UNIVERSITY OF SOUTHAMPTON

FACULTY OF ENGINEERING AND THE ENVIRONMENT

CIVIL, MARITIME AND ENVIORNMENTAL ENGINEERING AND SCIENCES

SMARTPHONE ASSISTED PEDESTRIAN LOCALIZATION WITHIN BUILDINGS

BY

SHASHANK KUMAR GUPTA

THESIS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

NOVEMBER 2016

UNIVERSITY OF SOUTHAMPTON

ABSTRACT

Faculty of Engineering and the Environment
Civil, Maritime and Environmental Engineering and Sciences

Doctor of Philosophy

Smartphone assisted Pedestrian Localization within Buildings

By
Shashank Kumar Gupta

Location based services are becoming an indispensable part of our life. The wide adoption of satellite based positioning - Global Positioning System (GPS) has practically solved the problem of outdoor localization for a wide range of scenarios. Unfortunately, satellite based positioning is not possible indoors because of weak radio signals and loss of the direct line of sight from the satellites. Therefore, significant efforts have been motivated towards finding a practical solution for indoor localization especially in regards to localizing pedestrian.

Certainly the topic of indoor pedestrian positioning does not lack research, there have been several research studies also various commercial solutions have been developed. What is common for all of them is that no approach has yet made a big impact within this area (e.g. GPS for outdoor localization). The reason behind this is that they either need an expensive infrastructure deployment (e.g. Wi-Fi access points) or have specialised hardware needs (e.g. network card), or have low accuracy and low reliability or have privacy issues such that pedestrians' location is continuously monitored without their consent. There is also a trade-off between accuracy and cost. Sensing infrastructures (e.g. Wi-Fi) involving higher investments provide better accuracy where as those involving lower investments (e.g. QR codes) provide lower accuracy.

Even worse, systems could not logically localize a pedestrian that is whether they are on this room or the adjoining room separated by a dividing wall and somehow if they do, they require large amounts of infrastructure to be installed into the environment.

Smartphones are little less to ubiquitous. Thus, this thesis investigates an alternative approach to indoor pedestrian localization that uses smartphones to provide accurate, reliable, low cost logical localization. A significant emphasis is given on user privacy and minimal usage of infrastructure or none at all. It is demonstrated that how the information from smartphone sensors can be used for positioning in an infrastructure free environment by means of a case study. An extension to the well-studied inertial navigation technique is implemented using smartphone mounted on a toy vehicle over an artificial testbed – Scalextric track.

Having learnt that infrastructure free positioning is possible using only the inertial navigation sensors embedded in smartphone, off the shelf stride estimation methods (foot step detection techniques and stride length estimation models) are applied to investigate the most suitable stride estimation method for smartphone based pedestrian dead reckoning (PDR) positioning system. Unfortunately, what was most noticeable in all the methods was that their performance was user specific and importantly, dependent on heuristic parameters. In addition, the position error grows overtime because of slowly accumulating errors in the measurement of inertial sensor.

To reduce the dependency on heuristic parameters we investigate the statistical approach – 'Kalman filter' to get a better estimate of the stride lengths. Nevertheless, drifts are mitigated by enforcing constraints from map using map matching technique – multiple uncertain routes engine (MURE). MURE is an extension to the Kalman filter that allows location to be described using multiple discrete Gaussian distributions bound to a map. The developed map aided pedestrian dead reckoning (PDR) system was field tested in different buildings. It yielded accurate matching results as well as a significant enhancement in positioning accuracy. Experimental results demonstrate that the mean absolute position error is less than 1.3 m and 95% confidence interval is between -3.16 m to 3.32 m.

To further improvise the performance of map aided PDR system an extension to map based positioning is proposed via using landmarks. Landmark based positioning uses human as a sensor to sense proximity to landmarks. Landmarks are nothing specific as such but objects that are unique enough in comparison to the adjacent items e.g. quick response (QR) codes. Experimental results demonstrate that when map based positioning is used in addition to landmark based positioning the mean absolute position error is less than 1.0 m and 95% confidence interval is between -2.0 m to 2.0 m.

Smartphones are mostly held in hands however these can be used as a lieu to dedicated wearable gadgets e.g. smart glasses that contain the similar set of sensors as smartphones. Hence, we investigate a scenario similar to smart glasses via smartphone mounted on helmet. The thesis concludes that in principle it is possible to logically localize a pedestrian within buildings using the inertial sensors embedded in smartphone. The algorithms developed in this thesis are suited to cases in which it is impossible or impractical to install large amounts of fixed infrastructure into the environment in advance. Also, methods proposed in this thesis are applicable in indoor tracking applications.

Contents

List of Figures

List of Tables

DECLARATION OF AUTHORSHIP

I, Shashank Kumar Gupta, declare that the thesis entitled 'Infrastructure Free Localization of the Pedestrians within Buildings', and the work presented in the thesis are both my own, and have been generated by me as the result of my own original research. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University,
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
- Where I have consulted the published work of others, this is always clearly attributed;
- Where I have consulted the published work of others, this is always clearly attributed;
- Where I have quoted from the work of others, the source is always mentioned. With the exception of such quotations, this thesis is entirely my own work;
- I have acknowledged all main sources of help;
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
- Part of this work has been published as listed in Appendix A**.**

Signed: Shashank Kumar Gupta

Date: **30th November 2016**

Acknowledgement

First of all I am extremely thankful to the ‘Almighty’, for always guiding me and giving me the inner strength to follow the righteous path.

I would like to express my deepest gratitude to my advisor, Dr. Simon Box, for his excellent guidance, patience, and providing me with an excellent atmosphere for doing research. Your broad knowledge and experience has been a great guiding light throughout this journey. Thank you for your never ending ability to find time to assist despite your extremely busy schedule. I could recall the ‘daily to-do list’, ‘setting up work-plan in Asana’ and ‘taking a short break’ as few of the bullet points that you have certainly taught me and would always remain imbibed in my work cycle.

I should like to thank my co-advisor, Professor R. E. Wilson for his support and advice. Your sincere advice and guidance in meetings have helped me a lot in solidifying my basics in this field and to think for a bigger picture. Also, understandably there was a phase shift for you from this university to the University of Bristol, so our meetings were little less. However, I am very thankful for your advice and guidance throughout.

I should like to specially thank Dr. Ben Waterson (my internal examiner) for providing me with the challenges and showing me the areas of improvement. Importantly, I would like to thank all the Transportation Research Group members especially Melanie Hallford for providing the needed support throughout.

During these years I have been lucky enough to have some of my friends as colleagues. I would also like to acknowledge them, especially Varghese Antony Thomas.

My parents have always been an inspiration for me, and thanks to both of you for always providing me the moral and financial support. I would like to thank my loving sister who has always been with me through my thick and thins. A very special thanks to both of my in-laws for their extraordinary support. Importantly, I would like to thank my loving and caring wife, Dr. Poonam Gupta without whose support this work may have not been completed.

And last of all, I would like to greatly thank my grandparents and dedicate this work to my late granny.

Acronyms

A-GPS	Assisted GPS
AOA	Angle of Arrival
APs	Access Points
BP	Back Projection
BTS	Base Transceiver Station
CDMA	Code Division Multiple Access
CID	Cell Identification
csv	Comma Separated Values
DR	Dead Reckoning
EDGE	Enhanced Data Rates for Global Evolution
FFT	Fast Fourier Transform
GBP	Great British Pound
GNSSs	Global Navigation Satellite Systems
GPRS	General Packet Radio Service
GPS	Global Positioning System
GSM	Global System for Mobile Communications
HMM	Hidden Markov Model
HSGPS	High-sensitivity GPS
ID	Identifier
IMU	Inertial Measurement Unit
INS	Inertial Navigation Sensors
IRNSS	Indian Regional Navigation Satellite System
ISM	Industrial Scientific and Medical
KF	Kalman Filter
KNN	K-Nearest Neighbour
LAN	Local Area Network
Lidar	Light Detection and Ranging

MAE	Mean Absolute Error
MEMS	Micro-electronics Mechanical System
MURE	Multiple Uncertain Routes Engine
1D	One Dimensional
PDA	Personal Digital Assistant
PDF	Probability Density Function
PDR	Pedestrian Dead Reckoning
PF	Particle Filter
POIs	Point of Interests
QR	Quick Response
QZSS	Quasi-Zenith Satellite System
RAM	Random Access Memory
RF	Radio Frequency
RFID	Radio Frequency Identification
RSS	Received Signal Strength
RSSIs	Received Signal Strength Indicators
SD	Secure Digital
SDK	Software Development Kit
2G	Second Generation
TDOA	Time Difference of Arrival
TOA	Time of Arrival
3D	Three Dimension
3G	Third Generation
2D	Two Dimension
USB	Universal Serial Bus
UWB	Ultra-Wide Band
USD	United States Dollar
VGA	Video Graphics Array
Wi-Fi	Wireless Fidelity

WLAN	Wireless Local Area Network
XML	Extensible Markup Language

1. Chapter 1: Introduction

With the growing trend towards a world where people, vehicles, and other mobile objects are getting more and more interconnected, location information is increasingly becoming a recognized need for providing rapid and timely information to the mobile workforce. In the context of security personnel, as stated by Callmer [1].

"Be it fire-fighters, soldiers or police officers, being able to track the position of each individual user in real time while in a building, is the dream of the operational management. In case something urgent happens, knowing where all the personnel are and where they have been, enables swift and accurate cooperation to solve the problem. Having a positioning system would therefore greatly enhance the safety and efficiency of the personnel." In the context of visitors and students, as stated by Wang *et al.* [2].

"University campuses have thousands of new students, staff and visitors every year. For those who are unfamiliar with the campus environment, an effective pedestrian navigation system is essential to orientate and guide them around the campus." The development of pedestrian navigation system necessitates an effective pedestrian localization system too [3]. In the context of health professionals, as stated by Want *et al.* [4].

"Hospitals, for example, may require up-to date information about the location of staff and patients, particularly when medical emergencies arise."

Evidence suggests that the location revolution has been underpinned by the deployment of Global Navigation Satellite Systems (GNSSs) such as GPS [5]. Google, Yahoo, Microsoft, Facebook almost every big player in the Web 2.0 economy have introduced a location based service [6]. Moreover, small start-up firms such as Foursquare [7], BrightKite [8] or RallyUp [9] all have launched distributed, mobile solutions based on a user's location thus starting what infamously became known as the "location war" [10].

However, all these positioning solutions are built using systems that provide only very coarse positioning, which, in practice, allow for outdoor use only. SatNavs [11] are now almost standard equipment for the motorist and even play a large role in applications as diverse as agriculture and surveying. Largely driven by the inclusion of GPS receivers in smartphones, global shipments are predicted to exceed 1 billion units per annum by 2020 [12]**.** However the growth of mass market `Location Based Services' is hindered by the technological limitations of GPS. Most significantly, GPS does not provide a robust and accurate position solution to a user trying to navigate where they spend most of their time – indoors [13]. As mentioned by Takahashi [14].

"*Indoor navigation is one of the last great technical problems that hasn't been solved."*

Certainly, the topic of indoor positioning does not lack research, several approaches and suggestions based on different technologies have been developed and introduced over the last two decades [15]. An extensive review of these is done in Chapter 2. What is common for all of them is that no approach has made a big impact within the area. A number of systems have even been commercialized, produced and installed in different quantities but none of these have yet made a big penetration into the society.

One important development has been the emergence of small, ultra-low power cellular mobile phones [16]. But, as we will see in more detail later on, while cellular enabled positioning works well in outdoor environments however indoors, cellular signals get scattered, absorbed and offer suffer from the problem of multipath. As a result alternative solutions are required for positioning inside buildings. Evidence of this development can also be found in the sudden rise of web-based positioning services like Navizon [17], Skyhook [18], etc. that allow to determine location using Wi-Fi signals. Even more so, many big players in the consumer market, e.g. Apple iBeacon [19], provide location-based services for their mobile platforms using Bluetooth, developing and running the required software in-house.

However, the question still remains that why none of the indoor positioning systems have yet made a big penetration into the society?

After evaluating the built systems, we came to conclusion that the main reason for this is that the proposed solutions, although can be very accurate (less than 1 m) and even low cost (less than 30 GBP), have one thing in common that it is very challenging to find a balance between accuracy and cost. On top of this, the more complex challenge is to develop a solution that is both low cost (affordable by all less than 30 GBP or may be free) and reasonably accurate (less than 1.5 m) too. Specifically, an accuracy less than 1.5 m because for most indoor applications e.g. search and rescue mission, indoor tracking, augmented reality applications, disaster management applications, visually impaired navigation, etc. accuracy better than room level (2 – 5 m) is adequate [20, 21].

Cricket system [22] locates pedestrians with an accuracy less than 1 m through a combination of radio frequency and ultrasound systems installed in the surrounding. Nokia deployed Bluetooth based beacon transmitters to locate pedestrians in different rooms of the lab [23]. Deploying special infrastructure adds an extra cost which was however taken care by the company. Navisens [24] have developed a software stack to estimate pedestrian's position with an accuracy less than 2 m using only the inertial measurements from smartphone sensors with zero infrastructure (no Wi-Fi, no GPS and no Bluetooth) cost. CEO Ashod Donikian however comments that if the position estimation is not done over a long time, the cumulative errors can grow which may make the positioning ambiguous. Thus the problem remains as such.

On further filtering the literature, we noticed that there are some other limitations too which prevent wide scale implementation of indoor positioning systems. These are as follows:

i. Localization using pre-existing infrastructure and specialised hardware: It is not always that the infrastructures such as Wi-Fi radios, Bluetooth beacons, etc. are already pre-installed across the buildings. Since, many a time building administrators may be reluctant to add additional infrastructures specifically for the sole purpose of indoor positioning as these deployments incur heavy costs on them. For an area of 312 m^2, King *et al.* [25] installed 9 Wi-Fi access points so that the Wi-Fi signal could be sensed everywhere in the building. The cost of Wi-Fi infrastructural setup was about 1300 GBP excluding the hardware cost – network card.

   In general, positioning by these infrastructure require an additional hardware to be carried by pedestrians for positioning e.g. network card in previous system [25] that further adds to the overall cost. Consequently, one of the challenge that this thesis would address is to how to reduce the usage of infrastructure (e.g. Wi-Fi radios, Bluetooth beacons, etc.) and dedicated hardware (e.g. network card, cameras, etc.). Users already carry mobile devices (a personal smartphone), and may be unwilling to carry an additional device specifically for localization. Localization technologies need to rely on minimum hardware and infrastructural requirements.

ii. Complexity: An indoor positioning system may be built to work all over the world, within city limits, throughout a campus, just in a particular building, or even just in one room and systems can often expand to a larger scale by increasing the infrastructure. For instance, a simple tag system like the Active Badge location system [4], which locates tags in a single room, can be used on a campus by equipping all buildings with the required infrastructure. But barriers to positioning do not only include infrastructure but also middleware complexity.

   This complexity is mostly associated to software complexity [5, 26] i.e., complexity of the positioning algorithm. Rai *et al.* [27] locates the pedestrian using Wi-Fi and inertial navigation system (INS). System combines technologies using particle filter. 2000 particles are used for each step update. Although system provides an accuracy less than 5 m in 90% of the time however position computation when performed with slight modifications in a different environment the positioning accuracy degrades (less than 9 m in 90 % of the time [28]). Hence, reliability is low. Such a system may be suitable for some laboratory testing but

not practical for actual situations e.g. firefighter tracking, patient tracking, etc. in indoor environment. The other challenge that this thesis would therefore address is to how to provide reliable location updates, while keeping the software complexity low. Algorithms that are light in nature, shall be developed in this thesis.

iii. Privacy: Lack of privacy of location information could provide knowledge of activities of any individual whose location can be unobtrusively tracked. Service provider who know location of users can exploit location information to provide location dependent services not wanted by users. This is sometimes referred as user personalization. Personalization combining logging information and location information can have serious outcomes [29]**.** Thus the challenge here lies in providing anonymous position estimates. This is necessary, as the potential of data mining is very high [30]. Using suitable heuristics, as for example correlating a person's often visited locations, even position data can be correlated [31].

For some applications e.g. Active Badge [4], it is possible to keep user's location undisclosed if the calculation is carried out in mobile device's side. However, the problem is that in case of positioning in complete building, the computation cost becomes too large because of which responsiveness of the device decreases [32] and as a result the performance of positioning system degrades. We feel that undoubtedly, there is a trade-off between performance and privacy and moreover location tracking poses a serious imminent privacy threat. Consecutively, the other challenge that this thesis would address is how to maintain the privacy of users confidential?, without degrading the performance of system. Methods, running the localization algorithm on the smartphone device would be used to keep user privacy intact in the thesis. Low energy sensors and smart sensor fusion techniques would be further used to keep the computation cost low.

iv. Logical localization: If physical localization is employed to compute the pedestrians's logical location, the technology used needs to provide very accurate location estimates. The reason is that two rooms (e.g. a Starbuck's coffee shop and a Costa coffee shop) can be very close to each other, separated only by a *dividing wall*. A physical localization error of just 1 m, can localize the pedestrian on the wrong side of the wall, and thus, place the pedestrian in the wrong logical context. Likewise, if a pedestrian is in some store (e.g. departmental store) or in a train where there are several parallel bays, an error margin of 1 m may place the pedestrian on a different bay.

While very accurate (less than 1 m) indoor localization is possible using specialized hardware [25, 33, 34] and dense infrastructure deployment [4, 35], these solutions are difficult to deploy on wide scale. Alternate solutions are required to identify the user logical location accurately while obviating the need for specialized hardware and infrastructure deployment. A pioneered work in this direction is SurroundSense [36] that combined effects of sound, light and colour to identify the logical location. They hypothesised that every store in a shopping complex have different ambience – sound, light and colour e.g. ambient sound in Starbucks from coffee machine and microwaves, may be different from the sound of spoons clinging from the adjacent restaurant, lighting style in a ZARA shopping store may be different from the NEXT shopping store, etc. that when combined together can be used to provide logical localization.

Their hypothesis can be true for some cases e.g. shopping complexes, airports, etc. but not always. Two similar hardware stores can be side by side or there can be Government policies to keep the lightings minimum and background colours similar in stores in shopping complexes moreover, in University buildings, trains or in hospitals most of the rooms and compartments have similar ambience except varying in the size and floor geometry [37, 38]. We argue that by using map matching technique in addition to smart sensor fusion techniques and inertial navigation techniques we may not only be able to identify which side of the *dividing wall* they are present (provided initial location is known) but also and more importantly, on which bay of the store they are present.

To summarize, we need a localization system that (i) operate on mobile devices (e.g. smartphones), (ii) requires minimum infrastructure installations (including minimal dedicated hardware usage) or none at all, (iii) do not intervene privacy, (iv) reliable, and (v) provide accurate logical localization while keeping the overall cost low. The main aim of this research work is therefore to investigate how the information from smartphone sensors can be used to provide accurate, reliable, low cost logical localization. A significant emphasis is given on user privacy and minimal usage of infrastructure (including zero usage of dedicated hardware). To achieve this, the following objectives are formulated:

- To critically asses different existing indoor positioning technologies, systems, and combinations of those.
- Analyse the performance characteristics of smartphone embedded sensors. The focus will be primarily towards inertial sensors.

- Understand whether the positioning is possible using inertial sensors embedded in smartphone.
- Understanding the movement pattern of pedestrians, since the unconstrained movement of pedestrians is often varied.
- To develop an intelligent sensor fusion technique for merging the multimodal data from inertial sensors.
- To develop a map database for indoor localization. Generally a building is represented by a 2D plan, missing alphanumerical data regarding doors, furniture, staircases and so on. This representation is not convenient for the needs of localization [39]. Specific map database must be created representing the building as a spatial graph.
- To develop a map matching algorithm for projecting the position fix from sensor fusion technique into the map database of studied region.

In addition to these objectives, we also formulated two more objectives to further enhance the performance of developed map matching algorithm. These are as follows:

- To develop a landmark matching algorithm for projecting the position fix from sensor fusion technique into the map database of studied region while integrating the knowledge of artificial landmarks – quick response (QR) codes, present in the route.
- Compare the developed map matching algorithm and landmark matching algorithm.

Our work extends state of the art by exploiting the multitude of sensors embedded in smartphones. Low-energy inertial sensors, such as gyroscopes and accelerometers, are employed to localize the pedestrian. Indoor maps, designed in form node-link model, are used to correct noisy sensor measurements. A map matching algorithm – multiple uncertain routes engine (MURE) has been designed to estimate and predict the likelihood of a pedestrian on various links in a map. In addition, a landmark matching algorithm is designed which further extends the performance of MURE. The key contributions of this research work are as follows:

- **Localization using single inertial sensor:** We devised a case study to investigate the usage of smartphone embedded single inertial sensor – gyroscope to position a moving object (toy vehicle) on an artificial test bed – 'Scalextric track'. This kind of case study is first of its kind. It was reasoned that if an object moves in a repetitive manner in a pathway, and moreover the geometry of pathway is known a-priori then we can localize the object using only the single inertial sensor. An experiment has been conducted to verify this concept. The obtained results show that position error does not grow with time unlike the other traditional inertial navigation systems (INS). We also found that the fidelity of the localization is higher at the turnings where the heading is more informative.

- **Analysis of techniques for processing the inertial signal from smartphone sensors and extracting human gait characteristics:** In contrast to the traditional pedestrian dead reckoning (PDR) positioning systems, the smartphone based PDR positioning system is different. Since, smartphone orientation is non-static and can vary with time; also the smartphone sensors are of relatively low quality [40, 41]. Therefore, off the shelf stride estimation methods (foot step detection techniques and stride length estimation models) do not work for this case. So we extensively studied the characteristics of acceleration signal in time and frequency domain to extract the human gait characteristics. Some of the results have been published in [42]. We also evaluated a case when the smartphone is placed farthest from head and torso. Such a case has never been reported before. In addition we also found that when an individual walks normally they vary their walking speed little less and also their body shakes a little less.
- **A novel probabilistic map based positioning algorithm:** The geospatial data model is designed in the form of a network map such that the main characteristic of human made indoor environments, namely the parallelism and perpendicularity between the walls are preserved. A complication of the network representation arises from the fact that pedestrians are constrained to be on one of a number of discrete edges of the network. This discretization does not immediately lend itself to the use of the Gaussian distribution and the Kalman filter, which are powerful tools for dynamic localization in continuous space. In particular, if there is an ambiguity over which edge a pedestrian is located on, a single unimodal probability distribution is not suitable to represent the position of the pedestrian. In this case we propose a scheme ‘multiple uncertain routes engine (MURE)’ to keep track of multi-modal discontinuous probability distributions in a network map. Several experiments have been performed to verify the feasibility of this scheme. The obtained results show that this scheme not only has the ability to localize the pedestrian but also it estimates and predicts the likelihood of a pedestrian on a link in a network map. In nutshell, it can logically localize the pedestrian that is it can differentiate on which side of the dividing wall a pedestrian is present.
- **A novel method for pedestrian localization using landmarks:** As our technique of map matching is dependent on detection of corner features, it is equally possible that the corner features are distantly separated or potentially absent. This may cause the potential ambiguity in distribution amongst links to persist over a long time and moreover position error to grow. To deal with this issue, we rely on opportunistic encounters with artificial landmarks – quick response (QR) codes that when observed by the pedestrian triggers the landmark matching algorithm to run in system. The landmark matching algorithm

behaves like a look up table that calibrates the positioning system on detection of an artificial landmark. In addition, it rules out the extraneous distributions, and enables merging while confirming the pedestrian whether they are on the actual path or not. Such a method has never been reported before. Several experiments have been performed to investigate the feasibility of this method. Results show that the position error is significantly less using this method in comparison to the other methods reported in the literature. A significant drawback of this method is that QR codes affect the user experience. In addition, it requires time and effort to articulate the environment with QR codes therefore, its usage should be minimised.

- **A helmet mounted PDR system**: In general most of the sensors in gadgets (e.g. Google smart glasses [43], Oculus Rift [44], etc.) that employ user's location, are similar to the smartphone sensors e.g. inertial sensors, camera, etc. In addition they have the similar connectivity e.g. Wi-Fi, Bluetooth, etc. that can successfully provide useful navigation information. However, it is not always possible for security officers in a hospital to carry such gadgets for positioning even a builder or a hospital owner may not always be willing to purchase such costly gadgets for every worker just for the sake of positioning. Instead most of the officers wear some sort of headgears therefore we investigate how smartphone could be incorporated into this headgear and thereby be used for the purpose of localization. Such a scenario has never been investigated before. We have emulated a scenario by rigidly attaching a smartphone to a bicycle helmet. We have found that smartphones mounted on helmet can be an alternative option to locate pedestrians (e.g. construction workers) at lesser costs in comparison to wearable gadgets e.g. Google glass.

1.1 Structure of thesis

The document is structured as follows:

- Chapter 2 (Literature review): An extensive literature review is provided in this chapter. The underlying existing positioning technologies, methods and systems relevant to pedestrian's localization are introduced and discussed. As will become apparent, accurate, reliable, low cost logical localization of the pedestrians is still a problem; although there have been prior works in this domain however these are constrained to some typical indoor settings and moreover require heavy deployment of infrastructures.
- Chapter 3 (Case study: feature based localization): In this chapter, I describe the smartphone architecture and the characteristics of each of the built-in sensor used in this work followed by the performance analysis of these sensors. Based on the analysis, a case study is presented to localize a smartphone mounted on top of a toy vehicle on an artificial testbed (a Scalextric track).

- Chapter 4 (Inertial pedestrian dead reckoning): In this chapter, I describe inertial pedestrian dead reckoning (PDR), a technique which has been used in many projects in recent years to build pedestrian tracking systems. Several algorithms used for detecting pedestrian's footsteps and stride lengths have been detailed, implemented and analysed on smartphone based PDR system.
- Chapter 5 (Helmet mounted PDR): In this chapter, I describe an alternative to smart glasses for positioning. A Smart Glasses simulator has been designed in this context using smartphones mounted onto helmet, considering the fact that smartphones have the similar set of sensors and connectivity as smart glasses.
- Chapter 6 (Map aided PDR): In this chapter, I describe different types of map and how these are used for positioning. A map based positioning technique has been designed in this context to curb the growth of positioning errors in the earlier developed smartphone based PDR system. In particular, a topological map of the building has been designed in form of nodes and links. Following this, a map matching algorithm – multiple uncertain routes engine (MURE) has been designed and implemented on the smartphone based PDR system to monitor the position of a pedestrian at several locations simultaneously.
- Chapter 7 (Landmark aided PDR): In this chapter, a landmark based positioning technique has been designed and implemented onto the smartphone based PDR system. Landmark based positioning extends the prior developed map based positioning. Landmarks in form of QR codes are stored in landmark database which when sensed by the pedestrian, initiates landmark matching algorithm to run and subsequently reset the position error.
- Chapter 8 (Discussion): In this chapter, I discuss how findings from this research study complement to the existing body of research.
- Chapter 9 (Conclusion and future work): In this chapter, I conclude this thesis by summarizing the key findings of this research work and, highlighting some advantages and disadvantages of the developed technologies. Also, I would be giving directions of the future work and potential applications of this research study.

2. Chapter 2: Background and related work

In this chapter we present an overview of different techniques and mechanisms used for indoor positioning. With the advent of GPS, compact computers and inertial sensor systems in the 1980's, it started to become possible to envision man-carried systems for accurate positioning. Since then, many techniques have been proposed to solve the problem of indoor pedestrian positioning. At the time of writing of this thesis it is not clear whether this problem of providing accurate, reliable, low cost logical pedestrian localization within buildings has been completely solved.

Therefore we present an extensive review of the technologies used for indoor positioning and identify their pitfalls. Several positioning technologies can be combined using sensor fusion methods. Over the course of this chapter, we also investigate and examine related work where it is appropriate and shall identify the research gap. The first section of this chapter presents the attributes that are used as a measure for positioning. Section 2.2 reviews the existing systems and methods used for positioning. Section 2.3 details about sensor fusion methods. Section 2.4 details about the commercial indoor positioning applications available in the market. Section 2.5 summarises the key findings of this chapter and identifies the research gaps in the state-of-art of indoor positioning.

2.1 Attributes

Naturally, the position should be as precise, secured and accurate as possible. However, since every positioning system inherently determines the location with a certain error, the user of this information wants to know how big this error actually is moreover, is the information secured or not. Speaking of location information, some of the important attributes are privacy, accuracy and precision [5]. In addition, to this, a user wants its positioning system's cost to be as low as possible. These attributes are described below.

2.1.1 Accuracy and Precision

Accuracy can be defined as the degree of conformance between the estimated or measured position of a point at a given time [5]. It is the error difference between the position estimated by the positioning system and the actual position. Precision however denotes the distribution of all measurements i.e. how consistently the system works. This is also alternatively referred as reliability in some studies [45]. Figure 2.1 illustrates the difference between the accuracy and precision. Nevertheless it is possible to trade less precision for increased accuracy [46]. Consequently, these two attributes must be processed in a common framework in order to compare and rate them [45].

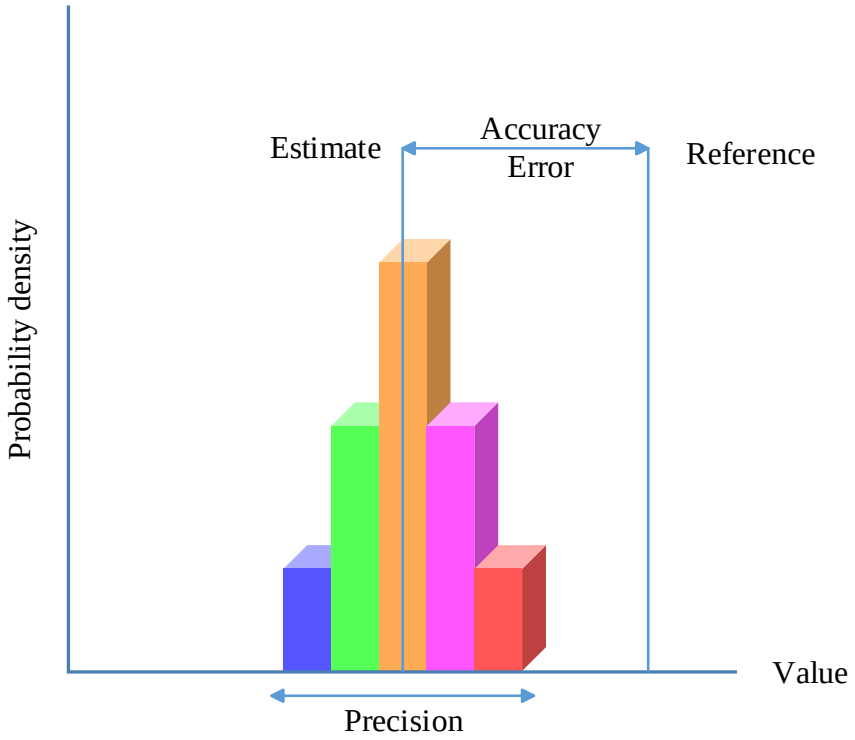


Figure 2.1: Location error, accuracy and precision.

2.1.2 Cost

One of the factors that make up the cost of an indoor positioning system, is the computation cost that is how much time is needed to report the location. Other time factors are the effort required to install and administer a system [5, 45]. Space is another important cost factor that is the extent and complexity of installed infrastructure and the used hardware's size and form factor. Energy is another important cost factor of a system [5]. Some mobile units (e.g. passive RFID tags, which are addressed later) are completely energy passive. These units only respond to external fields and, thus, could have an unlimited lifetime. Other mobile units (e.g., devices with rechargeable battery) have a lifetime of several hours without recharging. Above all, the capital costs, it includes factors such as the price of the devices used, infrastructural cost and salaries of support and maintenance personnel [47, 48].

2.1.3 Privacy

The level of privacy influences the approval by the pedestrian [49]: How comfortable are pedestrian with their data (e.g. trajectory) being stored? Do pedestrians have legal concerns about their privacy? If so, can pedestrians be motivated to provide personal data. Approval also includes the requirements for the system to allow certification by authorities. For e.g. if there is a need for admissibility in court, the requirements for the system to deliver evidence should be given.

2.2 Systems and methods for personal positioning

Smartphones integrate a variety of sensors that can be used to determine the position of a pedestrian. Specific positioning methods of how a position can be derived independent from the underlying technology is described. Later, key technologies using 1) radio frequency waves 2) magnetic fields 3) inertia 4) landmark and 5) sound waves are introduced.

2.2.1 Positioning methods

Positioning methods are independent from the used positioning technology [50]. Different positioning methods can be potentially employed by a specific positioning technology. Particular attention is paid to the capability of the implementation of these methods with sensors that commonly integrate into actual smartphones.

Three different categories of positioning methods are typically used for positioning systems. These are based on the Proximity detection, Triangulation and Scene analysis as illustrated in Figure 2.2. These will be detailed in the following sub-sections.

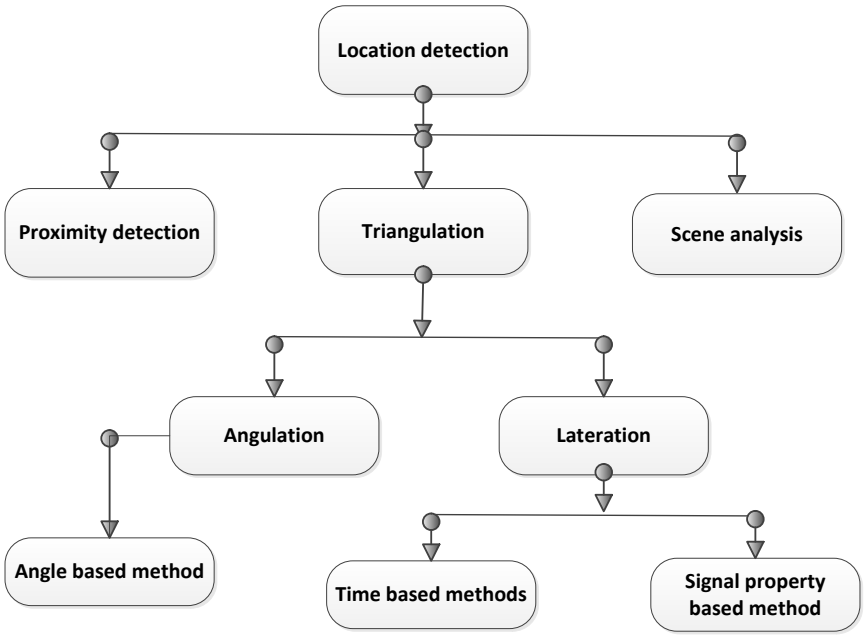


Figure 2.2: Classification of location detection methods (Source: [51]).

2.2.1.1 Proximity Detection

The proximity detection technique examines the location of a target object with respect to a known position or an area. A set of detectors are fixed at known locations; so when a target object is sensed by a detector. It is thereby considered to be present in the proximity area of that detector. In case multiple detectors sense the target the position of target is calculated by combining information of multiple detectors using different algorithms [52]. Figure 2.3 illustrates a simple scenario of proximity detection where two moving targets G and H are monitored by a detector D.

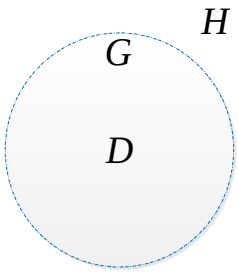


Figure 2.3: Illustration of proximity detection principle.

