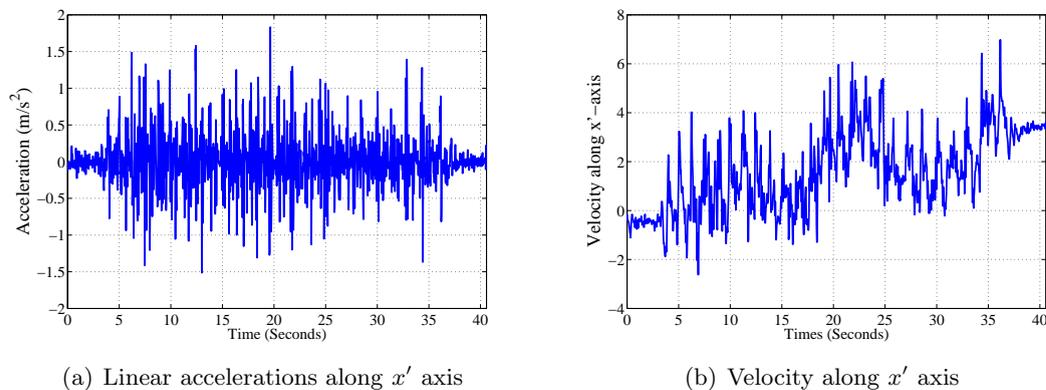


FIGURE 3.5: Showcase of the inferred acceleration and velocity in the global reference frame.

3.4 Step-based PDR and trajectory recognition

It is well known that the acceleration magnitude exhibits peak-like repetitive patterns when the user takes walks, as presented in [33, 51, 88, 89]. Thus, rather than inferring the user's moving trajectory by directly applying the IMU sensors outputs with the DR method, the peak-like repetitive patterns are taken advantage to recognise every new step that users take.

Besides the most common in-hand pose of phones, some papers such as [33, 51] also track user's trajectory when phones are in a trouser pocket. The first benefit could be providing the localisation service when the phones are in the users' pocket. Another benefit could be geo-tagging and crowd-sourcing users' collected in-pocket RSS fingerprints to calibrate or update the data stored in the RM. However, the necessity of such implementation remains doubtful. Firstly, users' localisation intentions levels, when the phones are in pockets, are relatively low. However, activating and utilising smartphones' inertial sensors to track the user's position, when keeping phones in pockets for long term, drains battery life very fast. Also, the RSS values are affected by the phone poses, since both the heights and orientations of the phones' antennas are varied dramatically across the two poses: in-hand and in-pocket. Therefore, taking advantage of the RM consisting of training RSS data collected by the technicians' phones in an in-hand pose will highly likely lead to inaccurate Wi-Fi on-line localisation results for users with phones in pockets. On the other hand, taking advantage of geo-tagged in-pocket

RSS fingerprints to calibrate the RM consisting of in-hand RSS fingerprints brings potential degeneration rather than enhancement to the accuracy of the localisation service. As a result, in the implementation of this thesis, only the cases of in-hand smartphone pose are considered for localisation service.

The task could be divided into several subtasks. The first goal is to discriminate between user's walking and non-walking state. Following that, step numbers and corresponding moments during the period of walking are counted and detected. Also, step length and yaw angle of every detected step is estimated. Finally, moving trajectory of the user could thus be, recognised.

3.4.1 Walking detection

As presented in [88], the short-term Fourier transform (STFT) technology as proposed in [90] is applied to transform the magnitude of the raw three-axis accelerometer output signal from the time domain to the frequency domain in order to detect the moving or silent status of users.

Firstly, the magnitude of the recorded data is separated into consecutive non-overlapped T_w one-second segments. Then the signal in every segment is converted to the frequency domain with the STFT.

A manually-set threshold is employed to compare with the absolute values of obtained spectral coefficients in a humans' stepping frequency range (1 to 3 Hz). Provided that the threshold is exceeded by any coefficient in the given frequency range, the corresponding acceleration signal segment is regarded as being collected in moving status. Otherwise, it is recognised to be at rest status.

Although the applied STFT-threshold method could be used to discriminate between movement and a stationary pose of the user, it might not be suitable for detection of walking vs. non-walking since high spectrum magnitudes of coefficients, in the range of 1 to 3 Hz, could also be registered by the aperiodic movements of smartphones, such as picking up the phone for calling, putting the phone in the pocket, etc. However, aperiodic movements are always found to be short-lived,

and can be filtered out by comparing the duration of every detected movement with a certain time threshold.

An example is given in Figure 3.6. The record is accomplished by a user transferring his movement status in sequence, as shown below:

1. Being stationary.
2. Walking steadily with the smartphone in hand.
3. Stopping and being at rest for while.
4. Putting the phone in the trouser pocket.
5. Walking steadily with the smartphone in a pocket.
6. Stopping and standing statically for a while again.
7. Picking up the smartphone from the pocket.

Note that the correspondence between each action included in the list and the variation pattern of the acceleration magnitude is highlighted with a number in the range of 1 to 7 in Figure 3.6. Also note that the in-pocket pose is included in the record only for the purpose of simple illustration: the peak-like periodic patterns could be found in the magnitude of the acceleration data collected from the in-pocket walking status as well. Here, the 4-th and 7-th actions in the list are regarded as two disturbing movements.

To discriminate between movement and stationary status, the value of the threshold employed on the STFT result of the acceleration magnitude is set as 20. The movement hinter is obtained and plotted as a red dashed line in Figure 3.6. In the figure, the status of the user is recognised as moving when values of the movement hinter are positive. Otherwise, the user is recognised to be at rest.

As demonstrated in the figure, both of the two disturbing actions (the 4-th and 7-th action) have short durations (less than two seconds). Thus, the movement duration threshold for discriminating walking from other movements could be set to two second. However, false negative errors are hardly avoidable when users undergo short-term walking. The resulting displacements of users cannot be found by the

method, if each of the walks lasts less than two seconds. However, as a common knowledge, the nature of the low frequency of human beings' normal stepping, in the majority of cases, constraints the moving range of the short-term walks in two seconds, and thus would only distort the step counting and location estimation results slightly. Also, as a common knowledge, the possibilities of human beings taking walks less than two seconds, in real life, is rare.

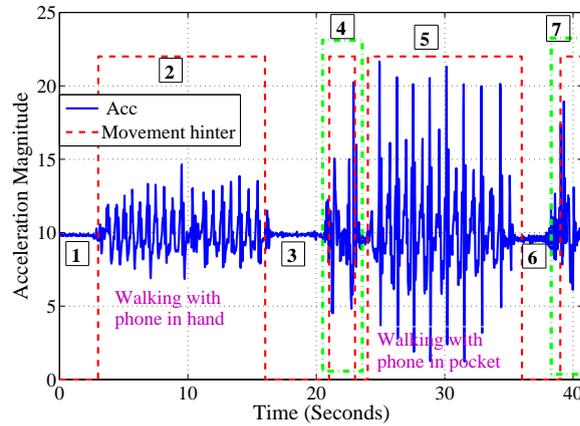


FIGURE 3.6: Raw acceleration magnitude and movement hinter

3.4.2 Estimation of angle difference between user's walking direction and phone's orientation

In reality, a user's walking direction and the orientation of the phone in his hand may not be aligned with each other exactly, as indicated in Figure 3.11. Therefore, in order to estimate the user's direction, besides the orientation of the smartphone, the angle offset between the user and his smartphone is required to be estimated as well. This section assumes that users hold the sides of their phones, as illustrated in Figure 3.7. Therefore, the angle difference is always smaller than 45° , as indicated in Figure 3.11. For the cases when users hold the top and bottom boundaries of their phones, for equation (3.22), x is replaced with y , and vice versa.

The waveforms of the linear accelerations recorded along the x and y axis of the smartphone's local reference frame, when a user is walking with a smartphone in hand, are indicated in Figure 3.9. In this case, the y axis is aligned with the moving direction of the user. From the figure, it may be noted that the acceleration



FIGURE 3.7: Both sides of the smartphone are held by the palm of a user.

along either the x or y axis exhibits a periodic pattern is found. In particular, the red arrows included in both of the sub-figures point to the bottom positions of the periodic patterns. In addition, each minimum value of the linear acceleration along the y axis highlighted by a red arrow included in Figure 3.11(b) corresponds to a stepping moment when a foot is about to be grounded totally, as indicated in Figure 3.8. Note that, by counting the number of red arrows, it could be figured out how many times the patterns have been repeated in the elapsed time (4 seconds) during which 6 steps have been taken by the user. The answers are obvious: 3 and 6 for the x and y axis, respectively, which implies that the occurrence frequency of the pattern for the y axis doubles the occurrence frequency of the pattern for the x axis. The underlying reason is stated as follows.

The change of the polarity of the linear acceleration along the moving direction in a one-step period is shown in Figure 3.8. When a human being lifts a foot and steps forwards, the forward acceleration is found to be positive. As a common knowledge, positive forward accelerations during such periods push this human to move forward. However, the forward acceleration is found to be negative when the

foot starts to land on the ground, which provides friction on the foot to decrease the velocity of it until the status of it has been transformed to stationary. Such acceleration change forms a unique pattern repeated for every step taken. On the other hand, when walking, a human being's pelvis will tilt sideways towards the stepping leg, while bending the knee of the supporting leg. This action will result in a sideways acceleration towards the supporting leg (the related video proof could be found in [91]). Because either of legs serves as the supporting leg in turn, the occurrence frequency of the periodic pattern of the linear acceleration along the sideways direction is half of the walking frequency.

Also, in Figure 3.10, the magnitude spectrum of the linear accelerations along the x and y axis shown in Figure 3.9 (the DC component is excluded) in the phone's location reference frame is illustrated. As shown in the figure, the frequency of the dominant component of the linear acceleration along the moving direction (y axis in this case), 1.5 Hz, doubles the frequency of the most significant component of the linear acceleration along the sideways direction (x axis in this case), 0.75 Hz. In addition, the magnitude of the component of the linear acceleration along the x axis at 1.5 Hz is low, compared to the other bins. This affords an opportunity to estimated the angle difference between the user's moving direction and the orientation of the device, $|\Upsilon|$, as:

$$|\Upsilon| = \arctan \frac{MP_x}{MP_y} \quad (3.21)$$

where $MP\{\bullet\}$ denotes the ratio of the spectral magnitude of the linear acceleration along $\{\bullet\}$ at the frequency on which the most significant frequency component of the linear acceleration along the y axis is observed to the sum of all spectral components in the range from $\frac{f_s}{N_{pt}}$ to $\frac{f_s}{2} - \frac{f_s}{N_{pt}}$. Here, N_{pt} denotes the the number of samples of acceleration along either axis in a limited duration, e.g, 1 second. Next, the polarity of the angle difference should be derived further. The proposed estimation method of the polarity of Υ is shown as:

$$PL_{\Upsilon}(T_w) = -\text{sgn}(\mathbf{a}_x(t_{bot})), \quad t_{bot} = \arg \min_{t \in T_w} \mathbf{a}_y(t) \quad (3.22)$$

Note that the linear acceleration along the x and y direction can be factored into two parts: the projection of the acceleration along the moving direction \mathbf{a}_{moving} and the projection of the acceleration along the sideways direction \mathbf{a}_{side} on the x and y axis, as expressed in vector form :

$$\frac{\mathbf{a}_y}{\cos(|\beta|)} = \mathbf{a}_{moving} \bullet \cos(|\Upsilon|) + \mathbf{a}_{side} \bullet \sin(|\Upsilon|) \quad (3.23)$$

and

$$\frac{\mathbf{a}_x}{\cos(|\beta|)} = \mathbf{a}_{moving} \bullet \sin(|\Upsilon|) + \mathbf{a}_{side} \bullet \cos(|\Upsilon|) \quad (3.24)$$

In the following analysis, $\cos(|\beta|)$ is ignored since it does not have influence on the estimation result.

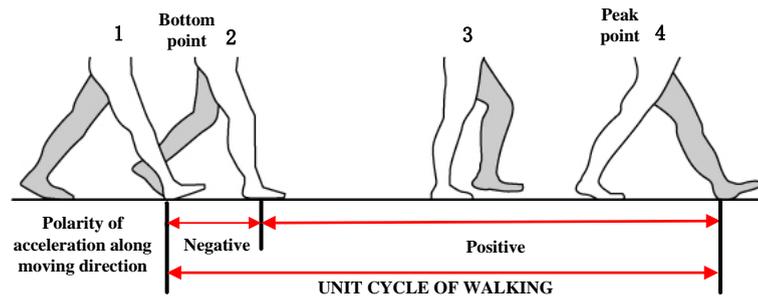


FIGURE 3.8: The variation trend of linear acceleration over the moving direction. Four gaits are shown particularly. The bottom and peak of the forward acceleration emerges in the duration of transforming the gait from the first one to the second one, and from the third one to the fourth one, respectively.

Primarily, note that, if holding the phone as shown in Figure 3.11(a), the polarity of the projection of \mathbf{a}_{moving} on the x axis is opposite to the direction of the x axis when the foot hits on the ground. However, if holding the phone as shown in Figure 3.11(b), the polarities of them will coincide with each other at the moment. These knowledges form the basis of estimation of PL_{Υ} . On the other hand, the direction of \mathbf{a}_{side} could be opposite to the one indicated in the figure, depending on which foot, right one or left one, hitting on the ground.

Since $|\Upsilon|$ is assumed to be smaller than 45° , $\cos|\Upsilon| > \sin|\Upsilon|$. When $\min \mathbf{a}_y(t)$ is detected, \mathbf{a}_{moving} should achieve the negative maximum value during T_w also,

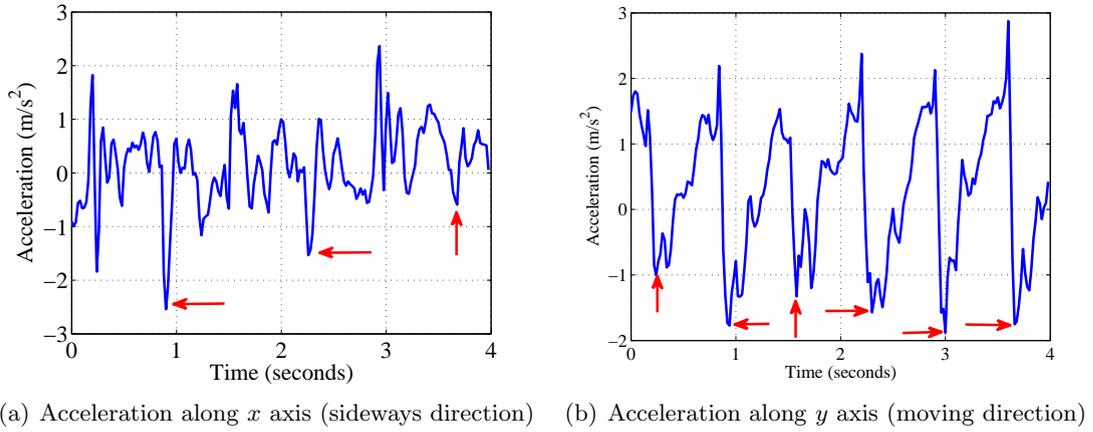


FIGURE 3.9: Showcase of the accelerations recorded along the x and y axis in the phone's location reference frame, when the user is walking. In this case, the moving direction aligns with the y axis

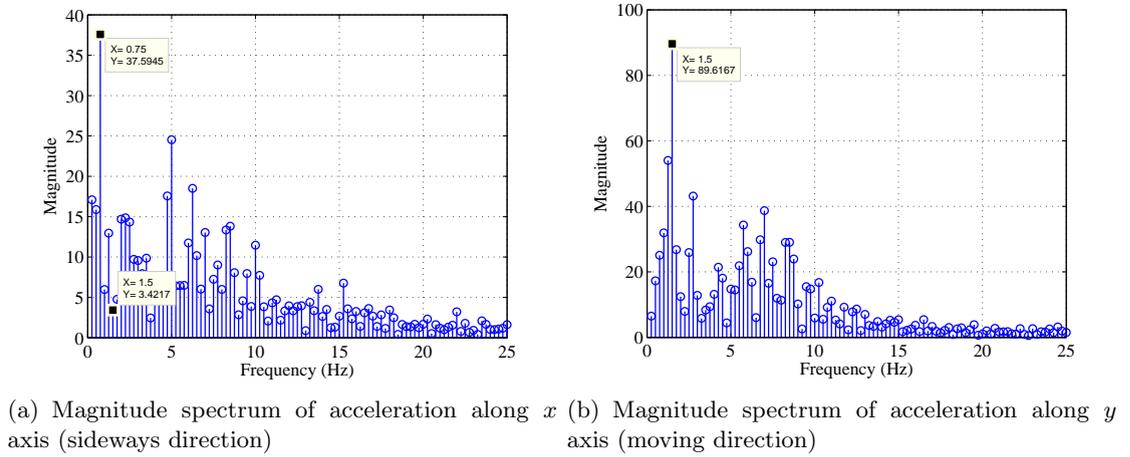


FIGURE 3.10: Showcase of the magnitude spectrum of the accelerations along the x and y axis in the phone's location reference frame, when the user is walking.

which is highly likely caused by hitting the stepping foot on the ground. As a result, at t_{bot} , if $|\Upsilon|$ is closer to 45° than 0° , the projection of \mathbf{a}_{moving} on the x axis will deny the projection of \mathbf{a}_{side} on it with the highest probability, if the direction of the projection of \mathbf{a}_{side} is opposite to that of \mathbf{a}_{moving} on the x axis. Under this situation, applying the knowledges mentioned before, PL_Υ will be estimated accurately with a high possibility.

However, if $|\Upsilon|$ is closer to 0° than 45° , on the other hand, the projection of \mathbf{a}_{moving} might be denied by the projection of \mathbf{a}_{side} on the x axis, if the direction of the projection of \mathbf{a}_{side} is opposite to that of \mathbf{a}_{moving} on the x axis. Under such a

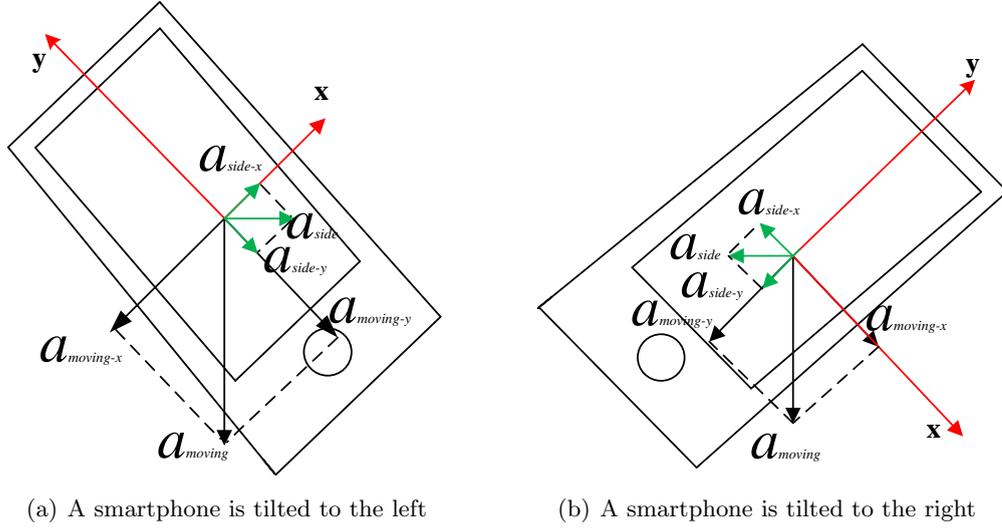


FIGURE 3.11: Vertical view of the accelerations \mathbf{a}_{moving} and \mathbf{a}_{side} on the x and y axis in the phone's local reference frame when a user's foot hits on the ground. In the two cases, a phone is held by the user with opposite angles to the moving direction. Note that the direction of \mathbf{a}_{side} could be opposite to the one indicated in the figure.

situation, a low accuracy of estimation of PL_{Υ} by observing the polarity of \mathbf{a}_x is expected. However, the effect of such wrong estimation of Υ on the pedestrian trajectory result is regarded as light because $|\Upsilon|$ is closer to 0.

After the polarity of Υ , PL_{Υ} , is estimated, the user's orientation could be estimated as:

$$\theta = \alpha - |\Upsilon| \bullet PL_{\Upsilon} \quad (3.25)$$

To evaluate the performance of the method, 10 pieces of continuous walks with the orientation of the phone biased to the left or right of the user's body, exactly as the same way depicted in Figure 3.11(a) and 3.11(b), exactly, have been taken.

The accuracy of the estimation of polarity of Υ is used to evaluate the performance of the method. The corresponding results are indicated in table 3.1. Overall, for the cases of $|\Upsilon| = 30^\circ$, the estimation accuracy is 85.7%; for the cases of $|\Upsilon| = 15^\circ$, the estimation accuracy is 70.8%.

Based on that, in addition, the accuracy of the estimation of Υ is also evaluated for the cases of $|\Upsilon| = 15^\circ$. In particular, for the cases of $\Upsilon = +15^\circ$, the probability histogram of the difference between the estimate and the true value of Υ , and

the fit bimodal Gaussian distribution function is indicated in Figure 3.12. In practice, since users always hold the phone with a small angle offset between the phone and themselves, e.g. 15° , to the orientation of their bodies, the fit probability density function shown in the figure is applied in the following particle filter implementation.

| Walk's index | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|--------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|-----------------|-----------------|-----------------|
| Accuracy | $\frac{10}{12}$ | $\frac{12}{15}$ | $\frac{14}{16}$ | $\frac{12}{13}$ | $\frac{12}{14}$ | $\frac{10}{14}$ | $\frac{9}{13}$ | $\frac{11}{16}$ | $\frac{11}{15}$ | $\frac{10}{14}$ |

TABLE 3.1: The accuracy of the estimation of the polarity of Υ , when the smartphone is held with an angle bias to the the tester's orientation. During the first five pieces of walks, $|\Upsilon|$ is approximately 30° ; $|\Upsilon|$ is about 15° for the last five pieces of walks. Numerators count the number of one-second periods during which the polarity has been estimated correctly in each walk; denominators count the total number of one-second periods in each walk.

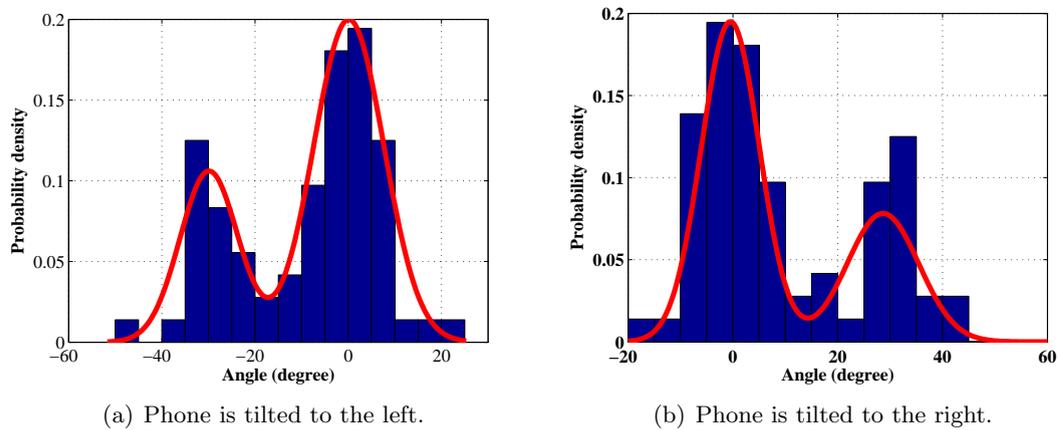


FIGURE 3.12: Illustration of the histogram of the difference between the estimate and the true value of (a): $\Upsilon = +15^\circ$ and (b): $\Upsilon = -15^\circ$; the red solid line highlights the fitted bimodal Gaussian distribution function.

3.4.3 Detection of steps and stepping moments

Since a repetitive pattern of the magnitude of a 3-axis accelerometer reading results from the naturally inherited rhythm of walking, the pattern is applied to count the number of the user's steps taken. In particular, the local peaks of the magnitude of the accelerations is found to be corresponding to the stepping moments, which is used for detection of both steps and stepping moments. Also, the fact that the walking frequencies of normal human beings are less than 3 Hz is utilised to prevent any peak-like pattern of the accelerometer readings occurring in a short

time ($\Delta t = 0.33s$) after the detection of the last step from being recognised as a new step.

The step detection algorithm proposed in [32] is applied in the implementation of this thesis. In the algorithm, peak detection is applied to find the local maximum by comparing the acceleration values with a pre-defined threshold, δ . Also, since the system may confuse the spark noise with step occurrences, as explained in Section 3.4.1, for every detected local maximum, the algorithm checks if the second condition could be achieved: the average of the acceleration magnitude values (except the local maximum itself) over a 0.14-second (7 samples) window around the local maximum is higher than a pre-defined threshold, ρ , when users take a step. In the case when two such requirements are met, a step is believed to be detected, and the corresponding occurrence moment of a local maximum is recognised as a timestamp of a step event. Specifically, δ and ρ are obtained from a bootstrap phase that will be shown below in detail. Then, the step detection algorithm could be used to count steps. Figure 3.13 demonstrates a zoomed-in version of the magnitude of 3-axis accelerometer readings of four consecutive steps when walking with smartphone in hand, and also the relationship between acceleration magnitude and δ , ρ .

While acceleration pattern may be different across different users, it remains similar for the steps taken by the same user. By analysing the trial walk, personalised parameters could be learned. Besides, for the purpose of another task (step length estimation as shown in next section), the user is requested to try to keep a relatively constant walking frequency in the walk as well. In the walk, only less than 20 consecutive steps are requested to be taken by the user. After a walk, the user needs to inform the smartphone manually of how many steps, in reality, have been taken. Also, before the start and after the end of the walk, the user is requested to inform the system of the start and end positions on the map shown on the screen of the smartphone.

When analysing, the algorithm changes relevant parameters in certain ranges in order to estimate the number of steps that the user has taken. By observing the difference between the estimated and the true number of steps, the algorithm

could, thus, choose the best set of parameter values as the option to be applied in the on-line localisation phase. Here, an example is presented to explain the implementation details of the step counting algorithm. To find the optimum values of δ and ρ , they are changed within the range of 12.3 to 13, with an increasing step index of 0.1. Figure 3.14 shows the difference between the real and estimated steps number with varying δ and ρ . The values of $\rho=12.5$ and $\delta=12.9$ are chosen as they are centred at the area where the step counting difference between estimation and ground truth is minimised to zero.

The performance of the step detection algorithm (with the chosen parameters) is evaluated with newly collected acceleration samples. The user repeats walking on a route for 10 times (in total 900 walking steps). The cumulative step-count error is 7 steps, which corresponds to 0.77% estimation error. Note that due to the previously mentioned constraints are applied, the noisy spark signal will be filtered always. Therefore, applying such algorithm result in false negative errors only: users' steps taken are not detected.

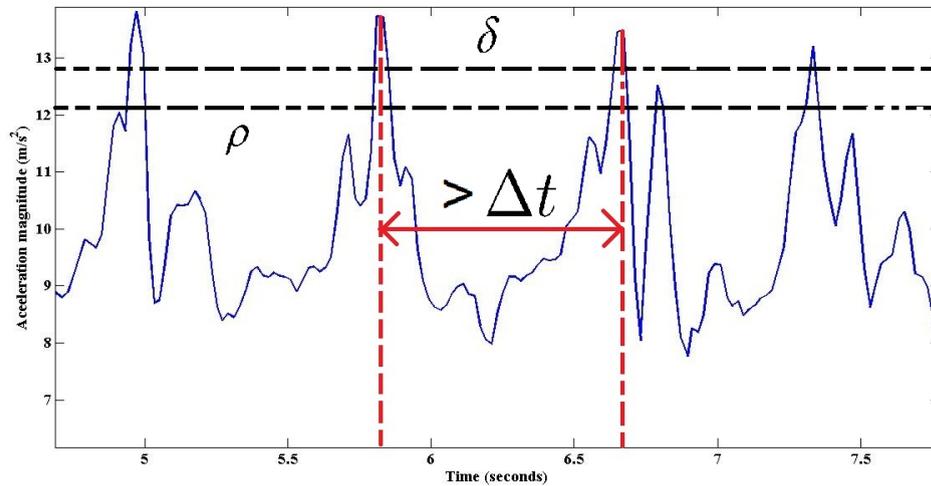


FIGURE 3.13: Acceleration magnitude for the four continuous steps with smartphone in hand.

| | | ρ | | | | | | |
|----------|------|--------|------|------|------|------|------|------|
| | | 12.3 | 12.4 | 12.5 | 12.6 | 12.7 | 12.8 | 12.9 |
| δ | 13.0 | 0 | 0 | 3 | 3 | 3 | 0 | 3 |
| | 12.9 | 0 | 0 | 0 | 0 | 0 | 0 | |
| | 12.8 | 0 | 0 | 0 | 0 | 0 | | |
| | 12.7 | 3 | 3 | 3 | 3 | | | |
| | 12.6 | 3 | 3 | 3 | | | | |
| | 12.5 | 3 | 3 | | | | | |
| | 12.4 | 8 | | | | | | |

FIGURE 3.14: The step-counting error for different values of ρ and δ with smartphone in hand.