The detector D reports target G to be in its proximity. While it reports target H to be in non-proximity considering the fact that target G is within the proximity area of detector D shown by a dotted circular ring. In case multiple targets are present in close proximity, the detector would report their positions based on the measured signal strength. The closest target would be one whose signal strength sensed by the detector is highest. One major advantage of this method is that infrastructure is already deployed (in form of cellular networks) so implementation cost is very low or almost zero. However accuracy is significantly affected by dimensions of the cell, the range of signals and the density of detectors in a network [52].

2.2.1.2 Triangulation

Triangulation is a way of determining the target's location using the geometric properties of triangles. Figure 2.4 illustrates the basic principle of triangulation for a 2-D position measurement. Consider the geographical coordinates of three reference beacons A, B and C to be known. The R_A, R_B and R_C are the distances of the moving target from the three reference beacons A, B and C respectively. Or alternatively, these are the centre of circles A′, B′ and C′ respectively. Then according to the principle of triangulation the position of target object M would be in the area where each of the circles A′, B′ and C′intersect [53]. The triangulation is subdivided into two categories: (1) Lateration and (2) Angulation.

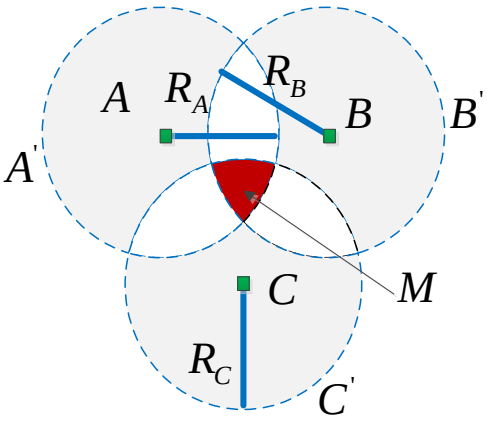


Figure 2.4: Basic principle of triangulation.

Lateration

Lateration computes the target's position by estimating the distance of the target from the reference beacons. If there are multiple reference beacons, it is multi-lateration. Trilateration is a specialised case of multi-lateration which uses three reference beacons [51]. Instead of measuring the distance directly using received signal strengths (RSS); time of arrival (TOA) or time difference of arrival (TDOA) is usually measured, and the distance is derived by computing the attenuation of the emitted signal strength or by multiplying the radio signal speed (usually the speed of light) and the travel time [5]. The lateration is further subdivided into two categories:

- Time based methods
- Signal property based methods

Time based methods

The time based methods involve the measurement of one or more parameters related to signal travelling time to compute the relative distance of the target from reference beacons. These are sub-grouped into two types:

- Time of Arrival (TOA)
- Time difference of arrival (TDOA)

The principle of TOA is based on the precise measurement of the arrival time of a signal transmitted from a mobile target to several reference beacons [5]. Considering the velocity of the signal to be known (usually the speed of light) the distance between mobile target and reference beacons is determined by multiplication of the signal travel time with the signal's speed. It is imperative that the clocks are very precise and synchronized at both the transmitter and receiver ends [54].

The principle of distance estimation by TDOA is also similar to the TOA. However, the receiver at TDOA does not need to know about the absolute time at which signal was transmitted - only the difference of arrival time from the transmitters is needed. To calculate the position of a mobile target hyperbolic lateration is used [55]. Figure 2.5 illustrates a scenario for 2-D position measurement of a mobile target. Consider a mobile target M that transmits message, which is received by two reference beacons A and B. If the message is received by A earlier than by B, then this implies that mobile target is closer to A in comparison to B.On the other hand, if B receives message earlier than A. This would imply the opposite i.e. B is closer to mobile target than A. As a result there is mostly a time difference between the arrival of message at A and B. This time difference between the arrival of message at A and B is mapped to a distance difference according to the velocity of the signal; which is represented by a hyperbola of possible locations of the mobile target. Identically, a third reference beacon (if present; consider it to be C) could form a second hyperbola. The intersection of both hyperbolas is the position of a mobile target.

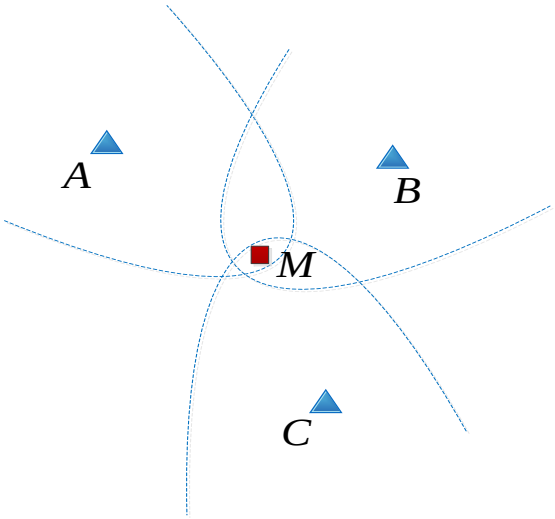


Figure 2.5: Illustration of TDOA method by hyperbolic lateration.

Signal property based methods

The signal property based methods measure the attenuation in received signal strength to compute the relative distance of a mobile target from reference beacons [51]. Various propagation models are used to translate the difference between transmitted signal strength and the received signal strength into a range (relative distance from the mobile target to beacon) estimate.

These propagation models are typically expressed in terms of path loss [56]. The path loss represents how much a signal is attenuated as it propagates through space. Total path loss is however a complex function with multiple components [57] such as propagation distance (free-space loss), signal properties (frequency), terrain (hills, mountains, bodies of water), etc. It is essential that propagation model should consider as many loss factors as possible to provide higher accuracy. However, it is often unfeasible indoors [58] because of the multipath effects from the surroundings. Consequently, there are multiple different propagation models proposed by various researchers varying in complexity and path loss components [59]. For instance, a model defined by International Telecommunication Union (for indoor settings) [60].

$$L_{total} = 20 \log 10\, f + N_L \log 10 d + Lf(n_f) - 28\ dB \quad (2.1)$$

where L_{total} is the total path loss, N_L is a distance power loss coefficient, d is the travel distance, f is signal frequency in MHz, $Lf(n_f)$ is floor penetration loss factor in dB, and n_f is the number of floors between transmitter and receiver.

Methods that are based on time measurements have to guarantee a good synchronisation between target and reference beacons. On the other hand, methods that are based on the signal strength have problems with interferences and reflection. Therefore, these are in general more suitable for outdoors than for indoors.

Angulation: Angle of Arrival (AOA)

Angulation computes the mobile target's position by estimating the angles relative to multiple reference beacons. In principle, it locates the target from the intersection of several pairs of angle direction lines, each formed by the circular radius from a reference beacon [54]. To measure the incident angle, the reference beacons are equipped with an antenna array capable of determining the angle from which the signal was received. If at least two reference beacons are able to determine this angle, the intersection of virtual lines drawn from the respective beacons heading towards the angle the signal was received from denotes the position of the emitter i.e. the mobile target [5].

In contrast to the other methods, the AOA requires only two reference beacons to estimate position in two dimension (2D) and three reference beacons for three dimension (3D) localization [58]. However, the AOA method is highly vulnerable to certain effects of signal distribution especially multipath effects of the signal. Since, this causes the signal to reach the reference beacons via different paths (i.e. across different angles). Consequently, direct line of sight is a prerequisite for flawless functionality of the AOA principle rendering it quite impractical for indoor scenarios [54, 61]. Additionally, AOA measurements require highly directional antennas to measure the angle of incidence at reference beacons. This increases both the cost of the system and beacons' size, making the system too large for indoors.

2.2.1.3 Scene analysis

The scene analysis refer to the type of algorithms that first collect fingerprints of a scene and then estimate the location of a mobile target object by matching some characteristics of that scene with the collected fingerprints. A fingerprint is a signature or unique characteristic that differentiates one scene from other [5, 51]. There are two stages for location fingerprinting 1) offline stage and 2) online stage. During the offline stage, signal strengths from access points are collected at pre-identified locations and stored in the form of a database. Next during an online stage, the currently observed signal strengths and previously collected fingerprints are matched to figure out the location of a mobile target.

Again the performance of this method is hindered by the multipath effects of signal [54]. Since, the fingerprint of a signal could change with the change in physical conditions of the environment [62]. As a result this would increase or decrease the signal reception paths at a particular point. Consequently, this causes the variation of received signal strength at a particular point, and thereby degrades the overall accuracy of positioning system.

2.2.2 Radio frequency (RF) based positioning

The RF based positioning employ radio frequency (RF) signals to position a mobile target. The RF signals are the electromagnetic waves generated by an oscillating electrically charged particle.

These waves are in practise transmitted by a fixed transmitter towards a moving target. One or more properties of these electromagnetic waves are measured at the receiver station to estimate the position of a mobile target [63]. Location acquisition is done using methods namely Proximity detection, Triangulation and Scene analysis. Few of the key positioning systems designed using RF waves are as follows:

- Global Navigation Satellite Systems (GNSSs)
- Cellular systems
- Wireless local area network (WLAN) systems
- Radio Frequency Identification (RFID) systems
- Ultra-wideband (UWB) systems
- Bluetooth

2.2.2.1 Global Navigation Satellite Systems (GNSSs)

Global Navigation Satellite Systems (GNSSs) aim to provide global positioning and navigation services. These systems allow mobile targets to determine their location (which is defined by longitude, latitude, and altitude) utilizing signals that are sent by satellites orbiting around the Earth. Few of the operational GNSSs are U.S.NAVSTAR Global Positioning System (GPS) [64] (see Figure 2.6), Glonass (Russia) [65] and IRNSS (India) [66]. Since GPS is the most widely used system and conceptually similar to the other systems, therefore this section details about it.

Originally designed and built to serve a military requirement for navigations, GPS is a satellite-based positioning technology that is currently being used by many civil users. Although only 24 satellites are needed to be considered as a full system, there are often more than 24 present as new ones standby to replace the older satellite [3].The GPS system works on the theory of triangulation. Each satellite broadcasts a RF signal including timestamp and the satellites coordinates to the receivers (mobile target) located on or close to the surface of the earth at every billionth of a second.

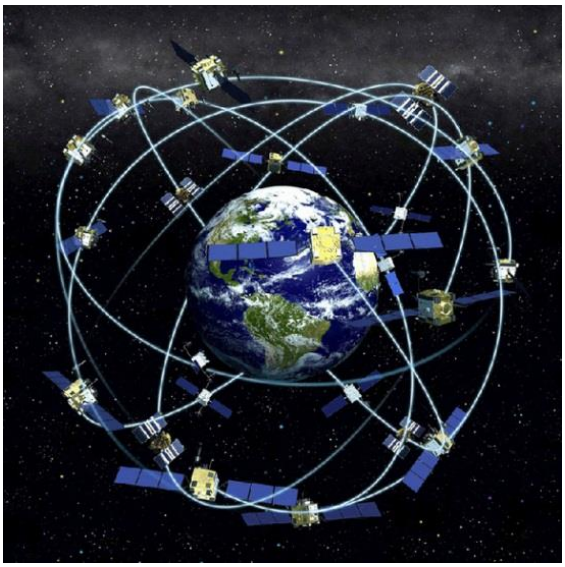


Figure 2.6: NAVSTAR: GPS Satellite Network (Source: [67]).

Receiver takes this information from the triangulation of at least four satellites and uses it to calculate its position. The mathematics of actual positioning calculations can be seen in [64]. Receiver, which is either a hand-held device, mounted on a vehicle, found in a plane or ship, compares its own time with the time sent by the satellites and calculates its distance between the satellites using the difference between the two timing. The final product of GPS readings, depending on the receiver, will pinpoint location in terms of longitude, latitude and altitude.

It has been observed in [68] that when GNSS signals are available, absolute position fixes within standard deviation of 2.5 m and mean error of 3.5 m can be attained using commodity, stand-alone, single frequency receivers outdoors. This performance is improved by integrating measurements from inertial measurement units (IMUs) and applying forward-backward smoothing in time as illustrated by [69].

Nevertheless, reception of GNSS signals indoors is difficult and error prone [70]. Assisted-GPS (A-GPS) techniques can dramatically reduce satellite acquisition times via the use of aiding information supplied by cellular network operators. However, this does not help in the acquisition of extremely weak signals typically found indoors. High-sensitivity GPS (HSGPS) receivers can often provide position estimates indoors by increasing the number of satellite observations [71] but the accuracy is generally poor (with errors often greater than 30 m) [72]. .

According to Lachapelle [73], even with new GNSS satellites (Galileo) signals (L2C and L5) as well as future low-cost micro- electronic mechanical system (MEMS) sensors and their ultra-tight integration with HSGPS, it will still be very difficult to get consistent accuracy indoors better than 10 m. Additional position aiding, such as from in-building instrumentation, short-range RF devices, and 3D maps of buildings will likely be required.

Locata Corporation [74], have developed "locata technology" to locate the receivers indoors and outdoors. It utilises a group of time-synchronized pseudolite transceivers that transmit GPS like signals to enable single point positioning using carrier phase measurements. Test experiments demonstrate positioning accuracy less than 20 cm. This can potentially provide logical localization but at the expanse of installing LocataLite devices (pseudolite transceivers, see Figure 2.7) in every room. The similar concept of pseudolites has been utilized by Xu *et al.*, in [75] to position a mobile receiver indoors. The pseudolites receive real world GPS signals, repeat each satellite signal and transmit them to the mobile receiver. The mobile receiver determines its position via four TDOA measurements with an accuracy of about 1 m in static conditions and slightly less than 1 m in dynamic conditions.

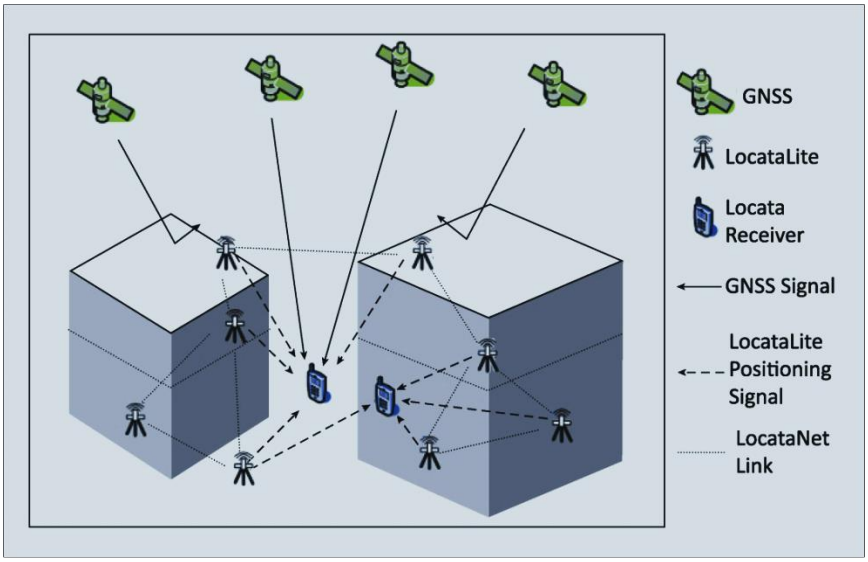


Figure 2.7: LocataLite positioning system (Source:[74])

2.2.2.2 Cellular systems

Cellular positioning systems are based on RF infrastructure transmitters that do not form contiguous network as GNSS however it covers a significant part of the Earth's landmass. The cellular networks are organized in cells (see Figure 2.8), which are built up by base stations called Base Transceiver Stations (BTS) and several mobile devices [76]. Each cell is identified by a unique cell Identification (CID) and covers a range from approximately hundred meters up to 35 km depending on the terrain and the estimated number of mobile devices the cell has to handle. The mobile device locates the nearest BTS by identifying CID and determines its position in a cell based on proximity detection method (see Section 2.2.1.1) or time based methods (see Section 2.2.1.2) [77]. For a review about cellular systems see Chapter 8 in [78]. Positional accuracy of these techniques typically depends on the density of base stations, network layout and the reliability of time of arrival measurements [55, 79].

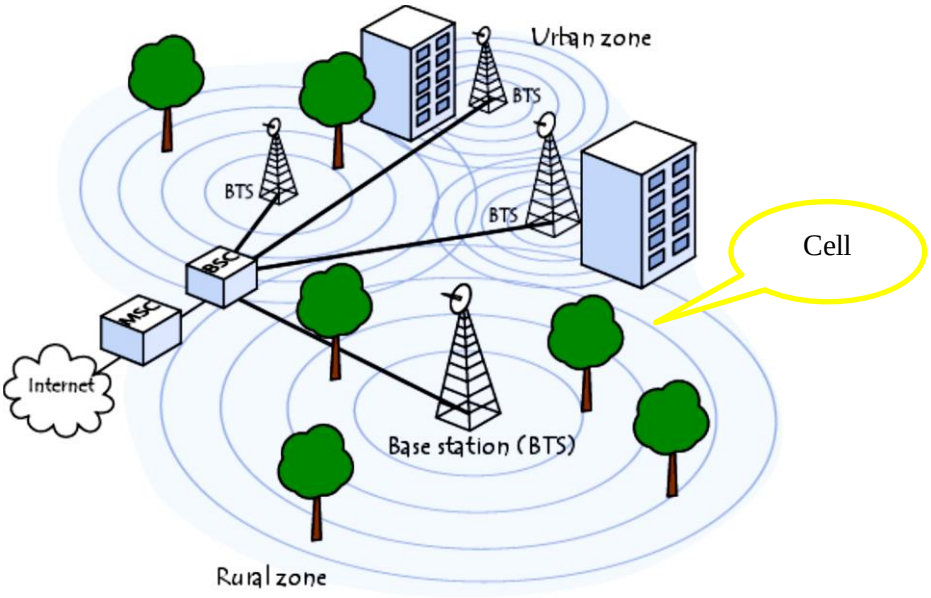


Figure 2.8: An illustration of cellular network (Source: [50]).

Angle of Arrival (AOA) is another algorithm adopted by cellular service providers, but is not as widely implemented. Technically, indoor positioning via cellular systems is practically possible only if signals from existing outdoor infrastructure can be detected which has proven to be difficult because of RF multipath and the need of direct line of sight from base stations [55, 80]. However, in few cases this has been achieved, but at the expense of location fingerprinting.

In [81] Otsason *et al.* localized pedestrians using existing cellular network – Global System for Mobile Communications (GSM) only. It involved measuring of wide signal fingerprints from 35 cells. These wide fingerprints included six strong GSM cells and readings of up to 29 additional GSM channels, most of which were strong enough to be detected but too weak to be used for efficient communication. The higher dimensionality introduced by the additional channel dramatically reduced the position error and moreover, improved the positioning accuracy but at the context of increased computational cost. The results showed that positioning system was able to differentiate between floors in both wooden and steel-reinforced concrete structures, and achieved median accuracy ranging from 2.48 m to 5.44 m in large multi-floor buildings.

Tian *et al.* [20] have combined Bayesian filtering along with machine classifiers to associate cellular telephone network received signal strength fingerprints for positioning. The authors used a sampling scheme in the training procedure, and the Bayesian filter to introduce information about room layout and pedestrian trajectories. Results indicate accuracy greater than 95% with Bayesian filtering and greater than 69% for non-Bayesian filtering over a period of 42 days, at the context of several days of former training. While in [82] authors have used the code division multiple access (CDMA) networks for identifying different floor levels and positioning

a pedestrian. The system works on the same principles of fingerprinting i.e. firstly the offline phase of collecting fingerprints and then online phase of determining location by searching for the closest matches of the current measurement to the set of measurements collected in the training phase.

Whatever specific positioning algorithm is used, the performance of cellular positioning depends greatly on the propagation conditions imposed by the wireless channel [3]. Consequently, reliability of the systems can vary with change in physical conditions. Nevertheless, cellular positioning remains of interest since it may be available when other signals are not, as well as for older devices that are not A-GPS and/or Wi-Fi enabled. As a result, the refinement of algorithms for cellular positioning remains an active area of research for indoor environment [83].

2.2.2.3 Wireless local area network (WLAN) systems

WLAN positioning systems are those RF based systems that aim to provide positioning services temporarily on an emergency sites or it could be part of a wider, ad hoc tactical communication network [84]. It operates in the 2-4 GHz Industrial Scientific and Medical (ISM) band with a gross bit rate of 11, 54 or 108Mbps over a range of 50-100 m, particularly dominated by Institute of Electrical and Electronics Engineers (IEEE) 802.11 standards – Wi-Fi [5].

The setup of WLAN is very much similar to the cellular networks (see Figure 2.9); where WLANs are subdivided into several cells called Basic Service Sets (BSS), and each cell is controlled by a base station which is referred as an access point [50]. Each access point broadcasts a signal that has a higher transmission power than GPS plus special signal structures allow for better penetration into walls of the buildings and less mitigation to multipath effects. As a result these are often preferred for positioning indoors [5, 85]. The broadcasted signals are received by wireless receivers. These receivers have a capability to measure the strength of received signal.

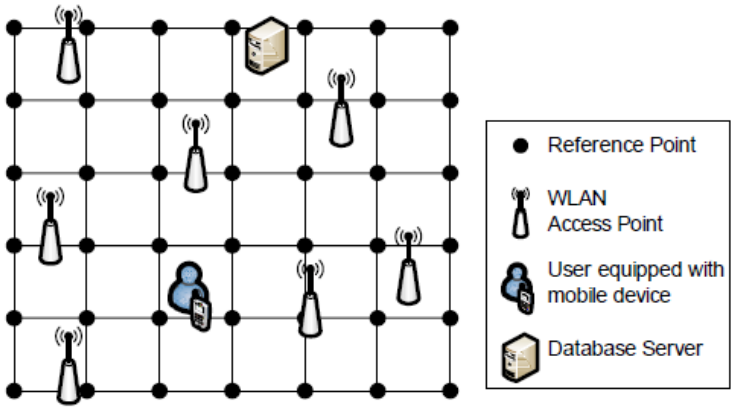


Figure 2.9: Basic WLAN setup (Source: [86]).

The received signal strength is converted into a number known as received signal strength indicator (RSSI) in a wireless receiver. Based on the RSSI values the receiver locates itself using proximity detection method (see Section 2.2.1.1), trilateration method (see Section 2.2.1.2) and scene analysis method (see Section 2.2.1.3) [51, 87]. The methods – trilateration and scene analysis rely on knowledge of the exact location of APs and/or the ability to model signal strength as a function of distance from the AP's location. This is however difficult for indoor implementation [54]. As a result, fingerprinting techniques have emerged as the preferred method for Wi-Fi positioning in the indoor environment. Instead, fingerprinting employs a calibration phase (also referred to as the training or offline phase) in which Wi-Fi signals are observed at known locations. The set of APs and their respective signal strengths presents a "fingerprint" that is unique to that location. In the positioning phase (or online phase) the observed Wi-Fi signals at an unknown location are compared to the database of previously recorded fingerprints to determine the closest match. Several matching techniques have been developed for this, including k-nearest neighbor estimation; support vector regression, Bayesian modeling, neural networks, etc. [88].

Performance varies with AP density and distribution, reliability of the positional reference database, and the positioning algorithm employed, among other factors. For a single building with a substantial number of APs, median horizontal accuracy between 1 - 5 m has been reported in [89, 90] provided the radio map is stationary over time. Bahl *et al.* [91] proposed an in-building pedestrian location and tracking system – RADAR using signal propagation models and pattern matching approach - K-Nearest Neighbour (KNN) method [92]. The results depicted an accuracy of about 3 m over 50% of the time using the pattern-matching approach and 4.3 m over 50% of the time using lateration techniques. Performance was however dependent on the number of data points considered and the orientation and speed of the pedestrian. Horus system [89], improved upon RADAR, employs a stochastic description of the RSSI-location relationship and uses a maximum likelihood-based method to estimate locations. Experimental results show that the average accuracy of the Horus system is better than the RADAR system by more than 82%. In both the systems Wi-Fi calibration was done frequently at many physical locations. Also, the performance of system varied with density of APs.

WILL [90] presents an indoor logical localization approach without site survey or knowledge of AP locations. Fingerprints are partitioned into different virtual rooms based on RSSI stacking difference. Fingerprints with high similarity are put into one virtual room. Different virtual rooms then construct the logical floor plans, which are mapped into physical floor plans using centrality of a vertex and shortest paths length between vertices. The implementation results show that WILL can achieve an average room-level accuracy of 86%.

Besides, WILL involves complex mathematical computation in addition it is also indicated that some virtual rooms are indistinguishable.

King *et al.* [25] proposed a probabilistic indoor positioning system – COMPASS based on the digital compass and WLAN infrastructure. The system estimates the position of a pedestrian by comparing the received signal strength fingerprints (from the access points) in the offline stage with the measurements done in an online stage. A pedestrian's orientation is measured by a digital compass to reduce the human body's blocking influence to the positioning process. Since as addressed by authors, the human body consists of more than 50 percent of water which blocks the radio signals and influences the measurement accuracy. As a result they have increased the number of signal strength samples in each position with different orientations. Experimental results show after a training of about 10 hours, the COMPASS system achieved an average distance error of less than 1.65 m in an area of 312 m^2. Locating multiple users at the same time has not been discussed moreover measurement errors always occurred close to electromagnetic objects such as high-voltage power lines and electronic devices. Thus, reliability is low.

An important limitation of WLAN is the attenuation of signal at wireless receivers due to movement of furniture, doors, human, etc. [93]. In [94] and [95] wireless location fingerprinting was performed to achieve a positioning accuracy less than 3.5 m, 90% of the time. The most pressing challenge was however the non-stationarity of radio map. This was reflected as the differences in the measured signals during the online and offline phases of same exact location.

Another emerging WLAN technology is Zigbee [96]. It operates on unlicensed bands including 2.4 GHz, 900 MHz and 868 MHz. The signal range coverage of a ZigBee network in indoor environments is typically 20 to 30 m. It has low energy consumption and is designed for multi-channel control systems, alarm systems, and lighting control. Distance calculation between two ZigBee nodes is usually carried out from RSSI values [51]. One of the concerning challenges with Zigbee networks is that it operates in an unlicensed band therefore it is often susceptible to interferences with the other radio networks. Hu *et al.* [97] have deployed a ZigBee network for indoor environments and have proposed an algorithm to mitigate such interferences. The system is sufficient to logically localize in a room. They have incorporated methods such as arranging reference nodes' position, calibrate RSSI values and lower the output power level including shutting down of irrelevant wireless nodes for accurate positioning. Experimental results show that in an empty closed room (with no human present) of dimension 56mx30m, the stationary node could locate the other moving node with a root mean square error less than 1 m and when humans were present the root mean square error was increased to 1.8 m. The cost of whole system (excluding prior time and effort in setting up the system) for achieving such an accuracy in a single room was approximately 400 GBP.

2.2.2.4 Radio Frequency Identification (RFID) systems

RFID positioning systems are those RF based systems that aim to provide positioning service on a complex indoor environment such as hospitals, offices, etc. [53] to locate movable or non-movable objects [98] including pedestrian, food items, drugs, etc. A summary scheme of how RFID positioning system works is shown in Figure 2.10.

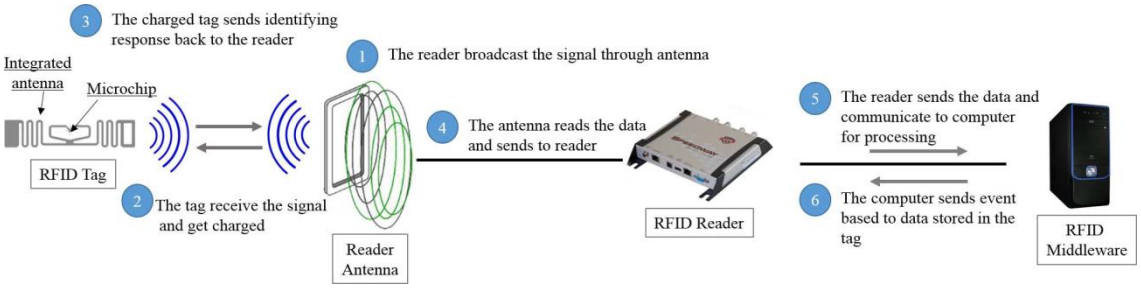


Figure 2.10: A schema of RFID positioning systems (Source: [98]).

The data is stored in form of RFID tags that comprise of a chip and an antenna. RFID tag acts as a transponder, responding to queries from a nearby transceiver (reader) by transmitting back its own unique 64-bit or 128-bit identifier via radio waves. The tags generally do not include a power source therefore power required for its operation is transferred by electromagnetic induction [99, 100]. Principally, RFID systems operate in three frequency ranges: low frequency (125 Hz-134 Hz), high frequency (13.56 MHz) and ultra-high frequency (860 MHz-960 MHz). Their performance is unhindered by non-line of sight conditions; since transmitted radio waves have the ability to penetrate solid materials. However, signal strength received at reader depends upon density of objects [98]. Based upon the received signal strength at reader, reader locates itself or tags using proximity detection method (see Section 2.2.1.1) or lateration method (see Section 2.2.1.2) [101, 102]. In case of reader localization, the positioning accuracy is highly dependent on the density of tags. Tesoreiro *et al.* [103] introduced a localization system based on proximity to RFID tags to position a museum visitor. The museum visitor carried a personal digital assistant (PDA) that obtained the identifier (ID) of RFID tags in the vicinity. Each tag was associated with an exhibit in the museum. The ID of the detected tag was subsequently transmitted to a server, which returned information about the exhibit in proximity to the user. The localization accuracy of this system was not reported. However, the accuracy was related to the density of tags in the environment. In addition, the privacy of a pedestrian was an issue since the position computation was done at server end. So there was always a possibility of leakage of the position information

The navigation system ‘ways4all’ developed by Kiers *et al.* [104, 105] included an array of RFID tags. These tags were deployed under the carpet to provide positioning and route guidance for visually impaired and blind people. RFID-tags send their code to the user‘s mobile

phone through a mobile RFID reader. The phone sends the code to a RFID-database server where all the tags together with some additional information are saved as location points. The mobile phone receives real-time routing information (including interruptions, delays and platform changes) from the database server. Experiments show, that the current maximal reading range of 30 cm needs to be extended to 60 cm in order to obtain a sufficiently high tag detection rate. This can be a potential option for logical localization. However again maintaining the privacy of user confidentiality was an issue. The project highlighted a serious drawback of RFID based localization i.e. there is no RFID standard, so not every RFID reader can be used. Instead, selection of RFID readers should be done cautiously.

On the contrary, tag localization necessitates readers to be placed at known locations. This makes the overall process of positioning more expensive. Hightower *et al.* [106] locates objects woven with RFID tags indoors with metre level positioning accuracy. The approach is generally a good option for logical localization. It utilised multiple readers to collect signal strength measurements that were subsequently used to approximate the distance from the objects (woven with tags) via mathematical function defined with empirical data. Although authors claim localization accuracy of 87–95% for classifying regions at 3 m resolution and 67% at 1 m resolution, the performance was significantly limited because of the delay in processing measurements. Also, the cost of whole system was approximately 1000 GBP including infrastructure cost, SpotOn board and cost of tags.

While RFID based localization solutions are in-expensive in comparison to the earlier discussed RF based positing systems and can achieve fine positioning accuracy (less than 1 m), a wide scale deployment however requires a dense infrastructure of tags (for mobile reader) and readers (for mobile tags) for positioning. In addition, RFID positioning systems are often limited by the extent of supplied power, coverage and need of sporadic location updates. This makes it non-preferable for general purpose indoor localization [107].

2.2.2.5 Ultra-wide band (UWB) systems

The UWB positioning systems have been in existence since late 1890s however technological limitations pushed the research towards the development of transmissions employing narrowband signals [108]. It was only in the 1960s that baseband sub-nanosecond pulses found their applications for radar devices, and subsequently their implementation began for communication, localization and radar. The UWB systems use an electronic tag/device which emits short UWB pulses (typically <1 ns and duty cycle of approximately 1: 1000) to track an object or person, and in some cases the UWB systems monitor signal changes introduced by the tracked object or person in the environment [109] to locate their position. In principle there are two approaches to estimate the position of a moving target [110]: 1) direct approach and 2) two-step positioning

approach. The two step approach estimates the position of a moving target at central unit from the information gathered by reference tags. However, in a direct approach the moving target self localizes as illustrated in Figure 2.11.

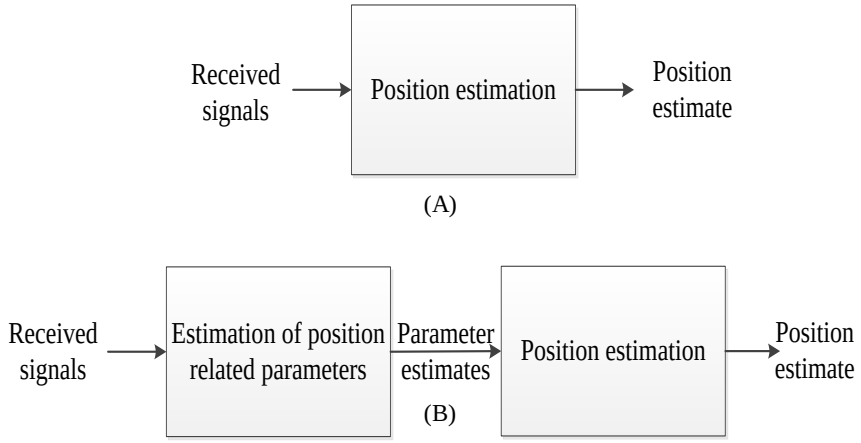


Figure 2.11: (A) Direct positioning (B) Two step positioning

Although the two-step positioning approach is suboptimal in general, it can have significantly lower complexity than the direct approach. UWB systems operate at multiple band of frequencies ranging from 3.1 to 10.6 GHz and use techniques that cause a spreading of the radio energy over a wide frequency band with a very low power spectral density (less than -41.3 dBm/MHz) [111]. This high bandwidth offers high data throughput for communication. The low frequency of UWB pulses enables the signal to effectively pass through obstacles such as walls and objects.

The high data rate of UWB can reach 100 Mbps, which makes it a good solution for near-field data transmission. Also, the high bandwidth helps in reducing the effect of multipath interference and other external noises caused by nearby physical devices such as doors, furniture, etc. This makes it suitable for general purpose indoor localization [112]. An important requirement for UWB positioning is precise clock synchronization between transmitter and receiver. This can potentially take long time. Therefore, the time based methods (see Section 2.2.1.2) and angle based methods (see Section 2.2.1.2) are commonly used to acquire range data for estimating position of the fixed or mobile target [113]. UWB systems can provide accuracy less than 1 m however the high accuracy comes with additional costs in setting up the system. Speaking up generally the UWB systems are ccommercially expensive compared to other technologies [114]. As a result their applicability is restricted to industries. Designing and implementation of UWB antennas is another challenging task that prohibits the usage of UWB indoors and outdoors [115].