3.4.4 Step length estimation

A linear relationship between step length and walking frequency, as shown in equation (3.26), is clearly found in [33, 75, 92]. As previously stated, the values of k_1 and k_2 are varied between different users. To learn k_1 and k_2 for every individual user much more efficiently than the method proposed in [33], a novel method is proposed. Note that unlike the method presented in [33], the proposed method has requirement of neither a general step length model sourcing from a large number of training data nor a time-inefficient convergence process for personalisation of k_1 and k_2 . Instead, the common knowledge that two distinct points determine a line is applied in the implementation of the thesis. In detail, besides the first trial walk taken for learning δ and σ for the step count estimation algorithm, another trial walk with a higher relatively constant walking frequency than the first is requested to be done by the user. For this walk, the user only needs to inform the system of the start and end positions. With the learned δ and ρ from the previous trial walk, every step taken in the current walk and the corresponding stepping moment could be recognised. Then, by analysing the two walks, a set of linear equations could be built, as:

$$\begin{cases} \bar{l}_1 = k_1 \frac{1}{\bar{t}_{step1}} + k_2 \\ \bar{l}_2 = k_1 \frac{1}{\bar{t}_{step2}} + k_2 \end{cases} \quad (3.26)$$

where \bar{l}_1 and \bar{l}_2 denote the averaged step length of the steps taken in two trial walks, calculated as dividing the Manhattan distance between the start and end positions by the detected number of steps; \bar{t}_{step1} and \bar{t}_{step2} denote the averaged time interval between successive steps and, as previously mentioned, $\bar{t}_{step1} \neq \bar{t}_{step2}$. In this way, k_1 and k_2 could be personalised for every individual user. The values of the parameters are stored in each user's smartphone for following tasks. Thus, step length could be inferred for every individual taken step, in the on-line localisation phase.

The trial consisting of two independent walks could be taken for every user only once and the parameters obtained could be used for subsequent localisation services for all the time. So it should not be regarded as a burden for users.

To verify the proposed method, an experiment is implemented. When a user walks and simultaneously records accelerations, videos of two trial walks are recorded by an observer. The position of every step taken is recorded in the video and the length of every step could be thus manually inferred as well. The experimental results show that the step length estimation errors are in the range of 4% to 9% of the values of the ground-truth lengths of the steps taken by the user, performance of which is considered to be better than the method proposed in [33]. Note that as mentioned in Section 2.5.2, 5 significant turns along the paths are required by the work done in [33] for personalisation of the parameters to have a accuracy which is lower than 10%. In Figure 3.15, the relationship between the inferred step length and the stepping frequency, $\frac{1}{\bar{t}_{step}}$, is presented. As could be observed from the figure, although the user tries to keep a constant stepping frequency in the walk, the stepping frequency is found to vary in a limited range during the walk in practice. However, assuming the variation follows the Gaussian distribution, applying mean value \bar{l} and $\frac{1}{\bar{t}_{step}}$ in equation (3.26) to derive k_1 and k_2 is a proper choice. As long as users try hard to keep relatively constant stepping frequency in

two trial walks, the proposed method works. Finally, k_1 and k_2 for this particular tester are found to be 0.35 and 0.20, respectively.

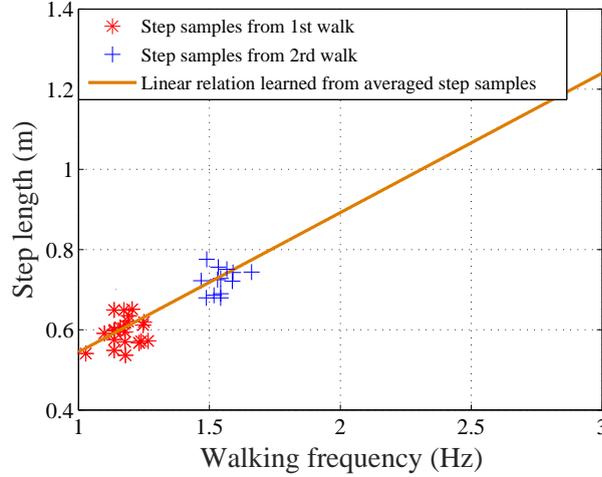


FIGURE 3.15: Linear relation between step length and walking frequency is learned

3.4.5 Recognition of the PDR moving trajectory

The overall scheme of the PDR trajectory recognition is indicated in Figure 3.16. As could be found in the figure, the trajectory is updated according to the IMU reports segmented into one-second pieces over the time domain. In the implementation of the thesis, each one-second piece of IMU report consists of 50 samples, since the sampling frequency is set as 50 Hz. Yaw angles of users' smartphone are obtained from the previously mentioned complementary filter while walking. Samples of yaw angles are stored in the buffer, and only processed at the proper time moments, when new steps are detected by the step detection algorithm. Moreover, Υ is taken into account deriving the orientation of the user's body, as $\theta = \alpha - \Upsilon$. Finally, θ is normalised into the range of $[-\pi, \pi]$ (coinciding with the range defined for α), according to the equation shown as follows:

$$\theta_n = \theta - 2\pi \left\lfloor \frac{\theta + \pi}{2\pi} \right\rfloor \quad (3.27)$$

Figure 3.17 demonstrates an example of deriving the moving position from the information of step length and the orientation of a user's body θ , when the user

experiences a turn. Here, the steps are assumed to be detected at two time moments: $t - \delta t$ and t . In addition, step length l for every step is estimated according to equation (2.21), k_1 and k_2 learned from the bootstrap phase. As a result, the user's displacement could be updated according to the following rules:

$$\begin{cases} \mathbf{p}_{x'}^t = \mathbf{p}_{x'}^{t-\delta t} + l \cos(\theta_n) \\ \mathbf{p}_{y'}^t = \mathbf{p}_{y'}^{t-\delta t} + l \sin(\theta_n) \end{cases} \quad (3.28)$$

Also, since the 2-D coordinates of the global reference frame is not aligned with the 2-D coordinate system applied in the floorplan, 2-D rotation matrix [93] is applied to transfer the coordinates of the user's moving positions in the global reference frame to the local reference frame of the floorplan. Note that the yaw orientation bias of the building to the north direction, measured as 10° , is also taken account into transferring the global coordinates system to the local reference frame.

An experimental trial is shown in Figure 3.18. The trial is done in the office area, where two corridors intersect with each other orthogonally. A tester with a smartphone in hand (Υ is close to 15°) walks along one of the corridors, experiences a turn at the corner shown on the figure, continues walking along the orthogonal corridor, and finally stops. The blue dashed line indicates the ground-truth moving trajectory of the user, and the arrow in the same colour indicates the final ground-truth stopping position of the user. Moreover, the red points represent the displacement updating results of the detected steps over the time domain obtained by applying the step detection algorithm, the estimation algorithm of θ and the personalised step length estimation algorithm. Note that the uncertainties in the estimation of both l and θ are not taken into account, since the trail walk is for demonstration purpose only. From the figure, it is easy to find that the displacement updating results, along the path, collide with the walls of the corridors for several occasions. After the experience of the turn, the tester's movement is tracked to be more deviated seriously from the corridor range. As a result, it is proven that implementation of the PDR moving trajectory tracking solely is not enough to afford indoor localisation services.

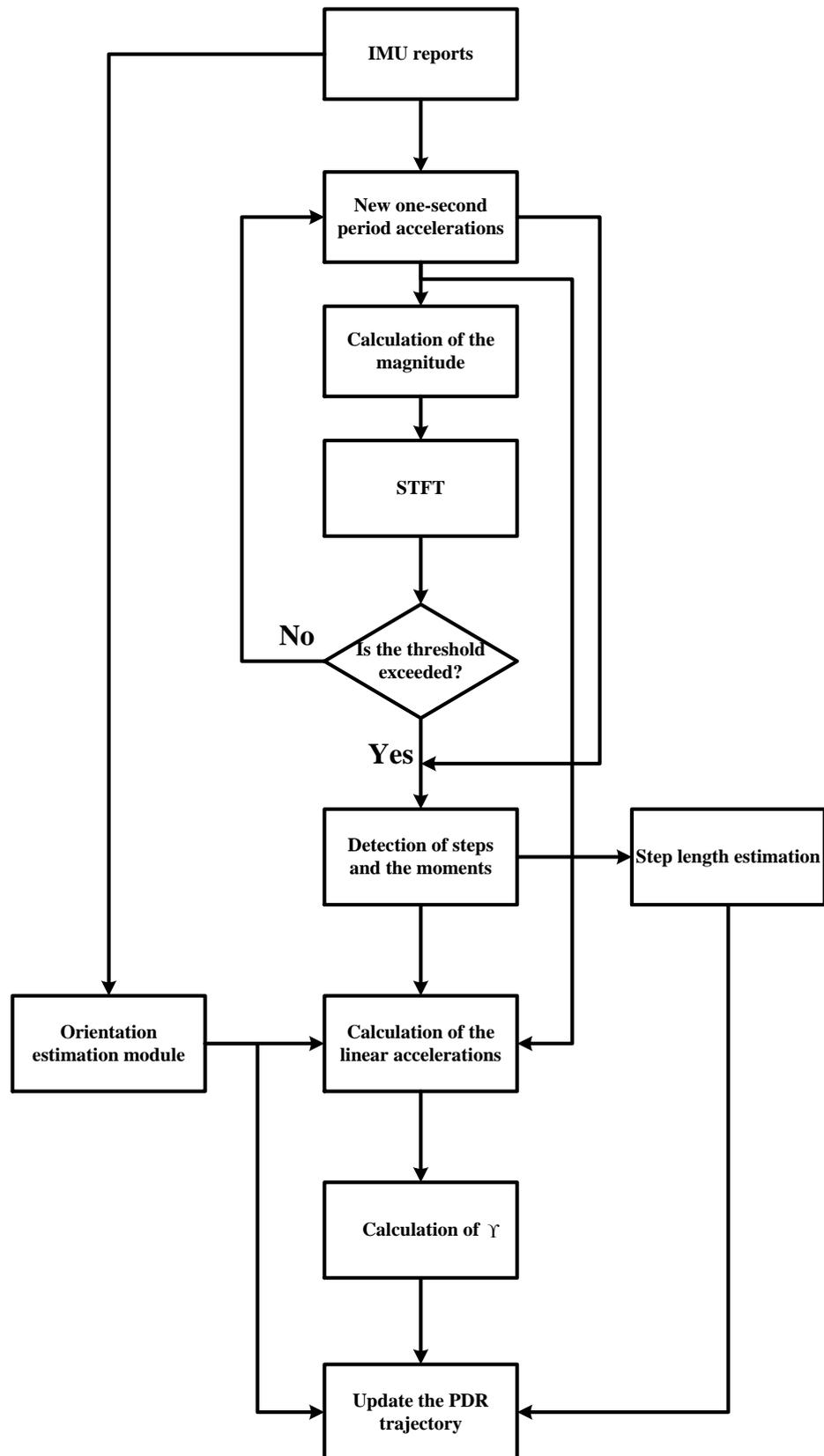


FIGURE 3.16: Overall trajectory recognition scheme.

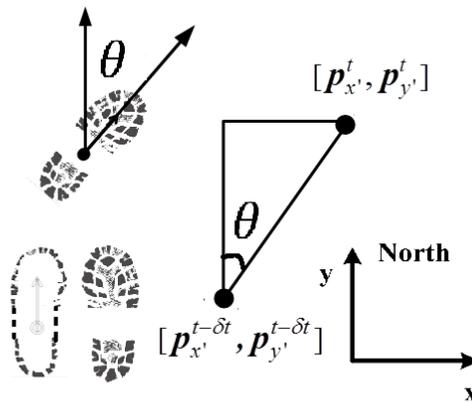


FIGURE 3.17: A user is turning his body with an angle of θ at the time t . Position changes are calculated using triangulation. Here, the global reference frame defined in Figure 3.2 is applied.

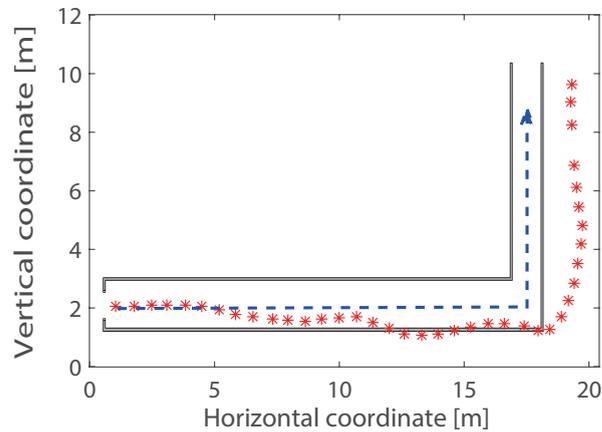


FIGURE 3.18: An example of estimation of the tester's moving trajectory.

3.5 Conclusion

The conventional inertial navigation method could be used to track the motion status of objects, including the attitude, the velocity and the position. However, drift errors are accumulated rapidly in the estimation. Therefore, rather than the conventional method, a step-based PDR tracking system is developed and applied in the thesis. Specifically, parameters for the step detection algorithm and the step length estimation algorithm are personalised for each individual user during a trial phase. Compared with the step length estimation algorithm proposed in [33], neither a convergence process (5 significant body turning are required along the paths to provide an accuracy lower than 10% are requested by [33]) nor a

significant amount of training samples (over 4000 training samples collected from 23 testers are requested by [33]) are required by the proposed method in this thesis. Step length estimation errors in the range of 4% to 9% of the values of the ground-truth step lengths could be expected by applying the proposed approach. Also, a complementary filter is proven to be capable of alleviating both the fast and slow variations in Euler estimation results. However, in the experiment, it can still be found that there exist serious biases in the location estimation results when applying the proposed PDR tracking algorithm. Hence, applying the algorithm standalone for providing localisation service is regarded as an improper choice. Constraints from wireless localisation and interior structure of of the building correcting the tracking error will be shown in Section 5.3.

Chapter 4

Wi-Fi indoor localisation

4.1 Introduction

This chapter presents the algorithms estimating the locations of on-line collected Wi-Fi RSS measurements. Firstly, in Section 4.2, the basic experiment setup including the floorplan of the area of interest and some observations about the collected Wi-Fi RSS measurements is presented. The employed indoor localisation algorithm consists of two phases: rough localisation and fine localisation. They are presented in Sections 4.3 and 4.4, respectively. In Section 4.3, the performance of unsupervised and supervised clustering algorithms applied for rough localisation in the area of interest is compared, and the final choice is made based on the comparison result. In Section 4.4, the parameters influencing the performance of the probabilistic NB algorithm including the bin width employed for discretisation and the number of training data collected on each RP are discussed. The experiment results show that the accuracy of the indoor localisation algorithm is varied with different values of the two parameters being applied.

4.2 Experimental Setup

4.2.1 Indoor area for experiments

Experiments are being carried out in C34 area of the first floor of the Portobello Centre in our university, where the communication research group is located. There exist several faculties and postgraduate research students' offices, as well as laboratories, printing room, etc.

As a common knowledge, the density of the RPs distributed within a grid influences the accuracy of indoor localisation system. While densely distributed RPs may result in high redundancy in the fingerprints and high computational load in real-time processing, sparsely distributed RPs may lead to insufficient resolution of the location estimation.

The floorplan and grid distributed RPs on it where RSS site-survey jobs are implemented are shown in Figure 4.1. Specifically, the positions of the RPs along corridors and in an open space are chosen by the designer of the localisation system. When distributing RPs, the target distance between every pair of neighbouring RP is 1.5 m. Generally, in most of the existing proposed indoor RSS localisation systems [1, 16, 23, 26, 34], the applied horizontal and vertical spacing between the neighbouring RP are varied from 1 to 3 m. Therefore, the selected distance between neighbouring APs in the implementation of this thesis coincides with them.

4.2.2 Wi-Fi fingerprints collection

RSS fingerprints used in this thesis were collected by a HTC One (M7) smartphone with a sampling frequency of at most 1 Hz. RSS fingerprint measurements are reported as integers in the range of $[-100, 0)$ in the units of decibels relative to 1 milliwatt (dBm). Along the corridors/in the open space, the Wi-Fi RSS fingerprints are collected over two opposite directions/four orthogonal directions to counteract the negative effect of RSS variation introduced by directional receiving antenna of the phone.

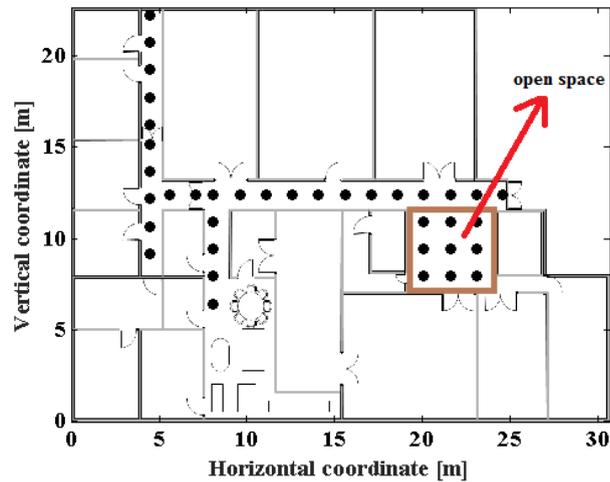


FIGURE 4.1: Map of the indoor area for experiments. Reference points are marked as black solid circles

Overall, there are 118 distinguished APs with different MAC addresses being detected in the RM built from the training phase. Figure 4.2 shows the number of occurrences of various RSS levels in a RM, including hundreds of RSS samples collected at each RP, categorised by the RSS level indexed from low to high. Note that the number of all distributed RPs in the floorplan is 37.

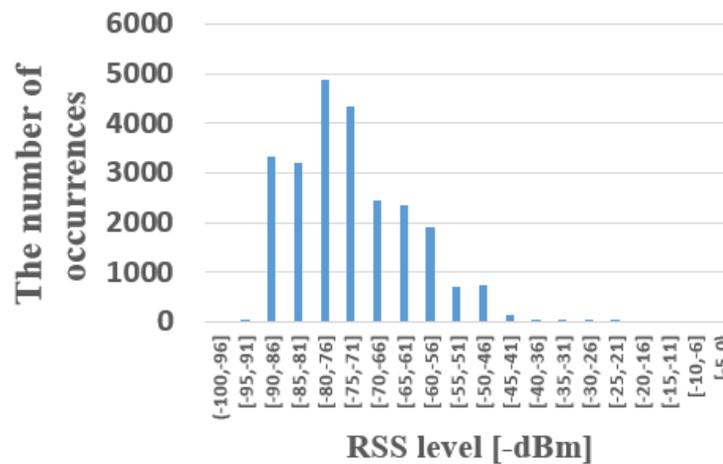


FIGURE 4.2: The number of occurrences of different RSS levels in a RM with hundreds of RSS samples collected on every RP, indexed from low to high.

It could be found in every piece of Wi-Fi RSS reported scan, the RSS information of more than one AP is missing. In addition, there exists the trend that a RSS with

lower mean power is prone to be missed by the phone, as indicated in Figure 4.3. In a practical on-line phase, missing RSS reports, specifically, from informative APs, will distort the RSS pattern, and may lead to biased localisation results. Thus, missing RSS reports is considered as one of the practical limitations and difficulties to Wi-Fi localisation systems.

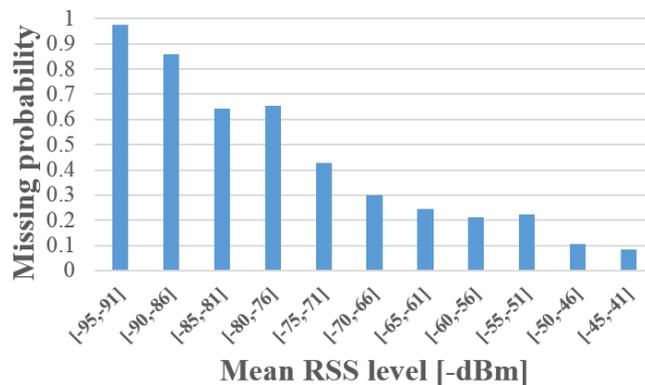


FIGURE 4.3: The averaged probability of missing a RSS VS. mean RSS level in every Wi-Fi scan result.

4.3 Rough localisation

In view of the time-varying characteristics of RSS measured in indoor environments, on-line RSS observations collected on the same position may deviate from those recorded in RM. In such cases, serious localisation errors will be made, if K positions of which the RSS training data possesses the highest similarity with them are distributed **sparsely**. Furthermore, computational load required for position updating is proportional to the number of candidate RPs. Therefore, constraining the number of candidate RPs could reduce the computations required for position updating. As a result, rough localisation becomes a necessary part of an indoor localisation system. Subsequently, the membering RPs in the estimated cluster from rough localisation are viewed as candidates for following fine localisation.

However, determination of the number of clusters and members of each cluster is challenging. A fair **requirement** is that members of each cluster should be spatially distributed compactly. In other words, the RPs in the range of every

cluster should be connected directly with each other. One such an example could be found in Figure 4.5(a).

Note that the algorithms which determine automatically the cluster membership for each RP (the range of all clusters) are termed unsupervised clustering algorithms. The affinity propagation clustering algorithm, the one proposed in [94, 95], functions as one of such unsupervised clustering algorithms. Also, it has been adopted in an indoor localisation system [34]. Another two candidates could be unsupervised k -means and one of its variant k -means++ algorithms. As mentioned in Section 2.4.1, according to [61], k -means++, one of the unsupervised clustering algorithms, aims to reduce the computational loads of Lloyd's algorithm applied by the conventional k -means algorithm, and improve the quality of the solution. Rather than random initialisation of centroids, the core concept of the k -means++ algorithm is to try to keep the initialised centroids being isolated from each other. Overall, k -means++ possesses better generalisation capability than the conventional k -means algorithm but consumes shorter convergence time. However, in the implementation of this thesis, it could be found that manual adjustment of clusters boundaries is needed, as the previously mentioned requirement of the compact distribution of members of clusters cannot be met. One such an example could be found in Figure 4.5(b). It could be easily found in the figure that some RPs marked with the same colour (belonging to the same cluster), for example the red ones and purple ones, are not directly connected with each other.

The number of clusters could be varied by changing certain parameters employed in running the unsupervised clustering algorithm. For the unsupervised k -means and k -means++ algorithm, the number of clusters expected by the operators is set explicitly. On the other hand, the number of clusters is determined by the affinity propagation algorithm, based on the preference values of the data points set manually. However, the number of clusters might not be increased linearly with the increase of the preference values of all data points in linear scale, according to the observation of the results of a experiment done by the author of the thesis. As a result, unsupervised k -means++ is chosen to be implemented for exploring the optimistic number of clusters. The criteria used here is the clustering accuracy of the algorithm, which is defined as below,

$$CA = \frac{1}{N_l} \sum_{i=1}^{N_l} \frac{N_{t-f}^i}{N_t} \quad (4.1)$$

Here, N_l denotes the running times of the unsupervised k -means++ algorithm with various randomised initial centroids of clusters; N_t denotes the number of total number of test RSS samples collected on the RPs in the area of interest; N_{t-f}^i denotes the number of test RSS samples clustered falsely in the i -th iteration. Note that the ground-truth collection RP of each test sample is known to the system priorly. Thus, it is easy to recognise the RSS samples clustered falsely, when each RP is already arranged a cluster membership by the clustering algorithm. In the experiment, the test is repeated for 100 times ($N_l=100$) for each value of the number of clusters in the range of 2 to 10; 10 samples collected on each RP for training the k -means++ algorithm are picked randomly from a pool of training data (70 in total) in each iteration; 50 test samples collected on each RP are employed for performance evaluation. The experimental result is given in Figure 4.4.

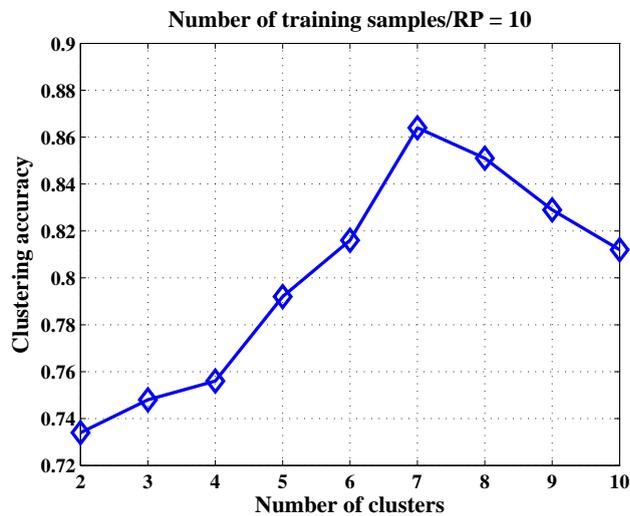


FIGURE 4.4: Clustering accuracy VS. number of clusters

From the figure, it could be found that the optimistic number of cluster is 7 resulting in the highest clustering accuracy. Note that the distribution of clusters shown in Figure 4.5(b) is also an example result of the implementation of the unsupervised k -means++ algorithm, when the number of clusters is set as 7. Also, majority of the clustering results from the other repeated tests, when the number

of clusters is set as 7, are found to be similar to this one. As a result, The cluster distribution shown in Figure 4.5(b) could function as a draft clustering pattern, on which manual adjustment could be done for obtaining a better clustering pattern. So, what is a better clustering pattern?

Since, in the implementation of this thesis, the main target is to provide localisation service to the pedestrian in the corridors in the building, if RPs in the same cluster are not distributed co-linearly, the pedestrian users may be localised to rooms incorrectly by applying the Wi-Fi LF localisation approach, resulting from the weighted sum calculation as indicated in equation 2.6. Under such circumstances, the location estimation results may be required to be projected to the corridor in the closest proximity to it. Naturally, additional computation is required, which is regarded as time-consuming. As a result, to avoid such non-linear distribution of RPs in the same cluster is the second **requirement** for manual adjustment of the clustering pattern shown in Figure 4.5(b).

Besides the two above mentioned requirements, the functionalities of RPs should also be taken into account. For example, the open space bounded by the brown rectangle drawn in Figure 4.1 is for demonstration and exhibition purposes, different from other RPs in the corridors. Bearing these requirements in mind, a possible adjustment result could be obtained, as shown in Figure 4.5(a).

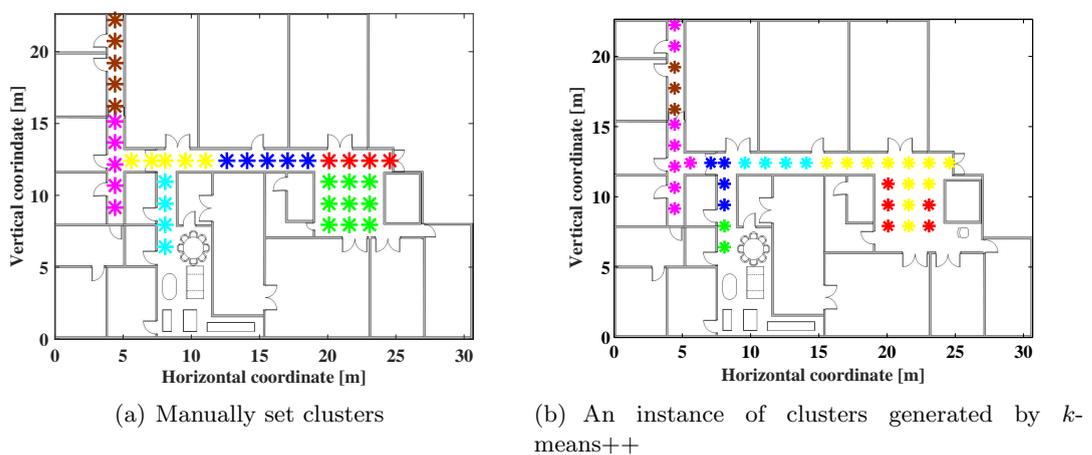


FIGURE 4.5: Distribution of member RPs of clusters

With labelled training RSS samples, one of OVO-SVMs and OVA-SVMs is then

chosen to be trained and employed as the clustering algorithm for rough localisation of the on-line RSS fingerprints. In the implementation of this thesis, the slack variable for implementation of SVM for error tolerance is set as 1. Also, the linear kernel model is employed in the implementation of the SVM. In addition, the performance of k -means algorithm whose initial centroids are explicitly set by the author, termed supervised k -means, is compared with OVO-SVMs and OVA-SVMs. Specifically, for the supervised k -means, the average of all training RSS vectors collected on all member RPs of each cluster is employed as the exemplar of each cluster. So, totally, there are 7 different exemplars in this case.

Also, two unsupervised clustering algorithms: affinity propagation as proposed in [94] and unsupervised k -means++ [61], are evaluated. Following that, their performance is compared with the supervised clustering algorithms.

A publicly accessible external library of the affinity propagation algorithm being available at ¹ is applied in the implementation of this thesis. Currently it is maintained by the authors of [94].

Similar to the above experiments, the same number of training and test samples are collected on each RP to form the pool of training data and dataset of the test samples. Also, the clustering accuracy is used to evaluate the performance of the clustering algorithms. In every test, only a specific number of RSS samples are picked randomly from the pool of training data, and used to train the clustering algorithms. In particular, the number of the selected RSS training samples varies from 10 to 50 with an increasing step of 10. The net clustering accuracy for each specific number of RSS training samples is also obtained by averaging the results of 100 repeated tests in which the training data is randomly picked from the pool. In the test, the number of clusters employed for all clustering algorithms is equally set as 7 (the preference values of all data points are set properly for implementation of the affinity propagation algorithm).

The evaluation result is shown in Figure 4.6. Overall, the performance of supervised k -means improves only slightly with increasing number of training samples, which is different from the cases of multi-class SVMs, and is proved to be the

¹<http://www.psi.toronto.edu/index.php?q=affinity20propagation>

worst among all. In addition, performance of both multi-class SVMs is shown to be higher than that of the unsupervised clustering algorithms shown. When the number of training samples per RP is set to be equal to or larger than 30, the performance of OVO-SVMs is better than OVA-SVMs, but worse otherwise. For OVO-SVMs, when the number of applied training samples per RP is insufficient, a large number of training samples collected outside a particular cluster could assist greatly for the algorithm to learn the boundary between the RSS fingerprints collected on the RPs in the cluster and the ones collected outside the range of the cluster. However, when more training samples per RP are applied, the total number of training samples for the RPs outside the range of a particular cluster becomes overwhelmingly large, so that the scales of the two classes labelled with +1/-1 become imbalanced [96]. Under such circumstance, the learned boundary to distinguish the RSS fingerprints collected in/out of a particular cluster is likely to be biased. Therefore, OVA-SVMs should be chosen as the clustering algorithm when the available number of training samples per RP is lower than or equal to 20. Otherwise, OVO-SVMs should be chosen.

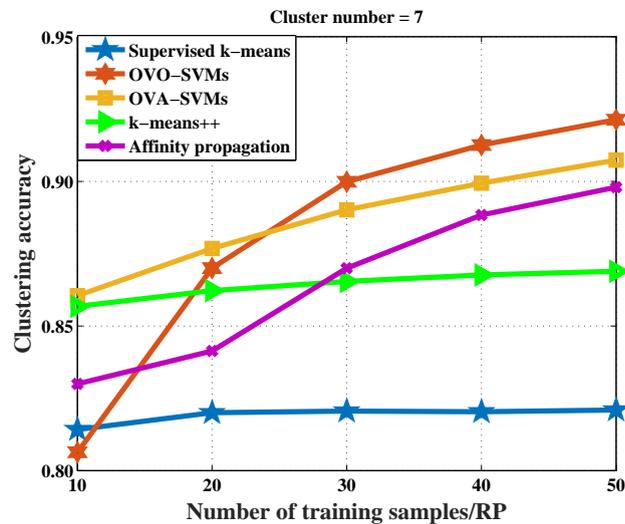


FIGURE 4.6: Clustering accuracy VS. number of training samples

4.4 Fine localisation

In this section, the performance of a NB localisation algorithm with empirical RSS distribution is tested. Specifically, the root mean squared error (RMSE) is employed to evaluate the performance, as:

$$RMSE = \frac{1}{N'_l} \sum_{i=1}^{N'_l} \sqrt{\frac{1}{N_R} \sum_{k=1}^{N_R} \|\mathbf{p}_k^i - \hat{\mathbf{p}}_k^i\|^2} \quad (4.2)$$

Here, N'_l and N_R denote the repetition number of the tests and number of testing RSS samples involved in each test; \mathbf{p} and $\hat{\mathbf{p}}$ denote the ground-truth and estimated 2-D position vector. $\hat{\mathbf{p}}$ is estimated as:

$$\hat{\mathbf{p}} = \sum_{j=1}^J Pr(\mathbf{p}_j | \psi(\mathbf{o})) \mathbf{p}_j, \quad \forall j \in \{1, 2, \dots, J\} \quad (4.3)$$

Since clustering accuracy is also influenced by the number of training samples/RP, in order to analyse only the effects of related parameters on the accuracy of the fine localisation, in the tests carried out in Section 4.4.1 and 4.4.2, only the RPs highlighted as green marks (belonging to one cluster) on Figure 4.5(a) are involved. Therefore, rough localisation is not implemented in Section 4.4.1 and 4.4.2.

4.4.1 Analysis of the effect of varying bin width

For probabilistic LF methods based on empirical RSS distribution, continuous RSS values are required to be categorised into discrete bins to overcome the high discretisation variance problem. Recalled in Section 2.3, bin width is adjusted to achieve such target.

With the same number of training RSS samples collected at one particular RP from one particular AP, the pattern of the probability distribution function (PDF) of the discretized RSS bins varies. An example is shown in Figure 4.7.

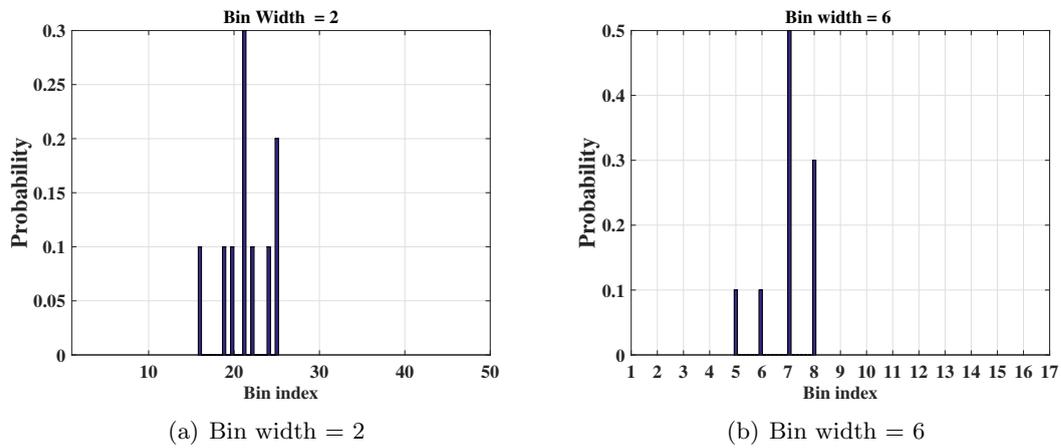


FIGURE 4.7: An example of probability distribution of 10 RSS training samples after discretization with different bin width

From the figure, when the bin width is set as 2, it can be found that probabilities of some existing bins are zeroes, while probabilities of neighbouring bins on both side of them are not. Under this circumstance, the probability density of the discretised RSS values is considered to be distributed sparsely. Also, as could be observed from the second figure, this situation disappears when the applied bin width is set as 6. The influence of employing different bin widths on the accuracy of the NB localisation algorithm is studied and indicated in Figure 4.8(a). The number of the resulting bins for each applied bin width is also shown in Figure 4.8(b). From Figure 4.8(b), it could be found that larger bin width will lead to fewer bins. Also, as could be found in Figure 4.8(a), the lowest RMSE occurs when the bin width is set at 4.

In the area of application of machine learning algorithms, the number of APs could be regarded as the number of features describing a particular object or event; the number of bins employed is related to the number of different classes that a particular feature may fall into. The sparse RSS distribution model presented in Figure 4.7(a) is regarded as an example of a too complicated mathematical model, and thus degrades positioning accuracy results. As mentioned in Section 2.3.1.1, if the bin width is enlarged to the extreme so that only one bin will be generated (all RSS values are categorised into the one and only one bin), it is expected that the mathematical model generated cannot differentiate any individual object between each other, and therefore also leads to degraded positioning accuracy (under-fitting

problem) that is even worse than the over-fitting problem, as presented in Figure 4.8(a). Overall, it is important to apply the proper bin width for discretising the continuous RSS values, when the probabilistic NB algorithm is used.

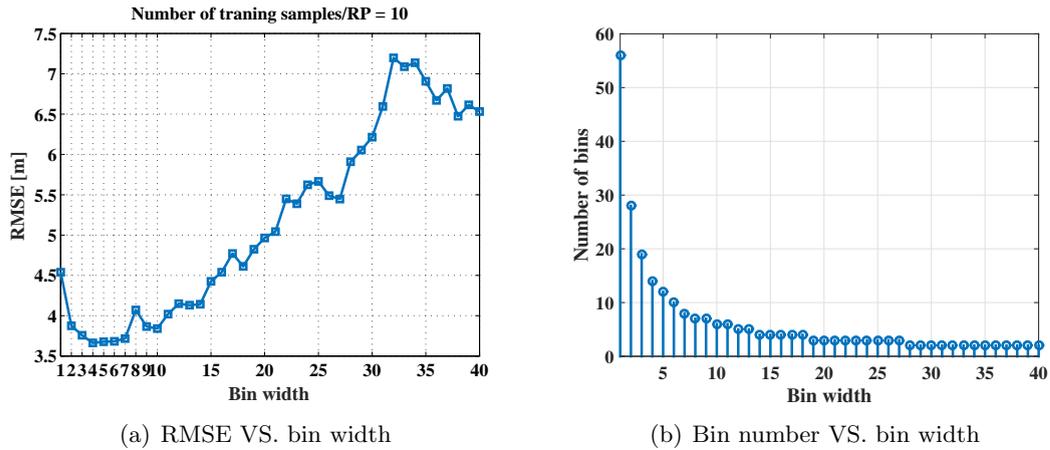


FIGURE 4.8: Analysis of effect of varying bin width

4.4.2 Analysis of the effect of varying the number of training samples collected on each RP

Generally, more training samples implies a more time-consuming RSS site-survey phase. But probability density distributions of discretised RSS samples are less likely to be sparse when the same narrower bin width is applied, if more training samples are available. As a result, the over-fitting problem is more likely to be tackled.

Such an example is given in Figure 4.9. In this experiment, more RSS training samples are collected on each RP. From the figure, it could be found that the lowest RMSE is approximately 2.6 m, which occurs when the bin width is set at 2. The lowest RMSE and the optimal bin width are lower than the previous case (indicated in Figure 4.8), which confirms that discretised RSS samples are no longer sparsely distributed when the bin width is set at 2, with more RSS training samples being available to the system. Furthermore, with more available training samples, the localisation algorithm is proved to achieve a better positioning accuracy. Tests have also been carried out with a set of different available numbers of training samples/RP in the range of 10 to 50 with an increasing step of 10. The related

experimental results including the lowest RMSE and the corresponding optimal bin width, are presented in Table 4.1.

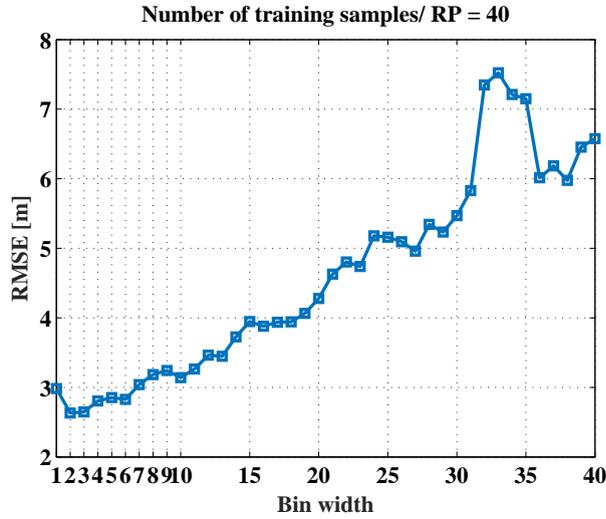


FIGURE 4.9: RMSE VS. bin width, when the number of training samples/RP is set at 40.

| Number of training samples/RP | 10 | 20 | 30 | 40 | 50 |
|-------------------------------|-----|-----|-----|-----|-----|
| Lowest RMSE | 3.7 | 3.1 | 2.8 | 2.6 | 2.4 |
| Optimal bin width | 4 | 4 | 3 | 2 | 2 |

TABLE 4.1: Experimental results of the best posing accuracy and optimal bin width VS. number of training samples/RP.

As could be found from the table, the lowest RMSE decreases monotonically, with increasing number of training samples collected at each RP being available to the system. Meanwhile, the optimal bin width decreases also. This observation coincides with the reviewed knowledges included in Section 2.3.1.1: with more training data being available to the system, the discretisation variance is reduced; discretisation of the RSS continuous values with smaller bin width will reduce the probability of the decision boundaries falling in to bins, which implies that the discretisation bias is reduced also. Furthermore, the different value of the optimistic bin width for discretion of the continuous RSS samples given by the experimental results, when different number of training samples are available to the system as indicated in the table 4.1, could be regarded as an empirical evidence to be applied by probabilistic indoor localisation systems deployed in the other scenarios in the future.

4.5 Chapter Summary

This chapter consists of three parts. The first part presents the setup of the experiment, including the floorplan and the distribution of the RPs for collection of Wi-Fi RSS measurements, information about the collected Wi-Fi RSS samples, etc.

In the next part, rough localisation for reducing the computational load in on-line Wi-Fi localization service is introduced. Several clustering algorithms categorised into supervised and unsupervised classes are included. Firstly, the clustering accuracy is employed for performance evaluation of a unsupervised clustering algorithm *k*-means++, with the number of clusters being varied. The experimental results show that the best clustering accuracy occurs when the number of cluster is set 7. One of the output clustering patterns is used as a draft example draft, on which additional manual adjustment of the cluster membership of the RPs is needed, according to the requirements presented in Section 4.3. Following that, one of supervised clustering algorithms is chosen for implementation of the rough localisation, based on the performance evaluation results. Also, the performance of other clustering algorithms included in the chapter is also evaluated. The related experimental results indicate that the performance of OVA-SVMs is the best, when the number of training samples on each RP is less than or equal to 20. Therefore, it is applied for the implementation of thesis for the rough localisation purposes.

Finally, the influences of several factors on the accuracy of fine localisation approach (probabilistic NB algorithm) are discussed. Firstly, the optimal value of the bin with for discretising the continuous RSS value is shown to be different, when different number of training RSS measurements are available to the system. Generally, a smaller value should be chosen, when more training RSS measurements have been collected. Secondly, the evaluation result confirms again that RM consisting of more training samples will result in a better localisation accuracy. Thirdly, the experimental results indicated in Figure 4.1, the dependency of the optimistic discretisation value of RSS samples on the scale of training data, could be used as an empirical reference evidence for the deployments of probabilistic indoor localisation systems in other scenarios in the future.

Chapter 5

Applying a particle filter for indoor pedestrian localisation and tracking

5.1 Introduction

In this chapter, the detailed implementation of particle filter to track user's position is presented. Firstly, preparatory works such as building mathematical models for walls in the floorplan, building adjacency matrix for the manually distributed RPs and therefore calculating the inter-RP distance, etc., are presented. Following that, the core of particle implementation including particle propagation, particle correction, particle re-sampling, initialisation of particles for the particle filter, etc., is presented. Particles are distributed initially into a constrained area by taking advantage of the Wi-Fi rough and fine localisation algorithm presented in the previous chapter. In particular, two or three RSS measurements collected at the starting positions are required to implement the clustering algorithm for rough localisation to reduce the possibility of incorrect clustering result. Also, the IMU reports are expected to become more stable when users start to walk, given two or three RSS measurements required to be collected at the starting position, which leads to more accurate orientation estimation results. Following that, analysis result of the IMU reports is applied to update the positions of particles; correction of

particles are accomplished by taking advantage of the wall constraints, and on-line RSS measurements when significant users' orientation changes are detected. Since some of the particles are filtered in the particle correction phase, new particles are generated by copying the remaining surviving particles to keep the number of employed particles constant, which is also named as re-sampling phase.

In the experimental evaluation part, firstly, it is confirmed that sufficient number of particles should be applied to remove the possibility that all particles are killed in any one step event, avoiding the problem of single point of failure. Secondly, the experiment results confirm that applying two/three RSS measurements for obtaining rough localisation result of the RSS measurements collected at the starting position could improve the correct clustering probability significantly. This chapter also confirms that the accuracy of the pedestrian location tracking results are highly dependent upon the clustering accuracies of the RSS vectors received on the starting positions. Thirdly, the experiment results show that better fine localisation result of the RSS measurements collected at the starting position could enhance the accuracy of the pedestrian tracking result, under the circumstance that the surrounding wall constraints around the first turning point along the path perform poorly in preventing the particles from propagating into wrong positions or the first turning point is far from the starting position. Finally, the experiment evaluation results confirm that taking advantage of on-line Wi-Fi RSS measurements to adjust the importance weights of the employed particles could reduce the negative influence of the multi-clustered particles to the on-line localisation accuracy.

5.2 Mathematical representation of environment constraints

5.2.1 Walls representation

In the work, corridors and an open area close to the entrance of the lab are taken into account in the experiment; thus, all doors access to rooms are ignored when building mathematical models for walls. A simplified example is given in figure 5.1: horizontal and vertical coordinates of both end points of every piece of straight

wall are employed to build the mathematical model. From the figure, it can be found that, in total, there are four walls illustrated in this example. So, the wall database of this scenario could be built as a matrix, if the left bottom corner is set as the origin of the coordinate system:

$$\begin{bmatrix} 0 & 0 & 8 & 0 \\ 0.5 & 0.5 & 8 & 0.5 \\ 0 & 0 & 0 & 4 \\ 0.5 & 0.5 & 0.5 & 4 \end{bmatrix} \quad (5.1)$$

Here, the number of rows indicates how many pieces of linear walls are recorded in the database. The arrangement of entities for each row (linear wall) from left to right is specified as: horizontal coordinate of the first end point, vertical coordinate of the first end point, horizontal coordinate of the second end point, vertical coordinate of the second end point.

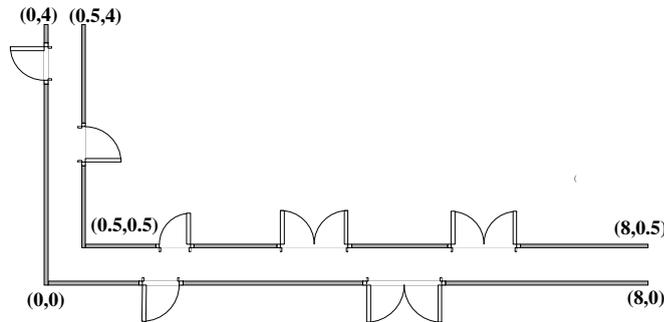


FIGURE 5.1: An example of walls representation

5.2.2 Calculation of inter-RP distance

As shown in Figure 4.1, in the implementation of this thesis, two RPs are regarded as being directly connected to each other if the geometric distance between them is lower than or equal to 1.5 m (the pre-defined distance between neighbouring RPs horizontally and vertically). Accordingly, the adjacency matrix is built.

With the knowledge related to the adjacency matrix, as presented in Section 2.6, a route between each pair of RPs could be built by taking advantage of route-planning methods such as Dijkstra algorithm. Also, according to the planned route, the Manhattan distance between every pair of RPs could thus be recognised automatically.

5.2.3 Automatic arrangement of cells to RPs

One of the applications of the adjacency matrix could be the arrangement of rectangular cells to RPs. For the purpose of defining the ranges of rectangular cells, boundaries for four orthogonal directions should be recognised.

An example is illustrated in Figure 5.2. In the figure, RP1 denotes the targeted RP. RP2 and RP3 denote its neighbouring RPs. As previously stated, the information of whether two RPs are neighbours or not could be known by exploring the adjacency matrix. Also, the walls presented in the figure could be found as the neighbouring walls of these RPs. It is easy to know that the range of the cell of RP1 is limited by line segments defined by the boundary points marked as blue circles. Thus, it is essential to learn the 2-D coordinates of these four boundary points. In this case, both of the vertical coordinates of BY1 and BY2 are specified as the mean of the vertical coordinates of RP1 and RP2; both of the horizontal coordinates of BY1 and BY3 are specified as the horizontal coordinate of the vertical wall; both of the horizontal coordinates of BY2 and BY4 are specified as the mean of the horizontal coordinates of RP1 and RP3; the vertical coordinates of BY3 and BY4 are specified as the vertical coordinate of the horizontal wall.

According to the flow chart illustrated in Figure 5.3 , cells and their boundaries are arranged to the RPs distributed in the floorplan of the experiment. The final results are shown in Figure 5.4.

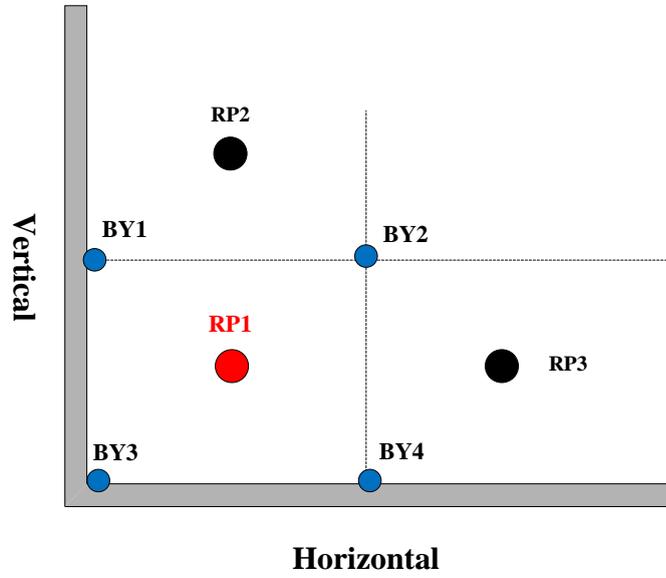


FIGURE 5.2: An example of arranging cell boundaries for a RP.

5.3 Implementation of particle filter

Recalled from Section 2.7.2, the particle filter aims to use a set of weighted samples termed particles to approximate the belief $Bel(\mathbf{x}^t)$ of the state at time t , as:

$$\Lambda^t = \{\mathbf{x}_i^t, \omega_i^t\}, i = 1, 2, \dots, N_s. \quad (5.2)$$

The particle update is accomplished by generating the set Λ^t from the most recent set $\Lambda^{t-\delta t}$. The process of particle update is indicated as follows:

1. Propagation: sample the particles using the proposal distribution:

$$\mathbf{x}_i^t \sim Pr(\mathbf{x}_i^t | \mathbf{x}_i^{t-\delta t}, \mathbf{u}^t) \quad (5.3)$$

2. Correction: the weight of each particle is updated as:

$$\omega_i^t = \omega_i^{t-\delta t} \cdot Pr(\mathbf{z}^t | \mathbf{x}_i^t) \quad (5.4)$$

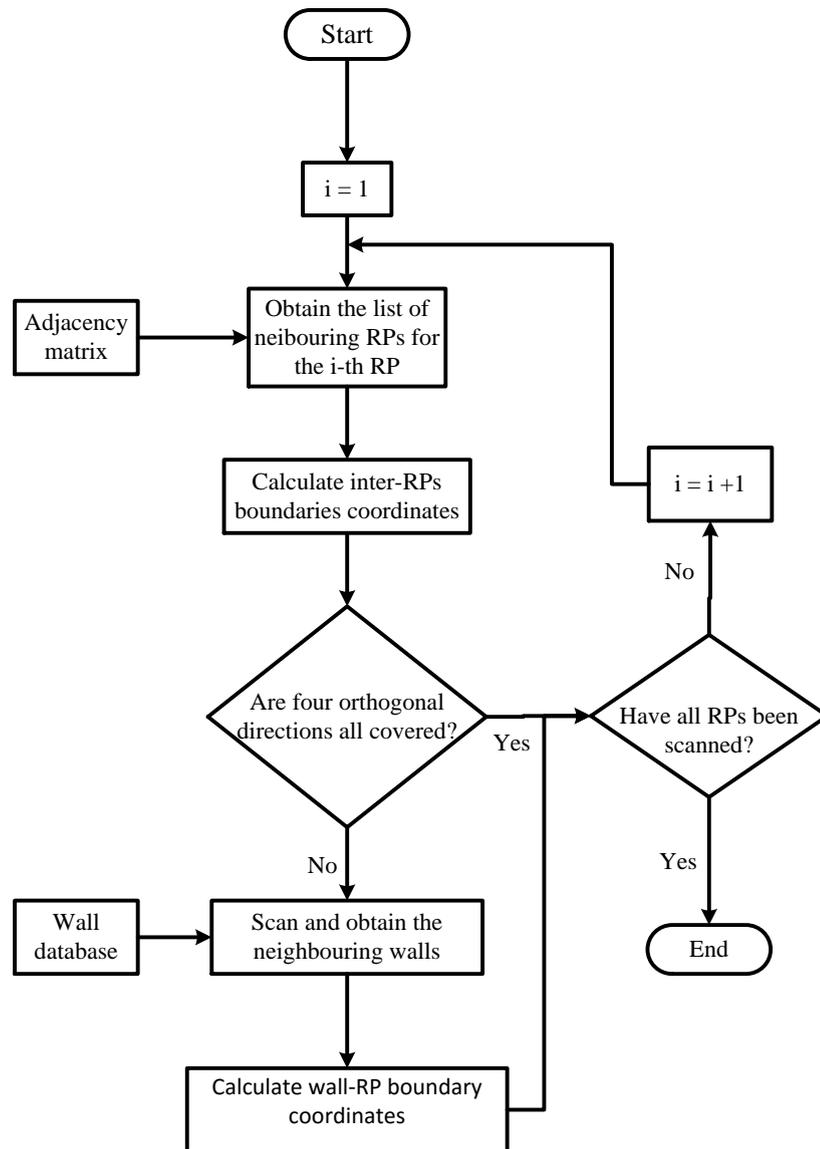


FIGURE 5.3: The flow chart of recognition boundaries of cells of all distributed RPs.

3. Re-sampling: suppose N_{fl} particles are filtered in a correction step, another new set of N_{fl} particles are required to be sampled from the set of $N_s - N_{fl}$ surviving particles. This process is for the aim to keep the number of the employed particles constant. The probability of each surviving particles being copied to generate the new set of particles is proportional to its importance weight. Also, partial importance weights of the copied particles will be assigned to its clones, so that the overall importance weight of this copied particle remains the same. With this importance weights re-assignment scheme, the propagated particles are not likely to be compactly distributed, which prevents the particles from being all killed

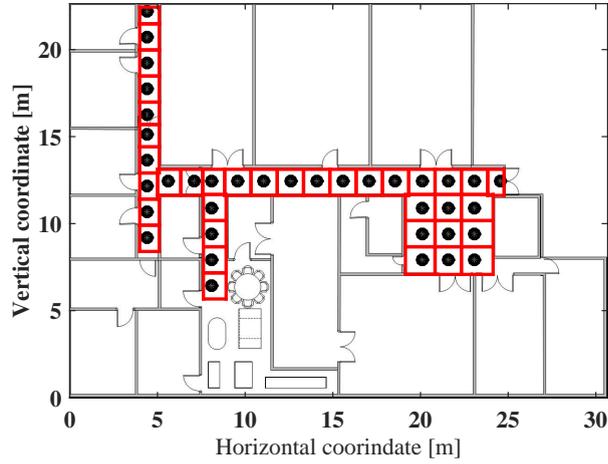


FIGURE 5.4: Arranged cells for the RPs distributed in the floorplan.

suddenly and also extremely biased location estimation result could be avoided.