Several indoor positioning systems are implemented commercially using UWB (see [112] for more details). One of the most known positioning systems that use UWB is Ubisense system [116]. The Ubisense system consists of three parts: the sensors, the tracked tags and the Ubisense software platform as illustrated in Figure 2.12. The tracked tags transmit UWB pulses that are received by sensors fixed at known locations. Then the location data of the tags is forwarded from these sensors via existing ethernet to the Ubisense software platform, which analyses and estimates the position of moving target (tags) via time based methods (see Section 2.2.1.2). The Ubisense sensors are organized into cells. In each cell, there are at least four sensors which cover an area of up to 400 m^2. The tracked tags are wireless, easily wearable, light weight (45 g) and have a battery life of about 1 year. The authors predict that Ubisense system can locate pedestrians with an accuracy of about 15 cm. However, the price of this high performance positioning system is also high. The whole package illustrated in Figure 2.12 costs about 11,000 GBP. UWB systems have also been used for important military applications where accuracy and reliability is important e.g. [117].

Figure 2.12: Hardware of the Ubisense System in an Academic Research Package: five tags (left) and four sensors (right) (Source: [53]).

Chen *et al.* [117] introduced a UWB short pulse radar for through-wall surveillance using the finite-difference time-domain. Results are based on simulated data for UWB short pulse radar trying to detect a box in a 2.36 m x 3.59 m simulated room with concrete walls. The range location of the target is determined by measuring the propagation delay time from the transmitting point to the target and back to the receiver point after accounting for the presence of the wall. Using the received field data at several observation points, an image of the target is constructed by using the Back Projection (BP) algorithm. They conclude that the proposed solution is a viable solution for a through-wall monitoring system however it is to be noted that directional antennas are required to collect the signals which itself is a costly venture.

Even though UWB has high immunity to multipath fading, the UWB signals get blocked too easily (40% of the time according to [118]) in various indoor environments, leading to the

possibility of the first pulse detected not being the line of sight path. The absence of the line of sight signal can easily create errors of a meter or two, which gives a localization accuracy similar to the other wireless-based solutions.

2.2.2.6 Bluetooth

Bluetooth is a wireless standard for Wireless Personal Area Networks (WPANs) and operates in the 2.4 GHz Industrial, Scientific and Medical (ISM) band. Bluetooth is functional only on a room or hall level, with a range of 10 to 15 m, and a bit rate of 1 Mbps. It is divided into three classes, each of which has a different range, as shown in Table 2.1. This range can potentially be affected by the surrounding environment, as the signals are susceptible to propagation effects [119]. This is especially true in an indoor environment. Although class 3 devices are ideal for indoor positioning purposes because of the small range, however such devices are very uncommon and the vast majority of available devices are of class 2.

Class	Range
Class 1	100 m
Class 2	10 m
Class 3	5 m

Table 2.1: Bluetooth range (Source: [119])

A device may connect to up to seven other devices simultaneously by setting up an ad-hoc network in which the device that initiates the communication acts as a master. Such networks can be set up without the need of user interaction. In order to avoid interference between devices, Bluetooth uses a technique known as spread-spectrum frequency hopping, in which a device transmits on one of 79 randomly selected frequencies, changing from one frequency to another 1600 times per second. This makes it unlikely that two devices transmit on the same frequency and therefore minimizes the risk of interference.

Positioning works on the similar principles as described for WLAN. Due to the shorter range a better accuracy might be expected, however it needs more beacons to cover an area [98]. Before setting up a network, a device must perform an inquiry in order to discover other devices that are within its range. This is a rather lengthy process that may last for up to 10 - 30 seconds and can significantly increase the power consumption. For this reason, Bluetooth device has latency unsuitable for real-time localization applications [119]. Another major concern is the privacy of user. Because the location data is consistently exchanged between beacons and user, between beacons, etc. so there is always a possibility of intercepting the data by external sources [120].

Usage of RSSI is significantly hampered by poor implementations of the hardware and firmware which may vary by manufacturer. Bargh and de Grote [121] used a fingerprint-based localization method which relies only on the response rate of Bluetooth inquiries. It was shown, that the measured response rate decreases with respect to the distance, e.g. for 2 m response rate = 97% and for 10 m response rate = 86 %. An extensive fingerprinting was necessary to achieve sub-room accuracy. Diaz *et al.* [122] developed a Bluetooth based positioning framework – Bluepass to locate multiple users at the same time. It allow users to locate and to be located by other users with the help of mobile devices. It is based on a server-client architecture where the detection of Bluetooth devices is done by a program running on personal computer with universal serial bus (USB) Bluetooth adapters. The program continuously scans for mobile devices, and maintains a connection to the server, to which it sends the RSSI and address of the devices it has found. The server is responsible for calculating the position. Users are required to log in to the system via a Bluetooth-enabled device on which a mobile application is running. This gives them access to a map showing the position of all the users in the system.

In [119] authors used fingerprints from Bluetooth beacons in addition to GPS for indoor and outdoor positioning. Three different algorithms – k-Nearest Neighbour, regression based on k-Nearest Neighbour and Naïve Bayes classifier were utilised, and performance was compared. Table 2.2 show the results for an office setting (3 rooms of dimension 6 m x 4 m and a corridor of dimension 35 m x 2.3 m), using 168 fingerprints.

	Accuracy	**Precision**		
		50%	**80%**	**95%**
k-NN	1.62	1.72	3.42	5.16
k-NN regression	1.60	1.42	2.40	3.79
Naïve Baye's	2.13	1.97	3.31	5.72

Table 2.2: Comparison of performance of different algorithms (Source: [119]).

At the context of 168 fingerprints, authors achieved positioning accuracy less than 2.2 m with 95% of the chances of the pedestrian to be within 5.72 m. An important conclusion was drawn from the study that for continuous and reliable positioning, Bluetooth beacons should be placed at requisite distances. Privacy of the user is maintained by preventing mobile devices to work in discovery mode. Also, the computations are performed on the mobile device itself.

Limitations: A major problem reported by some of the authors [63, 83] is the multipath problem of RF waves. Certain materials such as wood, rubber, etc. can absorb the RF waves. While some materials such as glass, metals, etc. can scatter the RF waves. This causes the RF

signal to delay and to deviate in multiple directions. As a result of this the received signal strength at a particular point fluctuates. Another problem reported by some of the authors [51, 123] is the decay of RF received signal strength with the change in physical conditions such as arrangement of furniture, structural modification or movement of personnel, etc. and distance [51].

Interferences and noises from surroundings are another major issue for RF based positioning [63, 96], since only some of the solutions operate on reserve band while others operate in unlicensed band. This further limits their applicability.

2.2.3 Magnetic positioning

The magnetic based positioning utilise ambient magnetic fields for positioning. It is categorised into two types depending on the way magnetic field is used [124]: 1) artificial magnetic field and 2) Earth's magnetic field. In case of artificial magnetic field, pre-deployed coils are required to create an artificial field. Theoretical models are used to translate the magnetic flux density into a range estimate. These models are typically function of the range and geometry of the coil therefore if multiple coils generate magnetic fields at known locations, the relative 3D position of the sensor is estimated [125]. In [126] Blankenbach and Norrdine have built an experimental model using a coil with 0.5 m in diameter and 140 loops. Magnetic field sensing was accomplished using three magneto resistive transducers placed in orthogonal directions. The magnetic coils are placed at some known distances (4.2 m, 12.6m and 16.6 m) from each other. The distances to at least three reference coils are estimated using the lateration principle to determine the position of static or moving target. Experimental results demonstrate a ranging accuracy of a few cm for short distances of less than 10 m. However, over longer distances the accuracy detonates by 10 cm at distance of 12.6 m and by 1.6 m at distance of 16.6 m. Thus, system can potentially logically localize. However, reliability over long distances is an issue.

On the contrary, in case of magnetic positioning using Earth's magnetic field; characteristics of magnetic field signatures present ubiquitously in the environment are identified as reference fingerprint to locations. This approach is mostly applied to determine the orientation of the pedestrian [127]. Liu *et al.* [128] proposed an indoor positioning system based on the smartphone in-built magnetic sensor – compass and accelerometer. Wi-Fi fingerprints are collected at the centre of room and at corridors. Digital compass measurements are taken at each step to compute heading while footsteps are detected by the accelerometer. To prevent the drifts and other disturbances from nearby surrounding magnetic materials the compass and other sensors are fused by particle filter. Results show that system achieves an accuracy of 1.62 m for a walk of 16 m but at the context of training of several hours.

In [129] IndoorAtlas have identified the anomalies in Earth's magnetic field to pinpoint the location of a pedestrian in a building with an accuracy ranging from 0.1 m to 3 m. The key

idea behind localization lies in the fact that every concrete structure leads to anomalies (see Figure 2.13) in the Earth's magnetic field. These anomalies are specific to locations which are used to position the mobile target. A significant advantage of this approach is that user privacy is secured because the position computation is done on the device itself. However, positioning involves prior mapping of the environment. This can be a problem in dynamic environments where magnetic fields change over time thus reliability is low.

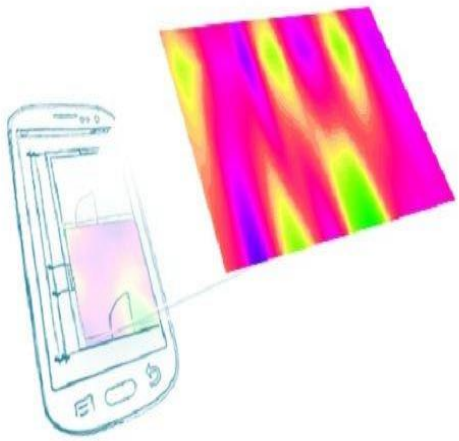


Figure 2.13: Earth's magnetic field anomaly (Source: [129]).

A significant advantage of this positioning technology is that it does not require a direct line of sight between source and sensor, unlike others [125].

Limitations: The magnetic flux generated by artificial coil are highly dependent on the number of loops in the coil, area of the coil and the current supplied in the coil [125] as illustrated by the following equation

$$B = \frac{\mu_o NIF\sqrt{1 + 3\sin^2\emptyset}}{4\pi r^4} \tag{2.2}$$

where μ_o is permeability of vacuum, N is the number of loops, I is the electric current, F is the area of the coil, $\emptyset$ is the elevation angle of an arbitrary point **P** in relation to the coil and r is the distance of that point to the coil. As a result proper selection of the coil and suitable infrastructural arrangements are needed for positioning. Another problem reported by some of the authors is fluctuating of magnetic fields indoors [124, 130]. This can be because of the presence of ferromagnetic materials or from an electrical device, such as a cathode ray tube, electric wires, etc. that generate electric and magnetic fields which disturb the Earth's magnetic fields. Authors have focussed on mitigating such perturbations [131]. But it is not an easy task as reported in [132].

2.2.4 Inertial positioning

Inertial positioning describes a methodology, where the position of a person or device is calculated relative to its last known position using dead reckoning (DR) [133]. The current position is derived from previous position using known or estimated movement information (speed, direction, acceleration, etc.) derived from the data acquired from inertial measurement unit (IMU). The IMUs consist of two orthogonal sensor triads – accelerometer and gyroscope and in some cases compass as well [134].

The IMUs yield relative position only, an absolute reference is however required to specify the displacement of a pedestrian [63]. IMUs are either fixed on a pedestrian's body such as in shoe or held freely such as in pockets [101]. The accelerometers measure linear acceleration, gyroscopes typically measure angular velocity and compass measures the direction relative to the geographic cardinal directions [134]. There are numerous accelerometer, gyroscope and compass designs, descriptions of some designs are given in [134-136]. A simple construction of a micro-electro-mechanical system (MEMS) accelerometer is shown in Figure 2.14, where it contains proof mass, usually held by a flexural support. It works by measuring the displacement of the proof mass, due to acceleration, using a pickup sensor.

Figure 2.15 (B) illustrates an example of a simple MEMS gyroscope. It contains a vibrating proof mass held by flexural support. It uses Coriolis acceleration effect on the vibrating proof mass to detect inertial angular rate [137]. Figure 2.15 (A) shows the basic principle of Coriolis acceleration [134] . The proof mass in Figure 2.15 (B) is made to vibrate with certain velocity by a drive motor. This velocity vector axis (x-axis) is perpendicular to the angular rate input axis (z-axis, out of the plane). When angular rate is applied on its input axis (gyroscope rotates), a Coriolis force is produced, which induces an oscillation of the proof mass in y-axis. Angular rate can then be estimated by measuring the amplitude of the oscillation in y-axis (Coriolis acceleration), which is proportional to the applied input rate.

Inertial measurement systems usually consist of a cluster of accelerometers and gyroscopes. Complete six-degree-of-freedom inertial measurement requires acceleration measurements in three orthogonal directions and angular velocity measurements in three orthogonal directions.

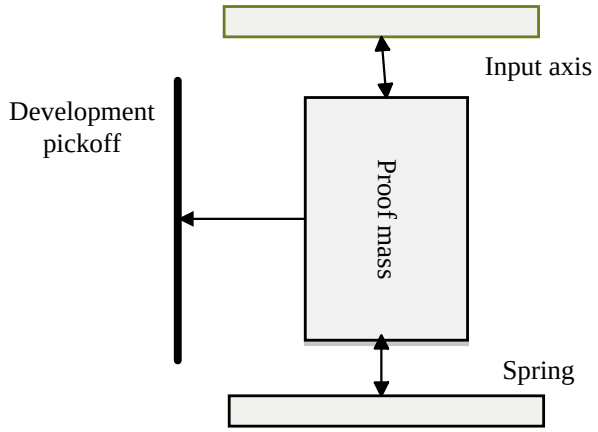


Figure 2.14: Basic principle of MEMS accelerometer (Source: [136]).

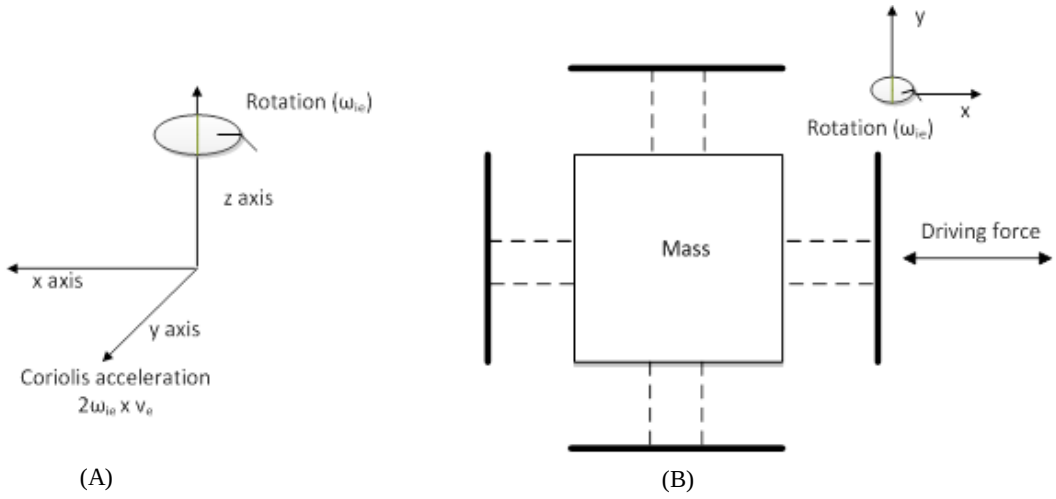


Figure 2.15: (A) Coriolis force generation and (B) Basic principle MEMS gyroscope (Source: [136]).

In some systems the sensor cluster is gimbaled and the gyroscope measurements are used in a feedback control loop to isolate the sensor cluster from rotations [134]. Thus the measured accelerations are in the Earth relative reference frame. Other systems use a strap-down design where the position and orientation of the sensor cluster are fixed relative to the moving target object [134, 138]. Thus measured accelerations are in a body fixed reference frame and the sensor signals must be integrated in real time to obtain the transformation to an Earth relative coordinate frame. This method is more computationally expensive but the hardware is simpler, cheaper and smaller without the gimbals. The strap-down system is more prevalent in modern IMU's.

To understand the process of obtaining the position using a strap-down IMU consider the following simplified example of how to integrate accelerometer and gyroscope measurements to obtain position and velocity estimates in six degrees of freedom for a moving object. First some coordinate frames must be defined (see Figure 2.16): the body frame is a Cartesian coordinate

system which is aligned with the object's body and hence the IMU. The global frame is the coordinate system against which rotations are measured; this can be defined as a non-rotating Cartesian frame with an origin at the centre of the Earth and axes aligned with the Earth's rotation axis. The Earth frame is similar to the global frame except that it is rotating with the Earth and is hence fixed with respect to a meridian.

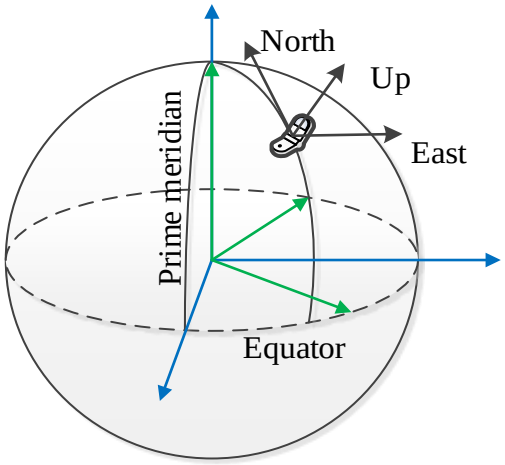


Figure 2.16: An illustration of the three frames. The blue arrows show the Earth frame, the green arrows show the global frame and the black arrows show the body frame.

Measurements from a strap-down IMU's accelerometers give acceleration in body coordinates $\ddot{x}_b$, where $\ddot{x}$ is an acceleration vector. And measurements from the gyroscopes give the angular velocity vector of the body frame with respect to the inertial frame ω_{ib}. We need to find the position vector of the target moving object in the Earth frame x_e. The acceleration of moving target object in the Earth frame can be found using

$$\ddot{x}_e = R_b^e \ddot{x}_b - 2\omega_{ie} \times v_e + g_l \quad (2.3)$$

where R_b^e is a rotation matrix which transforms body coordinates to Earth coordinates, $2\omega_{ie} \times v_e$ is the Coriolis acceleration due to the rotation between the inertial and Earth frames (ω_{ie}) and the velocity of the object in the Earth frame (v_e), g_l is the the local acceleration due to gravity.

The acceleration of moving target object in the Earth frame can be integrated once to obtain the velocity v_e and twice to obtain the position x_e. The local acceleration due to gravity g_l is found using a gravity model, and ω_{ie} is found using a Coriolis model. The two angular velocities ω_{ib} and ω_{ie} can be concatenated to obtain a rotation matrix derivative $\dot{R}_b^e$ between the body and the Earth frame. This can be done, for example, using quaternions [3]. The rotation matrix derivative can then be integrated to get R_b^e. Equation 2.3 can be solved dynamically. Values of x_e, $v_e$and R_b^e from the previous time step are used to calculate $\ddot{x}_e$ in the current time step and this is integrated numerically over the length of time step. Thus it is the change in the position of the

moving target object that is being tracked relative to some initial estimates of the position. Figure 2.17 shows a block diagram of the dynamic calculation.

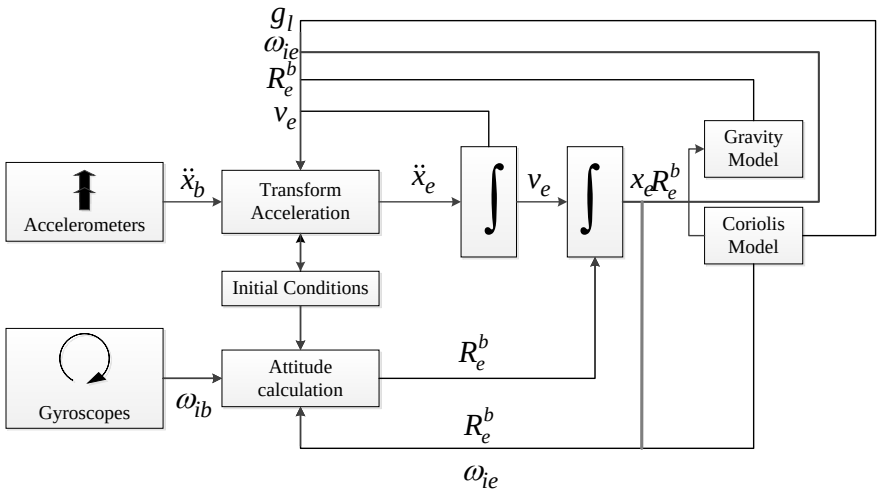


Figure 2.17: Strap down IMU state calculation algorithm (Source: [134]).

The IMUs provide high frequency (50 Hz is typical) position information with no latency under normal operation. They are autonomous i.e. not reliant on any external technology or references and thus work under all normal conditions [139]. A significant drawback of IMU assisted positioning is the presence of drift error. Because position is calculated by integrating the acceleration and angular rate over time the error in position is cumulative and as a result grows over time. It has also been observed in [140] that this error can even grow super linearly with time – 100 m after 1 minute of operation and 1000 m after 2 minutes of operation. The cost of an IMU is generally proportional to the rate of this drift error [133]. Table 2.3 reproduced from [141] shows typical errors in position with time and cost for IMU systems. These data were published in 1997, today the costs may be somewhat lower but the trend remains the same.

	Cost $(USD)		
	> 750, 000	~100, 000	~10, 000
Position	0.3 – 0.5 m	0.5 – 3 m	30 – 50 m
Attitude	0.3 – 0.5 mdeg	4 – 5 mdeg	0.2 – 0.3 deg

Table 2.3: Drift in position and attitude after 1 minute of operation

Two methods have generally been used to overcome this problem of drift error [142]: (1) frequently-calibrated pedestrian dead reckoning and (2) step-based pedestrian dead reckoning.

2.2.4.1 Step based pedestrian dead reckoning

Step-based PDR, commonly, depend on the walking movement of pedestrians. Human walking is indeed a periodical activity at its core that can be observed from the analysis of human gait

[143, 144]. The gait cycle consist of two main phases: the stance phase and the swing phase as shown in Figure 2.18.

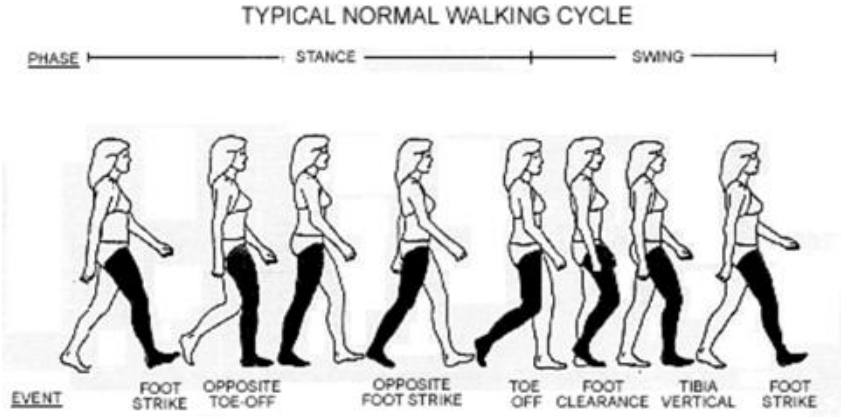


Figure 2.18: Typical Normal Walking Cycle (Source: [143]).

The foot is in contact with the ground for the entire stance phase until toe-off occurs. Toe-off initiates the swing phase when the foot is lifted off the ground and carried forward to begin the next stride. The swing phase ends when the foot is again placed on the ground, beginning the next stance phase, and completing the gait cycle. The opposite limb repeats the same sequence of events, but is 180^0 out of phase. The horizontal velocity of the foot varies with each stride from stationary to over twice the velocity of the torso. This leads to a cyclical periodic pattern of acceleration signal [143].

Pattern of human walking cycle is same for all normal humans where parameters can be different based on some factors like gender, height, and weight. Normally stance phase occupies 60% of the gait cycle. Step based PDR typically depends of three techniques: foot step detection, step length estimation and step heading estimation. The accelerometer signal is analysed for detecting footsteps. The linear displacement is calculated during each foot step by using stride length estimation models [145] and the pedestrian's heading is estimated using gyroscope or a compass or combination of both. This method is usually applied in pedestrian dead reckoning (PDR) systems as follows [146]:

$$X(t+1) = X(t) + SN\cos\theta \quad (2.4)$$

$$Y(t+1) = Y(t) + SN\sin\theta \quad (2.5)$$

if the displacement is in 2-D and if the displacement is in 1-D, this method is usually applied as follows [147]:

$$X(t+1) = X(t) + SN \quad (2.6)$$

where $X(t)$ and $Y(t)$ represent the pedestrian's location at time index t, $X(t+1)$, and $Y(t+1)$ represent the pedestrian's location at time index $t+1$, S is the displacement in one foot

step, N is the number of footsteps and θ represents the direction of motion or heading.. The pedestrian's heading is estimated via two different approaches [52]. These are as follows:

- An absolute direction is directly measured or estimated by sensors.
- A relative direction change between two time instants is measured by sensors; and the absolute heading of current epoch is propagated to the next epoch.

2.2.4.2 Frequently calibrated pedestrian dead reckoning

In contrast to step based PDR, frequently calibrated PDR decreases the accumulating error by applying a solution called zero velocity update (ZUPT) [139]. The ZUPT algorithm calibrates the velocity of the sensor based on the fact that the speed of a pedestrian's foot decreases to zero on the stance phase when a pedestrian simply steps on the ground during walking. If supposedly the sensor is affixed to the foot, the sensor's speed would also be zero.

Once 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 [139]. As a result, it is also suitable for distance updating when pedestrians runs, walks backward or even steps sideways. However, the requirement of mounting IMU on a foot has to be met, which is not convenient for smartphone users.

In comparison, a step-based PDR has no such device mounting position requirement, and is widely used in smartphone tracking applications. Also, in this research study we use step based PDR.

One of the earliest public descriptions of IMU assisted pedestrian positioning is the 1996 patent by Judd *et al.* [148]. The same concept was later applied by Levi *et al.* [149] to design the first commercial pedestrian dead reckoning (PDR) system. They developed a light weight and mobile PDR module that detected human walking steps and estimated the step length based on the peak detection of the vertical acceleration changes and its frequency analysis. The module also sensed the direction with in-built compass. In the same year, a DARPA project [150] proposed PDR technique using shoe-mounted inertial sensors with ZUPT. However, the results were never published. Later, the same concept was extended by NavShoe [151] to locate the pedestrian indoors and outdoors. The NavShoe software detected the stance (stationary) phase of the pedestrian and applied ZUPTs as pseudo measurements into the Extended Kalman filter (see [152], for more details on Extended Kalman filter). Experimental result show that for a walk of length 118.5 m the estimated pedestrian's position drifted by 0.3% of the travelled distance. Thus making the positioning ambiguous.

Khairi *et al.* [153] and Angermann and Robertson [154] designed a foot mounted PDR system. However, in addition to the foot mounted IMUs, they even applied environmental heuristics Khairi *et al.* [153] set the drifts in velocity to zero by applying ZUPTs while heading errors are mitigated by the information from building such as walking in corridors is restricted to straight pathways, 4 cardinal directions are usually possible in a rectangular floor, etc. Experimental results show for an average of walk of 24 minutes, the average error was approximately 5 m when building information was used however without using building information the error was approximately 150 m. It is to be noted the cost of IMU was 1700 GBP.

Angermann and Robertson [154] designed a foot mounted PDR system – FootSlam. The knowledge of all user intention was used to design a map of the building that was later used to guide the pedestrian through it. The more number of times places were visited within the building, the better information or map was built for that place. This information was fused with the human trajectory information using a Rao-Blackwellized particle filter [155] to represent the pedestrian's position in a 2-D hexagonal grid. Simulation results show an average accuracy of 1 – 2 m was achieved for 3 walks of 10 minutes. It is however unclear how these results would scale for longer walk. On an average 30,000 particles were used in computer simulation. A significant benefit of this approach was that pedestrian had full control on the location data, thus privacy of the user was kept confidential.

In conjunction with Wi-Fi, GPS and other sensors IMU's are also used in applications where high accuracy and frequency are important such as robotic vehicles [156]. Systems that combine DGPS and sensitive IMU's costing around 15,000 GBP can give localization accuracy of 0 – 2 m at a frequency of 100Hz [157]. Fallon *et al.* [33] have designed a wearable inertial indoor positioning system to localize pedestrians on the go. The system consists of an IMU, a Microsoft Kinect camera, a light detection and range finder laser and a button to tag the point of interests in map. The camera identifies the ceilings, wall junctions and other point of interests and subsequently localizes the pedestrian onto the designed map of a building (with unknown layout) at a go. Positioning results indicate that when a system is carried by a pedestrian for duration of 94 seconds, the particle cloud surrounding the pedestrian have a median error of 0.66 m. While the same system when carried on a wheelchair for duration of 180 seconds, the particle cloud surrounding the wheelchair has a median error of 0.48 m. This can be a potential option for logical localization however it is to be noted that the cost of whole system was around 5000 GBP.

Jin *et al.* [158] proposed a PDR approach suitable for any pedestrian carrying a smartphone or a tablet PC or a sensor set embedded in a customised key chain. The footsteps, were detected from magnitude of the acceleration, projected on the global frame and measured in the vertical Z direction. Step length was estimated by a heuristic model referred in [159].The heading was

estimated using gyroscopes. Specifically, principal controlled analysis was done on the gyroscope values over a sliding window of 1s to learn and recognize the pedestrian's direction indoors. Experimental results show that the average tracking error was reduced by 73.7% in comparison to the traditional PDR approach in the laboratory environment. However, the system had the drawback of poor scalability and incremental cost. Since, the pedestrian was required to carry multiple smartphones or a tablet PC and a smartphone for its effective implementation at any point of time. Moreover, it was required that they are all wirelessly connected and are synced.

Meanwhile, there have also been research studies that have designed waist mounted PDR system [160, 161], head mounted PDR system [162] and body wearable inertial PDR system [163, 164] using IMUs embedded in smartphones or as a direct sensing unit. But what has not yet been investigated is the usage of smartphones for helmet mounted PDR system which can be of a potential use to the construction workers or firefighters who wear helmets at all times. This would be investigated in this thesis.

Limitations: As mentioned before, one of the major problems is the presence of drift errors. The drift can be due to the thermal changes and inherent noise [133]. This will be discussed in detail in Chapter 3. Another major issue reported by some of the authors [3, 139] is that the inertial sensors provide (acceleration and angular velocity) measurements that when integrated report relative position. Therefore it needs an initial starting point for positioning.

Thus systems that provide the absolute position reference are required to report absolute location estimates. Although such systems can provide unique localization, the addition of inertial sensors could improve localization accuracies by working in parallel, as discussed earlier.

2.2.5 Landmark based positioning

The general procedure for performing landmark-based positioning is shown in Figure 2.19.

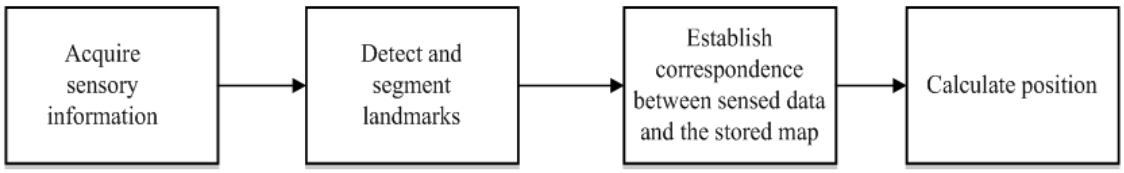


Figure 2.19: General procedure for landmark-based positioning (Source: [165]).

Firstly, the landmarks are selected, such that they are easy to identify; for example, there must be sufficient contrast to the background. Secondly, sensors are used to sense the environment and then extract distinct structures that serve as landmarks. Once the landmarks are extracted then a correspondence need to be established between the sensed landmarks and the stored database. Lastly, localization is done to compute the relative position of the target with respect to landmark.

Landmarks in general, can be an object or structure that marks a locality and is used as a point of reference. This concept is bound to the prominence or distinctiveness of a landmark from

the nearby objects in a large-scale environment or landscape [166]. The landmarks can be either 'naturally' present in the environment or can 'artificially' be attached to the scene. Based on the presence of these landmarks the landmark based positioning is classified into two categories [165, 167, 168]. These are as follows:

- Natural landmark based positioning
- Artificial landmark based positioning

The natural landmark based positioning involves objects in the environment that have a function other than navigation (importantly positioning) [165]. These landmarks can be objects such as doors, pillars, staircases, junction between two walls, etc., and even turning corners are a form of natural landmarks. A significant drawback associated with this positioning technique is that it works best when the environment is highly structured (e.g. museums, hospitals, etc.) [168]. Also, it is necessitates that complex signal processing is done to extract the natural landmark features from the environment [169]. The natural landmark based positioning is primarily performed using techniques related to Computer Vision [165]. Positioning using computer vision involves two stages: recognition, where beacons are detected in the images from the cameras and localization, where transformations are applied to determine the position of cameras relative to the beacons.

- Recognition: Object recognition algorithms are used to detect and identify beacons within the images. These beacons can also be termed as landmarks. The landmarks must be easy to identify and the probability of false positive must be low, therefore careful selection of landmarks is important [168]. One approach is to use artificial landmarks with easily identifiable shapes, patterns or colours. Lee *et al.* [170] used a triangular shaped marker. The shaded triangular sections were coated with film that was reflective to infra-red light. Tai *et al.* [171] used colours, geometric elements, and structural features in images to identify taxis, pedestrian walk symbols, zebra crossings, and, straight line tactile guide paths and junctions.

  Another approach is to use the natural features present in the scene as landmarks. The natural features can be such as floors, pillars, etc. SurroundSense [36] have identified the ambient sound, light, and floor colour to recognise the logical location of the pedestrian. It considers audio sample amplitude distribution as the ambient sound fingerprint for acoustic processing. Pixels of the floor pictures are translated to a hue-saturation-lightness (HSL) space, and the light intensity on L-axis as the ambient light fingerprint. Two simple states from the accelerometer readings, stationary and motion are used as the input to the support vector machines (SVM). The sequence of stationary or moving state are viewed as an abstraction of the user's movement pattern which are again

considered as a fingerprint. These fingerprints are merged by a multi-sensor filter to localize pedestrian with an average accuracy of 87%.

Tarzia *et al.* [172] have used acoustic background spectrum, an ambient sound fingerprint to recognise the logical location. The system first calculates the acoustic background spectrum of the room, and then classifies the room by comparing its acoustic background spectrum with the existing, labelled acoustic background spectrum values in the database. The system yields 69% correct fingerprint matches. It is however still a matter of question how would these results scale down in university buildings or hospitals where ambience of every room are almost similar. Or in a store with several bays side by side. This thesis would take up these challenges later on.

- Localization: When beacons have been detected in images from the camera the next step is localization. A camera cannot directly measure the range of objects in the images therefore this information needs to be inferred by post-processing the image data. The problem of localizing a camera in a field of beacons can be framed by considering the following simple example based on the perspective projection model [173, 174].