For the application of tracking pedestrian's position, the state of every particle includes several elements, as:

$$\mathbf{x}^t = (\mathbf{p}_{x'}^t, \mathbf{p}_{y'}^t) \quad (5.5)$$

where $\mathbf{p}_{x'}^t$ and $\mathbf{p}_{y'}^t$ denote the position coordinate of the user along the x' and y' axis in the global reference frame, respectively, at time t . These three steps will be presented in detail in the following sections.

5.3.1 Particle propagation

In this step, the i -th particle's state \mathbf{x}_i^t is updated according to the motion model distribution $Pr(\mathbf{x}_i^t | \mathbf{x}_i^{t-\delta t}, \mathbf{u}^t)$. Here, $\mathbf{u}^t = (l, \alpha, \Upsilon)$ denotes the step event detected at time t . Taking advantage of \mathbf{u}^t ensures that every particle is propagated in a way that coincides with the measurements made by the IMU module.

To take into account the uncertainties in estimation of step length and orientation, they are manually perturbed by variables, as:

$$\begin{cases} l' = l + \Delta l \\ \theta = \alpha + \Delta\alpha - (\Upsilon + \Delta\Upsilon) \end{cases} \quad (5.6)$$

Here, Δl and $\Delta\alpha$ denote the modelled uncertainties, which follow zero-mean Gaussian distribution with different variances. In the system, the standard deviations are defined to be 10 cm and 10 degree, respectively. If the polarity of the obtained Υ is detected to be positive, $\Delta\Upsilon$ follows $\varpi_1(p_1, \mu_1, \mu_2, \sigma_1, \sigma_2)$, otherwise it follows $\varpi_2(p_2, \mu_3, \mu_4, \sigma_3, \sigma_4)$. Note that $\varpi_1(p_1, \mu_1, \mu_2, \sigma_1, \sigma_2)$ and $\varpi_2(p_2, \mu_3, \mu_4, \sigma_3, \sigma_4)$ follow the bimodal Gaussian probability distribution shown in Figure 3.12(a) and 3.12(b), respectively. The mathematical representation of the probability distribution of ϖ_1 and ϖ_2 could be represented as:

$$f_{\varpi_1}(\Delta\Upsilon) = p_1 \frac{1}{\sigma_1 \sqrt{2\pi}} e^{-\frac{(\Delta\Upsilon - \mu_1)^2}{2\sigma_1^2}} + (1 - p_1) \frac{1}{\sigma_2 \sqrt{2\pi}} e^{-\frac{(\Delta\Upsilon - \mu_2)^2}{2\sigma_2^2}} \quad (5.7)$$

and

$$f_{\varpi_2}(\Delta\Upsilon) = p_2 \frac{1}{\sigma_3 \sqrt{2\pi}} e^{-\frac{(\Delta\Upsilon - \mu_3)^2}{2\sigma_3^2}} + (1 - p_2) \frac{1}{\sigma_4 \sqrt{2\pi}} e^{-\frac{(\Delta\Upsilon - \mu_4)^2}{2\sigma_4^2}} \quad (5.8)$$

The fit values of the parameters of the two bimodal Gaussian probability distributions are indicated in table 5.1 and 5.2. Then, the unit of θ is transformed to radians. Finally, θ is normalised to θ_n , according to equation (3.27).

| | |
|------------|------|
| μ_1 | 0 |
| μ_2 | -30 |
| σ_1 | 6.3 |
| σ_2 | 6.4 |
| p_1 | 0.72 |

TABLE 5.1: The parameter values for ϖ_1 .

| | |
|------------|------|
| μ_3 | 0.1 |
| μ_4 | 28.4 |
| σ_3 | 6.55 |
| σ_4 | 6.51 |
| p_2 | 0.69 |

TABLE 5.2: The parameter values for ϖ_2 .

With these knowledges, particles are propagated as:

$$\begin{cases} \mathbf{p}_{x'}^t = \mathbf{p}_{x'}^{t-\delta t} + l' \cos(\theta_n) \\ \mathbf{p}_{y'}^t = \mathbf{p}_{y'}^{t-\delta t} + l' \sin(\theta_n) \end{cases} \quad (5.9)$$

Note that the purpose of taking measurement uncertainties into account, when the propagation of the particles are implemented, is to scatter the distribution of the employed particles in a limited range, so that the implementation of the filter is immune to single point of failure: the on-line localisation system stops working, since all employed particles are concentrated on one position and are killed if an incorrect IMU measurement is made. Also, mapping of the coordinates of users' moving positions in the 2-D global reference frame to the local one is implemented, as implemented in Section 3.4.5.

5.3.2 Particle correction

In this step, the importance weights of propagated particles are updated. In the implementation of this thesis, both of environmental constraints and Wi-Fi on-line localisation results are employed to undertake this task. Firstly, how environmental constraints could be used to update the weights is explained.

If a particular propagated particle is found to have collided with any wall shown on the floor plan, the particle is filtered out, since the collision implies a possible user moving path intersecting with a wall, as:

$$\omega_i^t = \omega_i^{t-\delta t} \cdot 0 = 0 \quad (5.10)$$

The approach determining if a particle's moving trajectory collides with a specific wall or not, is applied as follows: given the 2-D coordinates of the particle's propagated locations at time t , $(\mathbf{p}_{x'}^t, \mathbf{p}_{y'}^t)$, and the locations before the propagation, $(\mathbf{p}_{x'}^{t-\delta t}, \mathbf{p}_{y'}^{t-\delta t})$, as 2-D coordinates of the motion endpoints, a linear equation representing the user's trajectory can be written, in the form $y = kx + b$. Similarly, linear equations for wall vectors could be built as well. Then, by substituting two

motion endpoint's 2-D coordinates for 'x' and 'y' in the wall linear function, respectively, two values termed M_1 and M_2 are obtained. Similarly, by substituting two wall endpoint's 2-D coordinates for 'x' and 'y' in the linear motion function, another set of values termed W_1 and W_2 could be known. Three different scenarios could result:

- M_1 and M_2 have the same polarities (they are both negative or both positive values) **AND** the polarities of W_1 and W_2 are the same as well: in this scenario a particle's trajectory does not collide with the wall. See black dashed line shown in Figure 5.5.
- M_1 and M_2 have different polarities (one of them is a negative value and the other is positive) **AND** W_1 and W_2 have different polarities as well: in this scenario the user's trajectory collides with the wall. See blue dashed line shown in Figure 5.5.
- Either M_1 and M_2 have different polarities **AND** W_1 and W_2 have the same polarities **OR** M_1 and M_2 have the same polarities **AND** W_1 and W_2 have different polarities: in this scenario the user's trajectory does not collide with the wall. See red line shown in Figure 5.5.

Moreover, the phenomenon of the multi-clustered distribution of particles could be observed when the tester takes turns on some paths. One such example is given in Figure 5.6. The user was walking on the path shown in Figure 5.7(a). Such a phenomenon occurs when the user takes a turn from a horizontal corridor to a vertical one on the path. Since some particles are in the meantime propagated into the left-most vertical corridor and there is no obstruction correcting these wrongly propagated particles, they are gradually separated from the accurately distributed particles, and thus a multi-clustered distribution phenomenon emerges. It is easy to find that it will lead to deviated on-line localisation results.

To tackle this problem, adjustments of the importance weights of the particles by utilising the Wi-Fi localisation results are triggered once upon such user's significant body turning is found by analysing the reports from IMU module. However, since the estimation of PL_γ is not accurate enough when the orientation

difference of human beings and their smartphones are low and the estimation of orientations of human beings' bodies are related to the estimation of PL_T , in practice, the change of yaw angles of users' phone are tracked instead for the purposes. Here, smartphones are assumed to be held steadily by users while walking. Suppose a user is estimated to be in the j -th cell by Wi-Fi LF localisation. The importance weight of each particle is updated according to the cell distance between the j -th cell and the occupied cell by the particle, as:

$$\omega_i^t \leftarrow \omega_i^t \cdot \frac{1}{DC(i, j) + 1} \quad (5.11)$$

,

Here, it is assumed that the particle is propagated into the i -th cell. and $DC(i, j)$ denotes the inter-RP distance between the i -th and j -th RP. The distance being zero implies the position of the particle coincides with the on-line Wi-Fi LF localisation result. In such a case, the weight remains the same. Otherwise, the importance weight will be reduced monotonically when the inter-RP distance is increased. In the end, particles' importance weights will be normalized so that the sum of them equals 1. With this method, the importance weights of wrongly clustered particles are expected to be reduced dramatically. Also, note that the inter-RP distance could be calculated and then stored in a database in a bootstrap phase with the aid of route-planning algorithm and automatic arrangement of cells, as shown in the previous sections. In the on-line phase, the inter-RP distance could be found instantly from the database, which reduces the computation loads in the on-line phase.

5.3.3 Particle Re-sampling

In [36], it can be found that an increasing number of originally generated particles collide with walls during users' walks. In other words, the number of particles employed from the particle correction phase will decrease over time in the on-line phase if no new particle is generated in the duration of the walk. Therefore, it is

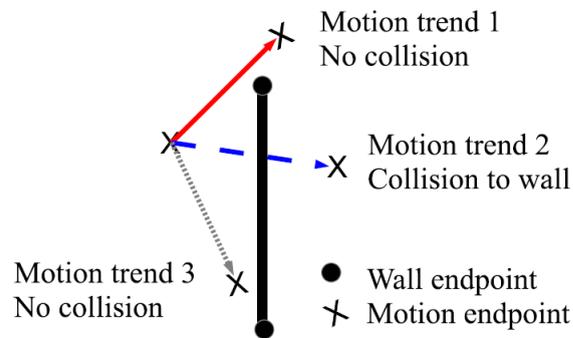


FIGURE 5.5: A collision is detected in the second scenario.

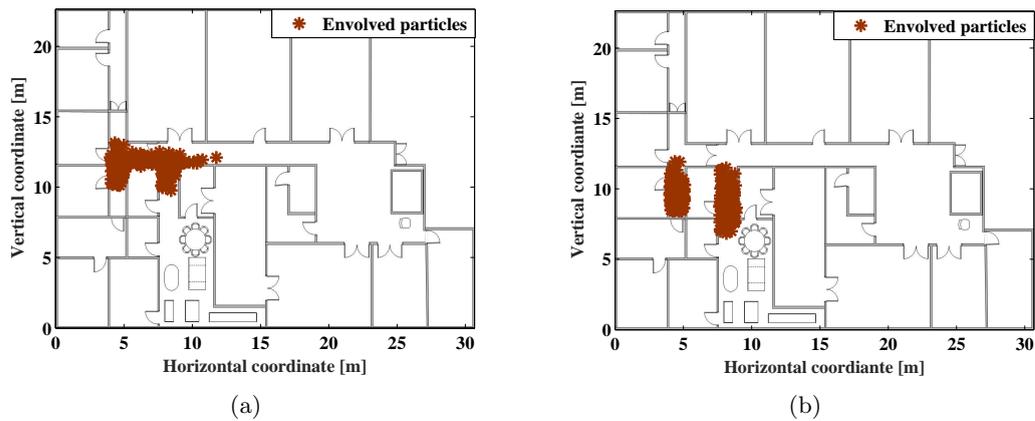


FIGURE 5.6: An example of a multi-clustered phenomenon

expected that location estimation could not be given in such cases when all particles are killed before the end of the user's walk (failure of service). Furthermore, occasionally noisy IMU measurements may lead to the filtering of a large number of particles which are close to the ground-truth states. In such cases, distribution of the remaining surviving particles might be skewed in following location estimation process. Imagining that many particles far from the ground-truth position could be filtered by taking advantage of map constraints and the Wi-Fi on-line localisation algorithm, if new particles are generated by copying the surviving particles, the skewed distribution of the on-line employed particles could be thus re-calibrated again.

As a result, it is necessary to implement the particle re-sampling scheme in the particle filtering algorithm. Especially, new particles are generated according to the set of surviving particles from the correction phase during every step event

detected. In the implementation of this thesis, the states of the new particles are sampled randomly from the set of surviving particles. The probability of each surviving particle being sampled is proportional to its updated importance weight. Also, note that surviving particles could be copied for more than once in one particular re-sampling step. The weight of each newly created particle is assigned according to the weight of its parent particle, and also the frequency of its parent particle being sampled in the current re-sampling implementation. For example, suppose the i -th particle with weight ω_i^t is sampled three times in a particular re-sampling implementation, the weights of both newly created particles and the parent particle are assigned or adjusted to be $\frac{1}{4}\omega_i^t$.

The details of the method of randomly choosing surviving particles from the particle correction phase for re-sampling implementation are given as follows. Firstly, a cumulative weight for each particle is calculated, as:

$$CL^t = (cl_1^t, \dots, cl_{N_s - N_{fl}}^t) \quad (5.12)$$

where

$$cl_i^t = \sum_{i'=1}^i \omega_{i'}^t \quad (5.13)$$

Here, the set of cumulative weights of all particles and the cumulative weight of the i -th particle at the time t are denoted as CL^t and cl_i^t , respectively. In the next step, totally N_{fl} float-point numbers in the range of $[0,1]$ are randomly picked out from a uniform probability distribution. For a particular randomly generated float-point number zz , the state of the j -th particle is sampled, where cl_j^t is the smallest number being larger than zz .

5.3.4 Initialisation of particles for the filter

If particles are distributed initially in the whole floorplan, a large number of particles are required to cover the whole area, which consumes loads of computational power for particle updating, and does not satisfy the requirement of real-time

implementation. Furthermore, adequate representation of $Bel(\mathbf{x})$ may only be obtained after a long-period user's walk. Thus, when users take short-time walk, the particle filter might not be capable of converging the particles to be representation of $Bel(\mathbf{x})$ adequately, which leads to deviated on-line localisation accuracies. As a result, it is necessary to distribute the particles into a smaller range initially.

Totally, there are two choices to designate range for distributing particles initially. The first one is to designate manually : the user clicks the screen of the phone to inform the phone of he his/her start position before the start of the walk, such as entrances of the lab, intersection area of two orthogonal corridors, etc.

However, a more common and convenient way to do so is the second choice: initialisation of the particle distribution range by taking advantage of the on-line Wi-Fi indoor localisation algorithm shown in the previous chapter. Specifically, multi-class SVMs algorithm is used for rough localisation. In the next step, the probabilistic NB localisation method is employed for fine localisation. Finally, totally N_s particles are distributed in the range of the arranged cells of all RP candidates in the estimated cluster, according to the inferred posterior probability of the RP locations given the on-line RSS measurement. The number of particles in corresponding cells are $N_s \cdot Pr\{\mathbf{p}_1|\mathbf{o}\}, \dots, N_s \cdot Pr\{\mathbf{p}_{J'}|\mathbf{o}\}$. Furthermore, particles are uniformly distributed in each targeted RP's cell for a scatter particle distribution. However, the localisation accuracy would be seriously deviated, especially, in the initial several steps of the user's walk given a wrong clustering result. To avoid the occurrence of such cases, more than one RSS sample is required to be collected continuously over time at the starting position before informing the user that he can start to walk and activating the module of particle updating. In addition, another benefit could be taking advantage of the time duration for stabilising the orientation estimation. Collected RSS samples over time will be all clustered, and finally the particles will be distributed into the cluster which is nominated by the clustering algorithm with the highest frequency. However, in case two or more clusters are nominated equally with the highest frequency, more RSS samples are requested until only one cluster is finally nominated with the highest frequency. Note that in the implementation of this thesis, the particles' initial distribution ranges are defined exactly as the clusters shown in Figure 4.5(a).

5.4 Experimental evaluation

Overall, the performance of the system is evaluated by a subject tester who walks along four different paths, as shown in Figure 5.7. The paths have different numbers of turns and different start/end positions. In the figure, arrows indicate the moving directions of the subject user. The characteristics of the routes are summarized in table 5.3. Except the experiments of Section 5.4.1, for the experiments of other following sections, for each route, the tester walks along it for once to collect the IMU reports to be used for evaluation of the proposed localisation algorithm from different aspects. Sixty Wi-Fi measurements are collected at each of the starting positions. Each of them is regard and used as a possible on-line RSS measurement collected by the tester when standing on the positions during the walks to initialise the distribution of the particles. Note that, due to the noise embedded in the acceleration measurements, one or more steps taken by the tester are missed by the step detection algorithm analysing the measurements. Also note that since the variation of the tester’s positioning error when he travels along the path is targeted to be analysed in this section, and the number of taken steps required to cover the same path is always varied in repeated walks, only one series of IMU reports collected in a trail walk along each path over the time domain is analysed

| Path index | # of turns | Distance [m] |
|------------|------------|--------------|
| 1 | 1 | 24.2 |
| 2 | 2 | 32.9 |
| 3 | 2 | 19.4 |
| 4 | 3 | 27.3 |

TABLE 5.3: Path summary

5.4.1 Analysis of the effect of different numbers of particles

If fewer particles are employed, the computational load of the smartphone is lower. However, the risk that all particles are killed in a stepping event increases. In case all particles are killed, re-initialisation of particles is required, which is absolutely opposite to the expectation of the system designer. Therefore, in this test, failure rate of tracking the user’s movement on the entire path is employed to evaluate how

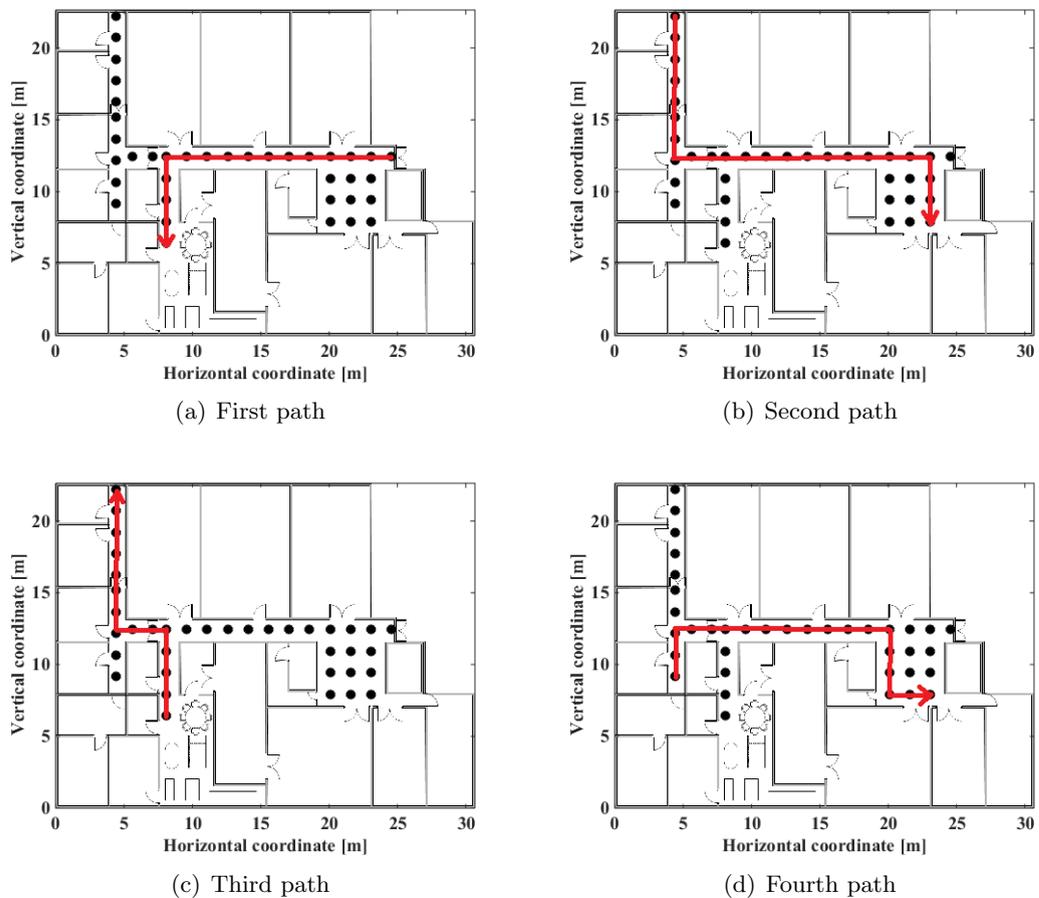


FIGURE 5.7: Four paths are evaluated specifically in the tests

many particles are required to be initialised. For this test, the tester walks along each path for 25 times, and as a result, 25 IMU reports have been collected on each of the route indicated in Figure 5.8. For each of the IMU reports, particles are initialised by the Wi-Fi localisation algorithm with different RSS measurements collected at the starting positions. Therefore, the evaluation result for each route is based on $25 \times 60 = 1500$ experiments in total. Also, in total of 10 RSS training samples on each RP are employed to build the RM.

From the figure, it can be found that the tracking failure rate generally decreases when an increasing number of particles is applied. Zero failure rate occurs when the number of applied particles achieves 240, 240, 200 and 160, respectively for the first, second, third and fourth path. Also, by observing table 5.3, a trend that the tracking failure rate is generally higher when the traversed distance of the whole path is longer, with the same number of particles being employed, could be found.

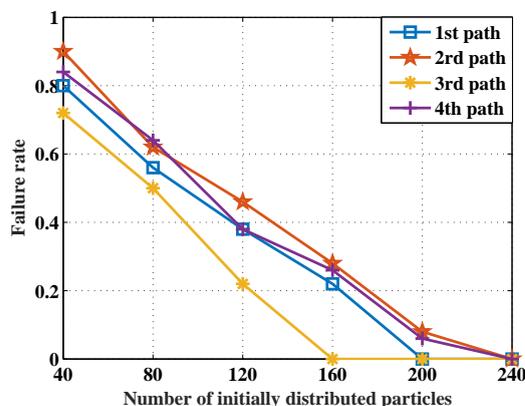


FIGURE 5.8: The failure rate of tracking the user on the entire path vs. number of employed particles

However, since the system has no idea how far is the user's expected destination from the origin position, the number of applied particles should meet the most strict requirement. In this floorplan, the most strict requirement is expected to be given by the second path with 2 turns and the longest traversed distance, as indicated in Figure 5.7(b). As a result, the decision of initialising the particle filter algorithm with 240 particles is made when the indoor localisation service is provided. Also, note that the experimental results given in Figure 5.8, and the different number of particles should be employed under different conditions (the number of turns along the path and the length of the path) could be taken advantage of as reference information for deploying such indoor pedestrian tracking system in other internal scenarios with various interior complexities and scales.

5.4.2 Analysis of the effect of applying Wi-Fi RSS clustering algorithm

As mentioned before, first, the clustering algorithm is employed to work on the Wi-Fi RSS receptions to determine which of the defined clusters is the user located in. In this section, the effect of the result of the Wi-Fi RSS clustering algorithm is evaluated. An example of an erroneous clustering result for the first path is shown in Figure 5.9. As indicated in this figure, particles are distributed initially in the open space, but not in the areas where the ground-truth starting position is located.

Firstly, let us evaluate the probability of correct clustering. Besides the cases in which only one RSS sample is used for rough localisation, the clustering algorithm is also evaluated by feeding set of two or more RSS samples to it. The details of the implementation are shown as follows.

In one particular test, the 60 RSS samples collected on the starting position of each route are separated randomly into 30 groups. Each group consisting of two RSS samples is applied for rough localisation. If the two RSS samples are localised into ranges of two different clusters, one more RSS sample is randomly picked from the remaining 58 ones and employed to obtain the final choice. The frequency of correct/incorrect clustering is counted. Such set of tests are repeated for 10 times to embrace more random RSS grouping results. Based on these tests, the overall correct clustering rate could be obtained. The related experiment results are given in Table 5.4, 5.5, 5.6 and 5.7, respectively. Note that the OVA-SVMs is applied in the rough localisation phase.

| # of initial RSS samples | Clustering accuracy | Averaged # of RSS samples |
|--------------------------|---------------------|---------------------------|
| 1 | 83.3% | 1 |
| 2 | 94.7% | 2.13 |

TABLE 5.4: The experiment results related to correctly clustered RSS samples, for the first path.

| # of initial RSS samples | Clustering accuracy | Averaged # of RSS samples |
|--------------------------|---------------------|---------------------------|
| 1 | 85.0% | 1 |
| 2 | 93.3% | 2.16 |

TABLE 5.5: The experiment results related to correctly clustered RSS samples, for the second path.

| # of initial RSS samples | Clustering accuracy | Averaged # of RSS samples |
|--------------------------|---------------------|---------------------------|
| 1 | 88.3% | 1 |
| 2 | 98.1% | 2.25 |

TABLE 5.6: The experiment results related to correctly clustered RSS samples, for the third path.

| # of initial RSS samples | Clustering accuracy | Averaged # of RSS samples |
|--------------------------|---------------------|---------------------------|
| 1 | 91.7% | 1 |
| 2 | 98.4% | 2.12 |

TABLE 5.7: The experiment results related to correctly clustered RSS samples, for the fourth path.

From the tables, a trend that clustering accuracy is improved when more on-line Wi-Fi RSS measurements collected at the starting position are applied could be observed. As a result, in the following implementation of the thesis, in order to provide satisfying rough localisation accuracy, the number of initial RSS receptions used for rough localisation is set as 2. In this way, the accuracy of rough localisation on the starting points of all paths is ensured to be higher than 93% at least. Also, note that 1 more RSS sample is required to be collected, provided that there are two clusters nominated by the two RSS samples, respectively, as mentioned before. Thus, in the tables, the averaged number of RSS samples for obtaining the clustering result is also listed. Among all, particle initialisation for the third path requires RSS samples at most: 2.25, which could be interpreted as: on average 3 on-line RSS samples are required in 25% of tests of particle initialisation for rough localisation on the starting point of the second path. Although applying more than one RSS sample for rough localisation takes more time, but in practice, at the beginning of the localisation service, a certain amount of time is also required to be consumed before the readings of IMUs become stable. In addition, taking several measurements continuously over time before providing an accurate initial location estimation is also a common approach of outdoor navigation applications available at the current market such as Google Map.

Secondly, the effect of the incorrect clustering of the RSS collected at starting position on the location tracking errors is evaluated on each ground-truth stepping position. Note that all of the groups of two RSS samples plus one potentially required RSS samples have been divided into two categories: being correctly and incorrectly clustered, from the previous test. In the current test, evaluation is carried out within the two categories, independently. Especially, initial distribution of the particles are designated by the fine localisation result. Detailed implementation of the fine localisation could be found in the next section. Following that, particle filter is repeatedly implemented with different initial particle distribution results. The mean positioning accuracy of the algorithm at the timestamp of each ground-truth steps taken by the tester are indicated in Figure 5.10.

From the figure, it can be found that, for all paths, the positioning errors are much higher when the particles are initially distributed into a wrong cluster. Thus, it

can be confirmed that positioning accuracy could be seriously distorted by an erroneous initial Wi-Fi RSS clustering result. As could be found from Figure 5.10(b), the positioning error even increases after the 15-th step has been taken on the path, after the initial Wi-Fi reception is wrongly clustered. To find the reasons leading to this situation, they are investigated. The fact that all particles are propagated to a corner of the floorplan, and the majority of them are killed by surrounding environment constraints is found. Due to the applied re-sampling scheme, new particles are generated by copying a few of the remaining surviving particles distributed in the same cell in every newly detected step event afterwards, and thus the estimated position and the ground-truth position becomes further and further away from each other when more steps are taken.

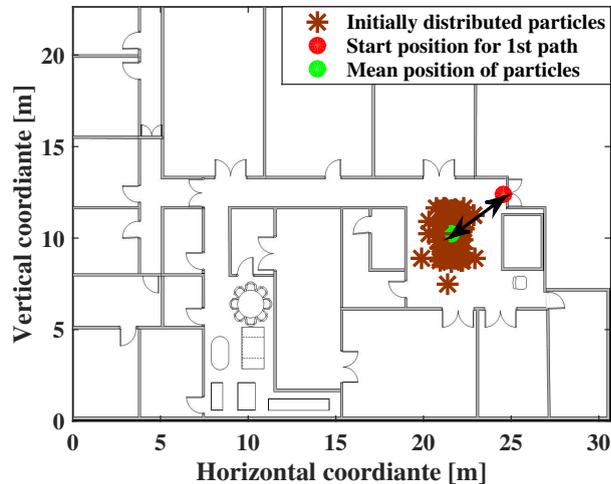


FIGURE 5.9: An example of particles being incorrectly clustered for the starting position of the first path

5.4.3 Analysis the effect of applying Wi-Fi RSS fine localisation algorithm

As mentioned before, in the fine localisation phase, the Wi-Fi RSS samples collected on the starting positions are used to determine how many particles should be distributed initially in each cell included in the targeted cluster. In Section 5.3.4, only the case of one Wi-Fi measurement being employed is discussed. However, as presented in the previous section, in the implementation of this thesis, two or three Wi-Fi measurements are required to be collected on the starting position

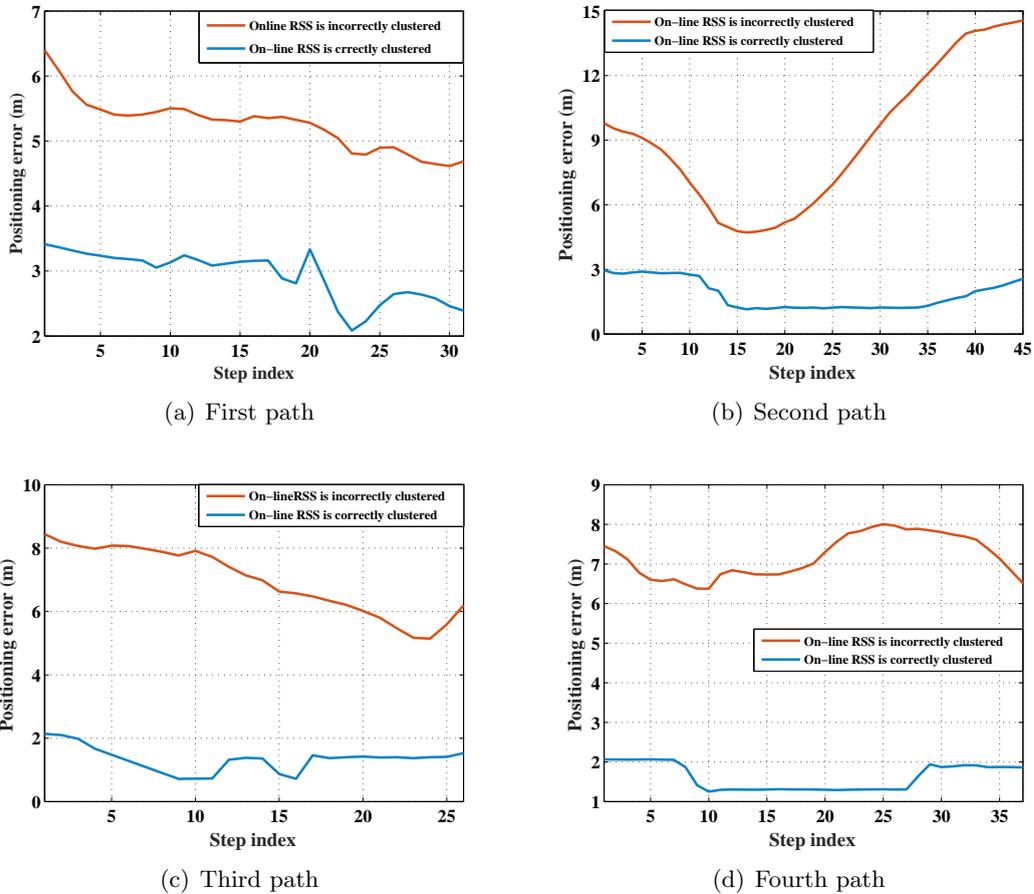


FIGURE 5.10: Comparison of positioning error of two scenarios: particles being correctly/incorrectly clustered.

when activating the location tracking service, which provides a way to overcome the time-varying wireless propagation effect: the RSS measurements which nominate the same cluster to be the final choice of the rough localisation are averaged and the result is fed to the probabilistic NB algorithm to obtain the fine localisation result. Accordingly, the number of particles distributed in each cell of the cluster is decided, as presented in Section 5.3.4. The Wi-Fi RSS measurement estimated to be collected outside the chosen cluster by the rough localisation is excluded from the fine localisation phase, since it is regarded as an outlier.

Figure 5.11 shows an example of a Wi-Fi fine localisation result with the Wi-Fi RSS measurements being clustered correctly. It can be found that the mean position of all particles is biased from the ground-truth starting position due to the erroneous fine localisation result in this particular example, which implies that many initially distributed particles are biased from the ground-truth starting location.

In this section, the target is to analyse if the performance of particle filtering is influenced by the fine localisation result of the averaged RSS measurements collected on the starting position. As demonstrated in Section 4.4.2, the accuracy of the fine localisation is generally improved when more RSS training samples are available to the system. In this experiment, the performance of the particle filter is evaluated by varying the number of training samples per RP in the range of [10, 50] with an increasing index of 10. The cases in which the sets of RSS measurements being clustered correctly (the same as what is applied in the previous section) have been analysed, and the cumulative distribution function (CDF) of the location tracking errors for the four paths are indicated in Figure 5.12. Note that for different numbers of RSS training samples being employed in the test, the corresponding optimal bin width, as presented in Table 4.1, is applied for discretisation of the applied Wi-Fi RSS samples.

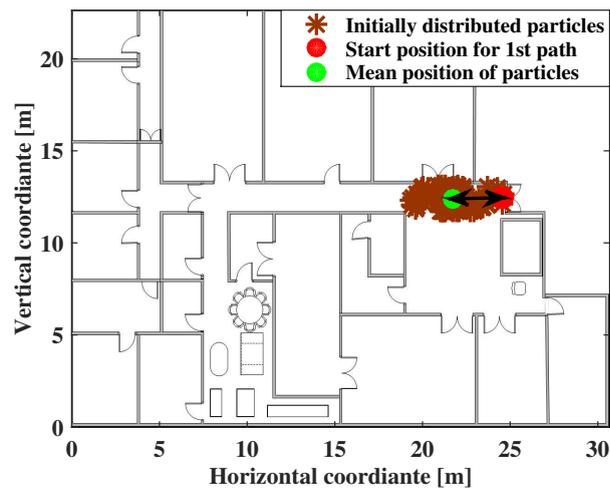


FIGURE 5.11: An example of particles being correctly clustered for the starting position of the first path

From the figures, a trend that the accuracy of location tracking results throughout the paths are generally higher, when the RM is built with more RSS training samples, could be observed. In particular, the performance gaps could be observed clearly, as indicated in the sub-figures showing the evaluation results of the tests implemented for the first, second and third path: The CDF of positioning errors drawn according to the different number of RSS training samples being employed are rarely over-lapped. However, it can also be found that for the fourth path, the

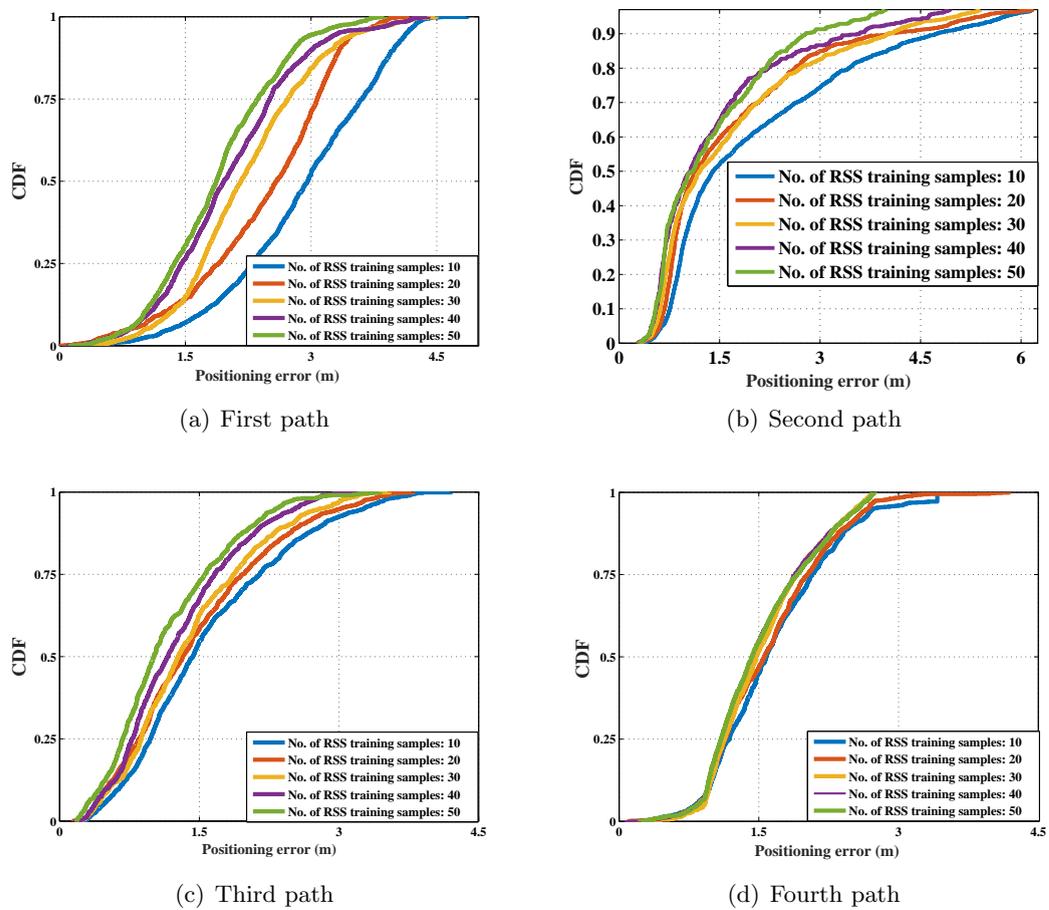


FIGURE 5.12: Performance of the localisation system with different number of RSS training samples being employed

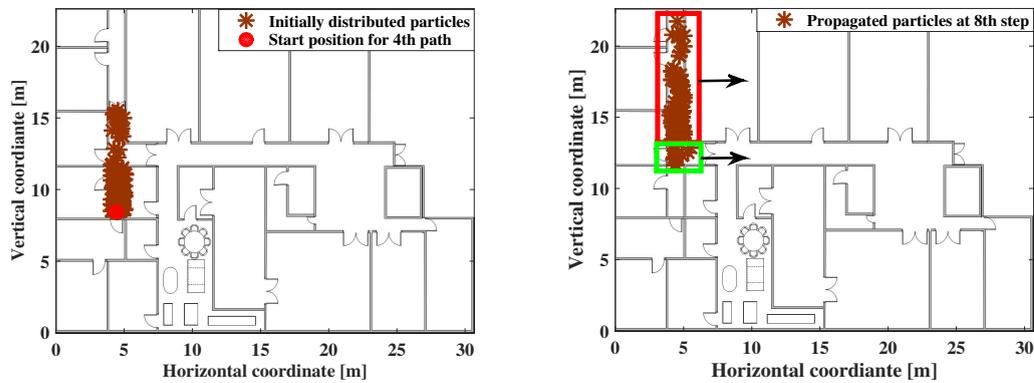
localisation performance is hardly impacted by the accuracy of the fine localisation result of the initial Wi-Fi RSS measurements: the CDF lines are overwhelmingly over-lapped with each other. The reason is presented as follows.

Among all of the first turning points of the paths, from Figure 5.7, it can be found that the first turning point of the fourth path is the closest to the starting position. Note that, initially, the particles are distributed on the RPs existing in the same corridor co-linearly, and the moving direction of a pedestrian user is parallel to the line determined by the RPs in the cluster before experiencing turning at the first turning point. Therefore, environmental enforcements being perpendicular to the moving direction of the pedestrian around the particles incorrectly propagated is found to provide a good opportunity to filter out the particles wrongly propagated. In the case that the initial particles are distributed from the starting position with

serious biases, the **closer** is the turning point and the starting position, the **earlier** in time that the propagations of them are filtered out. Such an example is given in Figure 5.13. Initially, particles are distributed in the same way as indicated in Figure 5.13(a): a result of erroneous localisation of initial Wi-Fi measurements. For this particular walk, both of the IMU module and a human observer record that the user turns around at the first corner along the path at the 8-th step. The distribution of propagated particles at the 8-th step is shown in Figure 5.13(b). In the figure, the black arrows indicate roughly the direction of the movement of the particles as horizontal in the next few steps reported by the IMU module. From the figure, it can be found that most of the particles which are inconsistent with the ground-truth state of the user (marked as the ones in the drawn red box) are expected to be filtered out by the surrounding vertical-standing enforcements shown on the floorplan shortly. However, the particles coinciding with the ground-truth state of the tester (in the range of the drawn green box) will survive and function as the samples for generating new particles in subsequent steps by the re-sampling scheme. In this way, the negative impact of erroneous fine localisation results of the initial Wi-Fi collections is cancelled out.

Overall, it is confirmed that though the pedestrian tracking accuracy along the paths is influenced by the fine localisation results of the initially collected on-line Wi-Fi RSS measurements, the impact is alleviated when the first turning point along the path is close to the starting position of the user. However, when the first turning point is relatively far from the starting position, when erroneous fine localisations of initially collected RSS measurements are made, biased location tracking result will last for a while until users pass through the first turning point. As a result, a robust RM is confirmed to be important to the indoor localisation service.

Note that in some cases, experiencing turns may result in multi-clustered distribution of particles. As shown before, in these cases, the location estimation error is expected to be lifted significantly. In the next section, the effect of on-line adjustment of importance weights of particles opportunistically on overcoming the problem will be analysed.



(a) Initially distributed particles for the fourth path (b) After passing the first turning point, the particles that are inconsistent with the true state of user will be filtered shortly

FIGURE 5.13: An example of biased distributed particles being filtered out after first turning point for the fourth path.

5.4.4 Analysis of the effect of adjustment of importance weights of particles by applying the on-line W-Fi localisation results after significant user's turns are detected

As mentioned in Section 5.3.2 and Figure 5.6, a phenomenon that the particles are multi-clustered distributed may be found after users take turns on the paths. Also, as shown in Figure 5.13(b), the distribution of propagated particles may have been seriously biased from the ground-truth state before the user experiences orientation turnings along the paths. Here, in this section, the purpose is to verify if opportunistic adjustment of importance weights of particles by applying the Wi-Fi on-line localisation results could aid the system to improve the localisation accuracy.

In the experiment, the timestamps of the **first** Wi-Fi RSS measurements being collected just after the successful detections of the significant orientation changes of the tester's smartphone (applied instead, as mentioned before) along the routes are recorded down. Accordingly, the collecting positions of the first RSS measurements are recognised. Then, 60 RSS measurements are collected on each of these positions and simulated as one of the possible on-line collected RSS measurements after the detection of the orientation change to adjust the importance weights of particles. Note that the adjustments of the importance weights of the particles are

triggered when such first Wi-Fi RSS measurement is collected. Here, a significant turn of the yaw angle of the smartphone is defined as: either the orientation difference of the held phone between the previous step and the current one is larger than 75 degree or the incremental turning angle of two consecutive steps is larger than 75 degree.

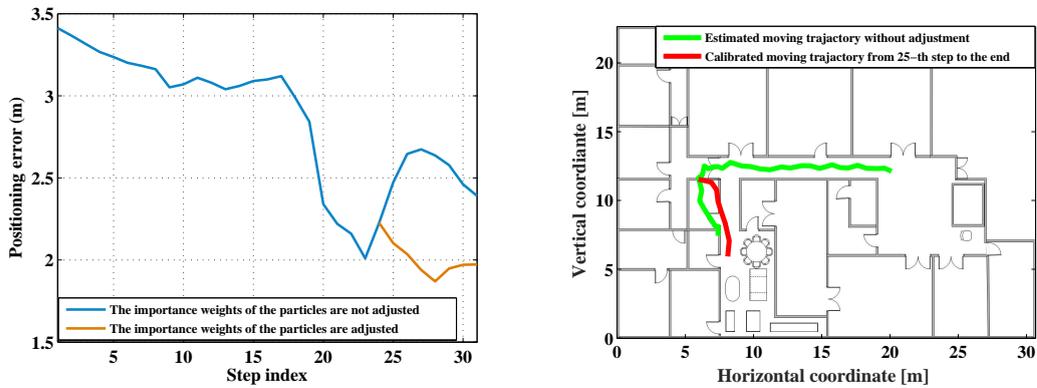
5.4.4.1 The first path

In the case of travelling on the first path, the tester is observed by a technician that he turns his body around at the 23-rd step. Also, the first Wi-Fi on-line measurement after the turn is obtained just before the occurrence of the 25-th step. Note that the 20-th step taken on the ground by the tester is missed.

Figure 5.14(a) shows the mean positioning error of the particle filter at the timestamp of each ground-truth step taken by the tester. Although the wall constraints could filter some of the incorrectly propagated particles, after the tester experiences the turn at the 20-th step, from Figure 5.14(a), it can still be found that accuracy of the localisation service becomes worse due to the multi-clustered phenomenon, as indicated in Figure 5.6(b). From both of Figure 5.14(a) and 5.14(b), it can also be found that Wi-Fi on-line adjustment could help the localisation system overcome the problem that the localisation accuracy is distorted by the multi-clustered phenomenon occurring in this case: the biased estimated trajectory (marked as green line) has been calibrated (marked as red line) after the 25-th step has been taken, and as a result the positioning error is reduced.

5.4.4.2 The second path

In the case of the second path, the tester turns along the path at the ground-truth 14-th and 40-th step, respectively. The corresponding on-line Wi-Fi measurements are collected just before the ground-truth 16th and 42-th step. Note that, according to the analysis of the IMU reports, the 12-th step taken on the ground by the tester is missed. The evaluation results are indicated in Figure 5.15.



(a) Comparison of positioning errors when impor- (b) An example of tester's trajectory with/without
tance weights are/are not adjusted for the first path. on-line importance weights adjustment being ap-
plied.

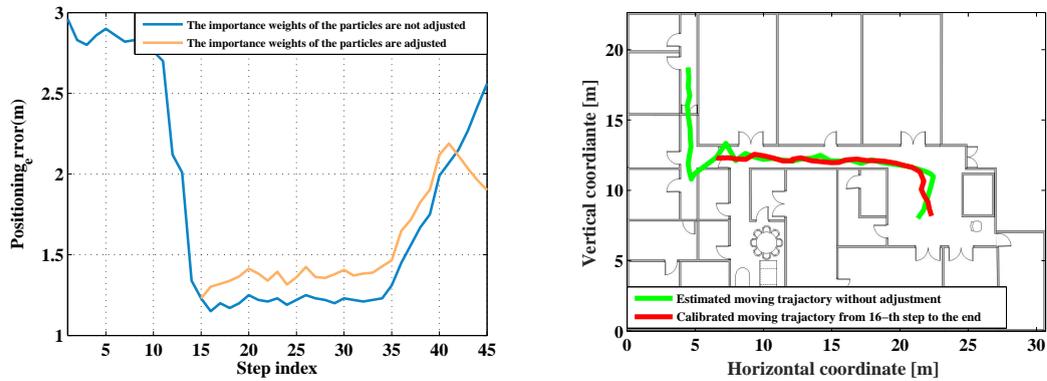
FIGURE 5.14: Evaluation results of on-line importance weights adjustment for the first path.

In Figure 5.15(a), it can be found that the positioning errors of the 42-th to 45-th steps have been decreased when the Wi-Fi on-line adjustment is applied. However, it could be also found that the adjustment does not reduce but improve the positioning error during the time when the tester takes the 14-th to the 41-th steps. One example of a trajectory tracking result is indicated in Figure 5.15(b), which also helps to confirm that only the positioning results after the 41-th step are properly calibrated by the on-line Wi-Fi localisation result. The reason is found to be that the influence of Wi-Fi on-line localisation is reduced, provided that the surrounding environmental enforcements are properly capable of calibrating the particles, such as the ones around the 14-th steps.

In Figure 5.15(b), in this example, it can be found that the estimation of the starting point of the tester is biased from his ground-truth starting location, since initially many particles are distributed in the wrong cells (but in the same cluster). However, some of these wrongly distributed particles could be filtered out before the tester reaches the first turning point of the path, due to the existence of the walls at the end of the vertical corridor. Moreover, majority of the remaining particles are killed, according to the movement of the 14-th and 15-th step. An example is shown in Figure 5.16(a): only a few of particles are left in the vertical corridor, and most of the particles are located in close proximity to the ground-truth location of the tester moving with the 16-th step, when the first Wi-Fi

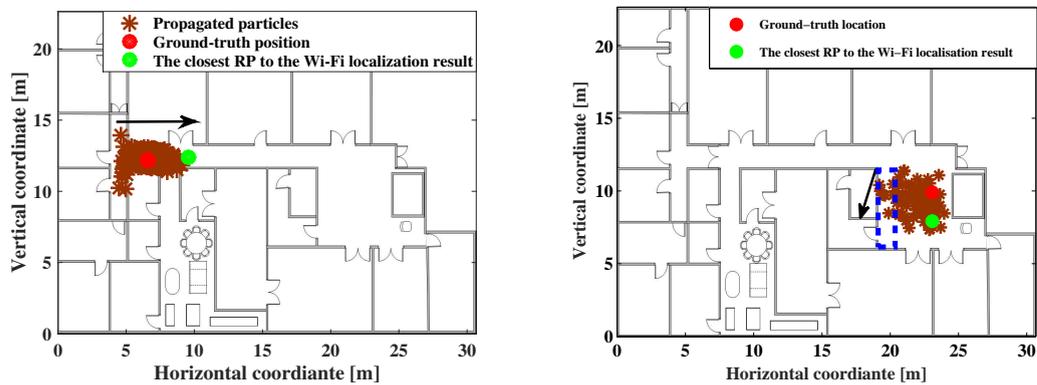
RSS measurement after the turn is just collected. But, in this example, since the Wi-Fi RSS localisation result collected at the 16-th step are right-biased from the ground-truth tester's position at the time, the adjustment of the importance weights of the particles by applying Wi-Fi on-line localisation result will lead to a right-hand skewed net localisation result, and therefore biased location tracking results throughout the walk in the long horizontal corridor.

Due to the uncertainties in the estimation of step length have been considered also, the geometric distribution of the particles is horizontally stretched (one purpose is for diverse distribution of the particles) when the tester goes through the horizontal corridor before turning around at the second turning point along the path. After the tester turns around at the second turning point, the particles are approximately uniformly distributed in the open space, as shown in Figure 5.16(b), which of course implies an erroneous localisation result if Wi-Fi on-line adjustment is not applied. Note that for demonstration purpose only, arrows are drawn in Figure 5.16 to indicate the rough real-time moving directions of the particles (without the modelled uncertainty being added to). In the case shown in Figure 5.16(b), the importance weights of particles are adjusted properly by taking advantage of the Wi-Fi on-line localisation result, marked as green point in the figure: the importance weights of the particles in close proximity to the lower right corner of the open space are improved while the importance weights of the particles far from it are reduced. However, when the on-line Wi-Fi localisation result is in the range of the dashed blue box drawn in Figure 5.16, the on-line localisation accuracy becomes worse, if the importance weights of the particles are adjusted by the Wi-Fi on-line localisation result. Related experiment is carried on and it shows that only 13.7% of the on-line localisation results of the RSS measurements collected on the ground-truth location of the 42-th step will be in the range of the drawn dashed blue box. Therefore, on average, applying on-line Wi-Fi localisation result to adjust the importance weights of the particles is confirmed to be capable of reducing the on-line localisation error of the second turn and the steps taken afterwards along the second path.



(a) Comparison of positioning errors when impor- (b) An example of tester's trajectory with/without
tance weights are/are not adjusted for the second on-line importance weights adjustment being ap-
plied.

FIGURE 5.15: Evaluation results of on-line importance weights adjustment for the second path.



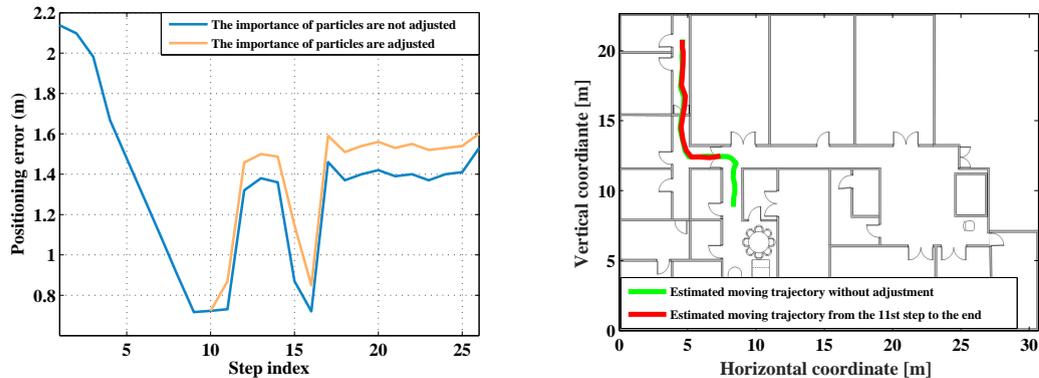
(a) Distribution of propagated particles at the 16- (b) Distribution of propagated particles at the 41-
th step of the tester.

FIGURE 5.16: Distribution of the particles propagated along the second path.

5.4.4.3 The third path

The indices of the ground-truth turning steps in the test walk of the third path are the 10-th and 15-th steps. Analysis of the applied IMU report of the tester travelling along the path shows that the 12-rd and 17-th step taken by the tester on the ground are missed. On the walk, the firstly sensed Wi-Fi on-line RSS measurements after the detected turns occurs at the time just before the 11-st and 16-th steps. The evaluation results of the test walk for the third path are indicated in Figure 5.17. From Figure 5.17(a), it can be found that the mean positioning errors after the 11-th step are slightly higher when Wi-Fi on-line localisation result

is applied for adjustment of the importance weights of the particles. The reason is found that the surrounding environmental enforcements of the steps from the 1-st to 15-th are well effective in filtering the biased propagated particles in this case. Taking advantage of the Wi-Fi on-line localisation result to adjust the importance weights in this case is highly likely to distort the net localisation result.



(a) Comparison of positioning errors when impor- (b) An example of the tester's trajectory with-
tance weights are/are not adjusted for the third /without on-line importance weights adjustment
path. being applied for the third path.

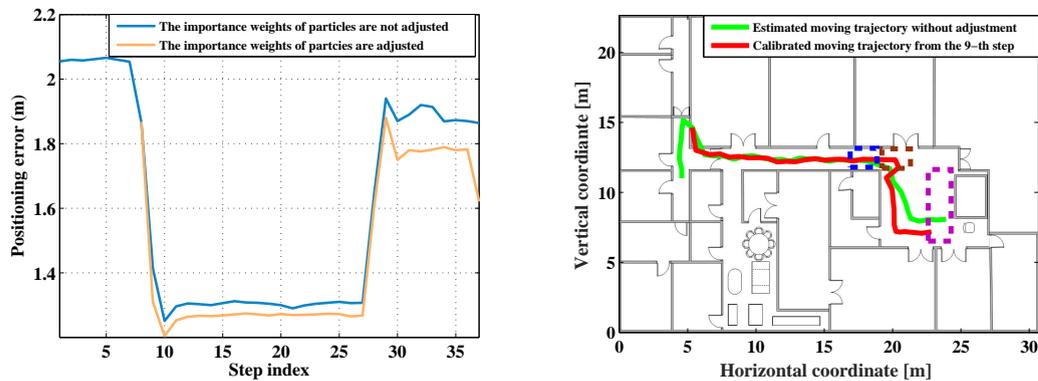
FIGURE 5.17: Evaluation results of on-line importance weights adjustment for the third path.

5.4.4.4 The fourth path

Finally, let us move on to the final path. Similar to the previous tests, the mean positioning error along a test walk for the fourth path is obtained by applying different particle initialisation results, and the experimental evaluation result is indicated in Figure 5.18(a).

For the example walk, the turns occurs at the ground-truth 8-th, 28-th and 34-th steps, respectively. Also, the on-line Wi-Fi RSS measurement for importance weight adjustment for the first user's body turning is collected just before the 9-th step taken by the tester. For the other two turnings, the on-line Wi-Fi RSS measurements for importance weight adjustment are collected just before the 30-th and 36-th step. The walls surrounding the ground-truth position of the second turn filter the particles in the range of the blue box in Figure 5.18 but leave the particles in the range of the brown box survive. As a result, the on-line localisation result is shifted to the right significantly. Also, the localisation

accuracy drops a lot, as can be found in Figure 5.18, but then improves again after the on-line Wi-Fi RSS measurement collected just before the 30-th step is applied for adjustment of importance weights. Note that, similar to the second orientation turning of the second path, importance weights might be adjusted in a wrong way leading to distorted localisation accuracy, when the applied on-line collected Wi-Fi measurement is localised in the range of the drawn dashed purple box. The on-line Wi-Fi localisation results of the RSS measurements collected at the position shows that only 26.6% of them are localised into that range. As a result, on average, adjustment of the importance weights of the particles by taking advantage of on-line Wi-Fi measurements is expected to improve the accuracy of the localisation service for the second turn and the steps taken afterwards.