  The perspective projection model in camera centred coordinates is illustrated in Figure 2.20. The focal point of the camera is located at the origin of the camera frame $[x_c, y_c, z_c]$. The camera is oriented so that it is pointing in the z_c direction. The image plane is parallel to the $[x_c, y_c]$ plane and a distance f along the z_c axis, f is the focal length. Lines can be drawn between points in space and the focal point, then where these lines intersect with the image plane is where the points will show up in the image. The coordinate transformation between a point in the camera frame $[x_c, y_c, z_c]$ and the coordinates of the point in the image $[x_i, y_i]$ is given by equation 2.7

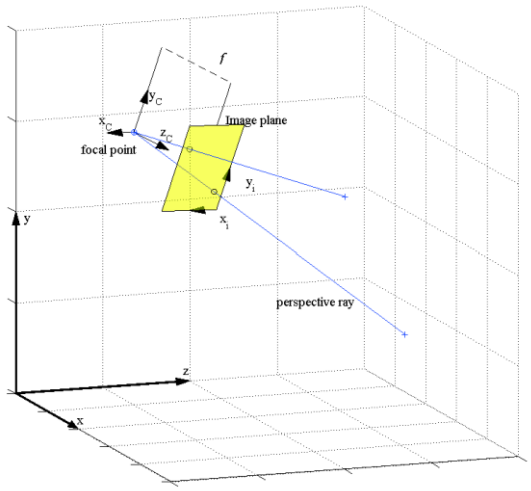


Figure 2.20: Perspective projection model for a monocular vision system.

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \frac{f}{z_c} \begin{bmatrix} x_c \\ y_c \end{bmatrix} \tag{2.7}$$

This transformation is not reciprocal and given the image coordinates, it is only possible to define a ray in camera space on which the point lies. The ray can be described using homogeneous coordinates as

$$x_i = \begin{bmatrix} \omega \frac{x_i}{f} \\ \omega \frac{y_i}{f} \\ \omega \end{bmatrix} \tag{2.8}$$

where ω is an arbitrary scalar. However with a binocular system like that illustrated in Figure 2.21 there is enough information in the two images to reverse the transformation, as illustrated by equation 2.9.

$$x_c = \frac{b}{2} \frac{x_l + x_r}{x_l - x_r}, y_c = \frac{b}{2} \frac{y_l + y_r}{y_l - y_r}, z_c = \frac{bf}{x_l - x_r} \tag{2.9}$$

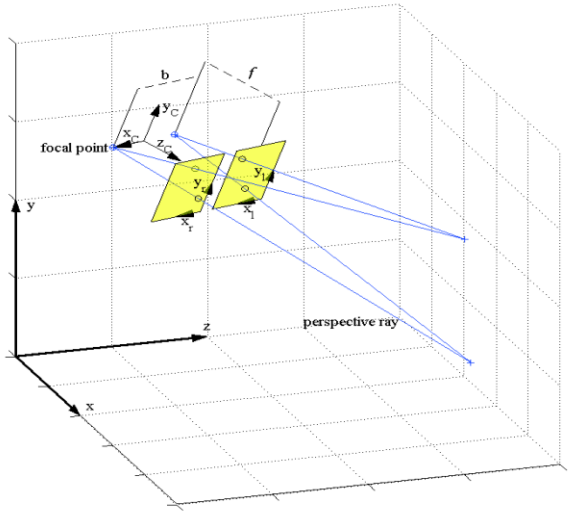


Figure 2.21: Perspective projection model for a binocular vision system.

In order to relate points in global space to the rays or points in camera space described above, the pose of the camera must be known. This can be described by a rotation **R**, where **R** is a 3x3 rotation matrix, followed by a translation **T**, where **T** is a 3x1 vector. A point in global space $X = [x, y, z]'$ can also be represented in homogeneous coordinates as $X_h = [x, y, z, 1]'$ and transformation between this point and the perspective ray of a monocular camera is given by

$$x_i = [RT]X_h \tag{2.10}$$

X can be transformed to a point in camera space X_c by the simple Euclidean transformation

$$X_c = RX + T \tag{2.11}$$

Now the problem to be solved is to find **R** and **T** given point correspondences between X and X_c (in the case of a binocular system) or x_i (in the case of a monocular system). For a direct linear transformation there are 12 unknowns in **R** and **T** and there are three equations in 2.11 or two equations in 2.10. This means 4 point correspondences between global space and three-dimensional camera space in the case of a binocular system. Or 6 point correspondences between global space and two-dimensional image space in the case of a monocular system.

In fact although there are 12 unknowns in **R** and **T** it can be shown that, ignoring scale, there are only 11 degrees of freedom [174] meaning that strictly a direct linear transformation only requires $3\frac{2}{3}$ (binocular) or $5\frac{1}{2}$ (monocular) point correspondences. However practically this still means having 4 or 6 points in the image. This is the case for a direct linear transformation but it is still possible to get localization with fewer points when line correspondences (between points) are used.

Computer vision based localization has often achieved accuracy less than 1 m in closed monitored zones using robots, however it is unclear whether the same accuracy level can be achieved for pedestrians walking freely indoors and outdoors [175]. In some cases pre-placed beacons have been used but in others a more advanced process called simultaneous localization and mapping [176] has been used. Here the robot is initialized using beacons with a known position but from then on the system finds and identifies new beacons through the relative position of points in images as the robot moves around. Experimental results show an accuracy of less than 1 m.

In comparison, artificial landmark based positioning involves objects (landmarks) that are specially designed for the sole purpose of enabling navigation (importantly positioning) [165]. These landmarks are specially designed patterns or markers such as Quick response (QR) codes [177], triangle-shaped landmark [170], specific light patterns [178], etc. Detection of these landmarks is much easier in comparison to the natural landmarks, since these are designed for optimal contrast and their identification is unique and, shared in a reference database. Necessarily artificial landmark based positioning require hardware installation while this is drawback is some applications (such as robot exploring unknown regions e.g. on an unknown planet), it is however suitable for localization of pedestrians [167]. Computer vision is again one of the preferred positioning technique for artificial landmark based positioning [179, 180]. Some of the key elements of the two landmark based positioning techniques are illustrated in Table 2.4.

	Natural landmark based positioning	**Artificial landmark based positioning**
Common targets of positioning	Mobile robots and vehicle [167]	Pedestrian [167]
Accuracy achievable indoors	• Average (1m – 3m) [181, 182] • Best (< 0.2m) [183]	• Average (1m – 5m) [184, 185] • Best (< 0.1m) [320]
Strengths	• No additional hardware or specialized infrastructure requirements	• Make navigation easier, real time and steady.
Limitations	• Selecting the natural landmark is hard (depends on individual's prudence).	• High installation cost. • Specialized hardware requirements.

Table 2.4: Characteristics of two landmark based positioning techniques.

A project by Microsoft research group – Easy Living [187], proposed the usage of computer vision to provide context awareness within intelligent environments. The architecture of system was built with 3 computers and two colour cameras. Each camera was connected to a computer, and both the computers were connected to the third computer. The third computer was primarily used for running person tracker algorithm. The algorithm combined colour and depth of the images taken from two cameras to provide position of the moving target. Experimental results depict positioning error less than 10 cm however partial occlusion was observed. This may be sufficient for logically locating pedestrians. However, a major concern is whether an employer or building owner may be willing to invest a hefty amount for setting up such a system.

In [188] the authors presented a navigation system for a humanoid robot, which can be extended to the case of human walking. Location determination was accomplished in two stages. In the off-line stage, images were captured. The on-line stage involved autonomous robot navigation between two arbitrary points while capturing images. To achieve localization, an algorithm correlated freshly captured images with images in the route database. In this way, the unprocessed raw image was considered as a large set of features. Correlation analysis yielded the position deviation between the learned route and the current position. Temporary occlusions were detected as sudden drops in the correlation. The maximum position deviation for a 17 metre long route, was 0.9 m. This represents a deviation of 5.3% of the walked distance. A significant advantage of proposed system was that privacy of the pedestrian was maintained in addition to the high accuracy (less than 1 m) achieved. Nevertheless, it is not known how these algorithms would fit in memory constrained devices (e.g. smartphones).

Fallah *et al.* [189] proposed a system – Navstar for visual impaired navigation. The system uses a virtual representation of the indoor environment that uses tactile landmarks (such as doors,

walls, and hallway intersections), that the human can sense. Feedback from the pedestrian upon confirmation of landmarks in the environment is used as ground truth to periodically update location data. In between landmarks, dead-reckoning (using smartphone sensors such as the accelerometer) was used to perform localization. Ten particle filters were executed in parallel. Each one of them employing a different set of assumptions regarding the capabilities of the user depending on pedometer and compass data from the smartphone and landmark confirmations by the pedestrian. What was most noticeable that each filter required 50-100 particles for every iteration. This implied on an average 500 – 1000 particles were simulated on a single instant. In addition, a pedestrian was required to worn an infrared camera, a belt carrying a battery, a tablet personal computer at all time during the walk.

Wang *et al.* [62] combined dead-reckoning, urban sensing, and Wi-Fi-based partitioning into a common framework – UnLoc for pedestrial localization. Pedestrians move naturally in the building collecting accelerometer, compass, gyroscope, and Wi-Fi readings. By assimilating data from these sensors, UnLoc identifies signatures of landmarks (e.g. corridor turn, stairs, stopping or drinking of water, etc.) present in the environment that are subsequently used a reference point for dead reckoning and also for localization. Results show median localization accuracy of 0.89 m when system runs offline, and 1.69 m in case when system runs online. This can be an option for logical localization however higher accuracy necessitates 2 hours of prior training. Furthermore, the implementation of UnLoc requires the smartphone and server are always wirelessly synced. Again, maintaining user privacy was an issue since the position computation was done at server end and there were no arrangements made to keep the position information secured.

Mulloni *et al.* [180] explored the usage of fiducial markers and maps; in addition to the signpost application for navigating a pedestrian across a venue. The fiducial marker served as a reference point. Whenever a signpost application detects a marker, the application updated the pedestrian's current position on a map and subsequently allowed the pedestrian to decide how to reach to the destination. For an area of roughly 100 m × 200 m, 37 markers were installed, the accuracy as such is not reported by the authors. However, authors highlight that privacy is an issue for the developed system given the application has a provision of storing the pedestrian's position in a server.

Limitations: Common problems reported for landmark based positioning solution using computer vision sensor are the ambient noise in the form of light or thermal radiation, signal reflections and illumination variability [63]. The ambient noise is usually overcome by filtering techniques [190] or by combination of different modulation techniques [191]. Another problem that is referred by most of the authors [192, 193] is the inefficiency of landmark based positioning

solutions (especially, computer vision based) to work under thick smoke, gust of air particles and dark conditions. This limits their applicability and makes them less appropriate for positioning tasks. Few authors have also reported the problem of occlusion caused by movement of dynamic objects. For instance, in the experiments performed in [191], the introduction of new objects or humans was specifically avoided during the experiments. Therefore installing of cameras in the ceiling or deploying of camera sensors with overlapping coverage areas has been suggested by authors in [177].

2.2.6 Sonic based positioning

Sonic waves are mechanical vibrations transmitted over a solid, liquid or gaseous medium. These waves are produced by vibrations below and above the threshold of human hearing. In particular these waves can be either infrasonic waves (<20 Hz) or ultrasonic waves (>20,000 Hz). The principle behind sonic based positioning is similar to the RF based positioning. The sonic waves are transmitted into the medium, and from the time it takes to revert back to the receiver is used to calculate the relative distance of the mobile target [125] according to the following equation:

$$r = \Delta t * v_{us} \tag{2.12}$$

where Δt is the time of flight, v_{us} is the velocity of wave (equal to $344\ m/s$) and r is the relative distance of the target. Or alternatively, the mobile target's position can be estimated by multi-lateration (see Section 2.2.1.2) from three or more ranges to fixed receivers deployed at known locations. The sonic waves operate optimally at a range of 10 m or less [125]. Doubling the distance causes the signal's sound pressure level to attenuate by almost 6 dB. Accurate logical localization is possible, however positioning solutions are limited by the need of infrastructures. For e.g. Ward *et al.* [194, 195] designed an Active Bat location system to monitor the position of pedestrians and objects. The Bats are carried by mobile objects as illustrated in Figure 2.22. These transmit ultrasonic signals that are received by several RF transmitter ultrasound receiver units, mounted at the ceiling. The ceiling mounted units are interconnected and synchronized. Once the bat sends an ultrasonic pulse, these are received by ceiling mounted units to estimate the position of a moving target. The authors report accuracy less than 10 cm for 95% of the measurements however using ultrasound this way require a large fixed sensor infrastructure throughout the ceiling.

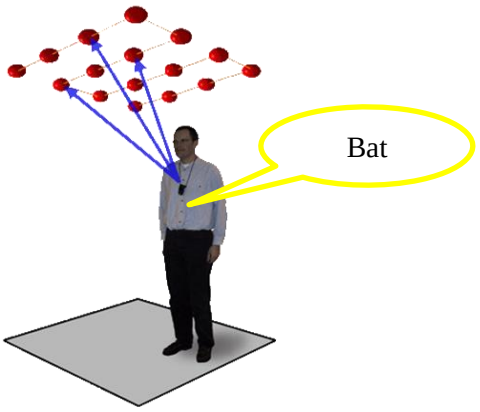


Figure 2.22: Active Bat location system (Source: [53]).

Complementing the Active Bat system, in [22] Priyantha *et al.* have designed Cricket Location Support System to position the mobile target. The ultrasound emitters were used to create an infrastructure while receivers were attached to the object being located. The emitter sends periodic information to the receiver containing its ID, range of coverage and its coordinates. Based on the received information, the receiver calculates its position using the time based methods (see Section 2.2.1.2). The emitters also transmit RF messages for synchronization of the TOA measurement in a decentralized fashion. However, when there are not enough emitters for the location calculation, the receiver uses the semantic string forwarded by the radio link to get proximity location information. Experimental results show that Cricket system can provide a position estimation accuracy of 10 cm and an orientation accuracy of 3^{o}. A significant drawback with the proposed system was that receivers performed many operations simultaneously. For e.g. it received both ultrasound and RF signal at the same time and subsequently, performed location estimation calculation. As a result receiver consumed more power and there was a significant delay in positioning.

Sonitor [196] is another positioning solution that can locate humans including the other 3D objects present in a room e.g. a key ring or any hidden objects in a room with a room level accuracy. Motion activated tags are worn by the people or equipment. These tags transmit ultrasonic signals with unique identification codes to wall mounted receivers on a particular room. These receivers process the signal and transfer relevant information to a central unit via wired or wireless local area network (LAN). Again the deployment of Sonitor system necessitates installation of cables and mounting of receivers onto the walls. In oppose to this, Wan and Paul [197] designed an ultrasonic positioning system to track persons without the need for body-worn tags. Instead using ultrasound transducers (see Figure 2.23), the ultrasonic transducers captured analogue echoes, which are then digitized and analysed in order to calculate the 1D range

of the moving target. The tracking algorithm utilizes a number of signal processing techniques to remove interference from other objects in the room and data from multiple sensors are fused by Kalman filter based smoother to determine the moving target's 2D position with accuracy better than 0.5 m. Again it is to be noted that experiments were performed in a test lab (dimension: 6m x 5m) with walls augmented by 6 ultrasonic transducers. The cost of a single module was approximately 30 GBP. It is not known how many of such modules would be needed in a building.

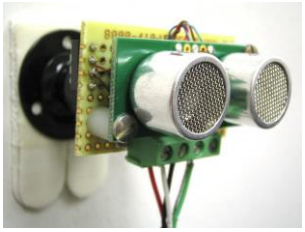


Figure 2.23: A sonar module with two separate transducers (Source: [197]).

Limitations: Some authors report that high levels of ambient noise commonly encumber the detection of the sonic signal [198, 199]. This is particularly important when considering the deployment of a sonic localization solution in densely populated regions such as hospitals. Another common issue is the dependency of sound speed on temperature T. The dependency is given by

$$v_{us} = (331.3 + 0.6T)\ m/s \quad (2.13)$$

where T is the absolute temperature in Celsius (^{o}C). At room temperatures this effect causes about 0.2 % systematic range deviation per degree Celsius. For a typical maximum range of 10 m, a change of 1°C in the temperature causes a deviation in the range estimation of 2 mm [125]. Minor influences on the speed of acoustic sound are also from air pressure, CO_2 content, etc. however it can be compensated by calibrating the measurements between known nodes.

Some authors [198] have also reported the co-interference of the sonic waves with multiple emitters. This prevents the isolation of a single source and lastly, the propagation properties of the waves pose a challenge for accurate positioning in indoor environments [199, 200]. Especially, with the change in physical conditions such as furniture, walls, etc. it can lead to echoes that result into localization inaccuracies.

2.3 Sensor fusion

As stated in the previous section (Section 2.2), each positioning technology have specific advantages and disadvantages. Therefore, the combination of different positioning technologies or systems that complement each other would be beneficial in regard to the quality of positioning

[201]. This section describes how sensor fusion can fuse data from multiple sources to get an improved estimate of the position.

The sensors complement each other best when their errors are independent of each other [50]. Because the position estimation relies on the probability theory a short introduction to the fundamentals of stochastics will be given in this section. Sensor fusion methods can be sub-divided into types with regard to the temporal dynamics: (1) static estimators and (2) dynamic estimators [202].

2.3.1 Fundamentals of Stochastics

"*The aim of stochastic filters is to estimate the state of a system on the basis of measurements. The measurements, as well as the system model itself, can be afflicted with uncertainties and errors that can be described with stochastic concepts*" [203]. In the following, some of the basic terms of related to stochastics will be introduced.

Random variable

Mathematically, a random variable is neither random nor a variable, it is a mapping from sample space into the real-line ("real-valued" random variable) or the complex plane ("complex-valued " random variable) [204]. There are two different ways to distinguish random variables: discrete and continuous random variables. Discrete random variables are mapped to values of a countable set, whereas continuous random variables are mapped to arbitrary values. These values are also termed as *realizations.*

Probability density

The probability density is a possibility to describe random variables. The function determines how likely is to obtain one or another outcome for each observation of the variable [205]. In the case of a discrete random variable X, its distribution is defined by means of a probability mass function, which reflects the probability of X to be exactly X_i, for the whole domain of possible values of X_i.

For a continuous variable x, the distribution is described by its probability density function (PDF), denoted by $p_x(k)$ which does not indicate probabilities, but probability densities for each possible value of x = *k*. Here, probabilities can be always obtained for the event that the variable x falls within a given range [a, b] by means of

$$P\,(a < x < b) = \int_a^b p_x(k)\ dk \tag{2.14}$$

The probability that the realization ρ occurs for the random variable x and the realization φ occurs for the random variable y is called compound possibility [50]. If a certain realization φ of y is given, the probability that realizations ρ of x will occur under this condition provided if it follows this relationship:

$$p_{x|y}(\rho|y = \varphi) = \frac{p_{xy}(\rho, \varphi)}{p_y(\varphi)} \tag{2.15}$$

This relationship is called as conditional probability or alternatively, Bayes' rule. If, in contrast, a realization ρ of x is given one has:

$$p_{x|y}(\varphi|x = \rho) = \frac{p_{xy}(\rho, \varphi)}{p_y(\rho)} \tag{2.16}$$

Normal or Gaussian distribution

In the area of stochastic filters, the Gaussian distribution is of particular interest because this distribution is an excellent model for the observed frequency distribution for many naturally occurring events. Furthermore, this distribution is important mainly because ("under mild regularity conditions") the sum of many independent and identically distributed random variables approaches normal, when many approaches infinity [206].

The distribution function of each random variable is independent of each other as well they can have any arbitrary values [50]. Under these conditions the distribution function tends to the normal distribution with an increasing number of random variables. A random variable x is said to be Gaussian distributed if its density function is of the form

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(\mu - x)^2}{2\sigma^2}\right) \tag{2.17}$$

where μ and σ^2 are the mean and variance of a random variable x respectively. Note that density function of a random variable is completely defined once the mean and variance of the random variable are specified. Figure 2.24 illustrates the probability density function for different values of mean μ and variance σ^2, the more the variance of random variable the higher the spread across bell curve.

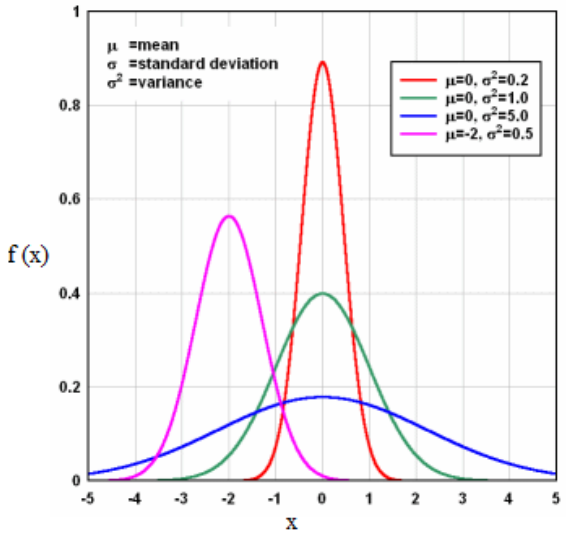


Figure 2.24: Different normal distribution PDFs with varying values of mean and variance.

2.3.2 Static estimators

Static estimators can be used to fuse data of multiple sources provided that the position estimation of the object to be tracked is nonrecurring or the object is stationary [50, 202]. The positioning sources are described as random variables via a PDF. PDFs of different independent sources if known then the optimal PDF can be estimated by multiplication of all the PDFs. For e.g. refer to Figure 2.25; the top two images (on right) depict the two independent disturbed sources with their respective PDFs adjacent to them. The third image depicts the outcome of multiplicative combination of the two PDFs. Both PDFs produces high values only for those areas, where both PDFs are indicating a high probability.

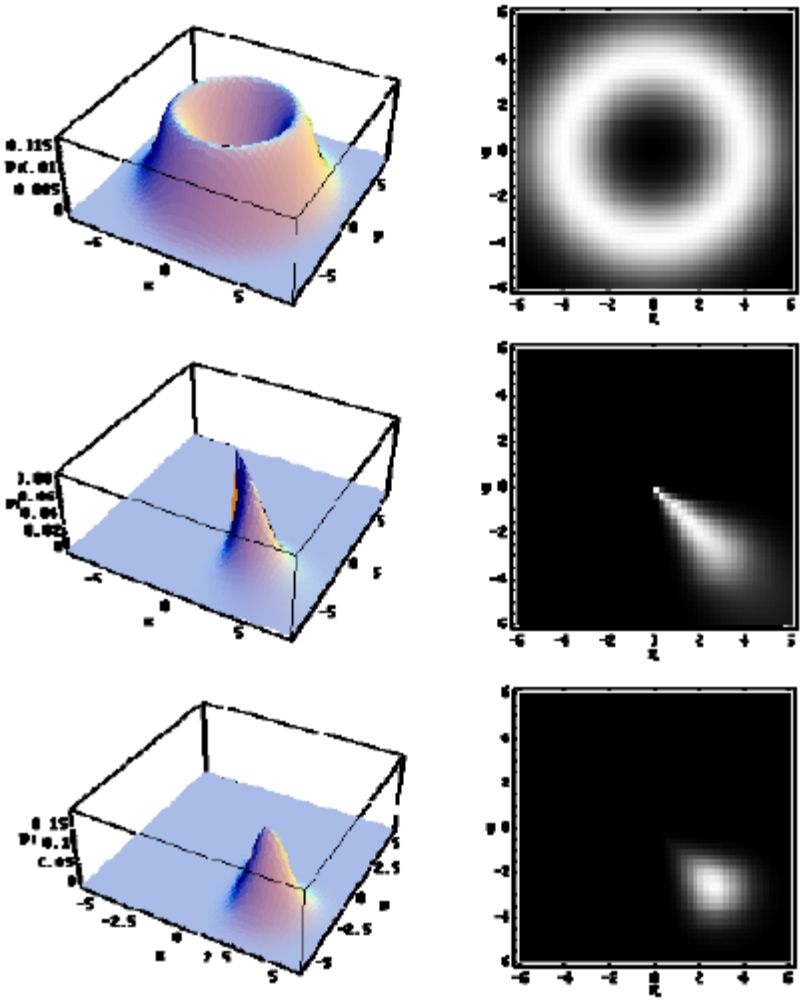


Figure 2.25: Example of the operation of two independent static position sources (Source: [202]).

2.3.3 Dynamic estimator

Static estimators cannot track moving objects continuously, so dynamic estimators are utilised. These take into account the possible physical movement, as well as the temporal characteristics of the sensors [202]. In the area of tracking objects and navigation few of the commonly implemented dynamic estimators are Kalman filter [152] and Particle filter [207] that combine data from different sensors. These will be introduced shortly. However, these are based on the Bayesian principles that would be introduced next.

Bayesian filter

Bayesian filtering is aimed to apply the Bayesian statistics and Baye's rule to the stochastic filtering problem [208]. It allows to model the uncertainty about the world and the outcomes of interest by incorporating prior knowledge and observational evidence. In principle, Bayesian filter probabilistically estimates the state of a dynamic system from noisy observations [209]. The state is not directly measurable instead it is estimated using sensor measurements.

Bayesian filters represent the state of a random variable at time *t* by x_t. At each point in time, a probability distribution over the state x_t is called a *belief*, denoted by *Bel(*x_t*)*. A belief represents the probability that state x_t is equal to the true state. The belief distribution *Bel (*x_t*)* is a posterior density over state x_t conditioned on all past available measurements $z_1, z_2, z_3 \dots z_t$.

$$\mathrm{Bel}(x_t) = p(x_t | z_1, z_2, z_3 \dots z_t) \tag{2.18}$$

Speaking in general terms, it represents the probability of a random variable to be present at state x_t given history of sensor measurements [210]. If the number of measurements increase over time, the complexity of computing such probabilities grows exponentially [209]. To avoid this, the Bayesian filter assumes that the dynamic system is a Markov process. A process is a Markov process when the present state x_t is sufficient to infer the next state x_{t+1} and no further information about previous states $x_{t-1}, x_{t-2}, \dots, x_1$ are needed.

Figure 2.26 illustrates such a hidden Markov process. The measurement at time *t* depends stochastically on the pedestrian's current physical location at time *t* (state x_t). The location at time $t + 1$ depends only on the previous state at time *t*, and previous states do not provide any further information [209]. This kind of model is also termed as Hidden Markov model (HMM).

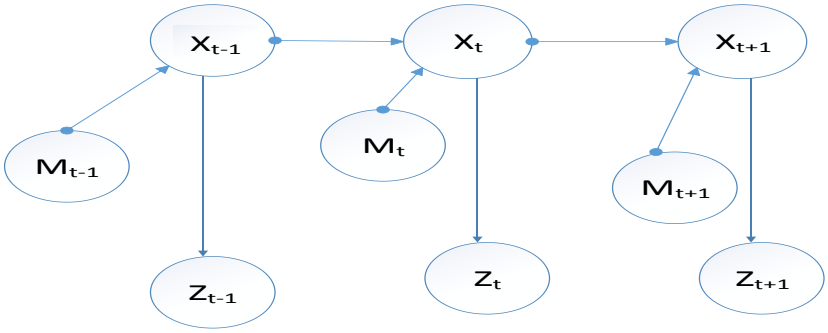


Figure 2.26: Graphical representation of the hidden Markov model for a localization problem.

HMM is a Markov process, where the parameters of the original system cannot be observed directly and the measurement might not represent the parameters exactly, hence they are only estimates [209]. Equation 2.18 can reformulated as equation 2.19. Here, z_t is the measurement data at time *t* and m_t is the motion data at time *t*.

$$\mathrm{Bel}(x_t) = p\,(x_t | z_{1:t}, m_{1:t}) \tag{2.19}$$

The state changes from x_{t-1} to x_t by applying the motion data. After that, belief can be calculated before incorporating the current measurements. This probability distribution is called *prediction* and is denoted by $\overline{\mathrm{Bel}}(x_t)$. Calculating $\overline{\mathrm{Bel}}(x_t)$ from $\mathrm{Bel}(x_t)$ using the measurement data at time *t* is called *correction* or *measurements update* (see equation 2.20). These are the two essential steps of Bayesian filters.

$$\overline{\mathrm{Bel}}(x_t) = p\,(x_t | z_{1:t-1}, m_{1:t}) \tag{2.20}$$

In general, there are two popular families of recursive state estimation techniques for realizing the Bayesian filter. The two families differ in how they represent the probability density over state x_t. The most popular family of Bayesian filter is Gaussian filters. Gaussian filters are unimodal with a single maximum and posterior focused around the true state with a small margin of uncertainty. "*They assume that the functional form of probability density over the state* x_t *is fixed and known, but its parameters are unknown"* [210]. The Kalman filter is the most widely used member of this family. They represent belief by multivariate normal distributions expressed by two set of parameters, the mean and the covariance.

The second popular family of Bayesian filter is nonparametric filters. They do not depend on a fixed functional form of posterior. Instead, they approximate belief by a finite number of values, each approximately corresponding to a region in state space. As the number of values approaches infinity, the approximation error converges uniformly to zero [205]. Particle filter is the most popular members of this group.

Kalman filter

The Kalman filter (KF) is a mathematical tool developed by Rudolph E. Kalman [152]. It is based on a Markov process and is used to estimate the state of a linear system. Measurements are processed that are linearly related to the system state [203].

The sensor fusion framework is modified to be implemented by linear system models and simple Gaussian distributions. System and measurement noise, which are assumed to be zero-mean and Gaussian distributed, are taken into account within a system and a measurement model [134]. Predictions of future states are based on the system model. These predictions are evaluated on the basis of the measurement model. There are five core elements of the Kalman filter [3]. These are as follows:

- State vector and error covariance matrix
- System model
- Measurement vector and measurement noise covariance matrix
- Measurement model
- KF algorithm

State vector and error covariance matrix: The state vector x_t is a set of quantities (states) that can sufficiently model the movement of a system. It may be a constant or time varying. An error covariance matrix P_t describes the uncertainties in the KF's state estimates.

System model: The system model encodes prior knowledge of how the KF's states and error covariance matrix vary with time.

Measurement vector and measurement noise covariance matrix: The measurement vector z_t is a set of simultaneous measurements of the system's properties, which are functions of the state vector. The measurement vector has an associated measurement noise covariance matrix R_t. It describes the statistics of noise on the measurements.

Measurement model: The measurement model describes how the measurement vector varies as a function of the true state vector (as opposed to the state vector estimate) in the absence of measurement noise.

KF algorithm: The Kalman filter algorithm utilises the measurement vector, measurement model, and system model to derive optimal estimates of the state vector. This is done by updating the states with weighted measurements recursively, based on their statistical information. The system and measurement model at time t are defined as follows:

$$x_{t+1} = A\,x_t + w_t \tag{2.21}$$

$$z_t = H\, x_t + v_t \tag{2.22}$$

where A and H are the known matrices, w_t denotes the system noise and v_t denotes the measurement noise. The noises are zero mean, white and Gaussian distributed such that

$$w_t \sim N(0, Q_t) \tag{2.23}$$

$$v_t \sim N(0, R_t) \tag{2.24}$$

Q_t and R_t are the covariance matrix of system noise and measurement noise respectively and determined experimentally. In this thesis, we use the Bayesian interpretation of the Kalman filter [211], which is based on assuming that the noises are Gaussian and that the states have a Gaussian distribution. Details of the implemented algorithm are discussed in Chapter 6. The prediction and measurement step of the Kalman filter are given as follows:

Prediction

$$\hat{x}_t^- = A\hat{x}_{t-1} \tag{2.25}$$

$$P_t^- = A\, P_{t-1} A^T + Q \tag{2.26}$$

Measurement update

$$K_t = P_t^- H^T (H P_t^- H^T + R)^{-1} \tag{2.27}$$

$$\hat{x}_t = \hat{x}_t^- + K_t(z_t - H\hat{x}_t^-) \tag{2.28}$$

$$P_t = P_t^- - K_t H P_t^- \tag{2.29}$$

These reiterate to find the optimal estimate. Here $\hat{x}_{t-1}$ is the previous best estimate at time $(t-1)$ with the corresponding covariance denoted by P_{t-1}. $\hat{x}_t^-$ is the predicted state at time t with the corresponding covariance denoted by P_t^-. K_t is the Kalman gain at time t. It is used to determine the weighting of the measurement information in updating the state estimates.

For detailed derivations of the above equations please see [134]. The equations above show the basic linear Kalman filter, however for most real world applications the dynamics of pedestrian are not actually linear nor are the sensor measurements truly linear. Therefore, several enhancements, such as the Extended Kalman filter (EKF) and Unscented Kalman filter (UKF), target these restrictions. EKF and UKF do not share the characteristic of being optimal estimators with the original Kalman filter [202].

Particle filter

The particle filter is a nonparametric implementation of the Bayes filter. It was first introduced in 1993 [212]. It represents the belief by a set of state samples, called *particles,* drawn from the posterior or belief

$$\mathrm{Bel}(x_t) = X_t = x_t^1, x_t^2, x_t^3 \ldots\ldots x_t^M \quad (2.30)$$

where M denotes the number of particles. Each particle x_t^M is a possible state at time t, that is, a hypothesis as to what the true world state may be at time t. The key point of Particle filter is *distribution* which is not limited to Gaussian only and second the number of particles. For an infinite number of particles the approximation would be optimal. In the case of a finite number of particles the approximation is not optimal, but as long as the number of particles is not too low, the difference is negligible [205].

Initially the particles are assumed to be distributed according to some distribution function and in case states cannot be narrowed down, particles are assumed to be uniformly distributed. At every time step each particle is moved randomly according to the movement model. As a result particles would be more likely performing anticipated movements rather than unlike movements. Following this particles are evaluated based on the measurements; and weights are computed according to the particle likelihoods. This is called the update step. The steps are performed iteratively.

Table 2.5 depicts the algorithm of a basic Particle filter. Lines 3 and 4 represent the prediction and update step. The input of this algorithm is the particle set X_{t-1}, along with the most recent motion m_t and the most recent measurement z_t.

Algorithm Particle filter ($X_{t-1}, m_t, , z_t$)

1. $\overline{X}_t = X_t = \varphi$
2. for i = 1 to M do
3. *sample* $x_t^{[i]} \sim p(x_t | m_t, x_{t-1}^{[i]})$: *motion model*
4. $w_t^{[i]} = p(z_t \mid x_t^{[i]})$: *measurement model*
5. $\overline{X}_t = \overline{X}_t + \left\langle x_t^{[i]}, w_t^{[i]} \right\rangle$
6. end for
7. for i = 1 to M do
8. *draw q with replacement from probability* $\propto w_t^{[i]}$
9. $X_t = X_t + x_t^{[i]}$
10. *end for*
11. *return* X_t

Table 2.5: Particle filter algorithm (Source: [205])

Line 3 generates a hypothetical state $x_t^{[i]}$, by sampling the state transition probability $p(x_t|m_t, x_{t-i}^i)$. In other words, a new particle (particle number *i* of prediction set) is generated using the *i*-th particle of the previous posterior distribution at time *t – 1* ($x_{t-1}^{[i]}$) and current motion data m_t. The line 4 incorporates the measurement z_t into the particle set. It calculates importance factor $w_t^{[i]}$, that is the probability of the measurement z_t under the particle $x_t^{[i]}$.

Lines 8 to 11 implements resampling or importance resampling step. The algorithm draws with replacement *M* particles from the temporary set $\overline{X}_t$ and fills in the set X_t. The chance of a particle to be drawn from the input set is proportional to its weight $w_t^{[i]}$ (importance factor). The idea is, that particles with low weights hardly add to X_t and therefore most likely will not be drawn whereas particles with high weights will do. In some cases, the distribution of particles may also change after the *resampling* step.

Many versions of the algorithm however never resample, instead maintain an importance weight of each particle that is initialized by 1 and updated multiplicatively. The accuracy of the estimation increases with the number of particles. Although the estimation technique is sequential, the number of particles in belief set depends on both the computational resources and timing of new data.