(a) Comparison of positioning errors when impor- (b) An example of the tester's trajectory with-
tance weights are/are not adjusted for the fourth /without on-line importance weights adjustment
path. being applied for the fourth path.

FIGURE 5.18: Evaluation results of on-line importance weights adjustment for the fourth path.

5.5 Chapter summary

This chapter consists of three parts. The first part presents the method of digitalisation the floorplan, including representing walls in vector form, presenting the spacial distribution of the RPs in a mathematical way using an adjacency matrix, and thereby recognising the inter-RP distances. With the knowledge of wall distribution in vector form and of the adjacency matrix of RPs, automatic arrangement of cells and recognition of their boundaries could be achieved for each RP.

The second part shows the details of particle filter implementation for indoor pedestrian location estimation. Firstly, particles are initialised into cells in the same cluster according to the localisation result of the Wi-Fi on-line RSS measurement at the starting position. Then, particles are real-time propagated in every detected step event by taking advantage of reports from the IMU module. Finally, particle propagations are corrected by taking advantage of surrounding environmental enforcements, and also importance weights of the particles are adjusted opportunistically by applying on-line Wi-Fi localisation results when significant changes of users' smartphones orientation are detected.

In the experimental evaluation part, it has been shown that the performance of the proposed indoor pedestrian localisation and tracking system is impacted by several factors, and also several conclusions are drawn.

Firstly, employing fewer particles in the particle filter implementation will increase the probability that all particles are killed before the end of the tester's walk. On the other hand, an overwhelming large number of particles being employed will drain the computational resource. Bearing this in mind, a trade-off is made: the number of employed particles is chosen to be 240.

Secondly, it is confirmed that particle initialisations resulting from incorrect clustering of initial Wi-Fi RSS measurements will lead to distorted location tracking results. To avoid occurrences of such situations, the thesis proposed to use a set of two or three initial Wi-Fi RSS fingerprints received by the user's phone for rough localisation in order to increase the probability of correct clustering.

Thirdly, the impact of the fine localisation result of the initial Wi-Fi RSS fingerprints to the proposed pedestrian localisation and tracking system is discussed. Specifically, the accuracy of the fine localisation result is influenced by the scale of the training data: generally, the accuracy is improved when more RSS fingerprints are used to build the RM, which coincides with the results of the experiments carried in Chapter 4. The result shows that the localisation accuracy of the system benefits from more accurate fine localisation result of the initial Wi-Fi RSS fingerprints. In the next chapter, the method of taking advantage of the user's historical

on-line Wi-Fi receptions to calibrate the RM for a more accurate fine localisation result of the Wi-Fi RSS fingerprints is discussed.

Finally, the effect of the proposed approach that importance weights of the employed particles are adjusted opportunistically by taking advantage of the on-line Wi-Fi localisation results, when significant orientation changes of users' phones are detected, is discussed. The results show that the effect is obviously positive when the surrounding environmental constraints around some particular turning points along the path perform poorly in preventing the particles from propagating into wrong positions. On the other hand, the effect is not obvious or even slightly negative on the performance of the localisation system, provided that the environmental constraints perform well. Also, more accurate fine localisation of on-line RSS measurements for adjustment of importance weights could improve the performance of the scheme. Nevertheless, the proposed approach should be applied in the localisation system, since the localisation accuracy of the system is improved significantly in the cases that the 'multi-clustered' phenomenons emerge and the posterior belief of users is hardly represented by the employed particle. Under such situations, the importance weights of the particles are required to be adjusted.

Chapter 6

Using users' on-line measurements to calibrate the RM

Building the RM database by collecting Wi-Fi RSS measurements manually on each RP is time-consuming and labour-intensive, especially when dozens of RSS measurements are required to be collected on each RP to ensure that high location estimation accuracies are achieved by the conventional RSS-based indoor localisation systems. It has been proved before that the performance of the proposed localisation approach is improved when more RSS training data has been collected [1, 55]. Also, rather than a location tracking service, some of the customers only require the localisation system to provide an accurate instant static wireless localisation result. Under these situations, a robust RM could aid a lot.

Some researchers proposed using an empirical indoor wireless propagation model to predict RSS fingerprinting values [25] or alternatively a hybrid approach combining a few sampled fingerprints, a mathematical propagation model and a data fitting algorithm [97]. However, most of these approaches require the knowledge of building materials, structures, the location of APs, etc, which, in most scenarios, is not available to the developers of the indoor pedestrian localisation systems. Recently, there exists active research on building the RM only with the fingerprints collected with the crowd-sourcing paradigm [50]. This approach takes advantage

of random traces of RSS measurements collected by people carrying smartphones as they walk around the area of interest during their daily routines. But obtaining a sufficient number of traces covering the whole indoor environment may take quite a long time. Therefore, applying the crowd-sourcing approach to construct a reliable and accurate RM is considered to be time-consuming, and challenging [32].

An alternative choice is to collect a limited number of geo-tagged Wi-Fi RSS fingerprints on each RP to initialise the RM, which takes a much shorter time than the conventional approach. Over time, the RM could be updated and maintained with crowd-sourced and non-geo-tagged on-line users' RSS collections to provide a more accurate localisation service. In [1], an EM algorithm functioning as an iterative algorithm to find the maximum *a posteriori* estimation of parameters in statistical models with a set of incomplete data [98] is applied for this aim. Note that the on-line users' RSS collections are regarded as incomplete data, due to their non-geo-tagged (unlabelled) nature. As mentioned in Section 2.4.5, the probabilistic NB algorithm is a generative approach. The generative model used in the probabilistic NB algorithm is calibrated by the EM algorithm. In this chapter, another method in which particle filter implementation results are back-tracked to calibrate the RM is proposed. It will be shown that this method could take a shorter time than the conventional EM algorithm but still provide more robust RM calibration results: better calibration performance is proven to be exhibited by the proposed method. Also, this chapter presents that the location estimation results of a hybrid generative/discriminative approach could benefit more from applying the calibrated RM with the proposed method.

Finally, the so called semi-automatic site-survey approach is proposed. It is designed to relieve the technicians with the site-survey task of mechanical repetitive operations on the recording device by taking advantage of IMU sensors and route planning algorithm to improve working efficiency, and initialise a more reliable RM. Details of the implementation will be explained in Section 6.3.

6.1 Implementation of the EM algorithm

With the probabilistic approach, localisation could be achieved by deriving the probability of each RP being the place on which the given on-line RSS measurement is collected, as indicated in equation (2.4). Since mutual independence between different features is assumed by the NB algorithm, $Pr(\boldsymbol{\psi}(\mathbf{o})|\mathbf{p}_j)$ could be factored in the way indicated in equation (2.5). Also, noting that after discretisation, $Pr(sl_i = \psi(o_i)|\mathbf{p}_j) = Pr(sl_i \in B_\xi|\mathbf{p}_j)$. Thus, it is easy to know that the key parameters for location estimation task consist of $Pr(sl_i \in B_\xi|\mathbf{p}_j)$ and $Pr(\mathbf{p}_j)$, which is denoted as:

$$\boldsymbol{\varepsilon} = \{Pr(\mathbf{p}_j), Pr(sl_i \in B_\xi|\mathbf{p}_j)\} \quad (6.1)$$

where $j \in \{1, 2, \dots, J\}$, $i \in \{1, 2, \dots, I\}$ and $\xi \in \{1, 2, \dots, N_{bin}\}$. As mentioned before, J, I, N_{bin} represents the total number of RPs, APs and discrete bins, respectively. Although limited number of labelled training data may lead to biased parameter estimation results, the EM algorithm is capable of calibrating both parameters by taking advantage of unlabelled users' on-line data.

The method of applying the EM algorithm is presented as follows. Firstly, $\boldsymbol{\varepsilon}$ is estimated from labelled RSS training data, which is also regarded as the process of building the RM from the labelled training data. Secondly, the NB algorithm is used to assign a probabilistically-weighted class label, $Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o}))$, for each unlabelled user's on-line RSS data $\boldsymbol{\psi}(\mathbf{o})$. Next, a new $\boldsymbol{\varepsilon}$ is estimated using all RSS data—both the originally collected RSS training measurements and newly labelled on-line RSS measurements. The second and third steps are iterated until $\boldsymbol{\varepsilon}$ converges. As demonstrated by [98], at each iteration, the algorithm is guaranteed to find parameters with an equal or increased marginal likelihood function.

Given a bundle of N_u unlabelled on-line RSS samples ($\boldsymbol{\psi}(\mathbf{o}^n) \in \boldsymbol{\psi}(\mathbf{O}_u), n \in \{1, 2, \dots, N_u\}$), the joint probability of them is expressed as:

$$Pr(\boldsymbol{\psi}(\mathbf{O}_u); \boldsymbol{\varepsilon}) = \prod_{n=1}^{N_u} Pr(\boldsymbol{\psi}(\mathbf{o}^n); \boldsymbol{\varepsilon}) = \prod_{n=1}^{N_u} \sum_{\forall j \in J} Pr(\mathbf{p}_j; \boldsymbol{\varepsilon}) Pr(\boldsymbol{\psi}(\mathbf{o}^n)|\mathbf{p}_j; \boldsymbol{\varepsilon}) \quad (6.2)$$

Similarly, given a bundle of N_{coll} geo-labelled training RSS samples ($\boldsymbol{\psi}(\boldsymbol{o}^n) \in \boldsymbol{\psi}(\boldsymbol{O}_l), n \in \{1, 2, \dots, N_{coll}\}$), the joint probability of them is expressed as:

$$Pr(\boldsymbol{\psi}(\boldsymbol{O}_l); \boldsymbol{\varepsilon}) = \prod_{n=1}^{N_{coll}} Pr(\boldsymbol{\psi}(\boldsymbol{o}^n); \boldsymbol{\varepsilon}) = \prod_{n=1}^{N_{coll}} Pr(\boldsymbol{p}_{\boldsymbol{\psi}(\boldsymbol{o}^n)}; \boldsymbol{\varepsilon}) Pr(\boldsymbol{\psi}(\boldsymbol{o}^n) | \boldsymbol{p}_{\boldsymbol{\psi}(\boldsymbol{o}^n)}; \boldsymbol{\varepsilon}) \quad (6.3)$$

Here, $\boldsymbol{p}_{\boldsymbol{\psi}(\boldsymbol{o}^n)}$ denotes the one and only one place where the training RSS measurement $\boldsymbol{\psi}(\boldsymbol{o}^n)$ is collected. The difference between the factored results of equation (6.2) and (6.3) could be found as simply whether or not all RPs should be accounted for in calculating $Pr(\boldsymbol{\psi}(\boldsymbol{o}^n); \boldsymbol{\varepsilon})$. This is due to the inherited nature of geo-tagged/non-geo-tagged data. Since for non-geo-tagged RSS data the collection place is unknown, every RP in the targeted area should be considered as a candidate, but this is not so for the cases of geo-tagged RSS measurements.

The logarithm function of the joint probability of both geo-tagged and non-geo-tagged data could be expressed as:

$$\begin{aligned} \log(Pr(\boldsymbol{\psi}(\boldsymbol{O}_l) \cap \boldsymbol{\psi}(\boldsymbol{O}_u); \boldsymbol{\varepsilon})) &= \log(Pr(\boldsymbol{\psi}(\boldsymbol{O}_l); \boldsymbol{\varepsilon}) \bullet Pr(\boldsymbol{\psi}(\boldsymbol{O}_u); \boldsymbol{\varepsilon})) = \log(Pr(\boldsymbol{\psi}(\boldsymbol{O}_l); \boldsymbol{\varepsilon})) + \log(Pr(\boldsymbol{\psi}(\boldsymbol{O}_u); \boldsymbol{\varepsilon})) \\ &= \log\left(\prod_{n=1}^{N_u} \sum_{\forall j \in J} Pr(\boldsymbol{p}_j; \boldsymbol{\varepsilon}) Pr(\boldsymbol{\psi}(\boldsymbol{o}^n) | \boldsymbol{p}_j; \boldsymbol{\varepsilon})\right) + \log\left(\prod_{n=1}^{N_{coll}} Pr(\boldsymbol{p}_{\boldsymbol{\psi}(\boldsymbol{o}^n)}; \boldsymbol{\varepsilon}) Pr(\boldsymbol{\psi}(\boldsymbol{o}^n) | \boldsymbol{p}_{\boldsymbol{\psi}(\boldsymbol{o}^n)}; \boldsymbol{\varepsilon})\right) \\ &= \sum_{n=1}^{N_u} \log\left(\sum_{\forall j \in J} Pr(\boldsymbol{p}_j; \boldsymbol{\varepsilon}) Pr(\boldsymbol{\psi}(\boldsymbol{o}^n) | \boldsymbol{p}_j; \boldsymbol{\varepsilon})\right) + \sum_{n=1}^{N_{coll}} \log(Pr(\boldsymbol{p}_{\boldsymbol{\psi}(\boldsymbol{o}^n)}; \boldsymbol{\varepsilon}) Pr(\boldsymbol{\psi}(\boldsymbol{o}^n) | \boldsymbol{p}_{\boldsymbol{\psi}(\boldsymbol{o}^n)}; \boldsymbol{\varepsilon})) \end{aligned} \quad (6.4)$$

Here, note that for the above derivation, the total probability theorem is also applied: given J mutually exclusive events A_1, A_2, \dots, A_J whose probabilities sum to unity, suppose B is an arbitrary event, then $Pr(B)$ could be factored in such a way:

$$Pr(B) = \sum_{j=1}^N Pr(B \cap A_j) = \sum_{j=1}^N Pr(B | A_j) Pr(A_j) \quad (6.5)$$

Taking the derivative of a logarithm function of a variable is a common way to obtain the maximum likelihood (ML) estimation of it. However, this equation contains a log of sums for the users' unlabelled data, which makes finding the maximum value by taking partial derivatives of it computationally intractable. Instead, the EM could be applied to avoid such complicated computation. Parameters are updated by the EM iteratively, involving two steps named E-step and M-step, respectively.

In E-step, probabilistic labels $Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o}^n); \boldsymbol{\varepsilon})$ are estimated for the users' collected on-line unlabelled RSS data by using the current estimate of the parameters $\hat{\boldsymbol{\varepsilon}}$ and the NB algorithm, as shown in equation (2.4). The M-step aims at updating the parameters, $\hat{\boldsymbol{\varepsilon}}$, using the current estimates of $Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o}^n); \boldsymbol{\varepsilon})$, with the equations shown below:

$$Pr(\mathbf{p}_j; \boldsymbol{\varepsilon}) = \frac{1 + \sum_{n=1}^{N_{coll}+N_u} Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o}^n); \boldsymbol{\varepsilon})}{N_{coll} + N_u} \quad (6.6)$$

and

$$Pr(sl_i \in B_\xi | \mathbf{p}_j; \boldsymbol{\varepsilon}) = \frac{\sum_{n=1}^{N_{coll}+N_u} \delta_D(sl_i \in B_\xi | \boldsymbol{\psi}(\mathbf{o}^n)) Pr(\mathbf{p}_j | \boldsymbol{\psi}(\mathbf{o}^n); \boldsymbol{\varepsilon}) + 1}{\sum_{\xi=1}^{N_{bin}} \sum_{n=1}^{N_{coll}+N_u} \delta_D(sl_i \in B_\xi | \boldsymbol{\psi}(\mathbf{o}^n)) Pr(\mathbf{p}_j | \boldsymbol{\psi}(\mathbf{o}^n); \boldsymbol{\varepsilon}) + N_{bin}} \quad (6.7)$$

Firstly, the iteration process is initialised with an advanced M-step, where only the geo-tagged RSS training measurements are used to estimate the parameters, according to equations (2.8) and (6.8). Then, each iteration starts with an E-step using the estimated parameters to label the users' on-line RSS collections, and is followed by another M-step.

$$Pr(\mathbf{p}_j; \boldsymbol{\varepsilon}) = \frac{1 + \sum_{n=1}^{N_{coll}} Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o}^n); \boldsymbol{\varepsilon})}{N_{coll}} \quad (6.8)$$

The cycles are not terminated until the algorithm converges to a point when the gradient of the change of the value of $Pr(\boldsymbol{\psi}(\mathbf{O}_u); \boldsymbol{\varepsilon})$ in equation (6.4) is lower than a pre-determined threshold.

6.2 Calibrating the RM by taking advantage of the profiles of final surviving particles

The flow chart of the RM calibration by taking advantage of final surviving particles from the particle filter implementation is indicated in Figure 6.1. As shown in the figure, the key inputs for the server to implement the RM calibration in an off-line analysis phase are: the time-stamped on-line collected Wi-Fi RSS measurements, the historical positions of particles of all detected step events, the importance weights of final surviving particles and 'reflection tables'. Note that the server could be installed anywhere. The only one requirement is that users' smartphone could communicate with the server via wireless communication techniques, such as Wi-Fi or the current mobile communication standards. The IP addresses of the smartphones and the server should be included in the data frame for the convenience of the routers on the road to forward the information to the following intermediate points or the destination correctly. In this way, it is easy for the operator to maintain the servers, since the servers could be just installed in the operator's office area, but not in the buildings being served with such localisation system.

The details of usage of the RSS samples collected in on-line phase will be presented in Section 6.2.2. All the information is recorded in real-time fashion, and analysed by the server. Among them, 'reflection table' is applied to figure out clearly the inheritance relationship of particles involved in neighbouring step events. It will be explained in Section 6.2.1 in detail. After a pre-determined number of RSS measurements have been stored in the server, the process of RM calibration is activated.

6.2.1 Recording the particle evolving information in real-time

As mentioned in the previous chapter, particles are initialised, propagated, and then corrected in every newly detected step event. While some particles are killed in the process, other particles may be copied for once or several times in the re-sampling phase of any detected step event. A reasonable assumption is that some

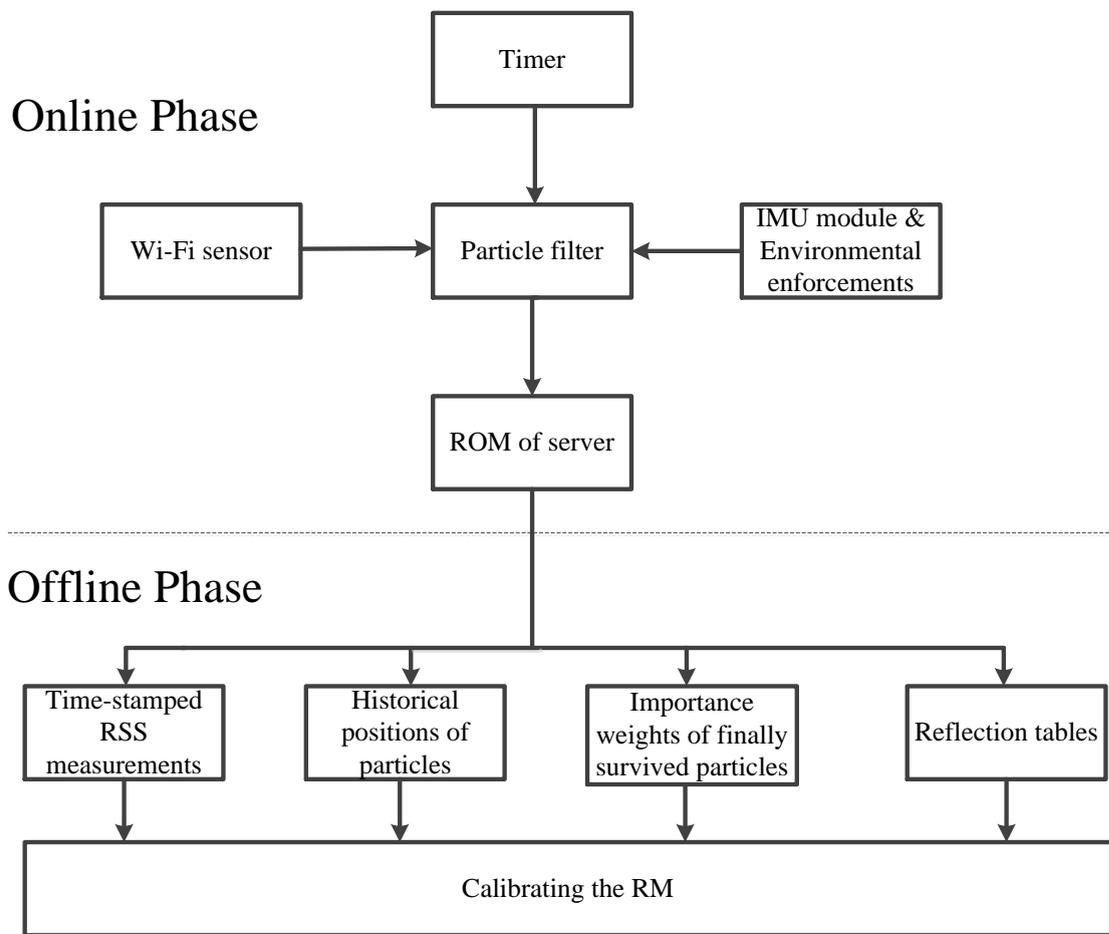


FIGURE 6.1: Flow chart of RM calibration with particle filter implementation

or all final surviving particles are the descendants of the particles generated in the initialisation phase. Then, to track the ancestor particles of the final surviving particles successfully in the off-line phase, the inheritance relationship of these particles should be recorded clearly in the on-line phase when implementing the particle filter .

In particular, the so called ‘reflection table’ figuring out the inheritance relationship of the particles are applied for this aim. An example showing the details of the implementation is indicated in Figure 6.2 for demonstration purpose only. In this example, for the purpose of making the presentation clear, only 16 particles are assumed to be involved in particle filter implementation in every step event experienced. Firstly, the particles propagated from the previous step to the current step or from the user’s initial position to the first step are indexed in order.

Secondly, some of the particles are filtered out, since the movements of them are detected to have collided with environmental constraints. In this example, only the particles originally indexed as '2', '4', '5', '8', '12', '13' and '14' survived and are stored in the memory in order. Thirdly, re-sampling is achieved by making copies of the surviving particles according to their importance weights, as explained in the previous chapter. After deleting the indices of the filtered particles, the indices of the surviving particles copied in the re-sampling phase are filled into the empty memory slots one-by-one. Finally, the indexing result is stored in the reflection table of this particular step event. When a new step event is detected, the same process repeats again. Noting that in the beginning of any new loop, the surviving particles and their copies inherited from the previous loop are re-indexed from 1 to 16 with an increasing step of 1. To achieve the purpose of back-tracking the particles, geometric positions of all resulting particles given by the particle filter implementation of every step event are required to be recorded as well. In contrast, only the importance weights of the surviving particles from the particle filter implementation of the last step event are needed by the system. The importance weights of the surviving particles of all other previously detected step events are not required.

Here, the definition of the reflection table is given: the index of the ancestor of the m -th particle in the k -th step event is given as $RT_k(m)$ in the $(k - 1)$ -th step event.

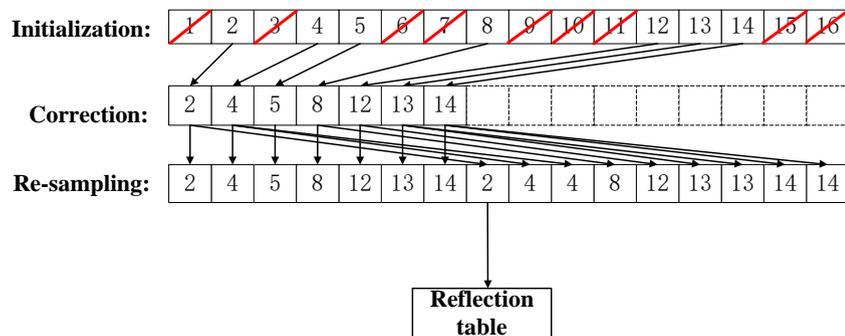


FIGURE 6.2: An example of particle indexing within the reflection table

6.2.2 Back-tracking the final surviving particles, and calibrating the RM

With the knowledges such as the reflection tables of the detected step events along the users' walking path, the positions of particles involved in the detected step events and the importance weights of the final surviving particles of the last detected step event, etc., the tasks of back-tracking the final surviving particles and calibrating the RM could be accomplished. An example is given, in which only 16 particles have been involved, and only 4 steps have been assumed to be taken by a user along a path, for demonstration purposes only. Also, the IMU module is assumed to detect the four step events completely. Under these assumptions, an instance of the combination of the four reflection tables of the corresponding detected steps in sequence is indicated in Figure 6.3.

Firstly, let us back-track the 1st, 6th and 7th particle in the last step (4th step). From the reflection table of the last step, their common point can be found: the indices of their ancestors in the previous step (3rd step) are the same. Suppose their weights are denoted as ω_1^4 , ω_6^4 and ω_7^4 , then the importance weight of the 6th particle in the 3rd step is updated by them as: $\omega_6^3 \leftarrow \omega_6^3 + \omega_1^4 + \omega_6^4 + \omega_7^4$. Continuing the tracking, the indices of their ancestors for the second, first step and the initialisation period are found to be $RT_3(6) = 12$, $RT_2(12) = 4$ and $RT_1(4) = 8$, respectively. In Figure 6.3, the indices of the particles mentioned are all marked as red, for the convenience of readers. The pseudo-code for back-tracking all particles is indicated in algorithm 1. Noting that the task of back-tracking particles should be done in the off-line analysis phase, e.g., by the server.

Following that, the cells where the ancestor particles of one or more final surviving particles are located are identified. Bearing in mind that the probability of a user being in the range of a particular cell during the time-being of a particular step event should be equal to the sum of the importance weights of the particles (ω as defined in equation 5.2) that are identified as belonging to the cell. Here, the importance weights of the particles are obtained by back-tracking their final surviving descendant particles. Note that the values of the importance weights of the final surviving particles may vary between each other, since the values

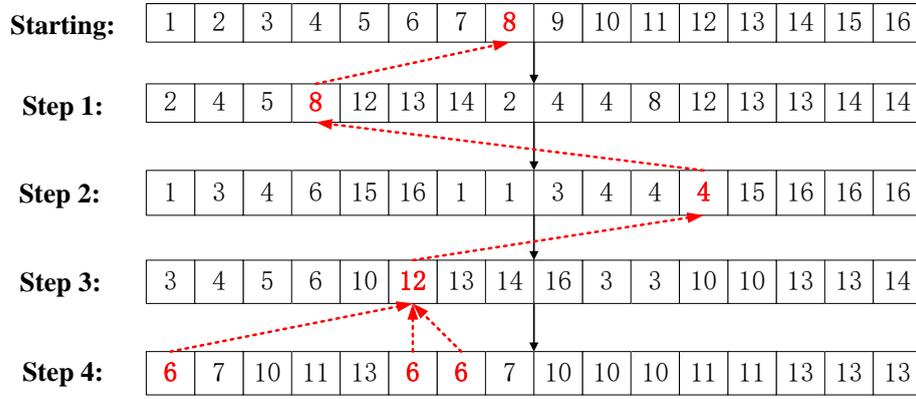


FIGURE 6.3: An example of the sequential reflection table

Algorithm 1 Back-tracking final surviving particles and Back-assignment of their importance weights to ancestors

Require: $\omega_i^k, \forall k \in \mathcal{SS}_1 = \{0, 1, \dots, N_{step} - 1\}, \forall i \in \mathcal{SS}_2 = \{1, 2, \dots, N_s\}$.