The main advantages of Particle filter in comparison to Kalman filter is that it avoids linearization errors and moreover, represents other PDFs apart from Gaussian distributions [205]. However, the major drawback of the particle filter is its high computation complexity. For e.g. 1600 particles are needed for each filter update for a 40m×40m experimental area to achieve the best performance [213]. This large computation workload cannot be handled by the memory constrained devices to give real-time updates to the user. Hence, this thesis chooses the Kalman filter to post-process the estimates.

2.4 Commercial products

There are a number of tailor made applications that guarantee to provide positioning information over different indoor environments using various sensors and technologies. The Navizon application [17] estimates a mobile device's current geographic location, latitude and longitude, by triangulating Wi-Fi signals, in addition to GPS. They propose to provide reliable position information anywhere – indoors and outdoors. Tests performed in [214] on Navizon 2 demo application showed that positioning accuracy ranging from 3 m – 5 m is possible. However, the application provides accurate positioning only in case when following conditions were compiled: accurate fingerprinting, precise mapping and good Wi-Fi coverage.

IntraNav [215] leverage sensory data reported by inertial sensors, including accelerometers, gyroscopes and digital compasses to track locations as device moves, and later

integrate UWB signals to compensate for accumulated errors in motion tracking. Based on the similar concept of positioning few other commercial vendors are Navigine [216], Skyhook [18] that use WI-Fi, Bluetooth, etc. instead of UWB. One major problem faced by such applications is precision is low because of the non-stationarity of radio map. Navisens [24] in comparison use only the information from inertial navigation system for positioning. The vendor claims it can trace the location of a person with an accuracy less than 1 m and the direction they are facing with an accuracy of 1 cm. The positioning algorithms and sensor fusion techniques that they apply are not disclosed. So, it is more of a black box.

In 2006, Infsoft [217] developed an integrated prototype of software and hardware. It makes use of multiple wireless technologies – Wi-Fi, Bluetooth and UWB to localize pedestrian and objects. These technologies can work in synchronous and asynchronous mode. In case of UWB, the locator nodes measure the distance to several APs. The locator node processes the data it receives from the APs and sends them to the backend where the position is displayed on a map. In case positioning data should be immediately displayed on a mobile device, the locator tags communicate with the smartphone via Bluetooth directly. In case of Bluetooth, the beacons are installed across the building that transmit continuous frames which are received by smartphone application. For an area of 370,000 square metres, it necessitates installing of 1200 beacons which is itself a laborious and time taking task. In case of Wi-Fi, positioning is accomplished in the similar manner as done by Navizon. Company claims to achieve an accuracy of 5 - 15 m which is however too low for indoor localization [218].

Ericsion Labs' introduced a hardware independent positioning solution in 2011 [219]. It gave developers a tool to create their own indoor maps and enable personal positioning. As reported by the vendors no third party is needed to design the maps, however pedestrian is needed to map each wireless AP manually. Smartphone application scans for APs within reach and sends the data to central server, via an application programming interface, which returns a position using trilateration. Note, a location can only be provided within an area with access to at least three APs as illustrated in Figure 2.27. It is not specified how user privacy is kept secured, importantly the position computation is done on server end thus there is a possibility of the data being mishandled by others at server end.

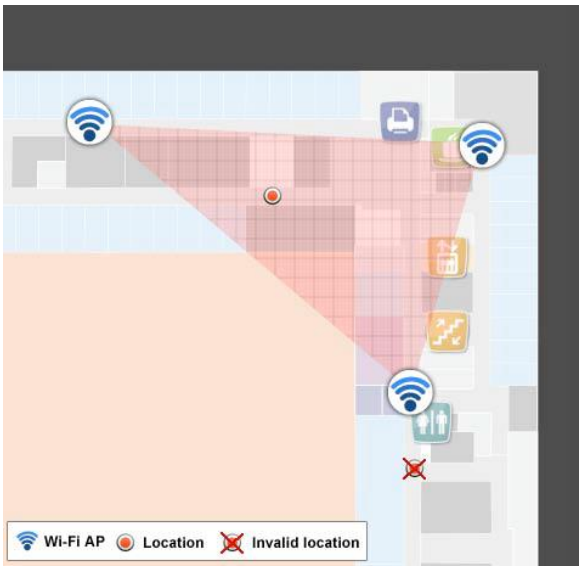


Figure 2.27: Illustration of valid and invalid locations for positioning, within a floor plan in Ericsson Labs' system (Source: [219]).

Meanwhile Google Indoor Maps [220] reduced the burden of designing maps. It uses the same interface as Google Maps [220] and is accessed through it. Figure 2.28 illustrates the working of Google map indoors and outdoors. Its indoor navigation algorithm is based on triangulation of received signal strengths from nearby Wi-Fi APs and cellular towers to determine user's position. In some mapped venues, position is enhanced by three-dimensional view of parts of the interior constructed with photographs and walks through it. Accuracy is not given as such, however it is limited by the signal broadcast strength and potential interferences from thick walls.

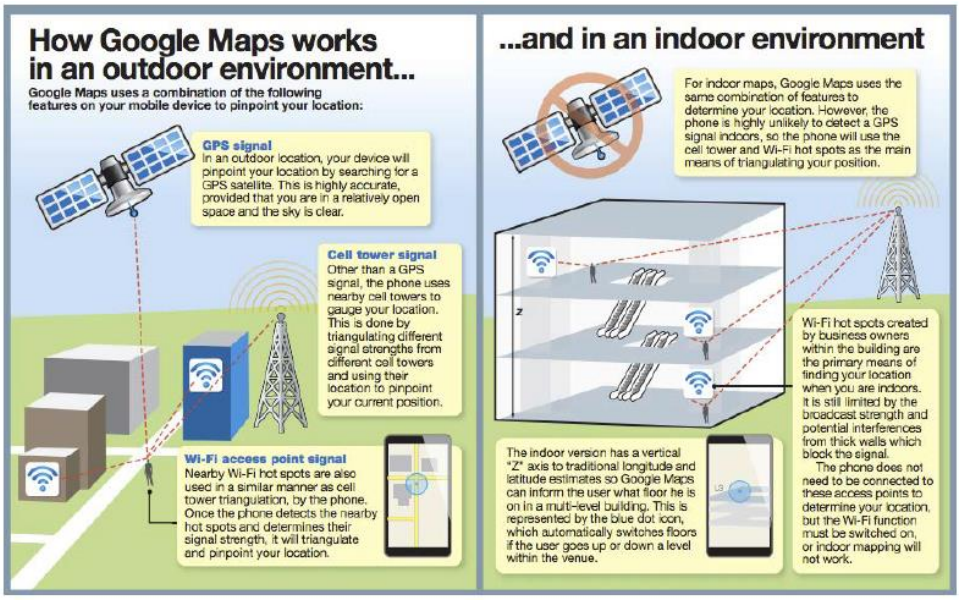


Figure 2.28: How Google map work indoors and outdoors (Source: [220]).

Another classical product by Google Inc. – Google glass [43] which has often been used for augmented reality applications. It uses complex mathematical algorithms to fuse the data from INS, GPS and other sensors to accurately localize the pedestrian. The positioning algorithms are not known however Google Inc. claims to achieve accuracy less than 2 m. The cost of gadget is 1000 GBP.

In 2012, Quuppa introduced High Accuracy Indoor Positioning (HAIP) system [221]. It uses Bluetooth smart as the core technology and exists in two different variants. The first is a mobile-centric system that analyse RSSI parameters to determine user's position. An application developed for cell-phones broadcasts a special Bluetooth smart request, which causes all nearby HAIP-locators (Bluetooth smart beacons) to respond to the inquiry. When the application receives the response, a request is sent to a locator database server that responds with the position of the discovered beacons. The information about the locator position together with the RSSI parameters which were received in the broadcast response allows the application to estimate pedestrian's position. Company claims to achieve an accuracy of 5-10 m using this application. This accuracy may be sufficient for outdoor localization however for indoor environments again it is too low. The second variant is a network-centric solution which also uses Bluetooth smart for sending radio packets however it differs from the prior in the sense that it requires special tags for positioning. The system does not rely on signal strength, instead angular estimation is used where the locator's beacon is equipped with an antenna capable of determining the angle of arrival of incoming radio packets. This angle together with information about locator position and the known height of the tag allow the system to determine pedestrian's position. Company claims to achieve an accuracy of 0.3 – 1 m using this variant. Accuracy is sufficient for indoor localization however an important requirement for the positioning to work by this solution is that the tag should not deviate from height of more than 1 m. In addition, environment should be evenly augmented with Bluetooth beacons. This may require time and effort.

In 2013 Apple introduced iBeacon [19] that uses Bluetooth low-energy wireless technology to provide location-based services. The iBeacons are low energy beacons that continuously transmit packets which are received by Bluetooth enabled devices (e.g. smartphone, PDA, etc.) to enable positioning. These are usually installed at requisite distances across the building. The positioning is typically made against a single beacon and will thus provide an approximate position in a large area if several iBeacons are deployed. Virgin Atlantic airlines trialled the technology at Heathrow airport, however results are unpublished. The company however claims that positioning accuracy would increase with the density of beacons where as it could subside with distance from beacons or in the presence of obstacles e.g. humans, etc.

A common issue reported in all the above discussed wireless (Wi-Fi and Bluetooth) enabled consumables is that user privacy is not secured (even though the identity of the user is not exposed) given that the APs or beacons can monitor the network physical address of a mobile user at any time.

Senion's StepInside [222] nevertheless keeps the user privacy confidential. Every user is assigned a random ID in the system. This means all user position data is treated anonymously and cannot be linked to a physical smartphone. StepInside makes use of multiple wireless technologies (Wi-Fi and Bluetooth) and INS to localize the user with an accuracy less than 2 m in 95% of the cases. It is however required that wireless technologies cover the entire area, this implies that beacons are installed at requisite distances across the venue. Also, beacons should be provided by StepInside – no third party beacons can be used. In a way, limiting the user's freedom.

In oppose to varying signal strengths and beacon installations, GeniusMatcher [223] developed a positioning application – Mally using computer vision and 3D technology. The user needs to take a picture of their surroundings and Mally compares it with the images stored in its database to compute the user's current location and orientation. On the same lines, researchers at TU Munich have designed an application – Navvis [224] using similar technologies but differing in the sensor fusion part. Tests are still in progress; however researchers report to achieve sub-metre level accuracy in experimental conditions. Given the complexity of computer vision algorithms [174], it is not at all clear whether it would be fast enough to report the pedestrian's location with no latency. In addition, their applicability is limited e.g. Mally can only provide location based services across Israel and some malls of UK.

There are a several other indoor location positioning initiatives, but these discussed above are the major players in indoor positioning and cover a range of technology approaches discussed earlier in the Section 2.2 e.g. Indoo.rs is working on solutions that leverage similar technologies to that of HAIP. It guarantees accuracy less than 5 m radius in 95% of the cases.

2.5 Summary

This chapter presented a comprehensive review of the existing indoor positioning technologies. Also, it discussed their limitations, benefits and systems designed using these technologies. Table 2.6 summarizes the salient features of these technologies. Several sensor fusion techniques were also discussed. It is however becoming apparent that there is no single technology that can be ubiquitously accurate and reliable, as a result integrated systems are been designed to cater such needs.

Notably, majority of such systems use some or the other form of infrastructures or dedicated hardware for accurate positioning (less than 1.5 m). Systems that provide high reliability in general have difficulty in deployment. Also, it is observed that there is a trade-off

between accuracy and cost. Sensing infrastructures involving higher investments provide better accuracy where as those involving lower investments provide lower accuracy.

INS can provide infrastructure free positioning. Furthermore, they offer an additional degree of privacy since the user can choose either to share or not to share their location information with any third party. However, the cost associated with this positioning is that errors accumulate over time. As a result, instead of localizing a pedestrian in a particular room, it often localizes the pedestrian in the adjoining room separated by a dividing wall. By integrating with other sensors, pedestrians have however been localized to a room 1 but studies still show that problem persists in estimating whether a pedestrian is present on this side of the dividing wall or the opposite of the dividing wall [36, 172, 225].

Bhattacharya *et al.* [225] implemented a tailor made Ekahau prototype (based on Wi-Fi fingerprinting) in addition to a grid-based location model to position the pedestrian. The model represents indoor environment using a grid constructed on top of a floor plan. Whenever the system overestimates or underestimates the position, the grid system incorporates the inaccuracies of system to reduce the errors to the nearest cell. Out of 40 tasks, three navigation errors were recorded while using the instruction aided navigation. Here an error means that the user entered the wrong aisle, or walked past the correct one. Thus, an error of about 10% was recorded at the context of tailor made solutions.

Taking these problems as a motivation, the research efforts are therefore directed towards investigating a method that is reliable, low cost (with minimal infrastructure deployment and low computation cost), do not intervene privacy, and can provide localization with that much accuracy to differentiate a pedestrian on either side of the *dividing wall*. This is also what we foresee as a research gap and which we would fill in by this thesis.

Technology/ Sensors	**Accuracy**	**Range**	**Infrastructure cost**	**Strength**	**Weakness**
GNSS	95 % within 6 – 10 m	Anywhere outdoor	Not applicable	High availability outdoors.	1. Minimal availability indoors. 2. Processing time is slow.
WLAN	95% within 1 – 5 m	Indoor: ~ 32 m Outdoor: ~ 95 m	> 200 GBP	Readily available in most buildings.	1. Signal strength degrades because of changes in physical conditions. 2. Precision is low. 3.High power consumption
RFID	95% within 1 – 3 m	<= 100 m	> 650 GBP	1. Quick response time. 2. Unaffected by non-line of sight condition.	1. Low power supply. 2. Sporadic location updates.
Cellular	Cell based	<= 35.0 km	>300 GBP	Presence of infrastructure.	1. Infrequent availability of the signal indoors. 2. Affected by multipath.
UWB	95 % within 15 cm – 4 m	50 m (with reduced accuracy) 9m: (normally with chips due to power cap)	> 900 GBP	1. High bandwidth. 2. Low operating frequency.	Precise clock synchronization required between transmitter and receiver.

Bluetooth	95% within 10 m – 20 m	<=100 m	> 200 GBP	High speed data transfer.	Explicit links between devices required.
Magnetic	95% within 0.1 – 2 m	Available everywhere.	Not applicable	Available everywhere.	1. Flux density varies due to interference from the surroundings. 2. Precision is low.
Inertial	95% within 1 – 5 m	Indoor and outdoor	Not applicable	1. High update rate. 2. Portable and easy to handle.	1. Error accumulation with time. 2. Reference is linked to initial point. 3. Performance depends on the quality of inertial sensor.
landmark	95% within 1 – 3 m	1-10 m	> 100 GBP	High reliability and high precision.	1. Inability to work under thick smoke, gust of air particles and dark conditions. 2. Ambient noise in the form of light or thermal radiation, signal reflections and illumination variability 3. Time and effort needed to augment environment.
Sonic	95% within 3 cm – 4 m	2-10 m	> 1000 GBP	1. Robust to multipath fading. 2. Privacy is maintained.	1. Noise from the surroundings. 2. Variation with temperature.

Table 2.6: Summary of the positioning technologies and systems.

3. Chapter 3: Case study – feature based localization

This thesis is about investigating techniques that can be employed for low cost accurate, reliable, logical localization of the pedestrians in buildings with minimal usage of infrastructure. Based on the research gaps identified in the previous chapter, it is understood that the key to solving the problem of indoor pedestrian localization is a combination of different positioning techniques and sensor modalities. This chapter provides a first step towards developing such a system. Smartphones have redefined the notion of mobile computing platform. Being low cost, nearly ubiquitous, easily available, portable, and affordable by all these can also support positioning [226]. Also, as stated by Kothari *et al.* [227].

"In recent years, smart-phones have re-defined the notion of mobile computing platforms. Ever improving features of affordability, ubiquity, and portability, increased sensory and computational power along with low power consumption fuelled by readily available batteries, have opened up a number of interesting applications. One such application is software that is location-aware."

Therefore this research's first focus was to investigate how the smartphones can be used for positioning. In principle, this chapter investigates the performance characteristics of smartphone embedded INS – accelerometer and gyroscope and also investigates how a single sensor – gyroscope can be used for positioning. Parts of this chapter are based on the paper entitled "Vehicle localization based on heading data using low cost sensors", which was presented in 46th Annual Conference of the Universities Transport Study Group (UTSG) held in Newcastle, UK [228] (copied in Appendix A).

To begin with, this chapter first gives an introduction about the Android platform. Next, it describes few of the salient features of the used smartphone – HTC Sensation Z710e in Section 3.2. In Section 3.3, it investigates the performance characteristics of INS embedded in this smartphone. Third, this chapter presents a proof of concept study to localize a smartphone mounted on top of a toy vehicle in Section 3.4. And lastly, findings from this chapter are summarized in Section 3.5. The key finding of this chapter is that feature detection in an INS enables error bounded localization in a feature rich map.

3.1 Android

Android is an open source Linux based software stack for mobile devices that is created by open handset alliance (OHA). The OHA is an alliance of several handset manufacturers, software providers, mobile operators and integrated chip manufacturers like Google, Intel, HTC, etc. [229]. The Android is available in different versions. The Android 1.x and 2.x versions are for smartphones and low-end tablet computers, whereas the Android 3.x version is only for tablet

computers. The recent version of Android is the 4.x series. It combines the functionalities of the 2.x and 3.x series, and is also applicable to latest smartphones and tablets [230].

The Android operating system is supported by a software development kit (SDK) to create new applications. These applications are written in Java programming language [231]. The architecture of an Android operating system is illustrated in Figure 3.1. It is sub-grouped into five categories which are as following [232]:

- Application layer: This layer is the top most layer and the applications are written in this layer.
- Application framework: The framework consists of blocks with which the application interacts with. These blocks are responsible for managing the basic functions of phone like resource management, voice call management etc.
- Libraries: The available libraries are written in C/C++ and they are called through a Java interface.
- Android runtime: The Android runtime is sub-divided into two components. These are as follows:
    - Core libraries: The core libraries provide most of the functionalities defined in the Java core libraries.
    - Dalvik virtual machine: The machine operates like a translator between the application side and the operating system. It is optimised for low processing power and low memory requirements.
- The kernel: The kernel is based on Linux operating system. It is responsible for managing the hardware and contains essential device drivers. Moreover, it is also responsible for memory management, process management and networking.

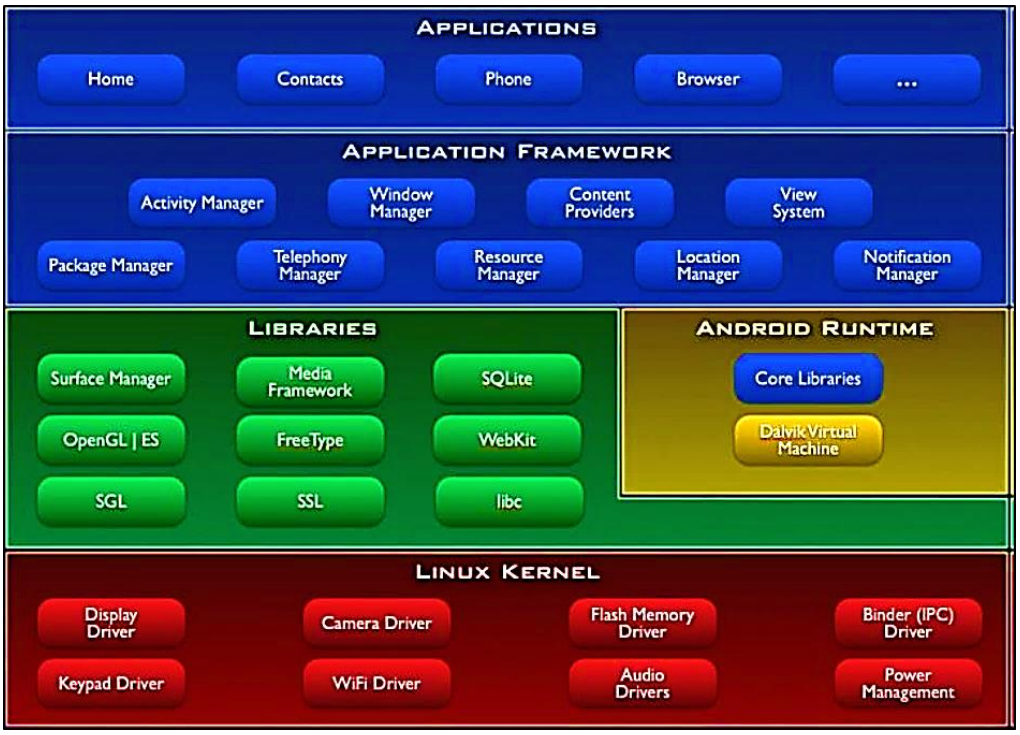


Figure 3.1: Android architecture diagram (Source: [232])

As a part of the project, I was given HTC Sensation Z710e, to work on. Some of the salient features of the smartphone are discussed in the following section – Section 3.2. However, on comparing the features of the smartphone – HTC Sensation Z710e with the other smartphone present in the market (see Table 3.1), we noticed that the HTC Sensation Z710e had rich set of sensors in comparison to the other smartphones that can provide positioning at lesser price range with the average acceptability. Therefore, we planned to carry on this research work with HTC Sensation Z710e. It is to be noted that the given costs of smartphones and ratings in Table 3.1 are of the year 2012 when we started off this research study.

Smartphone	Features	Threats	Ratings (out of 5)	Cost (£)
Motorola Atrix 2	Sensors: Fingerprint, accelerometer, proximity, compass, camera. Connectivity: WLAN, Bluetooth, GPS, USB Processor: 1.5 GHz dual core	No radio	4.0	250
Samsung Galaxy S2	Sensors: Accelerometer, gyro, proximity, compass and camera. Connectivity: WLAN, Bluetooth, GPS, USB, Near Field Communications (NFC) and Radio Processor: 1.2 GHz dual core	Java via MIDP emulator	4.5	350
Motorola Razr XT910	Sensors: Accelerometer, proximity, compass and camera. Connectivity: WLAN, Bluetooth, GPS and USB. Processor: 1.2 GHz dual core	No radio	4.0	270
Samsung Galaxy Nexus 19250	Sensors: Accelerometer, gyro, proximity, compass, barometer and camera. Connectivity: WLAN, Bluetooth, GPS, USB, NFC. Processor: 1.2 GHz dual core Cortex-A9	No Radio and No Java support	4.5	240
HTC Sensation Z710e	Sensors: Accelerometer, gyro, proximity, ambient light sensor, compass and camera. Connectivity: WLAN, Bluetooth, GPS, USB, NFC and radio. Processor: 1.2 GHz dual core Scorpion	Not applicable	3.5	320
Sony Ericsson Xperia X12	Sensors: Accelerometer, proximity, compass and camera. Connectivity: WLAN, Bluetooth, GPS, USB, NFC and radio. Processor: 1.0 GHz Scorpion	Not applicable	3.0	280

Table 3.1: Top Android smartphones launched in 2011 [233-238].

3.2 Salient features

HTC Sensation Z710e is co-developed by Google and HTC. It uses Android version 2.3.4 (Gingerbread), and is also upgradable to version 4 (Ice cream Sandwich). Table 3.2 summarizes few of the key features of this smartphone [239, 240].

S. No.	Properties	Specifications
1.	Network	2G & 3G
2.	Physical Dimensions	Height: 126.1mm; Width: 65.4 mm; Depth: 11.3 mm; Weight: 148 gram
3.	Memory	1. Internal phone storage: 1 GB, 768 MB RAM 2. Expansion slot: micro-SD, up to 32 GB, 8 GB included
4.	Connectivity	1. GPRS: Up to 114 kbps 2. EDGE: Up to 560 kbps 3. WLAN: Wi-Fi 802.11 b/g/n, and Wi-Fi hotspot 4. Bluetooth: Yes, version 3.0 5. USB: Yes, micro USB version 2.0 6. Near field communications 7. GPS: Yes
5.	Sensors	Accelerometer, gyroscope, digital compass, proximity sensor, ambient light sensor and camera (8MP + VGA)
6.	CPU	Dual-core 1.2 GHz Scorpion
7.	Battery	1. Stand-by: Up to 350 hours (2G) / Up to 400 hours (3G) 2. Talk time: Up to 8 hour 20 minutes (2G) / Up to 6 hour 40 minutes (3G)
8.	Display	540 x 960 pixels, 4.3 inches

Table 3.2: Some of the key features of HTC Sensation Z710e.

Table 3.2 shows that this smartphone is inlaid with various essential technologies and components that can not only assist in making phone calls but can also assist in the process of positioning. For instance, this smartphone has 2G and 3G network connectivity. This enables us to make voice and video calls, etc. over the network. While the inbuilt sensors such as INS – accelerometer, gyroscope and digital compass, and technologies particularly RF based such as Wi-Fi and Bluetooth can be used for positioning.

As discussed in the previous chapter (see Chapter 2, Section 2.2.2), RF based positioning technologies were sufficiently able to locate the pedestrian indoors. However, their performance was primarily limited because of two key constraints. First, they required deploying infrastructure in terms of Wi-Fi APs, Bluetooth beacons, etc. for positioning and second they required a data connection between the transmitter and receiver. Also, the quality of data connection significantly

degraded with the change in infrastructure of the indoor environments. As a result, the positioning solution behaves unexpectedly.

On the other hand, INS particularly suffered from drift errors (see Chapter 2, Section 2.2.4). This made the inertial based positioning solution unreliable. However, it provided an autonomy in positioning, since these did not required the support of any form of infrastructure or any other facilities for positioning [134]. Also, as it is known from the literature review (see Chapter 2), that there is always an accuracy and cost trade-off. The systems that result in minimum localization error employ expensive sensors. So the following experimental tests (detailed in the next Section) are devised to investigate the performance of smartphone embedded INS – accelerometer and gyroscope.

3.3 Performance analysis of smartphone embedded INS

The smartphone HTCZ710e has an ultra-low power digital three-axis accelerometer sensor and an ultra-low power digital three axes gyroscope sensor manufactured by Invensense [234]. The accelerometer sensor measures the acceleration of device (in $\mathrm{m/s^2}$), by measuring the forces affecting the sensor. Let F be the forces affecting the sensor and m is the mass of the device so

$$F = m\,\ddot{x} \tag{3.1}$$

$$\ddot{x} = {}^{F}/_{m} \tag{3.2}$$

where $\ddot{x}$ is the acceleration of the device. The gyroscope sensor measures the turn rate (angular velocity) of the device (in $\mathrm{rad/s}$). Integrating the gyroscope sensor measurements over time provides the total change in angular position during the integration time. While double integrating the accelerometer sensor measurements over a period of time provides the relative change in linear position during that time [231].

These inertial sensors – accelerometer and gyroscope are continuously subjected to errors which limit the accuracy to which the correct value can be measured [241]. Primarily, there are two types of error [133] – systematic and random. The systematic errors are deterministic in nature. These can be removed by calibration while random noise errors are stochastic in nature and often have a Gaussian distribution [3]. The systematic errors include the bias errors and the scale factor errors. These errors govern how well the positioning system would perform. The actual value of the gyroscope angular velocity $\dot{\theta}(t)$ and the accelerometer linear acceleration $\ddot{x}(t)$ differ from the measured values by addition of these errors as illustrated by the following equations

$$\text{Gyroscope: } \dot{\theta}(t) = s\,\hat{\dot{\theta}}(t) + \epsilon + \mu \tag{3.3}$$

$$\text{Accelerometer: } \ddot{x}(t) = s\hat{\ddot{x}}(t) + \epsilon t + \mu \tag{3.4}$$

where $\dot{\theta}(t)$ and $\ddot{x}(t)$ are the actual value of the angular velocity and acceleration at time t respectively. $\hat{\dot{\theta}}(t)$ and $\hat{\ddot{x}}(t)$ are the measured value of angular velocity and acceleration at time t respectively. ϵ is a constant bias error, s is a scale factor error and μ is a random noise error with mean zero and Gaussian distributed such that

$$\mu \sim N(0, \sigma^2) \tag{3.5}$$

where σ is the standard deviation. A short discussion about these errors is detailed below.

3.3.1 Bias error

The bias of a sensor is the average error measured over a specified time at specified operating conditions. The bias generally, remains fixed throughout the INS operating period but may vary between different operating periods [133]. In a gyroscope, a constant bias error of ϵ when integrated causes the angular position $\theta(t)$ to grow linearly with time t. In contrast, a constant bias error of ϵ in an accelerometer causes the linear position x(t) to grow quardratically with time *t* when double integrated. Integrating equations 3.3 and 3.4, considering scale factor error to be zero to study the effects of only bias error on the actual values of the sensors.

$$\text{Gyroscope: } \theta(t) = \int \hat{\dot{\theta}}(t)\, dt + \epsilon t \tag{3.6}$$

$$\text{Accelerometer: } x(t) = \iint \hat{\ddot{x}}(t)\, dt + \epsilon \frac{t^2}{2} \tag{3.7}$$

where $\theta(t)$ is the angular position in radians and x(t) is the linear position in metres at time t. Therefore, it is essential to remove this error before proceeding further. It can be measured by taking a long term average of the sensor's output when the smartphone is neither accelerating nor undergoing any rotation. Table 3.3 lists the constant bias errors present in the accelerometer and the gyroscope, when the smartphone was kept at rest on a table for 7 hours. For an accelerometer, the bias error was calculated by aligning each of the device's sensitive axes parallel to gravity *g* and then subtracting the mean value of long idle log along that axes from *g*. In the case of gyroscope, when the device is lying still the expected output is zero so therefore any offset from zero is a bias in the sensor, and the mean value from a long idle log is the constant bias error from the sensor. So the mean value from a long idle log was taken to calculate the constant bias error of the gyroscope sensor.

Axis	Accelerometer (m/s^2)	Gyroscope (rad/s)
X	-0.4791	-0.2895×10^{-5}
Y	-0.3021	-0.0502×10^{-5}
Z	0.1228	-0.0533×10^{-5}

Table 3.3: Constant bias errors present in the two sensors - accelerometer and gyroscope.

3.3.2 Scale factor error

The scale factor error is the multiplicative error that represents the ratio of the change in output to the input from unity. The scale factor varies non-linearly with temperature but for a limited temperature range the scale factor could be considered as constant [3]. In an accelerometer and gyroscope, the scale factor error scales the true value by a factor s on integration as illustrated by equations 3.8 and 3.9 respectively. Integrating equations 3.3 and 3.4, considering bias error to be zero to study the effects of only scale factor error on the actual values of the sensors.

$$\text{Gyroscope: } \theta(t) = s \int \hat{\dot{\theta}}(t)dt \tag{3.8}$$

$$\text{Accelerometer: } x(t) = s \iint \hat{\ddot{x}}(t)\,dt \tag{3.9}$$

where $\theta(t)$ is the angular position in radians and $x(t)$ is the linear position in metres at time t. The scale factor error can be measured by affecting the sensor with a fixed input signal for a period of time while measuring the output signal. Table 3.4 lists the scale factor errors present in the accelerometer and gyroscope respectively. In an accelerometer, the scale factor error was calculated by aligning the smartphone's sensitive axes parallel to the gravity g after the bias error has been removed by calibration and subsequently, measuring the corresponding acceleration along the three axes.

Axis	Accelerometer (m/s^2)	Gyroscope (rad/s)
X	1 .0481	0.9828
Y	1. 00687	0.9780
Z	0.9781	0.9858

Table 3.4: Scale factor errors present in the two sensors - accelerometer and gyroscope.

In contrast, the scale factor error along the gyroscope's sensitive axis was calculated by comparing the cumulative change in the angles to the true value when the gyroscope is rotated by fixed number of turns along the three axes. The smartphone was placed on a rotary tripod, with the axes of gyroscope aligned to the rotational axis of the tripod. The tripod was then rotated 5 full turns (31.41 radians) in clockwise direction. The accumulated rotation along the smartphone's coordinate axes - pitch, roll and yaw (see Figure 3.2(B)) were 30.8774, 30.7272 and 30.9712 radians respectively. So the scale factor error was determined according to equation 3.10.

(A) (B)

Figure 3.2: (A) Global Coordinate System (B) Smartphone's Coordinate system

$$s = \frac{\text{Accumulated rotation value}}{\text{True value}} \tag{3.10}$$

3.3.3 Random noise error

The output of the accelerometer and gyroscope is perturbed with thermo-mechanical and electrical noise which fluctuates at a rate much lower than the sampling rate. The spectrum of such a noise signal for frequencies below 1Hz is white so it cannot be calibrated and compensated. The result of integrating the random noise causes the bias to fluctuate and creates angular random walk in the integrated gyroscope output and velocity random walk in the integrated accelerometer output. The characteristic of such a noise is modelled by the Allan Variance method [133, 241]. The steps of computing Allan Variance are as follows:

1. Assume N samples of data from the sensor output.
2. Divide the data into n equal clusters of length τ_o with $n < N/2$
3. Average the data in each cluster to obtain a list of averages $a(t_1), a(t_2), a(t_3) \ldots \ldots a(t_n)$
4. Compute Allan Variance using (3.11)

$$AV = \frac{1}{2\,(n-1)} \sum_{i=1}^{n-1} (a\,(t)_{i+1} - a\,(t)_i)^2 \tag{3.11}$$

where n is the number of clusters and N is the number of data samples from the sensor. The Allan variance is plotted as a function of averaging time on a log-log scale and the slope of Allan variance plot represents the random process which helps to identify the random noise characteristics in the sensor. The random walk introduced by the white noise appears as an asymptote with slope −0.5 and the bias instability appears as a flat region with slope 0 in the Allan Variance plot. Numerically, the bias instability is measured by reading the minimum value on the Allan variation plot and the random walk is measured by drawing a line through the measured points on the Allan variation plot that form the downward slope at τ_o = 1 [241]. Table 3.5 and

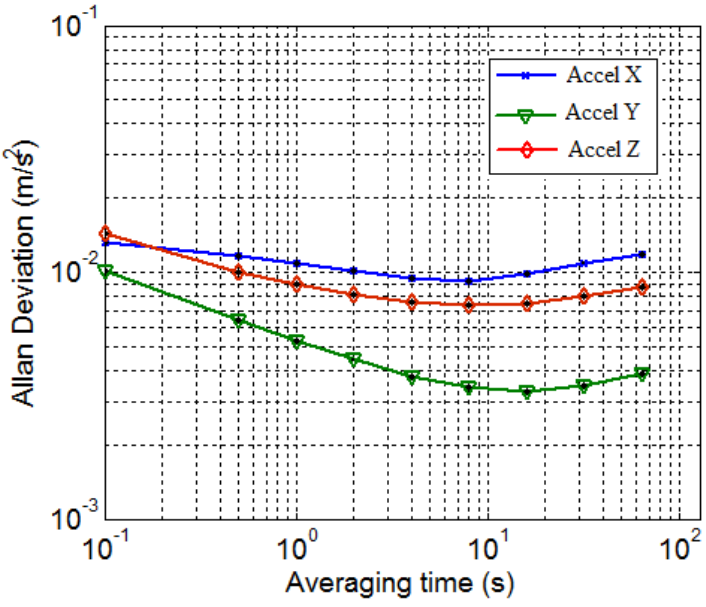
Table 3.6 lists the random noise errors present in the accelerometer and gyroscope sensor respectively derived from the Allan Variance plot as shown in Figure 3.3((A) and (B)).