- 1: Input: importance weights of all final surviving particles: $\omega_i^{N_{step}}, \forall i \in \mathcal{SS}_2$; Reflection table $RT_n, n \in \mathcal{SS}_3 = \{1, \dots, N_{step}\}$. Initialise $\omega_i^k \leftarrow 0, \forall k \in \mathcal{SS}_1, \forall i \in \mathcal{SS}_2$.
 - 2: **for each** $i \in \mathcal{SS}_2$ **do**
 - 3: $k \leftarrow N_{step}$
 - 4: $c \leftarrow i$
 - 5: **while** $k > 0$ **do**
 - 6: $b \leftarrow \omega_c^k$
 - 7: $c \leftarrow RT_k(c)$
 - 8: $\omega_c^{k-1} = \omega_c^{k-1} + b$
 - 9: $k = k - 1$
 - 10: **end while**
 - 11: **end for**
-

have been updated just after the user experienced significant body's orientation changes along the traversed path, according to equation 5.10. Suppose a piece of RSS measurement $\psi(\mathbf{o})$ is made in the closest proximity to the occurrence time of the k -th particular step event, then:

$$Pr(\mathbf{p}_j | \psi(\mathbf{o})) = \sum_{m=1}^{N_s^k(j)} \omega_m^k \quad (6.9)$$

In the equation above, as mentioned before, j denotes the index of the RP (cell); m denotes the index of one of the particles identified as belonging to the cell labelled as j and $N_s^k(j)$ denotes the total number of particles being identified as belonging to j -th cell in the k -th step event.

Since the sum of the importance weights of the final surviving particles is equal to 1, and the importance weights of all final surviving particles are also propagated back to any group of peer particles involved in any particular historically detected step event, $\sum_{j=1}^J Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o})) = 1$ could be guaranteed for any group of peer particles involved in any particular historically detected step event. Finally, with the probabilistic back-tracking localisation results of a bundle of N_u on-line RSS measurements, the RM is updated with equations (6.6) and (6.7).

Noting that the main difference between applying the EM algorithm and back-tracking the final surviving particles to calibrate the RM is the approach applied to obtain $Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o}))$ ($\forall j \in \{1, \dots, J\}$). For the EM algorithm, $Pr(\mathbf{p}_j|\boldsymbol{\psi}(\mathbf{o}))$ is estimated in the E-step of every iteration by exploring the similarity between the on-line RSS measurements and the training RSS data stored in the RM with the NB algorithm. However, In the proposed approach, such estimations are achieved by taking advantage of traced locations of ancestor particles of the final surviving ones and the importance weights of the final surviving ones. Another difference is that the proposed approach in this chapter is not required to go through iterations for dozens of times to accomplish the task of calibrating the RM. All on-line RSS measurements obtained are required to be applied to equations (6.6) and (6.7) only once, which saves significant computational power and running time in the off-line analysis, compared with the EM algorithm.

6.3 The semi-automatic site survey method

The conventional site survey method of building the Wi-Fi RSS database requires the technicians to notify the recording system of the geometric information of the current RP on which the RSS site-survey work is being implemented. Commonly, when the floorplan of the area of interest is shown on the touch screen of the recording device, this is achieved by clicking particular places on the screen, e.g., marked black circles, as shown on the map included in Figure 4.1, on the screen of the recording device. Also, the technicians are required to inform the recording system of both the starting and ending time points of the RSS collection process on each RP. Similarly, this could be achieved by clicking corresponding buttons of the

device to start or end the RSS collection process. However, in the implementation of this thesis, the inherited nature of these operations is found to be mechanical and repetitive, which potentially makes the technicians tired with the site-survey work. The technicians could also be distracted by something else in the mind during the site-survey work. In some extreme cases, the process of the site survey has to be reset, since the technicians are no longer aware of the site-survey tasks on which RPs has been finished, which is definitely intolerable in terms of consuming additional working time and labour costs.

Self-guided robots typically are equipped with various sensors, e.g., laser-based ranging, IMU sensors, cameras. They are capable of roaming around and exploring the space of interest, and collecting the RSS training data to build the RM [99]. Manually inserted landmarks (e.g., barcodes pasted on walls, RFID tags deployed on the ground, a particular pattern painted on the ceiling) and on-board sensors are taken advantage to localise themselves. To save humans' labour efforts from site-survey tasks, such robots could be used instead. However, this process is not fully automatic and still human interventions and efforts are required to be involved. Moreover, using a robot is currently not a global economic approach. Therefore, the proposed method focuses on reducing the quantities of human intervention in the site survey, in order to bring convenience to the technicians, improve the working efficiency, and ensure the robustness of the developed RM at a higher level. The related diagram is indicated in Figure 6.4, and will be explained as follows.

In the off-line phase, initially, the recording system is informed by the technicians of the desired starting and ending RPs of the path on which all RPs are required to be covered in the route. Similar to the traditional approach, this could be achieved by clicking the touch screen of the recording device. Once the information is received, the routing algorithm introduced in Section 2.6 is applied to provide a linked list, in sequence, including the indices of the RPs on the technicians' journey in sequence. Finally, the linked list is stored in the RAM of the recording device. Also, the path is highlighted on the screen of the recording device.

In the on-line phase, the technicians, initially, stand on the starting RP, and begin

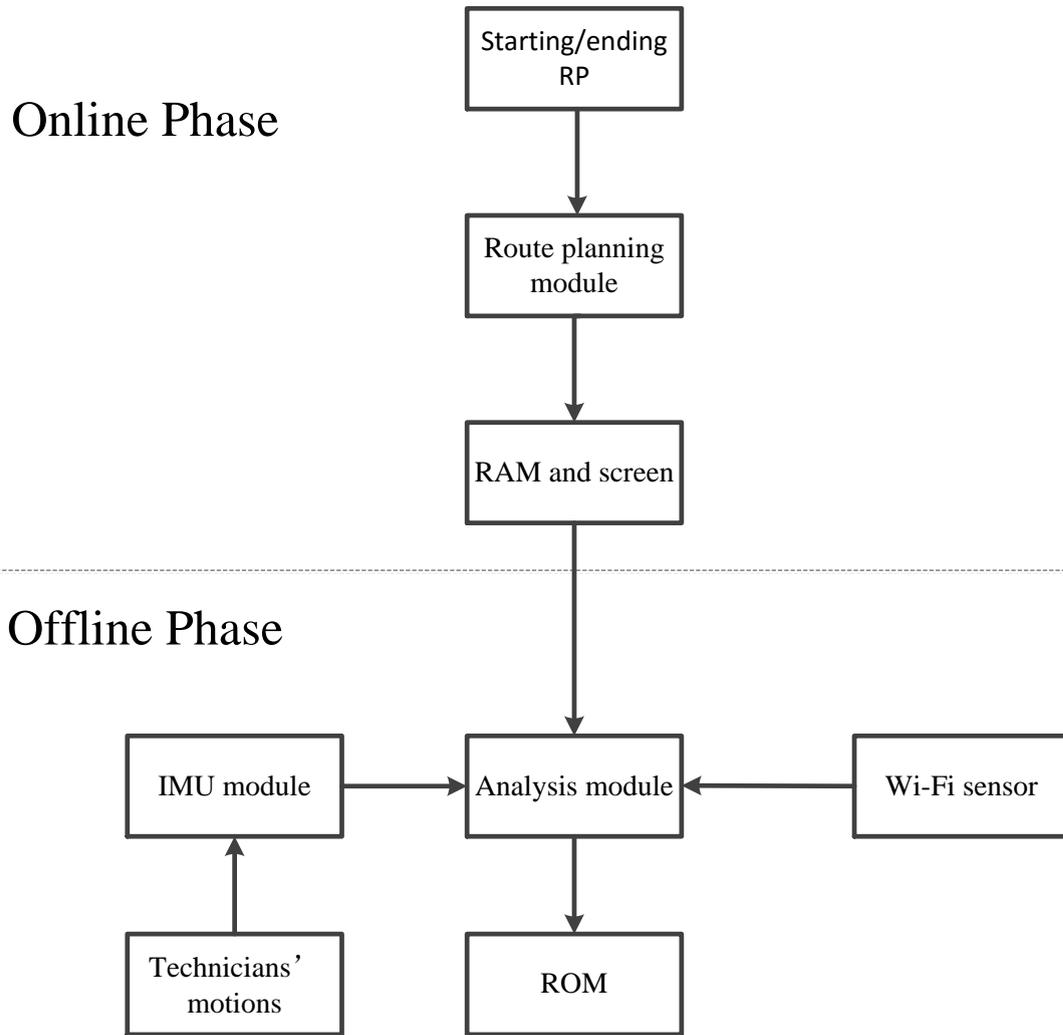


FIGURE 6.4: The diagram of semi-automatic site survey

the RSS collection process after clicking the 'start' button on the screen. Once the Wi-Fi RSS collection task on the starting RP is accomplished, the technician is free to move to the next RP with no need to give notice of either the end of RSS collection process on the current RP or the start of the RSS collection process on the next neighbouring RP. In particular, the analysis module applies data reports from the IMU module to judge the motion status of the technician, according to the walking detection algorithm presented in Section 3.4.1. When the technician is on the way to move to the next RP, the technician's motion status will be detected as moving again, which could be used as a sign notifying that the RSS collection work on the previous RP is finished. Following that, if the technician's motion status is detected as silent again, the technician will be recognised as stopping

at the next RP on the path, and Wi-Fi RSS measurements collected afterwards are geo-tagged automatically. Note that some of the collected RSS measurements are dropped, since they are recognised as being collected when the technician is moving forward to the next RP. The process repeats until finally the technician finishes the RSS collection task on the final RP and thus terminates the running of the recording system by clicking the 'end' button shown on the screen.

The proposed site-survey approach is capable of reducing the load of humans' intervention dramatically, which aims to achieve the goal of improving working efficiency and guaranteeing the reliability of the built RM.

6.4 Experimental evaluation

In the experiment, for the purpose of approaching the scenario of the experiment to the situation of humans' daily indoor localisation service in reality, a fair assumption is made in advance: the chance of every RP being covered by users' walks in daily life is approximately the same. The paths on which the experiments are carried out are indicated in Figure 6.5. Since two opposite directions are required to be experienced when the testers travels on these paths for localisation and RSS data collections, no 'arrow' is drawn on these paths to indicate the moving direction. Another main difference is that the RPs in the open space are excluded from the experiments of this chapter. When the paths connecting the RPs in the corridors with the RPs in the open space are taken into account, some RPs on the middle of the paths, especially the ones in the corridors, will be covered by the testers' walks for many times, which is the opposite to the assumption. Note that the paths used in the experiments of this chapter could cover every RP in the corridor. In addition, provided that the frequencies of the users walking on each path is the same, every RP on the path is approximately visited for the same frequency: among a total of 28 RPs, only 8 of them are visited more often (twice or three times) than the remaining ones, which coincides with the assumption of fair RP visiting opportunity.

Totally, three testers with different heights have been involved in the experiments. They walk along three paths for both direction to collect on-line RSS measurements and IMU reports on them. For each of the testers, the parameters for estimation of step length and step number have been personalised in two trail walks, as proposed in Section 3.4.4 and 3.4.3.

Also, the three paths shown are applied to test the performance of the semi-automatic site survey approach. The number of RSS measurements collected on each RP for either direction are varied from 5 to 20 for building RM.

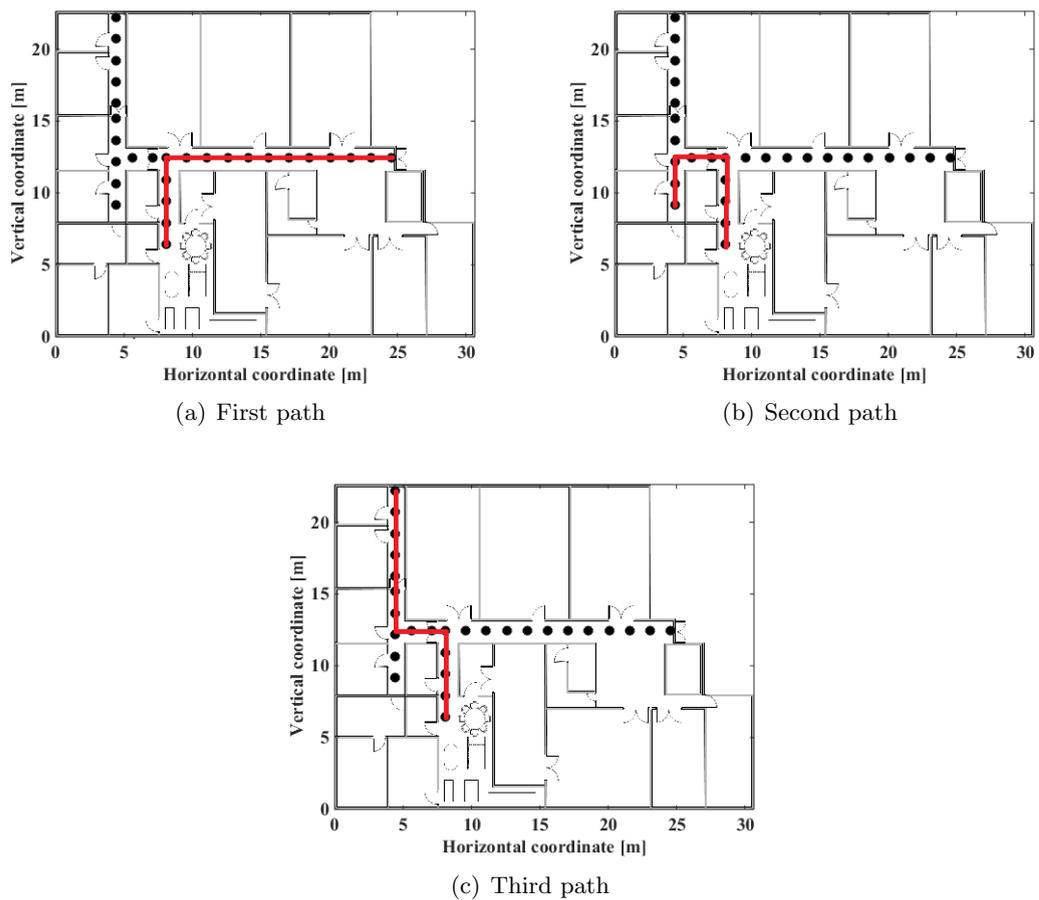


FIGURE 6.5: Showcase of the three paths utilised in the experiments

6.4.1 The evaluation of the back-tracking method in the off-line phase

The proposed particle filter is implemented when the testers take walk on each path for both directions. After the testers stop at the end and terminate the

localisation service, the server will start the off-line analysis, as indicated in Figure 6.1. An instance of the moving traces of the ancestor particles of the final surviving ones for all paths is indicated in Figure 6.6. Although the positions of some particles are found to be biased from the ground-truth locations during some particular steps, note that their impact on the net localisation results are limited by their importance weights. In this section, the performance of two on-line localisation approaches (Wi-Fi probabilistic NB approach and on-line weights-adjusted particles filter) and one off-line trajectory analysis approach (back-tracking the final surviving particles) is evaluated. The comparison of the CDF of localisation errors of these approaches are indicated in Figure 6.7. In the experiment, 10 RSS Wi-Fi RSS fingerprints (5 for either direction) collected on each RP are employed to build the RM. The employed bin width is set as 4 for the applied on-line Wi-Fi probabilistic NB approach. Then, the walk on each of three paths for either direction to collect data is repeated for 25 times. Each tester undertakes 7 or 8 times of repeated walks for each case. On average, the smartphone collects one RSS measurement every 1.5 second. Approximately, thirty-five on-line RSS measurements being collected in proximity to each RP.

From Figure 6.7, it can be found that the off-line location analysis results given by back-tracking the final surviving particles and back-assigning corresponding weights achieves the best accuracy (RMSE = 2.24 m); localisation results given by real-time particle filtering achieves the second best accuracy (RMSE = 2.58 m); performance of the on-line Wi-Fi RSS probabilistic NB approach is the worst (RMSE = 4.23 m).

6.4.2 Comparison of the two RM calibration methods

The on-line collected RSS measurements used in the previous section are also regarded as the testers' historically taken unlabelled Wi-Fi RSS measurements, and used for the RM calibration. In this section, part or all of them are applied to calibrate the RM. As mentioned before, in each iteration of the EM algorithm, probabilistic location labels of historically collected on-line RSS measurements are obtained by applying the Wi-Fi probabilistic NB localisation approach. On the

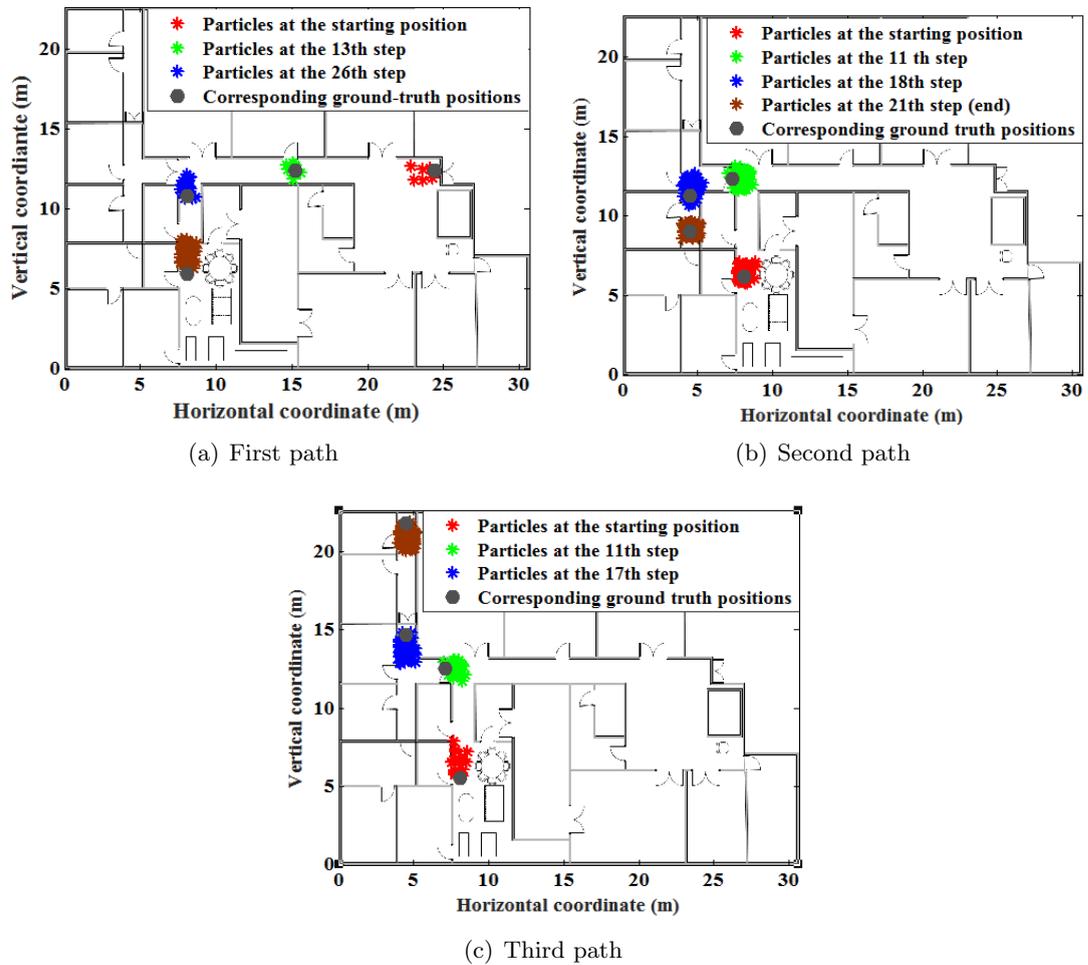


FIGURE 6.6: Three paths are evaluated specifically in the experiments for RM calibration (one direction for each path).

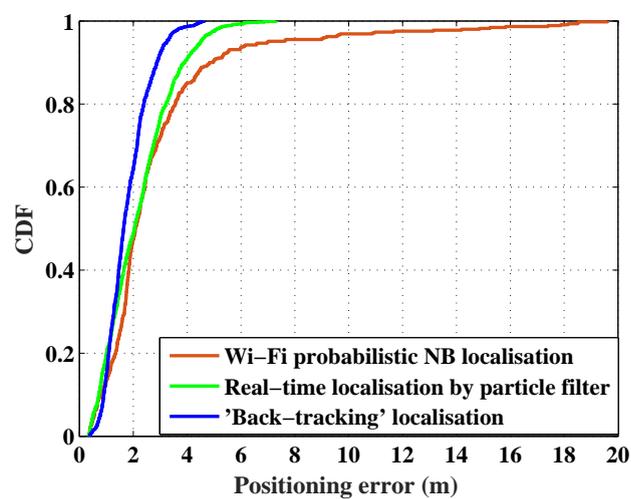


FIGURE 6.7: Comparison of the CDF of localisation errors when probabilistic NB approach, particle filter and back-tracking finally surviving particles are applied, respectively.

other hand, in the algorithm proposed in the chapter, probabilistic location labels of on-line RSS measurements collected historically in all experienced step events are estimated in the way indicated by equation (6.9). Here, only the ancestor particles of the final surviving ones in the last detected step event are taken advantage of. Also, importance weight back-assignment method, as indicated in Section 6.2.2, is applied.

In this experiment, the testing set used to evaluate the RM calibration results consists of 20 RSS measurements collected on each RP (10 for either walking direction). Firstly, the probabilistic NB approach is used for on-line localisation, based on the original RM. Secondly, a discriminative approach (multi-class LS-SVMS) is applied to calibrate the estimation result given by the probabilistic NB algorithm with the aid of the Fisher kernel. Here, the probability density distributions stored in the RM have already been calibrated further by two approaches: the EM algorithm (the same implementation as [1]) or back-tracking the final surviving particles, respectively. Finally, the localisation results obtained by applying the two approaches are compared in Figure 6.8.

From the figure, as could be found, the localisation accuracy is improved further when the RM database is calibrated by the on-line Wi-Fi RSS measurements taken historically. Secondly, similar to [1], the method of applying the combination of generative/discriminative approaches (PDF calibrated by the EM algorithm, Fisher kernel and multi-class LS-SVMS) on the RM calibrated by unlabelled on-line RSS data is proved to afford more accurate on-line Wi-Fi localisation results than solely applying the NB, the generative approach, as well as in the experiment results. Thirdly, the robustness of the RM calibrated by the proposed method is proved to be better than the one calibrated by the EM algorithm: the lowest RMSE that could be achieved by applying either the combination of generative/discriminative approaches or solely the NB algorithm based on the RM calibrated by the proposed method is lower than its counterpart. Fourthly, the RMSE difference between the dashed and solid green line is lower than the difference between the dashed and solid red line, which implies that less performance improvement space is left by the RM calibration results given by the proposed method to the discriminative approach than its counterpart. Also, from Figure 6.8, it can be

found that the performance gap between the implementation of the combination of generative/discriminative approach on the basis of the RM calibrated by the EM algorithm and the implementation of solely NB on the basis of the RM calibrated by back-tracking the finally surviving particles is small. Since the tasks such as updating the Fisher kernel matrix for the RM, building the Fisher kernel vector for on-line user's RSS measurements, and feeding the fisher kernel vector into multi-class LS-SVMs consumes additional computational power and running time, which is considered to be a major concern to real-time location estimation. Provided that the performance gap with/without the discriminative approach being applied is small, the discriminative approach could be avoided to be applied, in order to save computational resources. Finally, it could be found that the lowest RMSE occurs at a narrower bin width when the RM is calibrated further by the testers' on-line RSS measurements collected historically. The underlying reason is similar to the statement made in Section 2.3.1.1. When the bin width is set narrower, $Pr(sl_i \in B_\xi | \mathbf{p}_j; \boldsymbol{\epsilon})$, the PDF of the discrete signal level of the wireless signal received from the APs is highly likely no longer sparsely distributed, provided that the tester's historically-taken on-line RSS measurements are available to calibrate the RM (high discretisation variance is avoided). Also, inherently, the discretisation bias is low when the bin width for discretisation is set narrower. Therefore, the lowest RMSE achieved by the system could occur at a narrower bin width.

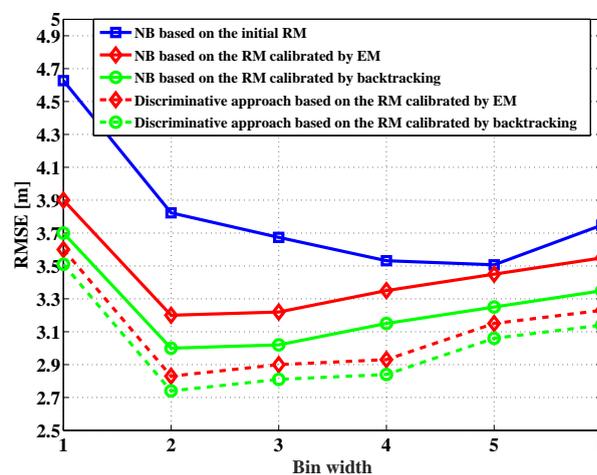


FIGURE 6.8: RMSE with respect to the bin width

In another test, the performance of localisation approaches with respect to the number of available on-line RSS measurements collected historically applied for RM calibration are evaluated. The details of the implementation of the test are shown as follows.

Firstly, the on-line Wi-Fi RSS measurements collected historically during each individual walk on each path as indicated in Figure 6.5 are put into different groups. Each group is indexed as $GA_{r,s,t}$, where r, s, t are integers, and used to indicate on which path, in which turn and in which direction has the tester recorded the set of RSS measurements. Thus r, s, t vary in the range of $[1, 3]$, $[1, 25]$ and $[1, 2]$, respectively. Following that, each one group so called 'GB' and indexed with s is formed in the way shown below:

$$GB_s = \{GA_{1,s,1}, GA_{2,s,1}, \dots, GA_{3,s,2}\} \quad (6.10)$$

Therefore, totally, there are 25 GB groups. Formulated in this way, each GB group consists of one and only one set of RSS measurements collected along a particular walk on every path for either direction. In this way, the RSS measurements included in each 'GB' group are guaranteed to be unique (being collected at different times) from the ones included in other groups.

For the purpose of testing the performance of the RM calibration result vs. number of historically-taken on-line RSS measurements applied for RM calibration, the scale of the RSS calibration database is varied. In particular, the scale of the RSS calibration database could be tuned by employing a different number of 'GB' groups. However, there may exist tens of thousands of combinations of 'GB' groups for building the RSS calibration database for tests. For example, if $\frac{3}{5}$ of all RSS measurements are taken advantage of, totally $\binom{25}{15} = 3268760$ combinations could be made. Therefore, taking advantage of all combinations of 'GB' groups for testing the performance of RM calibration approaches is obviously computationally unrealistic. Another alternative way could be integrating every randomly-chosen five 'GB' groups into a so called 'GC' group, and thus only $\binom{5}{3} = 10$ combinations exist. Based on the RM calibrated by each combination of the groups of the

historically collected RSS measurements, the localisation accuracy on the testing RSS measurement is evaluated, respectively.