Axis	**Bias Instability (m/s^2)**	**Random Walk ($m/s/\sqrt{s}$)**
X	0.0092349 at 8s	0.01094
Y	0.0032846 at 16s	0.005288
Z	0.00736629 at 8s	0.008921

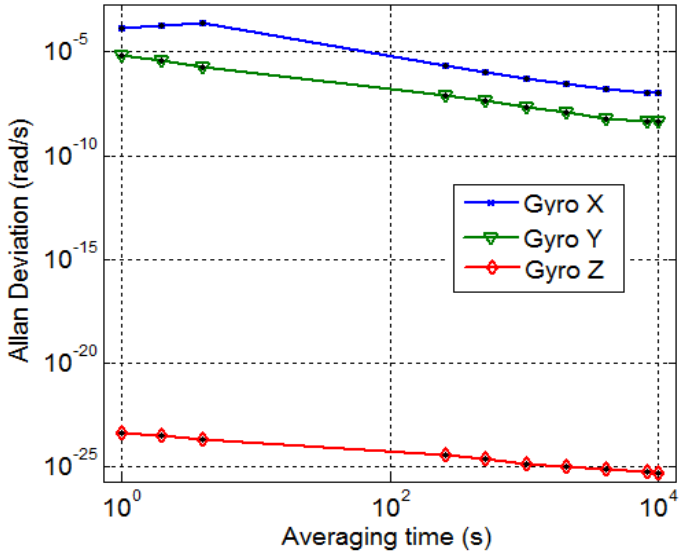
Table 3.5: Random noise characteristics of the accelerometer.

Axis	**Bias Instability (rad/s)**	**Random Walk ($rad/\sqrt{s}$)**
X	1.07821×10^{-7} at 8192s	1.344×10^{-4}
Y	4.45602×10^{-9} at 8192s	6.641×10^{-6}
Z	5.20406×10^{-26} at 10000s	4.264×10^{-24}

Table 3.6: Random noise characteristics of the gyroscope.



(A)



(B)

Figure 3.3: Allan Variance characteristics of the two sensors: (A) accelerometer and (B) gyroscope.

Interestingly, there are two key findings from this investigation of the performance characteristics of the smartphone embedded INS. These are as follows:

1. Bias instability and random walk can have a significant impact on the INS performance in measuring position and orientation. This instability and random walk will be manifested as a 'drift' in the overall positioning process. Therefore 'feature detection' to periodically correct drift would be desirable.

2. Bias instability and random walk are significantly less for the gyroscope sensor. Also, single integration is done on the gyroscope signal to obtain the angular position (see equation 3.6 or equation 3.8). In comparison to the double integration of the accelerometer signal to obtain the linear position (see equation 3.7 or equation 3.9). So drift will be much less severe in the case of a single integration i.e. for a gyroscope signal. Therefore, methods of localization that employ the smartphone's gyroscope sensor may be more accurate than methods that use the smartphone's accelerometer sensor.

Based on these key findings, a case study was devised, involving the usage of a single smartphone embedded inertial sensor – gyroscope to localize a toy vehicle in an artificial test bed – Scalextric track. Furthermore, correcting the drift via 'feature detection'. This is detailed in the following section – Section 3.4. Scalextric is a track based slot car racing system invented by Fred (B F) Francis [242] in 1957.

3.4 Case study

Given the relatively robust error characteristics of the gyroscope sensor, a case study experiment was designed to further test the performance of the gyroscope sensor and also to prove the concept of gyroscope based localization. The gyroscope based localization approach will be discussed in Section 3.4.1. A test bed – Scalextric track was designed with a well-defined geometry (see Figure 3.4). Circular curved edges were of radii 0.33 m at angle $\pi/2$ radians. The straight edges were of length 0.17 m. A proof of concept experiment was carried out to localize a smartphone mounted on top of a toy vehicle (see Figure 3.4) driving around a Scalextric track. It allowed us to move a smartphone around in a control environment in a very repeatable way where ground truth is obtained relatively easily.

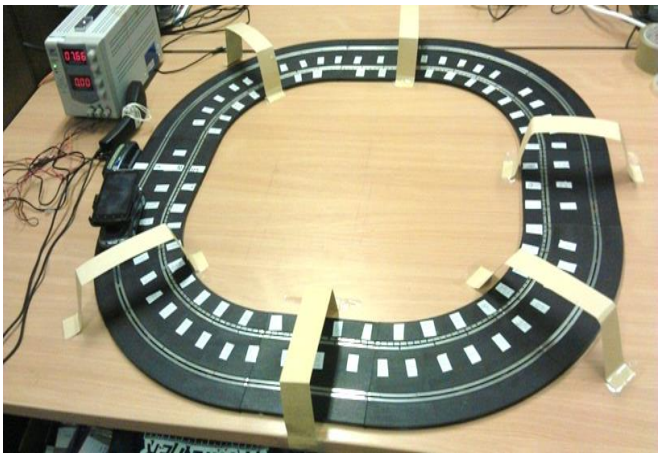


Figure 3.4: A slot car 'Scalextric' racing track powered by a DC voltage supply with smartphone mounted on top of the toy car kept at the top left. The track consist of several curved and straight edges with temporary bridges placed at some distance apart from each other.

The small footprint of the track combined with the high speeds of the vehicle was designed to create a significant challenge for a gyroscope based localization. To this end, a gyroscope based localization technique was designed specifically using the heading data obtained from gyroscope sensor. The heading data is then combined with a priori knowledge of the path's curvature to estimate the location of a toy vehicle in the map. The technique is discussed in the following sub-section 3.4.1. In sub-section 3.4.2 experimental setup and results are detailed. Following this, in sub-section 3.4.3 potential sources of error in the Scalextric model of localization are discussed.

3.4.1 Gyroscope Localization technique

The block diagram of system model is depicted in Figure 3.5. The system model assumes that the toy vehicle is always moving in forward direction only. Furthermore, it also utilises the fact that the Scalextric track is circular at curved edges and straight at the mid edges. The key component of this designed system is a gyroscope sensor embedded in smartphone and digital map of the track. The measurements from gyroscope are filtered and subsequently, mapped onto the digital map of the track using a data fusion algorithm to estimate the toy vehicle's position in the Scalextric track. Technical details of the algorithm and the working of components are discussed below.

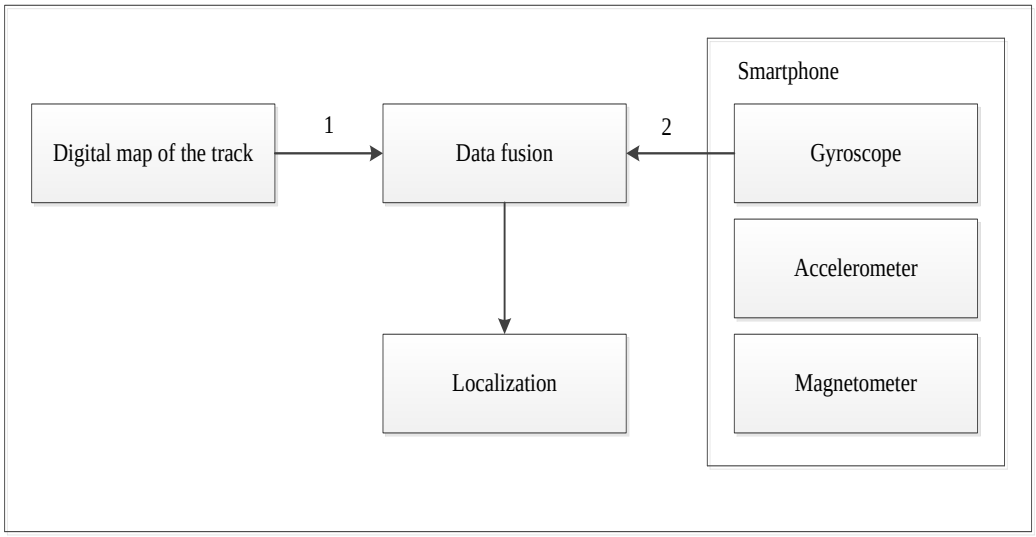


Figure 3.5: System overview showing the hardware and software components. Steps followed by the approach during localization. (1) The system obtains the information from digital map of the track. (2) The gyroscope provides the turn-rate of smartphone mount.

3.4.1.1 Gyroscope

As mentioned earlier (see Section 3.3), the gyroscope measures the turn rate of the device in three orthogonal directions. And integrating these measurements over time provides the total change in angle during the integration time. Since, we were only concerned with the heading of the toy

vehicle therefore we only integrated along the smartphone's z axis to derive the yaw (see Figure 3.2(B)); considering that the device's sensitive z axis is perpendicular to the ground.

3.4.1.2 Digital map of the track

A virtual map was constructed based on the geometry of the track. Continuous curved and straight edges were represented by smooth arc splines in the virtual map that are composed by joint circular arcs of radii 0.33 m and line segments of length 0.17 m as depicted in Figure 3.6. The map contains landmarks of low lateral dimension represented by blue triangles. These mark the position of bridges over the track that is used as reference points (for evaluation only, and not in the positioning algorithm). The straight edges are denoted by e_1, e_2, e_3 and e_4.

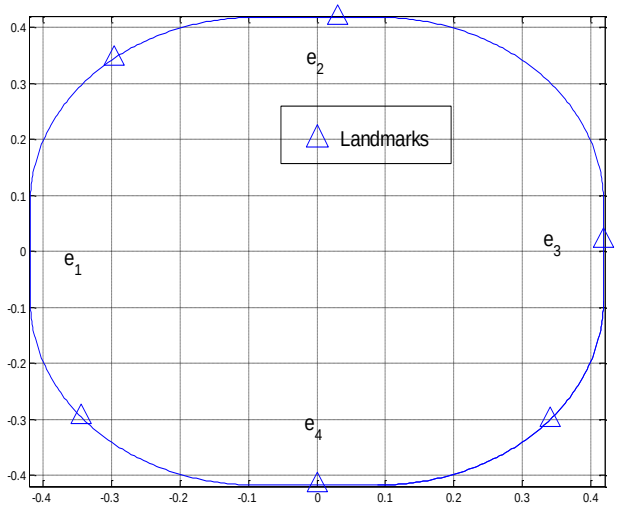


Figure 3.6: Virtual map of the track, depicting bridges with the triangle markings at some distance apart. The four straight edges are represented by e_1, e_2, e_3 and e_4.

3.4.1.3 Data fusion algorithm

The INS assumes a known initial position. This is also one of the primary reason for it being used along with other positioning systems such as GPS and Wi-Fi [243] rather than as independent positioning system. Similarly, in this case it is assumed that the initial conditions are known. The point at which the toy vehicle crosses the first bridge was considered as the starting position. The algorithm was designed based on the knowledge of the geometry of track and its main idea was to combine the direction of movement of the toy vehicle provided by the smartphone with the curvature of the track provided by the digital map. As mentioned by Mariani *et al.* [244] a turn is considered detected when the orientation around the vertical axis has changed by a sufficient value. Therefore based on this assumption a threshold limit of 0.025 rad was set to decide whether the toy vehicle was at the curved edges or at the straight edges as depicted by the following equations

$$\Delta\theta = \theta_t - \theta_{t-1} \tag{3.12}$$

$$\Delta\theta \begin{cases} > 0.025 \text{ rad, on curved edges} \\ \leq 0.025 \text{ rad, on straight edges} \end{cases} \tag{3.13}$$

where θ_t is the heading of toy car at time instant t and θ_{t-1} is the heading of toy car at time instant $t-1$. The threshold value was set by observing the difference in values of azimuth (θ) over 30 different runs of the toy car for voltages greater than 7.0 volts (see Figure 3.7 and Figure 3.8). Consider the toy car starts moving from the straight edge, its initial direction is saved as the direction, in which it last moved straight. All the following samples are compared to the previous samples in the straight direction. If one sample differs by more than 0.025 radians from the last sample in straight direction, the corner is detected and, to determine the end of a corner, for each new sample, the difference between the newest and the last sample is calculated. If two samples lie in between a previously specified boundary of constant size of 0.025 rad, the end of a corner is detected i.e. starting of a straight edge (see Figure 3.8).

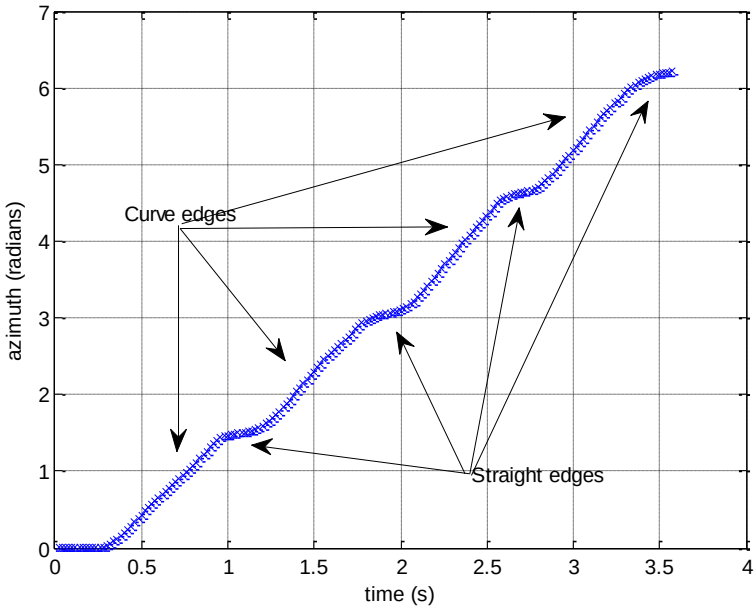


Figure 3.7: Example of a corner and curved edges detected in a single lap.

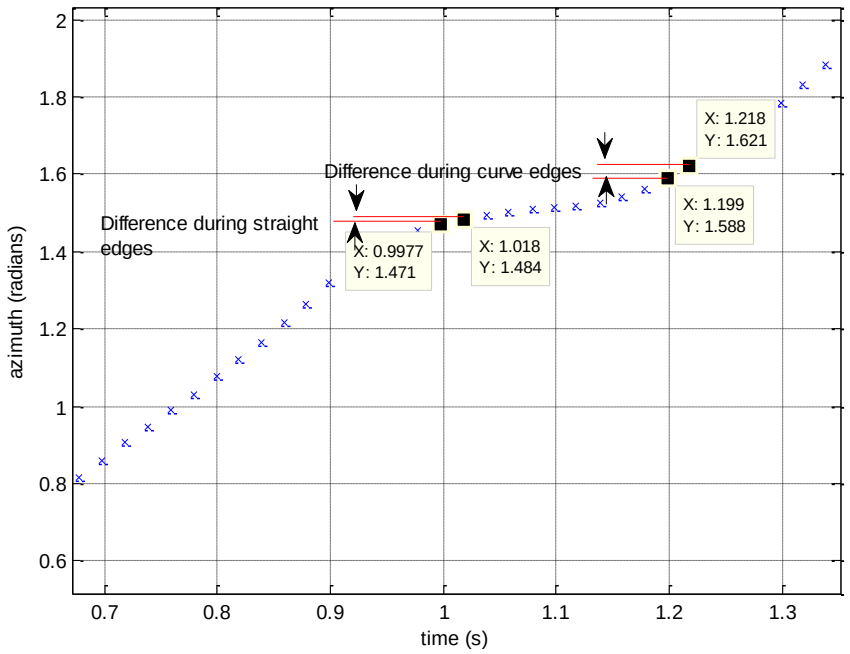


Figure 3.8: End of a turn example (sample data as in Figure 3.7).

The main steps of the algorithm are as follows:

Let the state vector **X** be defined as

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} \tag{3.14}$$

where x_1 and x_2 represent the position coordinates of the toy vehicle in the global coordinate system and, $\dot{x}_1$ and $\dot{x}_2$ represent the rate of the change of the position of the toy vehicle in the track. If the difference in the measured heading goes beyond the threshold limit, it would imply that the toy vehicle is at the curved edges, then

$$x_{1,t} = r\cos(-\theta_{t-1}) + \varepsilon_{1,c} \tag{3.15}$$

$$x_{2,t} = r\sin(-\theta_{t-1}) + \varepsilon_{2,c} \tag{3.16}$$

$$\dot{x}_{1,t} = \frac{x_{1,t-1} - x_{1,t-3}}{\Delta t} \tag{3.17}$$

$$\dot{x}_{2,t} = \frac{x_{2,t-1} - x_{2,t-3}}{\Delta t} \tag{3.18}$$

And if the difference in heading is below the threshold limit, it would imply that the toy vehicle is at straight edges, then

$$x_{1,t} = x_{1,t-1} + v_{1,s}\Delta t \tag{3.19}$$

$$x_{2,t} = x_{2,t-1} + v_{2,s}\Delta t \tag{3.20}$$

$$\dot{x}_{1,t} = v_{1,s} \tag{3.21}$$

$$\dot{x}_{2,t} = v_{2,s} \tag{3.22}$$

At transition, from the curved edges to straight edges

$$v_{1,s} = \begin{cases} \dot{x}_{1,t-1}, \text{if horizontal and on edge } e_2 \text{ or } e_4 \\ 0, \quad \text{if vertical and on edge } e_1 \text{or } e_3 \end{cases} \tag{3.23}$$

$$v_{2,s} = \begin{cases} 0, \quad \text{if horizontal and on edge } e_2 \text{ or } e_4 \\ \dot{x}_{2,t-1}, \text{if vertical and on edge } e_1 \text{ or } e_3 \end{cases} \tag{3.24}$$

Here $x_{1,t}$, $x_{2,t}$ and $x_{1,t-1}$, $x_{2,t-1}$ represent the position coordinates of the toy car in the track at time t and $t-1$ respectively. $\dot{x}_{1,t}$ and $\dot{x}_{2,t}$ represent the rate of change of the position of the toy car in the track at time t. θ_{t-1} is the heading of the toy car in the track provided by the gyroscope at time $(t-1)$ and r is the radius of the curvature of circular edges. $\varepsilon_{1,c}$ and $\varepsilon_{2,c}$ are the position coordinates of the centre of the circular curved edges. Δt is the time difference between the last two time steps i.e. equal to 0.04 seconds. $v_{1,s}$ and $v_{2,s}$ are the velocities of the toy car at the straight edges.

The three dimensional angular velocity vector φ_t provided by the gyroscope in the smartphone's reference frame is used to calculate an axis angle vector ψ_t which describes the orientation of smartphone in global frame of reference (see Figure 3.2(A)) at time t. Let

$$q_t = \begin{bmatrix} \cos\left(\frac{|\varphi_t|\,\Delta t}{2}\right) \\ \frac{\varphi_t}{|\varphi_t|}\sin\left(\frac{|\varphi_t|\,\Delta t}{2}\right) \end{bmatrix} \tag{3.25}$$

denotes the quaternion [245] formed from the angular velocity vector φ_t at time t and Δt be the sampling rate of the gyroscope. Then the current quaternion q_t based on the last time step is calculated by equations 3.26 and 3.27 [134].

$$q_t = 0.5 * q_{t-1} * p_{t-1} * \Delta t \; + q_{t-1} \tag{3.26}$$

$$p_{t-1} = [0 \; \varphi_{t-1}]^T \tag{3.27}$$

where $*$ represents the quaternion multiplication and p_{t-1} is the transpose vector of the quaternion q_t .So, the estimate of the axis angle vector is then obtained by the transformation of quaternion as given by equation 3.28

$$\psi_t = f_{q2e}\,(q_t) \tag{3.28}$$

where the function f_{q2e} denotes the transformation of the orientation stored as a quaternion to the axis angle vector via the direction cosine matrix. The resulting value of the axis angle vector ψ_t^Z, corresponding to global vertical axis is then taken as the estimate of the measured heading θ_t for the course of the toy vehicle i.e.

$$\theta_t = \psi_t^Z \tag{3.29}$$

3.4.2 Experimental setup & results

As depicted above, Figure 3.4 shows the experimental setup – a slot car racing track: 'Scalextric' consisting of several curved and straight edges. A smartphone (HTC Sensation Z710e) was mounted on top of a toy car and the toy car was driven repeatedly on the Scalextric track. The speed of toy car was controlled by power supply – BST PSD 30/3B [246] having an accuracy of ± 1% and noise root mean square (rms) error of 0.3 milli-volt. A current was applied to the track which powers the car's motor. The voltage was regulated, and this pushed the toy car in forward direction. The experiment was performed at three different voltage conditions – 7.2 volts, 7.65 volts and 8.1 volts, and the smartphone's gyroscope sensor logged the data at 50Hz.

The toy car was driven around the Scalextric track and the position estimated by the model (described in Section 3.4.1) is compared to the actual position (ground truth) of the known landmarks (bridges, illustrated in Figure 3.4). Specifically, these bridges were placed at three straight edges and three curved edges. The starting point of toy car was taken as the point where the toy car crosses the first bridge. To estimate the toy car's position, changes in the photo-sensor embedded in the smartphone were monitored. An event is generated by the photo-sensor when the toy car passes under the bridge. These events are time synchronised with the events generated by the gyroscope. The precise positions of the bridges were measured off-line. The position error p_e of the sample points in toy car's trajectory is then estimated using the bridges as reference points using equation 3.30.

$$p_e = \text{Actual position} - \text{Estimated position} \tag{3.30}$$

Figure 3.9 displays the result of eight laps for the three test cases, 7.2 volts, 7.65 volts and 8.1 volts. A cursory examination on the results show that the positioning error histogram curves across different voltages is symmetrical with negative bias.

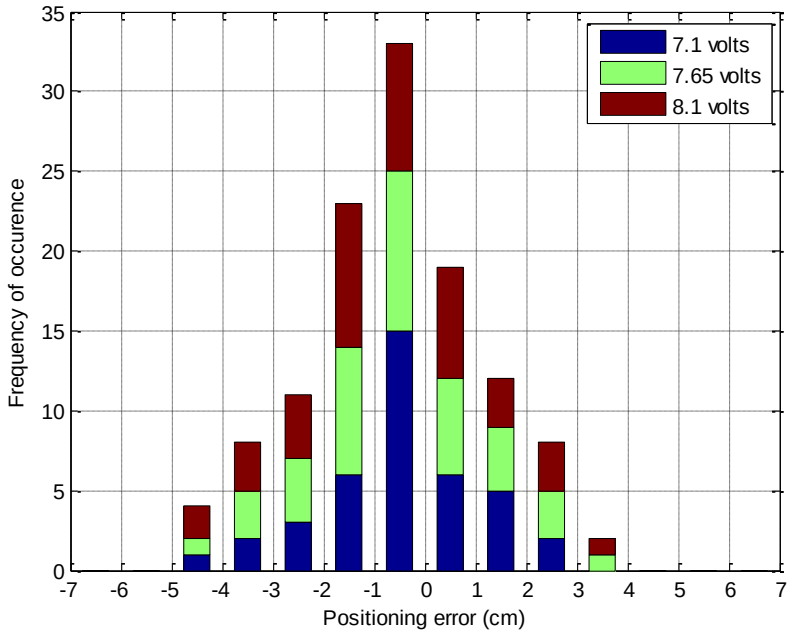


Figure 3.9: Position error histograms, corresponding to three test cases – 7.2 volts, 7.65 volts and 8.1 volts during eight laps of the track.

The results also indicate that the position error is independent of the toy vehicle's speed. This is as expected. Since, there is no correlation between the position error and the supplied voltage. The same characteristic can also be observed from Figure 3.10 and Figure 3.11. Specifically, from Figure 3.10 by looking at the times the toy car takes to complete 8 laps. It indicates the speed of toy car to be growing with voltage. However, position error appears to be random throughout. Figure 3.11 depicts the root mean square error as a function of number of laps, which again indicates the same i.e. the position error is independent of the supplied voltage (or speed). Moreover, it can also be observed that the position error does not grow with time or instead, with the number of laps. This behaviour is significant. As traditional INS positioning approach suffers from drift [133, 134], the method outlined here does not. This can be attributed because of the fact, that the algorithm detects the presence of 'features (corners)' on the known track and calibrates itself according to the equations 3.15 and 3.16. In this case, the corner is a curved edge. This 'feature (corner) detection' and 'map method approach' would be central to the methods of pedestrian localization discussed in later chapters (Chapter 6 and Chapter 7).

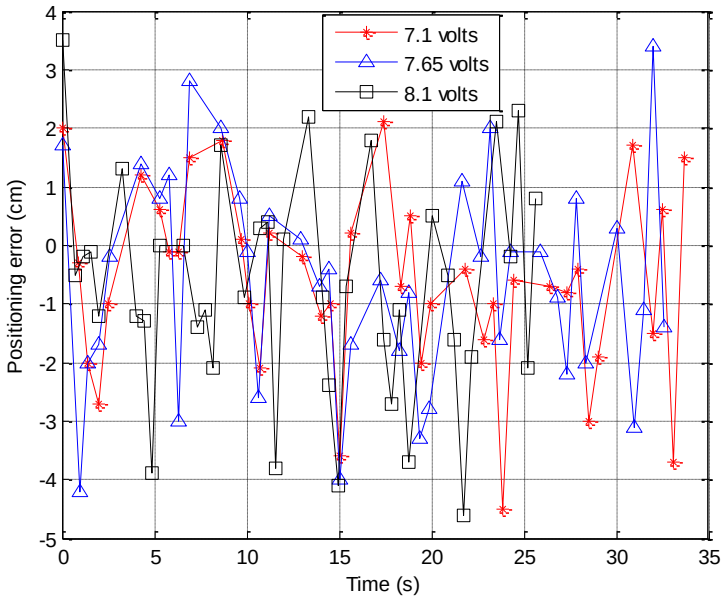


Figure 3.10: Positioning error, corresponding to three test cases - 7.2 volts, 7.65 volts and 8.1 volts with increasing time.

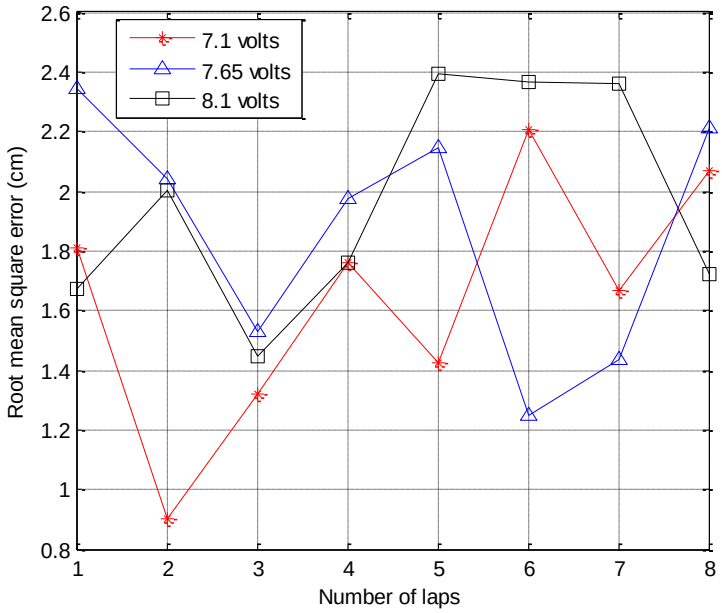
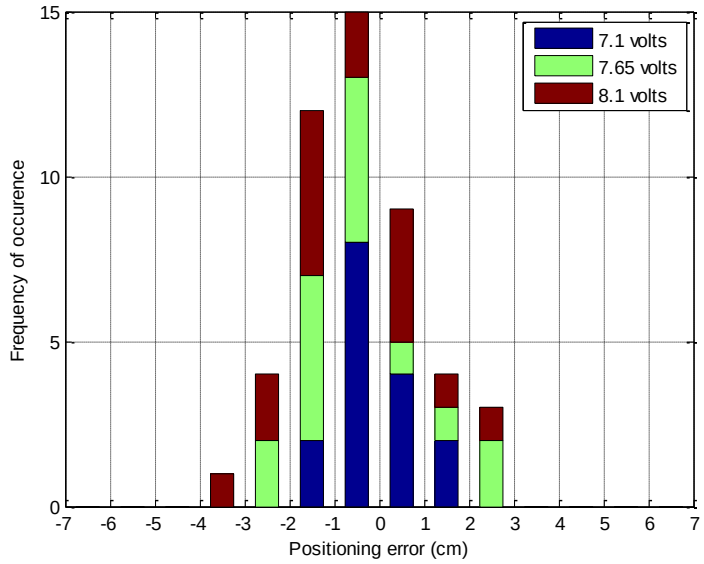


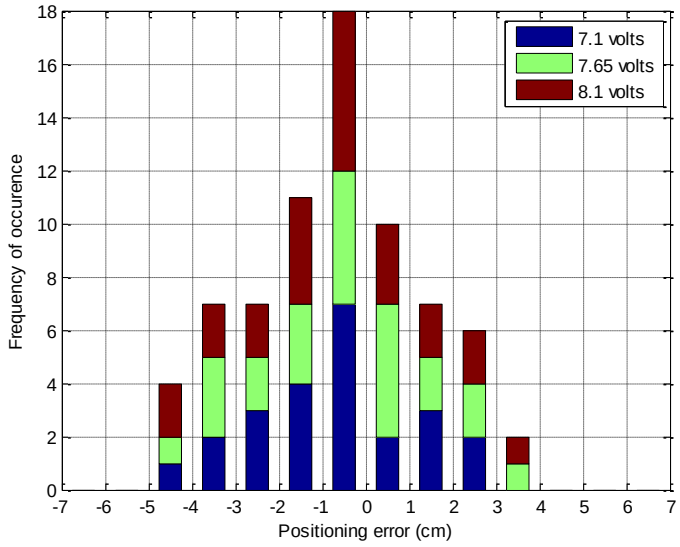
Figure 3.11: Root mean square error, corresponding to three test cases - 7.2 volts, 7.65 volts and 8.1 volts.

The performance of algorithm was also investigated separately in the curved edges and straight edges. It was our belief that the performance shall be better in the curved edges. The collected data was classified separately into the curved edges and straight edges. The positioning error was then evaluated separately at the curved edges and straight edges with the same process

as described above. As expected, it was observed that the performance of algorithm was better in the curved edges in comparison to the straight edges in all the three test cases (see Figure 3.12). Again, the behaviour could be explained with the same reasoning as mentioned before, i.e. the algorithm self-calibrates itself on detection of the 'corner feature – a curved edge'. Since all the measurements from gyroscope sensor were made on these curved edges. As a result of which the presence of drift was less severe in these curved edges in comparison to the straight edges.



(A)



(B)

Figure 3.12: Position error histograms in the (A) curved edges (B) straight edges corresponding to three test cases- 7.2 volts, 7.65 volts and 8.1 volts.

Table 3.7 summarises the precision (95% confidence level), and mean absolute error (MAE) for three test cases. The MAE is calculated according to the following equation

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|p_{e_i}\right| \tag{3.31}$$

where N is the number of data points i.e. equal to 40. The error appears to be slightly biased towards the negative side. The 95% confidence level lies in the region -4.5 to 3.1 cm with *MAE* less than 2 cm. This could be due to the presence of systematic errors (see Section 3.3) that were not completely filtered out. Also, there were errors in establishment of the proper ground truth. This would be detailed in the next section – Sources of error.

Volts	**95% confidence level (cm)**	MAE **(cm)**
7.2	-3.81 to 2.55	1.57
7.65	-4.27 to 3.09	1.60
8.1	-4.5 to 3.06	1.67

Table 3.7: Summary of the position error for three test cases – 7.2 volts, 7.65 volts and 8.1 volts.

3.4.3 Sources of error

There were several sources of potential error in this experiment that can be enumerated as follows:

i. Bridge detection: The lateral dimension of the bridges is 4 cm. This would cause the true position to deviate by ±2 cm of the ground truth target (centre of the bridge). Theoretically, the length of the bridge for proper detection by means of photo-sensor is less than 2.4 cm (calculations shown below), but it was observed by several tests that this length was insufficient leading to missed detection. Therefore, a wider bridge of 4 cm was used to ensure detection.

Calculations

Polling rate of the photo-sensor = 50Hz (3.32)

Length of the track = 280 cm (3.33)

Number of laps = 5 (3.34)

Total Distance travelled = 280x5 = 1400 cm (3.35)

7.2 Volts

Time elapsed in 5 laps = 14.6 seconds (3.36)

$$\text{Speed of the toy car} = \frac{\text{Distance travelled}}{\text{Time taken}} = 0.96\ \text{m/s} \tag{3.37}$$

Distance travelled in 20 millisecond = 0.96 x 100 x $\frac{1}{50}$ (3.38)

= 1.92 cm

7.65 Volts

Time elapsed in 5 laps = 13.2 seconds (3.39)

$$\text{Speed of the toy car} = \frac{\text{Distance travelled}}{\text{Time taken}} = 1.06\ \text{m/s} \quad (3.40)$$

Distance travelled in 20 millisecond = 1.06 x 100 x $\frac{1}{50}$ (3.41)

= 2.12 cm

8.1 Volts

Time elapsed in 5 laps = 12 seconds (3.42)

$$\text{Speed of the toy car} = \frac{\text{Distance travelled}}{\text{Time taken}} = 1.16\ \text{m/s} \quad (3.43)$$

Distance travelled in 20 millisecond = 1.16 x 100 x $\frac{1}{50}$ (3.44)

= 2.3 cm

ii. Mismatch in the timings between the events generated by the photo-sensor and gyroscope: Though the system clocks of the gyroscope and photo-sensor are time synchronised, it is however noticed that the timestamps associated with the gyroscope lags behind the occurrence of a photo sensor event by a delay of less than 10 ms. Since, the event listener responsible for handling the events generated by two sensors in the Android systems have a marginal time gap [247]. Furthermore, because of the limited resources on the device, and the existence of other processes such as garbage collection reduces the sampling rate of these sensors [232]. This causes the true position to deviate by ± 0.5 to ± 0.6 cm (calculations shown below) of the ground truth target.

Calculations

7.2 Volts

From (3.31), Speed of the toy car =0.96 m/s (3.45)

Distance travelled in 10 millisecond = 0.96 x 100 x 0.01 (3.46)

= 0.96 cm

7.65 Volts

From (3.31), Speed of the toy car =1.0 m/s (3.47)

Distance travelled in 10 millisecond = 1.0 x 100 x 0.01 (3.48)

= 1.0 cm

8.1 Volts

From (3.31), Speed of the toy car =1.16 m/s (3.49)

Distance travelled in 10 millisecond = 1.16 x 100 x 0.01 (3.50)

= 1.16 cm

i. Sensor noise and drift error: The systematic error largely determines the mean error while the random noise error has zero mean and mainly impacts the error variance. For instance a bias error of -0.0533 x 10^{-5} rad/s (see Table 3.3) in a gyroscope along z axis would cause the net decrease in the true orientation to decay by 3.2 x 10^{-5} radians in 1 minute along z axis. A random noise of 4.264 x 10^{-24} rad/$\sqrt{s}$ would cause the standard deviation of orientation error after 1 hour to be 4.264 x 10^{-24} . $\sqrt{60}$ =33.02 x 10^{-24} radians along z axis.