In Figure 6.9, the performance of the localisation approaches (the best RMSE that could be achieved when the employed bin width is varied from 1 to 5) with respect to the proportion of historical online RSS measurements being applied for RM calibration are indicated. As shown in the figure, overall, the lowest RMSE of the system decreases when more historically-collected online RSS measurements are available to the system for RM calibration, which was already proven by [1]. Furthermore, the result also demonstrates again that the proposed method achieves better localisation accuracy under the same condition. The optimal bin width vs. proportion of all historically-collected RSS measurements are indicated in table 6.1. Note that on average 35 Wi-Fi RSS samples have been collected on each RP, if all walks have been taken into account. So, the expectation value of the number of historically-taken Wi-Fi RSS samples collected on each RP for the tested proportion of all RSS measurements in the figure could be found as 7, 21 and 35. Therefore, according to table 6.1, provided that the RM calibration method is applied, the bin width for discretisation of the continuous RSS values of the on-line probabilistic Wi-Fi localisation system for further application could be tuned to a more proper smaller value than the initially set value by comparing the amount of accumulated historically-taken on-line RSS measurements with the products of those numbers and the number of RPs. In this way, the discretisation bias distorting the localisation accuracy could be reduced.

| | | | |
|-------------------|-----|-----|---|
| Proportion | 1/5 | 3/5 | 1 |
| Optimal bin index | 4 | 3 | 2 |

TABLE 6.1: The optimal bin index VS. Proportion of all historically-collected RSS measurements

6.4.3 Performance evaluation of the semi-automatic site survey approach

Totally, there are 28 RPs distributed on the paths indicated in Figure 6.7. The overhead time taken in notifying the recording system of the position of the RP

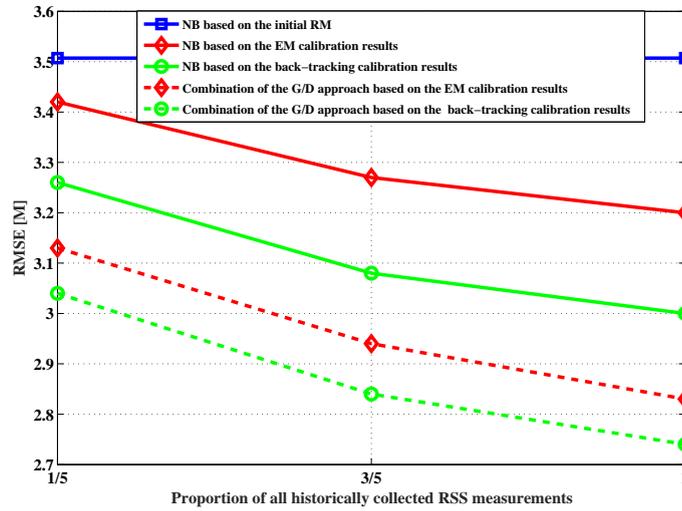


FIGURE 6.9: RMSE with respect to the proportion of historical on-line RSS measurements being applied.

where the technician is instantly located on and clicking the ‘starting/ending’ button is measured as 2 seconds approximately from the real experiment, on average. Also, as mentioned before, consecutive RSS measurements are obtained in every 1.5 seconds, on average. So, given 5 consecutive Wi-Fi RSS samples required to be collected from each of 28 RPs for either direction in the site survey, the ratio of the consumed overhead time to the total time consumed in site-survey work could be calculated as $RO = \frac{28 \times 2}{28 \times 2 + 1.5 \times 5 \times 28}$. Therefore, the ratio VS. the number of RSS collections on each RP is shown in the table as follows:

| Number of RSS collection on each RP | 5 | 10 | 15 | 20 |
|-------------------------------------|--------------------|-------|------|------|
| Total overhead time (s) | $28 \times 2 = 56$ | | | |
| RO | 21.1% | 11.5% | 8.2% | 6.3% |

TABLE 6.2: The ratio of overhead time to the total consumed time in site survey VS. the number of RSS collections on each RP

Note that another goal of this proposed approach is to reduce the additional potential time taken in repeating the whole RSS collection process resulting from the distraction of the technician from the boring and mechanical operations in the traditional site-survey task. However, the potential saved time is not accounted for in the calculation results in table 6.2, since it is hard to be quantified, depending on how often the technician wrongly localises himself due to the distraction, how many RPs the technician has been through until he wrongly localises himself, etc.

As a result, the ratio of the total time the proposed method could save shown in table 6.2 is regarded as the lower bound.

Since the proposed localisation approach requires only 10 Wi-Fi RSS measurements on each RP (5 RSS measurements for either direction) to be collected for initialising the RM, the proposed semi-automatic site-survey approach is expected to reduce by at least 21.1% of the total time required for a site survey.

6.5 Chapter summary

In this chapter, a novel RM calibration approach and semi-automatic site-survey approach are proposed.

In the proposed approach, when a particle filter is implemented to provide an on-line localisation service to users, the historical positions and inheritance relationship between re-sampled particles and ancestor ones are recorded simultaneously. After a certain number of on-line location tracking services are finished, the back-tracked historical positions of the final surviving particles and their corresponding importance weights are applied for RM calibration. As indicated in Figure 6.7, the localisation results of back-tracked final surviving particles holds the best performance. Therefore, the on-line localisation results based on the RM calibrated by the proposed approach are also proved to be more accurate than the approach presented in [1], as shown in Figure 6.8 and 6.9.

Note that only a small number of Wi-Fi RSS samples (e.g., 10 samples) are required to be collected on each RP distributed in the area of interest to initialise the RM for implementation of the proposed localisation method, The time spent in notifying the recording system of the starting/ending of RSS collection process and the position of the RP on which to collect the RSS measurements for building RM can be no longer ignored when comparing its duration with the time purely utilised in the RSS collection process in the site survey. Furthermore, the manual action for notifying the recording system of the information mentioned is regarded as trouble to the technicians, due to its inherited nature: mechanical and repetitive. A semi-automatic site-survey approach is proposed to relieve the technicians of the burden.

Here, 'semi-automatic' implies that the approach only requires the technicians to inform the desired starting/ending RP and notify the beginning/ending of the whole RSS collection process to the system. The experiment results show that at least 21.1% of the total working time could be saved by the proposed approach. In addition, potential humans' operation mistakes could be avoided, which results in a better working efficiency.

Chapter 7

Conclusion

In this thesis, an indoor pedestrian localisation and tracking system based on Wi-Fi RSS fingerprints and reports of the IMU modules was proposed.

Specifically, the initial position of the user is given by on-line probabilistic Wi-Fi RSS localisation results. An RSS fingerprint-based localisation system consists of two phases: an off-line phase and an on-line phase. In particular, the off-line phase is also known as a training phase, in which RSS measurements from APs in range are collected at grid-distributed known RP locations. Then, the PDF of the discrete signal levels of the RSS measurements collected on each RP are employed as reference data stored in the RM. Also, two phases are involved in on-line Wi-Fi RSS localisation: rough localisation and fine localisation. Firstly, the number of clusters are determined by the clustering accuracy of the implementation of the unsupervised k -means++ algorithm. Based on the clustering pattern obtained from the implementation of the algorithm, technician's manual adjustment of the cluster memberships for the RPs are needed, according to the requirements presented in Section 4.3. Following that, depending on the scale of the RM, a supervised clustering algorithm such as OVO-SVMs or OVA-SVMs is chosen for rough localisation of users into a smaller geometric range. Finally, the NB algorithm is applied for fine localisation. The localisation results are given in the form of the probability of each targeted RP being the current location of the user.

A particle filter is used to estimate the trajectory of the users by analysing the reports of the IMU modules. The particle filter models every trajectory as a sequence of hidden states in a continuous state space. Monte Carlo simulation is applied to solve for the most likely state sequence. Also, in the implementation of the particle filter, the distribution of the initially-employed particles are given by the on-line Wi-Fi localisation results. Specifically, Wi-Fi rough localisation determines the membering RPs of which cluster should be taken into account; Wi-Fi probabilistic NB fine localisation determines how many particles should be distributed exactly into the cells of the RPs in the cluster nominated by the rough localisation. Then, the IMU module and the wall constraints are taken advantage of to update and correct the positions of the particles, respectively. Since an increasing number of the originally-distributed particles are killed in the process, the localisation service will be no longer provided if all the particles are killed. Therefore, a particle re-sampling scheme is applied to keep the number of employed particles constant in every detected step event. In each detected step event, the newly-generated particles are sampled from the surviving ones, according to their respective importance weights. Also, the phenomenon that the employed particles possibly become multi-cluster distributed after users experience significant body turnings has been observed. To reduce the negative impact of that to the on-line localisation accuracy, after users' significant body turnings (significant phones orientation changes) are observed, on-line Wi-Fi RSS measurements are taken advantage of to adjust the importance weight of every particle, according to the distance between the on-line Wi-Fi localisation result and the position of the cell that the particle falls into. Analysis of the data obtained from the IMU module is also a key component to implement the particle filter for pedestrian localisation. Specifically, the unique repetitive pattern of the magnitude of 3-axis acceleration has been found to be suitable to count the number of steps taken by a pedestrian. Therefore, it is applied in this thesis. In addition, the time moments when the maximum values of the magnitude of the accelerations occur are regarded as the stepping moments, if a step is detected. Following that, the moving distance of every detected step is estimated by utilising personalised parameters obtained from a training phase. While a lot of high-frequency noise has been found in the readings of a 3-axis magnetometer, the type of the noise in the readings of a gyroscope is

found to be low-frequency. To reduce the impact of noises in the readings of two respective information sources to orientation estimation, a complementary filter is taken advantage of to provide the net orientation estimation results.

In the thesis, a new RM calibration method which labels users' historically-taken on-line RSS measurements by back-tracking the ascendant of the finally surviving particles and applying the importance weights of the finally surviving particles is proposed. Compared with the EM algorithm, no convergence process is involved and the RM calibration result is more robust. Bin width for discretisation of the continuous RSS values is suggested to be adjusted for providing more accurate localisation service, provided that increasing number of users' historically-taken RSS measurements are accumulated over time. Also, note that traditionally, in the off-line phase, dozens of Wi-Fi RSS training measurements on each distributed RP are required to be collected for building the RM. However, since the RM could be further calibrated by the on-line RSS measurements, in the off-line phase, much less training data is required to be collected. Thus, the workload involved in the off-line phase could be reduced. In addition, to reduce the quantity of human intervention in the site-survey task and the overall time consumed in site-survey task, a new site-survey method featured as 'semi-automatic' is proposed, in which both the route planning and walking detection algorithms are applied. The results show that the majority of human intervention actions, such as clicking on the screen of the device to notify the current position of the technician, the start/end of RSS collection on the current position, etc, could be saved.

7.1 Future work

Several problems and research topics could be explored to improve further the performance of the proposed indoor pedestrian localisation and tracking system.

Firstly, RSS levels of some APs at particular positions in the area of interest are found to be varied in a wide range. For example, some APs could be in the sensitivity range of the smartphones in part of collection time, but not in the rest of the time. However, RSS levels of other APs are relatively stable. Obviously, the

‘credibilities’ of such two groups of APs to the localisation system are different. Some researches such as [34] filter out some of the APs with low ‘credibilities’ in localisation tasks. However, the result shows that the localisation accuracy is dropped after such a scheme is applied. Therefore, a better way to leverage the knowledge that different AP exhibits different credibility should be explored for more accurate localisation results.

Secondly, in the thesis, exactly the same smartphone is used for collection of training RSS data and on-line RSS data in the off-line and on-line phases. In reality, users could be equipped with various types of smartphones with different antenna gain patterns. So, the RSS data collected by them on the same location at the same time may differ from each other. Therefore, the RM built from RSS training data collected by a particular cellphone model might not be suitable for the localisation tasks taken by other types of smartphones. Unfortunately, to the best knowledge of the author, by now, there is still lack of feasible approach to solve this problem. In future, much more research efforts should be focused on finding the solution, since it has already been recognised as one of the barriers preventing the indoor Wi-Fi RSS-based localisation system from being widely applied for commercial purposes.

Finally, note that nowadays applying the VLC technology with a pair of LED and light sensor on a smartphone has become an important prospective candidate member for providing an ultra-fast wireless network access service to normal users. The LEDs which are capable of such a communication purpose are expected to be deployed widely in indoor environments in future, as Wi-Fi routers are densely distributed nowadays. Also, the other features of the VLC such as short-distance and strict LOS transmission being required enables it to become a good candidate for providing localisation services in indoor environments. Therefore, it is expected that the initial distribution of the particles for implementation of the particle filter for indoor localisation and on-line adjustment of their importance weights could be more accurate once upon that the localisation results from on-line Wi-Fi RSS-based localisation technology and those from taking advantage of the installed LEDs are fused. As a result, the proposed localisation system should be extended to use the information provided by the light sensors to improve the performance.

Bibliography

- [1] Robin Wentao Ouyang, Albert Kai-Sun Wong, Chin-Tau Lea, and Mung Chiang. Indoor location estimation with reduced calibration exploiting unlabeled data via hybrid generative/discriminative learning. *Mobile Computing, IEEE Transactions on*, 11(11):1613–1626, 2012.
- [2] Hui Liu, Houshang Darabi, Pat Banerjee, and Jing Liu. Survey of wireless indoor positioning techniques and systems. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 37(6):1067–1080, 2007.
- [3] David S Moore, George P McCabe, and Bruce A Craig. Introduction to the practice of statistics. 2012.
- [4] Sayfa Ana. Introduction to ins. http://inertialnavigations.blogspot.co.uk/2012_03_01_archive.html.
- [5] Google inc. Android api guide on sensorevent. <http://developer.android.com/reference/android/hardware/SensorEvent.html>.
- [6] Jochen Schiller and Agnès Voisard. *Location-based services*. Elsevier, 2004.
- [7] Yue Chen. *Soft Handover Issues in Radio Resource Management for 3G WCDMA Networks*. PhD thesis, 2003.
- [8] Easytrails GPS. <http://www.easytrailsgps.com/>.
- [9] iCartel. <http://icartel.net/icartel-docs/>.
- [10] Alex Varshavsky, Eyal de Lara, Jeffrey Hightower, Anthony LaMarca, and Veljo Otsason. Gsm indoor localization. *Pervasive and Mobile Computing*, 3(6):698–720, 2007.

-
- [11] Roy Want, Andy Hopper, Veronica Falcao, and Jonathan Gibbons. The active badge location system. *ACM Transactions on Information Systems (TOIS)*, 10(1):91–102, 1992.
- [12] Nissanka Bodhi Priyantha. *The cricket indoor location system*. PhD thesis, Massachusetts Institute of Technology, USA, 2005.
- [13] Guang-yao Jin, Xiao-yi Lu, and Myong-Soon Park. An indoor localization mechanism using active rfid tag. In *Sensor Networks, Ubiquitous, and Trustworthy Computing, 2006. IEEE International Conference on*, volume 1, pages 4–pp. IEEE, 2006.
- [14] Sinan Gezici, Zhi Tian, Georgios B Giannakis, Hisashi Kobayashi, Andreas F Molisch, H Vincent Poor, and Zafer Sahinoglu. Localization via ultra-wideband radios: a look at positioning aspects for future sensor networks. *Signal Processing Magazine, IEEE*, 22(4):70–84, 2005.
- [15] Silke Feldmann, Kyandoghene Kyamakya, Ana Zapater, and Zighuo Lue. An indoor bluetooth-based positioning system: Concept, implementation and experimental evaluation. In *International Conference on Wireless Networks*, pages 109–113, 2003.
- [16] Moustafa A Youssef, Ashok Agrawala, and A Udaya Shankar. 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, 2003.
- [17] Anna Maria Vegni and Mauro Biagi. An indoor localization algorithm in a small-cell led-based lighting system. In *Indoor Positioning and Indoor Navigation (IPIN), 2012 International Conference on*, pages 1–7. IEEE, 2012.
- [18] Soo-Yong Jung, Swook Hann, and Chang-Soo Park. Tdoa-based optical wireless indoor localization using led ceiling lamps. *Consumer Electronics, IEEE Transactions on*, 57(4):1592–1597, 2011.
- [19] Liqun Li, Pan Hu, Chunyi Peng, Guobin Shen, and Feng Zhao. Epsilon: A visible light based positioning system. In *11th USENIX Symposium on*

- Networked Systems Design and Implementation (NSDI 14)*, pages 331–343. USENIX Association, 2014.
- [20] Ali Aassie Ali and AS Omar. Time of arrival estimation for wlan indoor positioning systems using matrix pencil super resolution algorithm. In *Proceedings of the 2nd Workshop on Positioning, Navigation and Communication, WPNC*, volume 5, pages 11–20, 2005.
- [21] Krzysztof W Kolodziej and Johan Hjelm. *Local positioning systems: LBS applications and services*. CRC press, 2006.
- [22] Kevin Roebuck. *Location-Based Services (LBS): High-impact Strategies-What You Need to Know: Definitions, Adoptions, Impact, Benefits, Maturity, Vendors*. Emereo Publishing, 2012.
- [23] Reetu Singh, Luca Macchi, Carlo S Regazzoni, and K Plataniotis. A statistical modelling based location determination method using fusion technique in wlan. *Proc. IEEE IWWAN*, 2005.
- [24] Navin Kumar Sharma. A weighted center of mass based trilateration approach for locating wireless devices in indoor environment. In *Proceedings of the 4th ACM international workshop on Mobility management and wireless access*, pages 112–115. ACM, 2006.
- [25] Paramvir Bahl and Venkata N Padmanabhan. Radar: An in-building rf-based user location and tracking system. In *INFOCOM 2000. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, volume 2, pages 775–784. Ieee, 2000.
- [26] Veljo Otsason, Alex Varshavsky, Anthony LaMarca, and Eyal De Lara. Accurate gsm indoor localization. In *UbiComp 2005: Ubiquitous Computing*, pages 141–158. Springer, 2005.
- [27] Vinko Erceg, KVS Hari, MS Smith, Daniel S Baum, KP Sheikh, C Tappenden, JM Costa, C Bushue, A Sarajedini, R Schwartz, et al. Channel models for fixed wireless applications, 2001.

-
- [28] Eraldo Damosso and Luis M Correia. *COST Action 231: Digital Mobile Radio Towards Future Generation Systems: Final Report*. European Commission, 1999.
- [29] André GM Lima and Luiz F Menezes. Motley-keenan model adjusted to the thickness of the wall. In *Microwave and Optoelectronics, 2005 SBMO/IEEE MTT-S International Conference on*, pages 180–182. IEEE, 2005.
- [30] He Zhao and Zheyao Wang. Motion measurement using inertial sensors, ultrasonic sensors, and magnetometers with extended kalman filter for data fusion. *Sensors Journal, IEEE*, 12(5):943–953, 2012.
- [31] Oliver J Woodman. An introduction to inertial navigation. *University of Cambridge, Computer Laboratory, Tech. Rep. UCAMCL-TR-696*, 14:15, 2007.
- [32] Yang Liu, M. Dashti, H. Claussen, and Jie Zhang. Semi-automatic site survey for indoor localization. In *Positioning Navigation and Communication (WPNC), 2015 12th Workshop on*, March 2015.
- [33] Fan Li, Chunshui Zhao, Guanzhong Ding, Jian Gong, Chenxing Liu, and Feng Zhao. A reliable and accurate indoor localization method using phone inertial sensors. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, pages 421–430. ACM, 2012.
- [34] Chen Feng, Wain Sy Anthea Au, Shahrokh Valaee, and Zhenhui Tan. Received-signal-strength-based indoor positioning using compressive sensing. *Mobile Computing, IEEE Transactions on*, 11(12):1983–1993, 2012.
- [35] Yang Liu, Mina Dashti, Abd Rahman, Mohd Amiruddin, and Jie Zhang. Indoor localization using smartphone inertial sensors. In *Positioning, Navigation and Communication (WPNC), 2014 11th Workshop on*, pages 1–6. IEEE, 2014.
- [36] Yang Liu, Mina Dashti, and Jie Zhang. Indoor localization on mobile phone platforms using embedded inertial sensors. In *Positioning Navigation and Communication (WPNC), 2013 10th Workshop on*, pages 1–5. IEEE, 2013.

- [37] Sebastian Sczyslo, Jens Schroeder, Stefan Galler, and Thomas Kaiser. Hybrid localization using uwb and inertial sensors. In *Ultra-Wideband, 2008. ICUWB 2008. IEEE International Conference on*, volume 3, pages 89–92. IEEE, 2008.
- [38] Jürgen Kemper and Holger Linde. Challenges of passive infrared indoor localization. In *Positioning, Navigation and Communication, 2008. WPNC 2008. 5th Workshop on*, pages 63–70. IEEE, 2008.
- [39] Moe Z Win, Robert A Scholtz, et al. Ultra-wide bandwidth time-hopping spread-spectrum impulse radio for wireless multiple-access communications. *IEEE Transactions on communications*, 48(4):679–689, 2000.
- [40] Cemin Zhang, Michael Kuhn, Brandon Merkl, Aly E Fathy, and Mohamed Mahfouz. Accurate uwb indoor localization system utilizing time difference of arrival approach. In *Radio and Wireless Symposium, 2006 IEEE*, pages 515–518. IEEE, 2006.
- [41] Mike Addlesee, Rupert Curwen, Steve Hodges, Joe Newman, Pete Steggles, Andy Ward, and Andy Hopper. Implementing a sentient computing system. *Computer*, 34(8):50–56, 2001.
- [42] Toshihiko Komine and Masao Nakagawa. Fundamental analysis for visible-light communication system using led lights. *Consumer Electronics, IEEE Transactions on*, 50(1):100–107, 2004.
- [43] Yuichi Tanaka, Toshihiko Komine, Shinichiro Haruyama, and Masao Nakagawa. Indoor visible light data transmission system utilizing white led lights. *IEICE transactions on communications*, 86(8):2440–2454, 2003.
- [44] J Grubor, OC Jamett, JW Walewski, S Randel, and K-D Langer. High-speed wireless indoor communication via visible light. *ITG-Fachbericht-Breitbandversorgung in Deutschland-Vielfalt für alle?*, 2007.
- [45] Jay Farrell. *Aided navigation: GPS with high rate sensors*. McGraw-Hill, Inc., 2008.
- [46] Subrata Goswami. *Indoor location technologies*. Springer Science & Business Media, 2012.

- [47] Jukka Talvitie, Markku Renfors, and Elena Simona Lohan. Distance-based interpolation and extrapolation methods for rss-based localization with indoor wireless signals. *Vehicular Technology, IEEE Transactions on*, 64(4):1340–1353, 2015.
- [48] Milan D Redzic, Conor Brennan, and Noel E O’Connor. Seamloc: Seamless indoor localization based on reduced number of calibration points. *Mobile Computing, IEEE Transactions on*, 13(6):1326–1337, 2014.
- [49] Mohamed M Atia, Aboelmagd Noureldin, and Michael J Korenberg. Dynamic online-calibrated radio maps for indoor positioning in wireless local area networks. *Mobile Computing, IEEE Transactions on*, 12(9):1774–1787, 2013.
- [50] He Wang, Souvik Sen, Ahmed Elgohary, Moustafa Farid, Moustafa Youssef, and Romit Roy Choudhury. 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, 2012.
- [51] Anshul Rai, Krishna Kant Chintalapudi, Venkata N Padmanabhan, and Rishu Sen. Zee: zero-effort crowdsourcing for indoor localization. In *Proceedings of the 18th annual international conference on Mobile computing and networking*, pages 293–304. ACM, 2012.
- [52] Timea Bagosi and Zoltan Baruch. Indoor localization by wifi. In *IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, pages 449–452, 2011.
- [53] Moustafa Youssef and Ashok Agrawala. The horus wlan location determination system. In *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, pages 205–218. ACM, 2005.
- [54] Teemu Roos, Petri Myllymäki, Henry Tirri, Pauli Misikangas, and Juha Sievänen. A probabilistic approach to wlan user location estimation. *International Journal of Wireless Information Networks*, 9(3):155–164, 2002.

- [55] Xiaoyong Chai and Qiang Yang. Reducing the calibration effort for probabilistic indoor location estimation. *Mobile Computing, IEEE Transactions on*, 6(6):649–662, 2007.
- [56] Azadeh Kushki. *A cognitive radio tracking system for indoor environments*. PhD thesis, University of Toronto, 2008.
- [57] Ying Yang and Geoffrey I Webb. Discretization for naive-bayes learning: managing discretization bias and variance. *Machine learning*, 74(1):39–74, 2009.
- [58] Ron Kohavi, David H Wolpert, et al. Bias plus variance decomposition for zero-one loss functions. In *ICML*, volume 96, pages 275–83, 1996.
- [59] Brendan Duncan. Bias, variance, and overfitting – machine learning overview part 4 of 4. <https://blog.fliptop.com/blog/2015/03/02/bias-variance-and-overfitting-machine-learning-overview/>.
- [60] Stuart P Lloyd. Least squares quantization in pcm. *Information Theory, IEEE Transactions on*, 28(2):129–137, 1982.
- [61] David Arthur and Sergei Vassilvitskii. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1027–1035. Society for Industrial and Applied Mathematics, 2007.
- [62] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, pages 144–152. ACM, 1992.
- [63] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.
- [64] Christopher JC Burges. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 2(2):121–167, 1998.
- [65] Hyeran Byun and Seong-Whan Lee. Applications of support vector machines for pattern recognition: A survey. In *Pattern recognition with support vector machines*, pages 213–236. Springer, 2002.

- [66] Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.
- [67] A Mathur and GM Foody. Multiclass and binary svm classification: Implications for training and classification users. *Geoscience and Remote Sensing Letters, IEEE*, 5(2):241–245, 2008.
- [68] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425, 2002.
- [69] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.
- [70] Tommi Jaakkola, David Haussler, et al. Exploiting generative models in discriminative classifiers. *Advances in neural information processing systems*, pages 487–493, 1999.
- [71] A Jordan. On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in neural information processing systems*, 14:841, 2002.
- [72] Andrew Ng. Cs229 lecture notes. *CS229 Lecture notes*, 1(1):1–3, 2000.
- [73] Eric Foxlin. Pedestrian tracking with shoe-mounted inertial sensors. *Computer Graphics and Applications, IEEE*, 25(6):38–46, 2005.
- [74] Khairi Abdulrahim, Terry Moore, Chris Hide, and Chris Hill. Understanding the performance of zero velocity updates in mems-based pedestrian navigation.
- [75] Diego Alvarez, Rafael C González, Antonio López, and Juan C Alvarez. Comparison of step length estimators from wearable accelerometer devices. In *Engineering in Medicine and Biology Society, 2006. EMBS'06. 28th Annual International Conference of the IEEE*, pages 5964–5967. IEEE, 2006.
- [76] Shane Colton and FRC Mentor. The balance filter. *Presentation, Massachusetts Institute of Technology*, 2007.

- [77] Shahid Ayub, Alireza Bahraminisaab, and Bahram Honary. A sensor fusion method for smart phone orientation estimation. In *Proceedings of the 13th Annual Post Graduate Symposium on the Convergence of Telecommunications, Networking and Broadcasting, Liverpool*, 2012.
- [78] Eric W Weisstein. Adjacency matrix. 2007.
- [79] Edsger W Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271, 1959.
- [80] Simo Särkkä. *Bayesian filtering and smoothing*. Cambridge University Press, 2013.
- [81] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *Journal of Fluids Engineering*, 82(1):35–45, 1960.
- [82] Greg Welch and Gary Bishop. An introduction to the kalman filter. https://www.cs.unc.edu/~welch/media/pdf/kalman_intro.pdf.
- [83] Arnaud Doucet and Adam M Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. *Handbook of Nonlinear Filtering*, 12:656–704, 2009.
- [84] General Use and Strategic Navigational Tactical Consumer. How good is your gyro? *IEEE Control Systems Magazine*, 1066(033X/10), 2010.
- [85] Myron Kayton and Walter R Fried. *Avionics navigation systems*. John Wiley & Sons, 1997.
- [86] William Premerlani and Paul Bizard. Direction cosine matrix imu: Theory. *DIY DRONE: USA*, pages 13–15, 2009.
- [87] Oliver Woodman. *Pedestrian localisation for indoor environments*. PhD thesis, University of Cambridge, UK, 2010.
- [88] Agata Brajdic and Robert Harle. Walk detection and step counting on unconstrained smartphones. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and ubiquitous computing*, pages 225–234. ACM, 2013.

- [89] Chenshu Wu, Zheng Yang, Yunhao Liu, and Wei Xi. Will: Wireless indoor localization without site survey. *Parallel and Distributed Systems, IEEE Transactions on*, 24(4):839–848, 2013.
- [90] JB Allen. Short term spectral analysis, synthesis, and modification by discrete fourier transform. In *IEEE Trans. on Acoust., Speech, and Sig. Proc.*, volume 4, pages 21–24, 1997.
- [91] Kevin Hall. Video analysis of pelvic tilt during walking and low back pain at kevin hall physiotherapy. <https://www.youtube.com/watch?v=2CKBgzulQ6E>.
- [92] Wiebren Zijlstra. Assessment of spatio-temporal parameters during unconstrained walking. *European journal of applied physiology*, 92(1-2):39–44, 2004.
- [93] Eric W Weisstein. Rotation matrix. 2003.
- [94] Brendan J. Frey and Delbert Dueck. Clustering by passing messages between data points. *Science*, 315:972–976, 2007. URL www.psi.toronto.edu/affinitypropagation.
- [95] Inmar E Givoni and Brendan J Frey. A binary variable model for affinity propagation. *Neural computation*, 21(6):1589–1600, 2009.
- [96] Yuchun Tang, Yan-Qing Zhang, Nitesh V Chawla, and Sven Krasser. Svms modeling for highly imbalanced classification. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 39(1):281–288, 2009.
- [97] Martin Klepal, Stéphane Beauregard, et al. A novel backtracking particle filter for pattern matching indoor localization. In *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, pages 79–84. ACM, 2008.
- [98] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the royal statistical society. Series B (methodological)*, pages 1–38, 1977.

-
- [99] Lun-Wu Yeh, Ming-Hsiu Hsu, Hong-Ying Huang, and Yu-Chee Tseng. Design and implementation of a self-guided indoor robot based on a two-tier localization architecture. *Pervasive and Mobile Computing*, 8(2):271–281, 2012.