3.5 Summary

This chapter provided a first essential step towards our aim. The principal behind this work was to show that the 'low cost smartphone embedded INS' system can be effectively used for positioning. However, there is always an associated accuracy and cost trade off. So, we devised tests to investigate the performance characteristics of smartphone embedded INS. The key findings of tests are as follows:

- Bias instability and random walk can have a significant impact on the INS performance in measuring position and orientation. This instability and random walk is manifested as a 'drift' in the overall positioning process.
- Bias instability and random walk are significantly less for the gyroscope sensor in comparison to the accelerometer sensor.

Based on these key findings, a case study was devised to investigate the usage of 'smartphone embedded single inertial sensor – gyroscope' to position a moving object (toy vehicle) on an artificial test bed – 'Scalextric track' with bridges placed at some distances apart. The bridges were used for evaluation only and not included in the positioning algorithm. Considering the fact that map geometry was well defined and known a-priori, the drift was corrected via 'feature (corner) detection'. This kind of case study has never been reported before. The outcomes of this case study are as follows:

- Position error is independent of the supplied voltage. Or instead, it is independent of the speed at which the toy vehicle moves around the whole Scalextric track.
- Position error does not grow with time, unlike other traditional inertial positioning techniques [134]. This behaviour can be attributed because of the fact that the algorithm self-calibrates on detecting the presence of 'corner features' on the track.

This 'feature (corner) detection' and 'map method approach' would be central to the methods of localization discussed in later chapters (Chapter 6 and Chapter 7). This case study also showed that, it is possible to localize an individual (in this case study: a toy vehicle) without carrying any form of dedicated hardware structures like camera, GPS receivers, etc. Furthermore, this case study showed that it is possible, in principle to localize an individual in an infrastructure free environment. Specifically, an infrastructure free environment is an environment that is devoid of

any specialised infrastructure deployment. This specialised infrastructure deployment can be in the form of any external hardware installations that is attached to the scene for the purpose of localization. For instance, Wi-Fi beacons, RFIDs, QR codes, etc.

The case study was performed in a control environment – Scalextric track. While in real scenario, the building floor plan is not as detailed as an artificial test bed. There are many long straight walks, U turnings, ramps, etc. in the building. Also, the movement of an individual (whether he/she is a pedestrian or it is a motor vehicle) is unlike the movement of a toy vehicle. A shock can occur at each pedestrial foot step or at every switching 'ON and OFF' of the motor vehicle engine. This can provoke errors in the smartphone embedded INS [248, 249]. The following chapter begins to address these concerns by introducing the concept of inertial pedestrian dead-reckoning (PDR), and subsequently details the tests for localizing an individual – pedestrian in the real indoor environment.

4. Chapter 4: Inertial PDR

The previous chapter was a proof of concept study carried on an artificial test bed – Scalextric track. In principal, it investigated whether an individual (toy vehicle) can be localized without any support of dedicated hardware or specialised infrastructure deployment using smartphone embedded inertial navigation system (INS) in a control environment. The results were encouraging – 95% confidence level lies in the region -4.5 to 3.1 cm with *MAE* less than 2 cm. It implied that bounded positioning error was possible using a single inertial sensor, provided the geometry of track is well defined. Therefore based on the findings of previous chapter, this chapter provides a reference implementation of the smartphone embedded INS to position a 'pedestrian' in a 'real indoor environment' provided there is no support of any kind of infrastructure or dedicated hardware.

Parts of this chapter are based on the paper entitled "Low cost infrastructure free indoor localization", which was published in proceedings of the 5th IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN) held in Busan, South Korea [42] (copied in Appendix A).The focus of this chapter is to analyse the inertial sensor measurements recorded from smartphone sensors to detect the occurrence of footsteps as well as to indirectly estimate step lengths. It also compares various off the shelf foot step detection techniques and stride length estimation models.

From the literature review (see Chapter 2), it was recognised that inertial positioning (see Chapter 2, Section 2.2.4) has several advantages. Out of which there are three key advantages, specifically for the pedestrian:

1) Portability: The inertial sensors are light weight and low power meaning they can be easily carried by a pedestrian for continuous positioning.
2) Suitability: Importantly, the inertial sensors are suitable for identifying and classifying different kinds of human motion. For e.g. jumping, crawling, running, sprinting, descending and ascending a ladder or stairs, etc.
3) Privacy: The inertial sensors offer a degree of location privacy to the pedestrian. It is totally dependent on the pedestrian whether he/she wants to share his/her location data to a third party or not.

This autonomy and versatility offers great potential for locating pedestrians in an infrastructure free indoor environment. In a PDR system, the accelerometer signal is usually applied for calculating linear displacement and the gyroscope or a compass or combination of both is used to calculate pedestrian's heading [145]. PDR are usually applied as frequently calibrated PDR or step based PDR (see Chapter 2, Section 2.2.4). Out of which frequently

calibrated PDR is often applied, when IMUs are placed on foot otherwise step based PDR are preferred choice. Thus, in case of smartphone based PDR system we apply step based PDR.

The PDR technique has been successfully applied previously to design many positioning systems and applications over the last two decades. These are summarized in paper [139]. In contrast to all the PDR positioning systems discussed, the smartphone based PDR positioning system is different. Since, smartphone orientation is non-static and can vary with time; also the smartphone inertial sensors are of relatively low quality [40, 41] in comparison to the other commercially available IMUs (see Table 4.1). Therefore, off the shelf stride estimation methods (foot step detection techniques and stride length estimation models) would not work for this case. So, the following experimental tests (detailed in the Section 4.2) are devised to investigate the most suitable stride estimation method for smartphone based PDR positioning system. Also, traditional heading estimation technique (detailed in Section 4.1.3) is applied on the smartphone PDR positioning system for estimating the course of pedestrian during the whole walk.

Sensor Manufacture	**Invensense**	**Xsens**	**Honeywell**
Model	MPU-9150	MTi – G	HG1700
Type	Smartphone IMU	Factory calibrated IMU	Factory calibrated IMU
Cost	100 USD	2500 USD	9,000 USD
		Accelerometer errors	
Bias	$\pm$80(x&y) milli-g 150(z) milli-g	0.02 m/s^2	1-2 milli-g
Noise	400 micro-g/$\sqrt{Hz}$	0.002 to 0.004 m/s^2/$\sqrt{Hz}$	-
Scale factor error	$\pm$3%	$\pm$0.03%	300 ppm
		Gyroscope errors	
Bias	$\pm$20 ^{0}/s	$\pm$1 ^{0}/s	1 – 10 ^{0}/hr
Noise	0.005 ^{0}/s/$\sqrt{Hz}$	0.05 to 0.1 ^{0}/s/ $\sqrt{Hz}$	-
Scale factor error	$\pm$3%	Not calibrated	150 ppm

Table 4.1: Comparison of an IMU usually embedded in a smartphone and a factory calibrated IMU [136, 250-253].

To begin with, the chapter first details the designed smartphone based PDR system in Section 4.1. Section 4.2 presents the experimental setup and results obtained from the smartphone based PDR system. Finally, Section 4.3 summarizes the key findings of this chapter.

4.1 Smartphone based PDR system

The algorithmic flow of smartphone based PDR positioning system is depicted in Figure 4.2. Initially, the systematic and random noise error (detailed in Section 3.3) is removed out from the sensor (accelerometer and gyroscope) data. The sensor (including magnetometer sensor) data is then filtered out using a low pass filter (or simply, a moving average) according to equation 4.1 [254], to remove the high frequency noise present in the signal waveform.

$$w(t) = \alpha w(t-1) + (1-\alpha)v(t) \quad (4.1)$$

Here $w(t)$ is the filtered signal at time t, $v(t)$ is the raw signal at time t and α is a constant equal to 0.7. The parameter α was set to 0.7, such that frequencies above 4 Hz (i.e. maximum strolling pace, including light running) [255] were suppressed. Figure 4.1 demonstrates that "pseudo" peaks are almost removed, resulting in a much smoother signal.

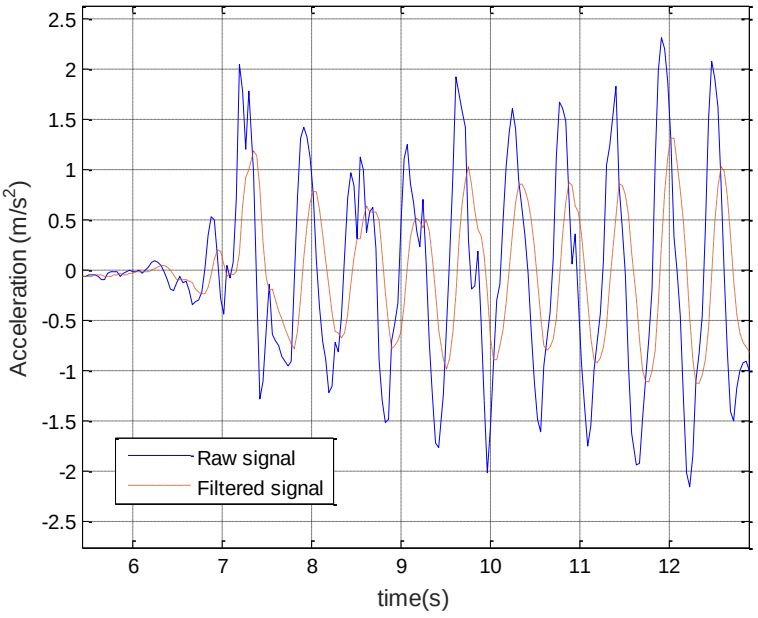


Figure 4.1: Raw acceleration signal (solid) versus filtered acceleration signal (dashed).

The filtered accelerometer signal is then forwarded to the footstep detection module. The footstep detection module detects the pedestrian's footsteps and also confirms whether the pedestrian is really walking. If the pedestrian's walking state is confirmed, the stride length is estimated at each detected footstep while the heading is determined using magnetometer and

gyroscope data. The detected footsteps, headings and stride lengths are substituted in equations (2.4) and (2.5) to estimate the relative position of pedestrian.

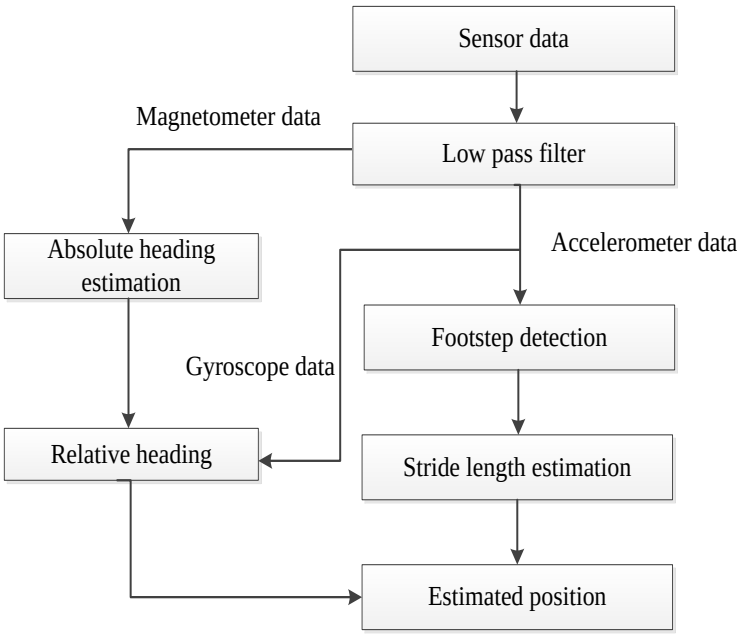


Figure 4.2: Block diagram of the smartphone based PDR positioning system.

4.1.1 Footstep detection

This section investigates footstep detection techniques suitable for smartphone embedded sensors. Also, it would present a method to classify whether the pedestrian was really walking or not. The pedestrian state is recognized whether he/she is walking or at rest (static). Based on the state recognized (refer Section 4.1.1.4 for details); if they are walking, the footsteps are detected. Accelerometer sensors have been commonly used for detecting footsteps and to classify different types of pedestrial motions – walking, running, jumping, etc. [256, 257]. The measured acceleration values represent a combination of the applied acceleration on the smartphone due to motion and the force of Earth's gravity [3]. Considering the fact that a smartphone can be at any position on the pedestrian's body, with time-varying orientation change, the magnitude of 3-axis accelerometer readings ẍ was used, instead of its x, y and z components $(\ddot{x}_x, \ddot{x}_y, \ddot{x}_z)$ as shown below

$$\ddot{x} = \sqrt{\ddot{x}_x^2 + \ddot{x}_y^2 + (\ddot{x}_z - g)^2} \tag{4.2}$$

where the component g is Earth's gravity (equal to $9.8\,m/s^2$) . Since, the Earth's gravity can have sudden changes in the measured acceleration values when the pedestrian switches their state of motion (static to walking state or to any other state and vice versa) therefore the Earth's gravity was subtracted out. There are different types of foot step detection techniques that can be

used to analyse the acceleration signal for confirmation of the occurrence of footsteps and estimation of the step frequency [139, 258, 259]. Here, three of state-of-the-art foot step detection techniques for smartphone based PDR positioning system are investigated:

- Peak detection [145, 260]
- Zero-crossing [261]
- Fast Fourier Transform (FFT) [257, 262]

4.1.1.1 Peak detection

The peak detection algorithm is based on filtering the magnitude of acceleration signal followed by applying a threshold on the acceleration signal [145, 263]. Its current implementation depends on high accuracy foot mounted accelerometers [260], which differ significantly from phone-embedded sensors. We quantify its performance when applied to smartphone embedded sensors in Section 4.2. The algorithm counts a valid footstep when local maximum peak (maxima) and local minimum peak (minima) are detected in sequence. The value of local maxima should be higher than that of the most recent valid local minima by at-least a threshold value Δ_o (equal to$+0.5\ \mathrm{m/s^2}$) as shown in Figure 4.3. Also, the value of the valid local minima should be lower than that of the most recent valid local maxima by a threshold value Δ_t (equal to $-0.5\ \mathrm{m/s^2}$). The detection thresholds are selected by analysing acceleration magnitudes of over 500 real step data points from 6 subjects while keeping in note that false peaks caused by acceleration jitters (that are too small in magnitude) are filtered. Table 4.2 shows the results.

Steps taken	Steps detected	False positive
85	84	2
83	80	4
88	89	2
86	86	1
84	84	1
86	85	2

Table 4.2: Performance of step counting algorithm.

Error of 1 to 2 steps is observed due to the highly unstable pattern obtained towards the end of walk. The red stars represent the footsteps detected in Figure 4.3.

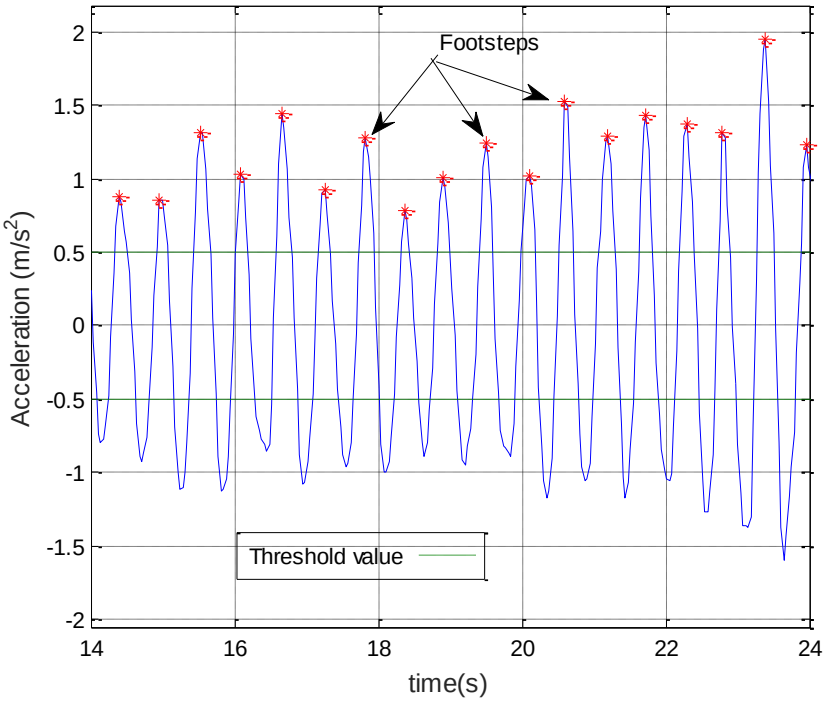


Figure 4.3: Enlarged view of the foot step detected via peak detection algorithm for (14-24) s.

4.1.1.2 Zero crossing

Another method for foot step detection based on the acceleration values is zero crossing method discussed in [261]. This method first computes the magnitude of the acceleration signal. The step boundaries are defined by either positive or negative going zero crossing of a filtered version of the acceleration signal as shown in Figure 4.4. Black dot points the occurrence of a valid footstep in Figure 4.4. One condition that needs to be fulfilled is that the number of samples between two zero crossings should be within certain thresholds. If they are greater than maximum threshold or less than minimum threshold, foot step is not counted. The maximum threshold was determined to be approximately 700 ms and minimum threshold was determined to be 300 ms approximately. This method has been effectively applied to the foot-mounted sensors and wearable sensors [151, 160]. We quantify its performance when applied to smartphone embedded sensors in Section 4.2. The choice of the 700 ms time difference threshold is due to the fact that, at normal walking speeds human approximately takes 1 – 2 footsteps per second whereas the choice of 300 ms time difference is due to the fact that at higher walking speeds human can approximately takes 3 – 4 footsteps per second [255].

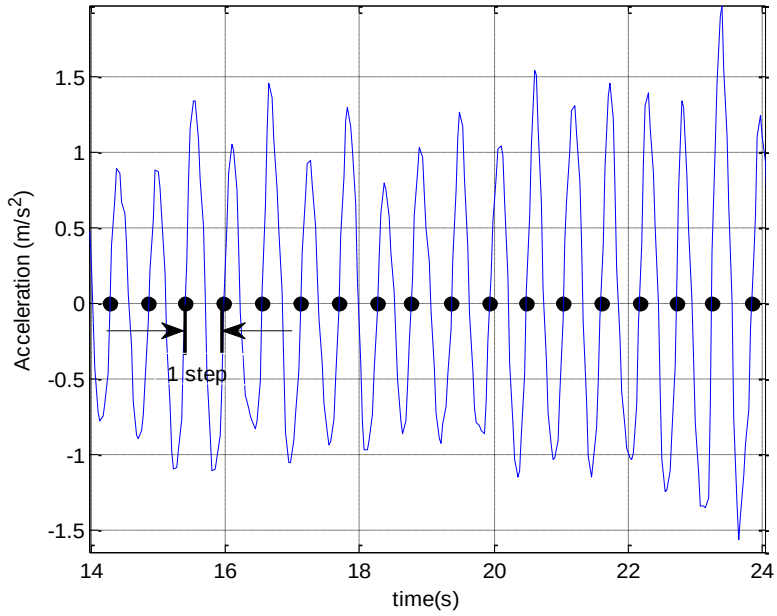


Figure 4.4: Enlarged view of the foot step detected via zero crossing algorithm for (14 -24) s.

4.1.1.3 Fast Fourier transform

Fourier transform is a mathematical transformation employed to transform signals between time and frequency domain [264]. So, it can be used for analysing the frequency component of the recorded accelerometer samples during pedestrial walk. Various researchers [257, 259, 262] have potentially applied the Fourier transform, specifically Fast Fourier Transform (FFT) to estimate the step frequency. We quantify its performance when applied to smartphone embedded sensors in Section 4.2. FFT is executed on the filtered accelerometer signal $\ddot{x}(t)$ as follows [265]:

$$\varphi(k) = \sum_{t=0}^{N-1} \ddot{x}(t) e^{\frac{-j2\pi kt}{N}}; k = 0,1 \ldots . N-1 \quad (4.3)$$

where $\varphi(k)$ is the frequency response of accelerometer signal at the kth spectral point, $\ddot{x}(t)$ represents the tth time sample of the accelerometer signal, $\sqrt{-1} = j$ and N is the window length over which FFT is executed. The window length N is a critical parameter. It determines the resolution in time and frequency domain [259, 266]. It should neither be too long nor too small. If window length is too long, the resolution in time domain is too small. Therefore, the local change in the periodicity of the signal would be missed. On the contrary, if the window length is too short the resolution in frequency domain is too high therefore close walking frequencies cannot be separated [267, 268].

The step rate for a normal walk varies between 0.7 Hz to 1.51 Hz [269], so to detect no less than 1 foot step in a second, we selected a window size of 1 second for this investigation.

However, in the later chapters (see Chapter 5 and Chapter 6) a window of bigger size is investigated and used.

As mentioned earlier, the pedestrian motion is a repeating process of placing the footsteps (walking) and non-placing of footsteps (rest). So, to know whether the pedestrian has really placed his/her footsteps activity classification was performed. It is detailed next in the following section.

4.1.1.4 Activity classification

The pedestrian's walking and rest states were recognized from the filtered accelerometer signal by observing several characteristics in the frequency domain like FFT amplitude and FFT energy via sliding window of 20 samples, with no overlap. Each sliding window covers a time interval of 1 s. The window of 1 s is used to sufficiently capture cycles of the pedestrian activities – walking and static [270]. Out of various frequency domain quantities – FFT amplitude was the most promising to identify between the start and stop of walking as shown in Figure 4.5. The figure depicts a pedestrian to be at rest for the first 2 s (see Figure 4.5(A) and Figure 4.5(B)) and walking for the remaining 2 s (see Figure 4.5(C) and Figure 4.5(D)).

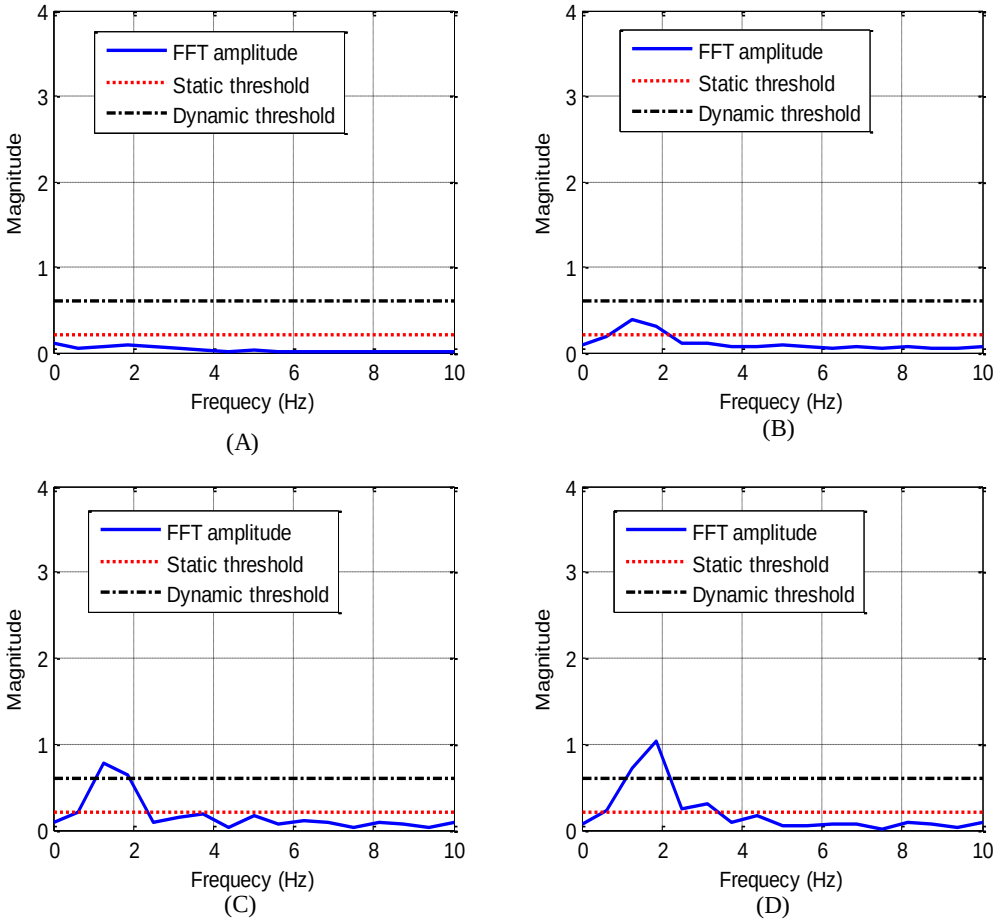


Figure 4.5: Variation of FFT amplitudes over different time periods: (A) (0-1) s (B) (1-2) s (C) (2-3) s and (D) (3-4) s.

Two FFT amplitude thresholds – T1 (equal to 0.6 m/s^2) and T2 (equal to 0.2 m/s^2) were set to identify the rest state and walking state respectively (as illustrated in Figure 4.6). These thresholds were set by analysing the waveform of over 800 steps from ten pedestrians. Table 4.3 illustrate the statistics of the two cases when the pedestrian was walking and secondly at rest. It was assumed that the pedestrian initially starts from rest. If FFT amplitude was greater than threshold T1, dynamic state was identified, indicating walking has started. On the other hand, if FFT amplitude was less than threshold T2, rest state was identified, indicating pedestrian has stopped walking. However, if the amplitude was in between static and dynamic thresholds, previous state is retained that is either walking or rest state. For e.g. in Figure 4.5(B) the pedestrian is considered to be at rest (the previous state from Figure 4.5(A)), although FFT amplitude threshold has crossed the threshold T2.

States	**Magnitude**		
	Absolute mean	Standard deviation	Minimum
Rest state	0.24	0.07	0.04
Walking	1.3	0.67	0.59

Table 4.3: Magnitudes of the two states.

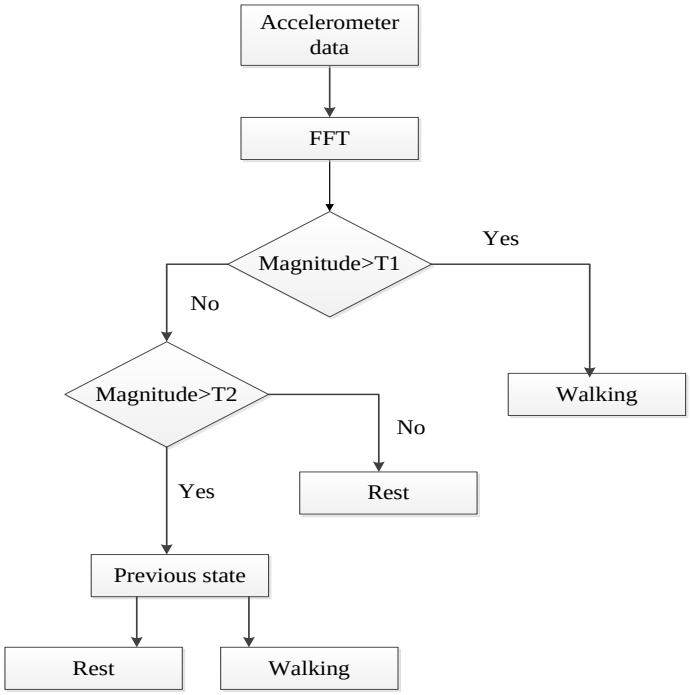


Figure 4.6: Activity classification: Walking or Static.

4.1.2 Stride length estimation

Once the footsteps were detected, the step size was still needed in order to compute the relative position of pedestrian. One way to estimate the step size was to assume that all steps have equal lengths as proposed by Groves [3]. This assumption can be true for some cases, but not always because the step size is not a constant value but related to walking speed and acceleration magnitude [271]. In a typical human walking behaviour, it has been observed that as step frequency increases, the peak acceleration difference increases and the time period between footsteps decreases while stride becomes larger [258].There are different models for step length estimation, but in most of them the sensor unit was attached to foot [272, 273]. However, the smartphones are mostly carried in hands or kept in pocket during walking [274]. As a result, we investigate three of the most common state-of-the-art stride length (SL) estimation models over pedestrians while they carry the smartphone in hand (see Figure 4.7). These models are as follows

- Weinberg model [275]
- Scarlett model [276]
- Kim model [271]

We quantify their performance, when applied to smartphone embedded sensors in Section 4.2. These models are detailed below:

Figure 4.7: Pedestrian walking with a smartphone in a straight aisle.

4.1.2.1 Weinberg model

The Weinberg model [275] is based on the principle that a vertical bounce in an individual's foot step is directly correlated to that person's stride length. This bounce is calculated from the

4

difference of the peaks at each foot step. SL is calculated using the filtered accelerometer signal by following equation.

$$SL = k \sqrt[4]{\ddot{x}_{max} - \ddot{x}_{min}} \quad (4.4)$$

where k is a constant. The peak acceleration values $\ddot{x}_{max}$ and $\ddot{x}_{min}$ were calculated separately for each footstep.

4.1.2.2 Scarlett model

The Scarlett model [276] calculated the stride length by deriving a correlation between the value of maximum, minimum, and average acceleration of a footstep, according to equation 4.5. Since, the motion of pedestrian is periodic and also it can be represented by the motion of a spring. Therefore, this model tried to solve the accuracy problem caused by the variation of spring in the steps of different people, or in the steps of one person using different paces from one measurement to another.

$$SL = k \frac{\frac{\sum_{i=1}^{N} |\ddot{x}_i|}{N} - \ddot{x}_{min}}{\ddot{x}_{max} - \ddot{x}_{min}} \quad (4.5)$$

where k is a constant and N is the window size.

4.1.2.3 Kim model

The Kim model [271] calculated the stride length by placing fixed markers at known locations (60 cm and 80 cm).The acceleration values were measured when the pedestrian walked through these marked locations. An experimental equation was derived for a step as shown by (4.6).

$$SL = k_3 \sqrt{\frac{\sum_{i=1}^{N} |\ddot{x}_i|}{N}} \quad (4.6)$$

where k is a constant and N is the window size. To estimate the k values for the different stride length models, a known distance of 10 m is walked by 10 volunteer subjects and the k is adjusted for each walk such that the estimated distance becomes the same as the real. The individual k values of the 10 walks by 10 volunteer subjects are then averaged to derive the actual value of constant k. Table 4.4 shows the results.

Weinberg		Scarlett		Kim		
$d_{estimated}$ (m)	k	$d_{estimated}$ (m)	k	$d_{estimated}$ (m)	k	d_{real}(m)
5.2	0.52	5.5	0.55	7.7	0.77	10
5.6	0.56	5.9	0.59	7.6	0.76	10
5.8	0.58	5.2	0.52	8.0	0.80	10
5.5	0.55	5.9	0.59	7.5	0.75	10
5.4	0.54	5.6	0.56	7.2	0.72	10
5.4	0.54	5.6	0.56	7.4	0.74	10
5.5	0.55	5.7	0.57	7.6	0.76	10
5.6	0.56	5.6	0.56	7.3	0.73	10
5.6	0.56	5.5	0.55	7.6	0.76	10
5.7	0.57	5.3	0.53	7.9	0.79	10

Table 4.4: Statistics of the ten subjects.

Here $d_{real}$and $d_{estimated}$ are the real and estimated distances. In the case of Weinberg model the value of constant k was estimated to be 0.55, and in the case of Scarlett model the value of constant k was estimated to be 0.56, and in the case of Kim model the value of constant k was estimated to be 0.76. It is to be noted that the constant value k was different from the actual values referred in [271, 275, 276] due to different placement of the sensor unit and at different places.

4.1.3 Heading estimation

Once the strides (footstep and stride length) were estimated, it was important to estimate the pedestrian's heading for the complete course of pedestrian's walk (see Figure 4.2). Therefore, in this section we discuss one of the traditionally employed heading estimation technique. We quantify its performance when applied to smartphone embedded sensors in Section 4.2. Traditionally, heading can be estimated from either the magnetometer sensor or gyroscope sensor [277].The magnetometer sensor measures Earth's magnetic field. While gyroscope sensor measures the angular velocity around the three axes of a smartphone (see Figure 3.2(B)) [134]. The characteristics of two sensors are summarized in Table 4.5.

Sensor	Characteristics
Magnetometer	• Absolute heading • Unpredictable external disturbances
Gyroscope	• Relative heading • No external disturbances • Short term accuracy

Table 4.5: Comparison of the (gyroscope and magnetometer) sensor characteristics (Source: [271]).

In summary, the magnetometer sensor has a long-term accuracy. Therefore, it is more suitable for estimating the absolute heading, while gyroscope sensor is more suitable for estimating the relative heading. Also, it has been observed that unlike the magnetometer sensor, the gyroscope sensor are more stable in indoor environments [278]. The primary reason of such behaviour is because of the fact that the ferromagnetic material such as steel rods or electrical devices such as electrical wires, etc. generate a field that disturbs the Earth's magnetic field [124, 130]. This subsequently causes the Earth's magnetic field to fluctuate in an unpredictable manner. As a result, smartphone embedded magnetometer sensor was used for estimating initial absolute heading θ_{abs} of the pedestrian and smartphone embedded gyroscope sensor was used for subsequent heading θ_{re} estimation. The initial absolute heading was estimated according to equation 4.7 [279, 280].

$$\theta_{abs} = \arctan\left(\frac{m_y}{m_x}\right) \tag{4.7}$$

where m_y and m_x are the magnetic readings along the smartphone's y and x axis (see Figure 4.8) respectively. Following this, the subsequent relative heading θ_{re} was estimated according to equation 4.8 and added to the absolute heading θ_{abs} (determined at previous footstep) to estimate the total heading θ_{total} of the pedestrian at the current footstep (according to equation 4.9).

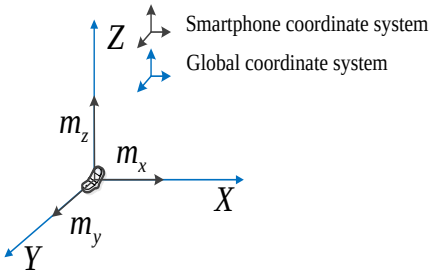


Figure 4.8: Illustration of the smartphone coordinate system and global coordinate system.

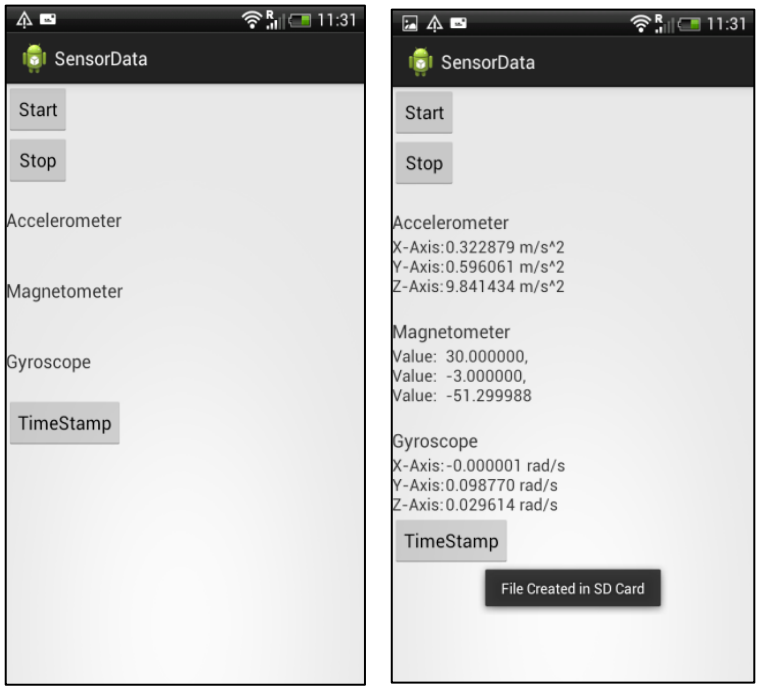
$$\theta_{re} = \int_0^t \hat{\theta}_z(t)\, dt \tag{4.8}$$

$$\theta_{total} = \theta_{abs} + \theta_{re} \quad (4.9)$$

where $\hat{\theta}_z(t)$ is the gyroscope measurement along the smartphone's z axis at time *t*. The smartphone's z axis is perpendicular to the ground therefore the global frame and the smartphone's coordinate frame are similar for this investigation.

4.2 Experimental setup & results

In order to validate the reliability of smartphone based PDR positioning system some walking experiments are done. The unrestricted human walking is complex and varied therefore walking was limited to horizontal forward walking [143] in the first instance. Twenty volunteer test subjects consisting of eight females and twelve males aged 20 - 40 years old were selected (selection criteria and ethics documentation are in Appendix B). The height of the test subjects ranged from 1.55 m to 1.85 m. There was no restriction on the type of footwear. All of them had different footwear – six of the women wore boots, two of the men wore leather shoes and the remaining wore sports shoes. The sensor data (accelerometer, gyroscope and magnetometer) was logged at a sampling rate of 20 Hz in a comma separated values (csv) file with our designed SensorData app, as illustrated in Figure 4.9.



(A) (B)

Figure 4.9: Screenshot of the designed SensorData app.

The csv file was stored in smartphone's external memory – secure digital (SD) card. It was then analysed in the MATLAB environment. It was noticed that because of the limitation of Android kernel the sampling events were generated at varied time instants. Therefore, the sensor data was re-sampled at 20 Hz by linear interpolation in MATLAB. It was assumed that there was

no obstacle in front of the pedestrian and the smartphone was assumed to be kept flat and stationary on the palm with screen facing upwards during the whole walk. Two set of experiments were performed, with the selected group of participants:

1. Straight line experiment
2. Square turning experiment

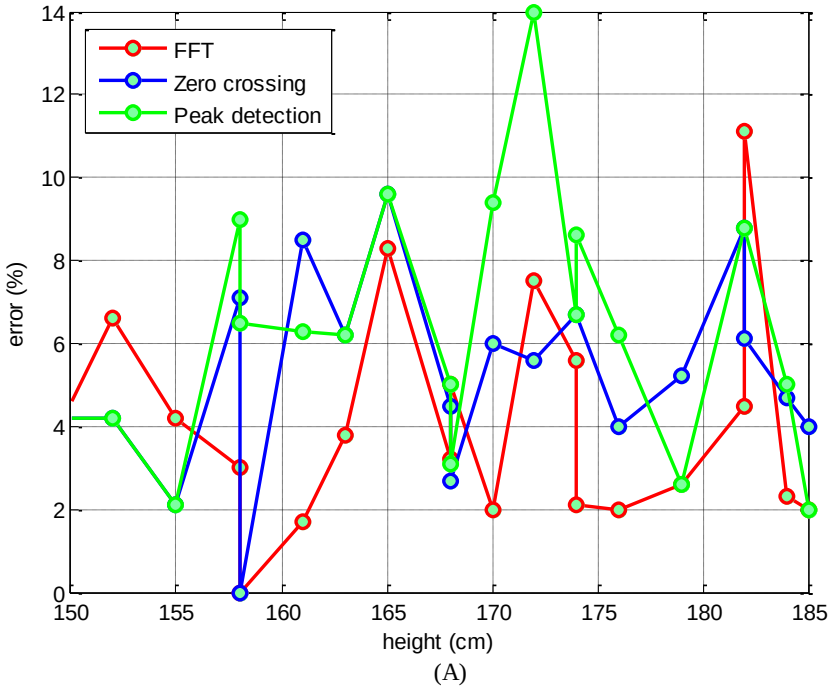
4.2.1 Straight line experiment

To evaluate the performance of foot step detection and stride length estimation algorithms, we performed – 'Straight line experiment'. The experiment was grouped into two sub-experiments – constant speed experiment and dual speed experiment. In the constant speed experiment, the subjects were asked to walk along a straight aisle of length 32.7 m at a pace which they perceived as a normal walking pace. In the dual speed experiment, the subjects were again asked to walk the same distance but at varying pace, half the distance (14.9 m) at normal walking pace, then pause and then remaining at a slow running pace. The pedestrian steps varied from 40 to 55 steps in the constant speed experiment. While the pedestrian steps varied from 17 to 24 steps in the first half and 12 to 27 steps in the second half on the dual speed experiment. The foot step detection error characteristics is evaluated by equations 4.10 and 4.11. RMAE is the relative mean absolute error (*RMAE*) percentage and ε is the error in footsteps detected.

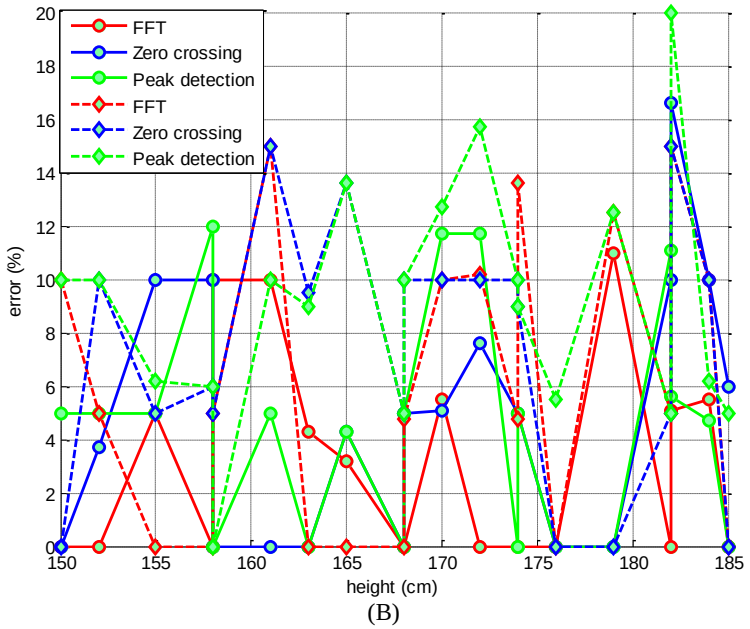
$$\varepsilon = |\mathrm{FS}_{\mathrm{estimated}} - \mathrm{FS}_{\mathrm{actual}}| \tag{4.10}$$

$$\mathrm{RMAE}(\%) = \frac{1}{\mathrm{N}} \sum_{\mathrm{i}=1}^{\mathrm{N}} \frac{|\varepsilon_{\mathrm{i}}|}{\mathrm{FS}_{\mathrm{actual}}} 100 \tag{4.11}$$

where $\mathrm{FS}_{\mathrm{actual}}$ is the actual number of footsteps taken and $\mathrm{FS}_{\mathrm{estimated}}$ is the number of footsteps estimated by the foot step detection algorithms, and N is the number of times the experiment was performed i.e. equal to 20. The $\mathrm{FS}_{\mathrm{actual}}$ was recorded in a csv file by asking the subjects about the number of footsteps they took to complete the whole walk. The step detection error percentage for all test subjects during the two set of sub-experiment is shown in Figure 4.10. The solid lines in Figure 4.10(B) represent footsteps detected over a length of 14.9 m and the dashed lines represent the footsteps detected over a length of 17.8 m in a dual speed experiment.



(A)



(B)

Figure 4.10: Comparison of foot step detection algorithms for (A) constant speed experiment and (B) dual speed experiment.

Error characteristics of the results for all the test subjects during two sets of sub-experiment are shown in Table 4.6 and Table 4.7.

Algorithm	**Length (m)**	RMAE **(%)**	**95% confidence level (%)**
FFT	32.7	4.10	1.44 to 6.76
Zero crossing	32.7	5.34	3.03 to 7.65
Peak detection	32.7	6.41	3.36 to 9.46

Table 4.6: Comparison of error characteristics for footstep detection algorithms for the constant speed experiment.

Algorithm	**Length (m)**	RMAE ***(%)***	**95% confidence level (%)**
FFT	14.9	2.98	-3.74 to 10.7
	17.8	6.04	-3.06 to 17.14
Zero crossing	14.9	4.41	-3.27 to 12.09
	17.8	7.40	-1.9 to 17.46
Peak detection	14.9	4.55	-4 to 13.09
	17.8	9.07	-1.09 to 18.05

Table 4.7: Comparison of error characteristics for footstep detection algorithms for the dual speed experiment.

The results indicate that the FFT based algorithm outperform the other foot step detection techniques during the two sets of sub-experiment. This behavior was as expected. Since the peak detection algorithm considers a valid foot step when the acceleration peak crosses a minimum threshold because of which several footsteps are missed and also falsely detected. In contrast, zero crossing algorithm rejects false foot step detection via time based thresholding. While FFT based approach employs the nature of the pedestrian's gait and the periodicity of placing the footsteps on ground to count the footsteps. As a result of this, the approach rejects the false foot step detection and avoids missing steps. Comparing the first half walk's performance to second half walk's performance in the dual speed experiment; all techniques perform better in the first half walk when a pedestrian was walking normally. This is due to the fact that when a pedestrian moves briskly they shake their body more. As a result more step misdetection occurs, and subsequently this behavior causes the accuracy of algorithm to be reduced.

After foot step detection, the stride length is computed to estimate the actual distance travelled. The total distance travelled is calculated by summing up the estimated step length of every detected foot step using equation 2.6 (see Chapter 2, Section 2.2.4). Dynamic methods were used, as described in Section 4.1.2, to estimate the stride length. Comparison of the error characteristics – mean absolute error (MAE) and 95% confidence interval for the total distance

travelled using these methods are summarized in Table 4.8 and Table 4.9. The MAE is calculated according to the equation 3.31 (see Chapter 3, Section 3.4.2).

	MAE**(m)**	**95% confidence level (m)**
Weinberg model	1.35	-3.12 to 3.08
Scarlet model	1.48	-4.13 to 2.19
Kim model	1.52	-3.71 to 3.65

Table 4.8: Comparison of error characteristics for the stride length models for constant speed experiment.

	MAE**(m)**	**95% confidence level (m)**
Weinberg model	1.62	-3 to 6.24
Scarlet model	1.73	-3.37 to 7.83
Kim model	1.90	-3.48 to 7.08

Table 4.9: Comparison of error characteristics for the stride length models for dual speed experiment

It is observed that Weinberg stride length model can estimate the total distance travelled better than others. This is indicated by the smallest mean absolute error (MAE) and the smallest 95% confidence level. On comparing the two experiments, it is observed that the stride length models perform relatively poor in the dual speed experiment. Again, this behavior is as expected. It can be explained with the same reasoning as before that is due to the fact when a pedestrian walks briskly their footsteps are often missed and falsely detected. As a result, the accuracy of the foot step detection algorithms decreases down and because of which the performance of stride length models also decreases. The similar results have also been observed in [281]. In particular, they used the Weinberg stride length model to estimate the total distance travelled during normal and varying pace walking. Also, Weinberg *et al.* [275] have reported that the step length can vary by as much as 40% between pedestrians walking at normal speed, and up to 50% across the range of walking speeds for an individual. Therefore it can be concluded that when a pedestrian walks normally, the position error is significantly less in comparison to when the pedestrians walks at varying pace.

4.2.2 Square turning experiment

After the investigation of stride estimation (foot step detection techniques and stride length models) algorithms, two set of sub-experiments were again performed on different routes for evaluation of heading estimation algorithm (see Section 4.1.3). The similar set of subjects were

again asked to walk along two different L shaped paths – path 1 and path 2 as illustrated in Figure 4.11. The green dot in Figure 4.11 represents the starting point of the walk and red dots represent the stopping point of the walk in two L shaped paths.

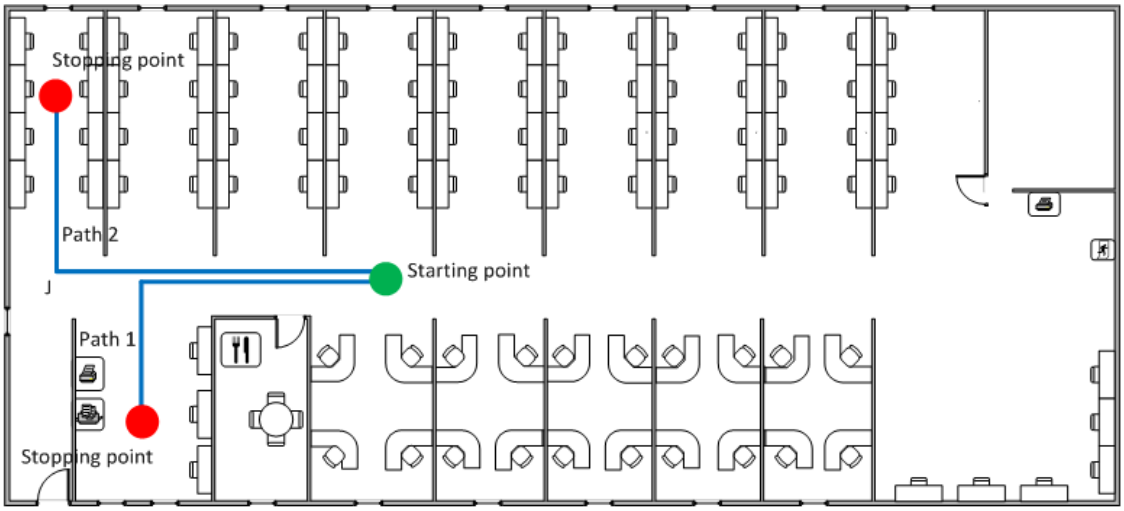
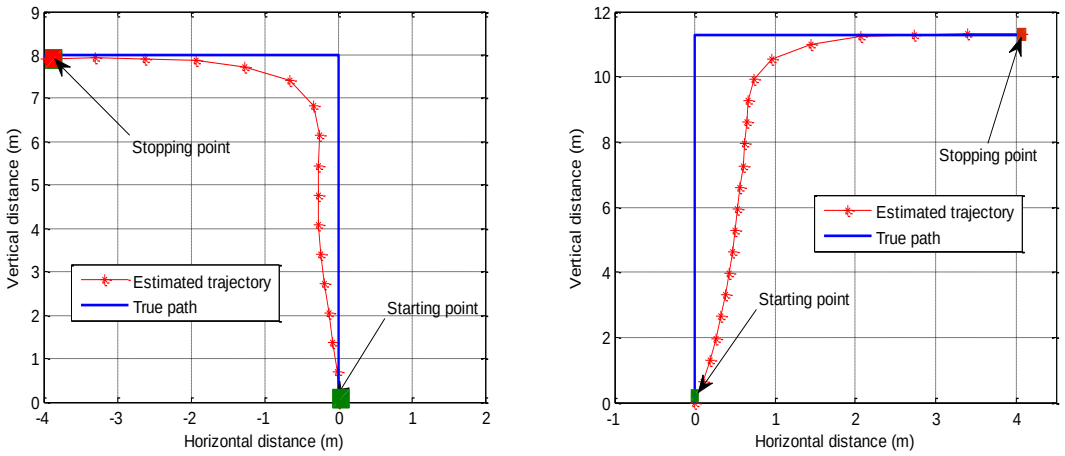


Figure 4.11: Indoor floor plan of the level 3, building 22 of the University of Southampton.

In the first set of sub-experiment, the subjects were asked to walk along the path 1 of length 12.1 m. The first straight path walk was of length 8 m then a 90 degree left turn and then the rest path was of length 4.1 m. In the second set of sub-experiment, the subjects were asked to walk the path 2 of length 15.4 m. The first straight path walk was of length 11.3 m then a 90 degree right turn and then the rest path was of length 4.1 m. In both the cases the subjects started at the same point and it was again assumed that there was no obstacle in the path.

Figure 4.12 (A & B) shows the actual and estimated trajectory of the pedestrian of height 1.79 m walking along the two L shaped paths. The trajectories were drawn using equations 2.4 and 2.5 (see Chapter 2, Section 2.2.4.1). The estimated trajectory does not have the same exact trajectory but it is close to the actual walking path. This can be because of the presence of positional drifts.



(A) (B)

Figure 4.12: True and estimated path in the two L paths.

The percentage error in heading at turning corners is evaluated by the following equation

$$\text{Relative Absolute Heading error (\%)} = \frac{|\theta_{\text{actual}} - \theta_{\text{estimated}}|}{\theta_{\text{actual}}} 100 \quad (4.12)$$

where θ_{actual} is the actual heading during turning at corners. It is equal to 90^0. $\theta_{\text{estimated}}$ is the estimated heading during turning at corners. Table 4.10 summarizes the results. As can be seen, the average relative heading error percentage is less than 11% in corners for both the cases. This is sufficient to trace the sharp changes in the heading when a pedestrian makes a left or right turn [282]. Comparing the path 1 with path 2, the relative mean absolute percentage error is less in path 2. This behavior was expected. Since, the path followed by the pedestrian was more constrained during turning in the path 2 which can also be observed from Figure 4.11. Therefore, as a result the pedestrian took a sharp turning in the path 2 when approaches near the turning point J.

L paths	Relative mean absolute heading error (%)	95 % confidence level (%)
Path 1	9.78	3.5 to 16.04
Path 2	7.07	0.33 to 13.81

Table 4.10: Comparison of the heading error characteristics in the turning corners along the two L paths.

4.3 Summary

This chapter focussed on investigating whether off-the-shelf stride estimation techniques (footstep detection techniques and stride length estimation models) and traditional heading estimation algorithm could be applied to the multi-modal sensor data from inertial sensors

embedded in a smartphone, to position a pedestrian on a real indoor environment. It was the first of its kind in the sense that smartphone was placed farthest from the torso and toe. Although, several research studies have applied such techniques however most of these have focussed on applying the techniques on foot mounted inertial module [260, 273]. Considering the fact that human motion is a periodic activity of placing and removing the toe from ground. So sensors placed nearer to toe can sense the optimal signal. In [161], authors have applied the techniques when senor unit was placed in waist and in [283] authors have applied the techniques on inertial sensors embedded in smartphone however smartphone was placed in pocket. The key findings of this chapter are as follows:

- For, an individual the walking speed varies less and also their body shakes a little less when they walk normally. Significantly, the performance of footstep detection techniques in a smartphone based PDR positioning system is best when the pedestrian walks normally.
- The FFT based footstep counting algorithm outperforms the other footstep detection techniques that are based on applying threshold to either the acceleration signal or the time. This behaviour is significant. Considering the fact, that human walking is a periodical activity and FFT based footstep counting algorithm employ the periodicity of acceleration signal to count footsteps, so as a result it avoids missing and falsely detecting steps.
- The Weinberg stride length model performs better in comparison to the other off-the-shelf stride length estimation models. This can be attributed because of the dependence on heuristic parameter 'k', that fitted on the selected set of data relatively better than others.
- The average percentage heading error is less than 11% in corners. This behaviour is significant. Since, it ascertains that it can be employed to sufficiently trace the sharp changes in heading when a pedestrian makes a left or right turn [282].

Although encouraging results were obtained in this chapter, there were some limitations. These are as follows:

- Presence of positional drifts: This drift introduces error in position estimates, either suddenly, due to a hardware glitch or an out-of-range measurement, or gradually, due to slowly accumulating measurement errors. Figure 4.12 showed the presence of positional drifts in the estimated trajectory. This drift can be because of the error in step direction estimation. Since the experiment was performed for less than a minute, the effect of drift was not dominant. However, for long walks the positional drifts can be dominant [133].

- Dependence on heuristic parameter ‘k’: In all the stride length estimation models (see Section 4.1.2) the parameter k is user specific. Therefore, the parameter k can be set individually for a user through calibration. This sets the average error of the estimated distance to be reduced [284]. However, one-size-fit-all model is going to produce large errors for people with more extreme walking characteristics. Also, for the same person, the step length can vary due to differences in walking speed [139, 275] and difference in shoes [285].

These shortcomings of the designed smartphone based PDR system make it unreliable as a standalone indoor positioning system. In particular, the presence of drift errors that tends to grow overtime. Figure 4.12 illustrated that if we know the map we could correct the trajectory. Moreover, Chapter 3 reiterated that features could be used to ‘correct drift’ and ‘locate on a map’. So we set out augmenting the PDR technology described in this chapter by incorporating the feature detection method in Chapter 6.

Also, in this chapter it was assumed that the orientation of smartphone relative to pedestrian was static. The position of smartphone was dictated to the pedestrians throughout the experiments to keep it straight and flat such that the orientation of pedestrian relative to the smartphone is static. However, at times this was not true because of sudden obstacles or glitches. A more realistic scenario to consider this assumption to be valid is a ‘Smart Glass’ scenario. This is discussed in the next chapter.

5. Chapter 5: Helmet mounted PDR

The previous chapter was primarily limited to smartphone handheld (see Chapter 4) scenario; with the assumption that orientation of the smartphone relative to the pedestrian was fixed during the complete walk. This assumption may not always be valid for a pedestrian walking on a pathway (either indoors or outdoors) while carrying a smartphone on their hand. However, this assumption can be considered as true for the case of smart glasses (e.g. Google Glass). Since, their orientation relative to the pedestrian is much more predictable and static [286]. Therefore, this chapter provides a reference implementation of smartphone enabled 'Smart Glasses simulator' for pedestrian positioning.

To begin with, the chapter first gives an introduction about the smart glasses in Section 5.1. It would also detail about the designed Smart Glasses simulator. Next the chapter presents the experimental setup and results obtained from the smartphone enabled Smart Glasses simulator PDR system in Section 5.2. Finally, Section 5.3 summarizes the key findings of this chapter.

5.1 Introduction

Smart glasses allow a user to augment reality by superimposing into the visual field a digital image. The device resembles the eye goggles found with virtual reality [287], however, unlike virtual reality, smart glasses display a virtual world in addition to what a user sees naturally by integrating the information from various sensors embedded in the device. An illustration of the smart glasses is shown in Figure 5.1. These contain the similar set of sensors as a smartphone as illustrated in Table 5.1.

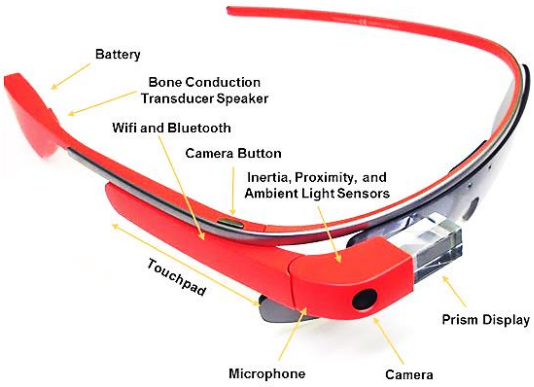


Figure 5.1: An illustration of Google Smart Glass (Source: [43]).

Hardware	Sensors	Connectivity	Cost
Google Glass	Accelerometer, gyroscope, digital compass, proximity sensor, ambient light sensor and camera (5 megapixel)	GPS, WLAN (Wi-Fi 802.11b/g) and Bluetooth	£1000
HTC Z710e	Accelerometer, gyroscope, digital compass, proximity sensor, ambient light sensor and camera (8 megapixel + VGA)	GPS,EDGE, WLAN (Wi-Fi 802.11 b/g/n and Wi-Fi hotspot), Bluetooth and USB	£300

Table 5.1: Comparison of the sensors present in a Smart Glass and smartphone.

In addition, smart glasses are provided with the similar connectivity like GPS, Bluetooth and Wi-Fi as a smartphone that can successfully provide useful navigational information to the pedestrians. However, the cost of purchasing smart glasses is roughly three times the cost of purchasing a smartphone (see Table 5.1). So, in lieu of purchasing smart glasses for positioning we have developed the "University of Southampton Smart Glasses Simulator" – see Figure 5.2 below.

Figure 5.2: The University of Southampton Smart Glasses Simulator

This kind of smartphone enabled 'Smart Glasses Simulator' has never been investigated before for positioning.

5.2 Experimental setup & results

To evaluate the performance of our smartphone enabled 'Smart Glasses simulator' based PDR positioning system some walking experiments are done. The experiment was focussed on analysing two different scenarios as illustrated in Figure 5.3.

- Earlier (handheld, see Chapter 4, Section 4.2) scenario: Pedestrian walking normally while carrying a smartphone (HTCZ710e) flat (with screen facing towards ceiling) on their hand.

- Smart Glasses Simulator: Pedestrian walking normally while smartphone (HTCZ710e) attached to the helmet.

Figure 5.3: Pedestrian holding smartphones during walk to evaluate the two scenarios: Smart Glasses Simulator and smartphone handheld.

Twenty volunteer test subjects consisting of eight females and twelve males aged 20 – 40 years old were selected (selection criteria and ethics documentation are in Appendix B). Each subject was given one smartphone and one Smart Glasses simulator for the complete walk. The subjects were asked to walk normally on a straight path of length 25 m in addition they were also asked to count the number of steps during the whole walk. Again, as earlier in the previous chapter (see Chapter 4, Section 4.2), the experimental conditions were same i.e. it was presumed that there was no obstacle during the whole walk and secondly, the starting point was known a-priori. Also, there was no restriction on the type of footwear and all of them had different footwear – two of the women wore boots, four of the men wore leather shoes, two of the men wore slippers and the remaining wore sports shoes. Table 5.2 summarizes the conditions of conducted series of experiment.

Conditions	Values
Number of subjects	20 (12 men and 8 women)
Distance	25 m
Weight	45kg – 90kg
Height	1.50 m – 1.85 m

Table 5.2: Conditions for the conducted series of experiment.

5.2.1 Step Counter

As discussed earlier (see Chapter 4, Section 4.1.1.3), we use FFT based step rate estimation technique to count the number of footsteps. The footstep detection error ε is evaluated by equation 4.10 (see Chapter 4, Section 4.2.1). The error statistics is evaluated in terms of the relative mean absolute error (RMAE) percentage via equation 4.11 (see Chapter 4, Section 4.2.1) and 95% confidence interval, as illustrated in Table 5.3. The step count error for all test subjects during the complete walk is shown in Figure 5.4.

Scenario	Length (m)	RMAE (%)	95% confidence level (%)
Smart Glasses simulator	25	2.74	-2.46 to 7.94
Smartphone handheld	25	1.67	-1.47\ to 4.81

Table 5.3: Step count error characteristics.

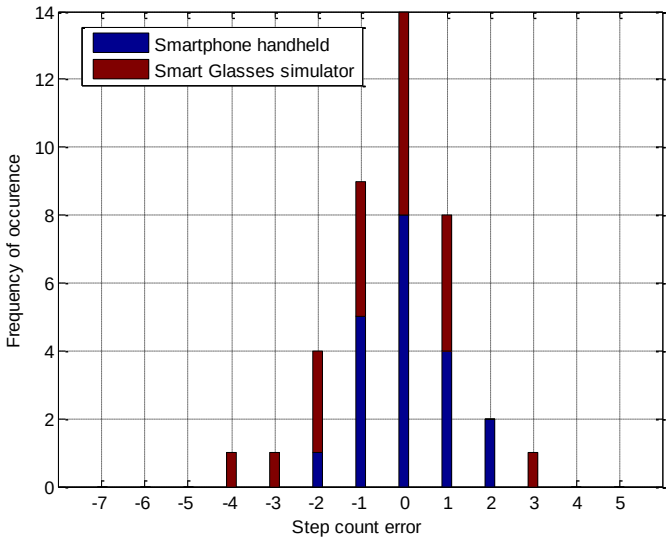


Figure 5.4: Histogram of the step count error for the two scenarios (A) smartphone handheld and (B) Smart Glasses simulator.

A cursory examination of the logged accelerometer data in csv file indicated that the Smart Glasses simulator performs relatively poor in comparison to the handheld scenario. This is also viable from Figure 5.4 and the stats shown in Table 5.3. The primary reason of such a behaviour can be attributed because of the fact that when a smartphone is placed on hand it is directly attached to the body, and also cautiously as well as continuously stabilized by the pedestrian throughout the walk. So, as a result of this the high frequency vibration and noises get damped. In comparison, the smartphone attached to the helmet undergoes several events of wobbling throughout the walk. This is likely due to the weight and fit of the helmet and may not be an issue for real smart glasses. As a result, there are several instances which cause the smartphone to move and, to detect false and miss steps. Thus, the step counter accuracy substantially degrades for Smart Glasses simulator.

Although the results appear to be counter intuitive, in general it is expected that the performance of smartphone attached to the helmet should have been better in comparison to its smartphone handheld counterpart. Since, internal human locomotor system tries to balance the head as stated and observed in the work by Blum *et al.* [288]. It is however to be noted that Blum *et al.* have tightly and directly mounted the sensor module onto the pedestrian's head. On the contrary, in the case of Smart Glasses simulator the smartphone is attached to the bicycle's helmet. While in the case of smartphone handheld scenario the pedestrian continuously stabilizes their hand. Arguably, if the smartphone would have held freely the results may have been different.

5.2.2 Position error

After foot step detection, the stride length is estimated to compute the distance travelled using equation 2.6 (see Chapter 2, Section 2.2.4). The total distance travelled is calculated by summing up the estimated step length of every counted footstep. Dynamic method was used, as described in Section 4.1.2 to compute the stride length. In particular we selected Weinberg stride length model since, it fitted our collected data set relatively better in comparison to other stride length models (as observed in the previous chapter, see Chapter 4, Section 4.2.1). This estimated distance travelled is then compared against the actual distance at end point to compute the position error d_{err}. The position error at known point of interest (POI) i.e. end point is computed as follows:

$$d_{err} = |\text{Estimated distance at POI} - \text{Actual distance at POI}| \quad (5.1)$$

The error statistics is evaluated in terms of mean absolute error (MAE) and 95% confidence level. The MAE is calculated according to the equation 5.2 at known point of interest as shown below:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|d_{err}| \quad (5.2)$$

where N is the number of times pedestrian walked across the straight path (i.e. N = 20). Table 5.4 illustrate the error statistics for the two scenarios, it is observed that the mean absolute

error is less than 2.5 m in either of the two scenarios. However, 95% confidence level is relatively less for the smartphone handheld scenario. This behaviour is likely to happen because the pedestrians continuously try to stabilize their hand during the walk. As a result, the hand held smartphone actually yield more reliable step length estimates than its head-mounted counterpart.

	MAE **(m)**	**95% confidence level (m)**
Smartphone handheld	1.68	-3.62 to 4.58
Smart Glasses simulator	2.31	-6.62 to 5.56

Table 5.4: Position error statistics for the two scenarios (A) smartphone handheld scenario and (B) Smart Glasses simulator.

A work by Shih *et al.* [289] have reported the similar results like this. In particular, smartphones placed slightly away from the body (i.e. smartphone is not attached to the body tightly) has shown higher position error. They have placed the smartphones on chest pocket and on waist. So, the smartphone placed on chest pocket showed higher positioning error in comparison to the smartphone firmly attached to the waist. However, in this research work the smartphones are handheld and attached to the Smart Glasses simulator.

5.3 Summary

The chapter focussed on investigating a scenario when the orientation of smartphone relative to the pedestrian was static and more predictable e.g. in Smart Glasses. A 'University of Southampton Smart Glasses Simulator' was designed in this context instead of purchasing a smart glass, considering the fact that smart glasses are roughly three times the price of a smartphone. This kind of investigation has never been reported before (as per our knowledge). The key findings of this chapter are as follows:

- The step count error for a Smart Glasses simulator is relatively high in comparison to the smartphone handheld scenario. This can be attributed because of the fact that they don't experience the damping effect means they experience high frequency noises. This can make signal processing activity like step detection more difficult. As a result performance of step counter algorithm degrades. This can be observed in Figure 5.4.
- The mean absolute error is less than 2.5 m in either of the two scenarios – smartphone handheld and Smart Glasses simulator. However, 95% confidence level is relatively less when the smartphone is handheld. This behaviour is likely to happen because the pedestrian continuously tries to stabilize his hand during the walk. This damps the high frequency vibrations and noises caused due to walking. As a result, the hand held smartphone actually yield more reliable step length estimates than its helmet mounted counterpart in a Smart Glasses simulator.

Although encouraging results are observed for smartphone enabled Smart Glasses simulator, there are some limitations. These are as follows:

- Weight unfit problem: In order to design the 'University of Southampton Smart Glasses Simulator', we have used the bicycle helmet as the base. While the helmet fitted relatively well to a set of pedestrians. However, it drooped for some of the pedestrians. So it had to be adjusted. This is a time consuming process. Nevertheless, this may not be a problem for real smart glasses.
- Position of smartphone: The position of smartphone is relatively static in either of the two scenarios – smartphone handheld and Smart Glasses simulator. In the smartphone handheld scenario, the pedestrian is dictated to keep the hand still throughout the walk. On the contrary, in the Smart Glasses simulator scenario the smartphone was mounted onto the helmet. It is however less likely that the smartphone's positon remains static for a long duration during walking in an actual scenario.

6. Chapter 6: Map aided PDR

In the Chapter 4, it was investigated whether off-the-shelf stride estimation techniques (footstep detection techniques and stride length estimation models) and traditional heading estimation algorithms could be applied to the data from inertial sensors embedded in a smartphone to position a pedestrian in an indoor environment. The experimental results showed interesting performance (see Chapter 4, Section 4.2). Step detection error was least using FFT based step rate estimation algorithm, mean absolute position error was least using Weinberg Stride length model and relative mean absolute heading error was less than 11% in turning corners.

Their performance was however limited due to variance in drift, users' walking characteristics and dependence on heuristic parameters. Therefore, this chapter would utilise the findings of Chapter 3 to counter the drift. In particular, it would employ the 'feature (corner) detection' and 'map matching method approach' introduced in Chapter 3, Section 3.4 to counter the drift. Also, it would investigate the statistical approach – 'Kalman filter' [152] to get a better estimate of the stride lengths; while reiterating the experiments over the two scenarios discussed earlier in the previous chapter (see Chapter 5, Section 5.2) – smartphone handheld and Smart Glasses simulator. As stated by Attia *et al.* [290].

"In indoor environment, where it is hard to provide a continuous reliable position estimate, map matching can bridge the navigation solution based on the floor alignments. The building information provide a logical threshold to bound the solution into a certain region, changing the main target of the navigation system from obtaining a high accuracy to position information to obtaining a position with enough accuracy allows the system to select the correct passageway."

To begin with, the chapter first gives an introduction about the basic principles of map based positioning technique in Section 6.1. A review of previous work is provided in Section 6.2. Next the chapter provides detail about the designed map based positioning technique aided with the smartphone based PDR positioning system in Section 6.3. The map based positioning technique is designed based on a probabilistic framework – multiple uncertain routes engine (MURE). It computes the probability of a pedestrian to be present on a particular link in a map provided the map is defined in form of nodes and links. In particular, it extends the continuous probabilistic framework of the Kalman filter to keep track of multi-modal discontinuous probability distributions in a map. Section 6.4 presents the experimental setup and results obtained from the smartphone based PDR positioning system aided with map. Finally, Section 6.5 summarizes the key findings of this chapter.

6.1 Introduction

Over the past two decades, many map based positioning techniques have been proposed and evaluated in various scenarios [291]. Most techniques have been developed for vehicle navigation, a few for outdoor pedestrian navigation, and very few have focused on pedestrian indoors. This can be attributed to the complex nature of building layouts and unconstrained motion of the pedestrian indoors [292-294]. Importantly, most of the map-based positioning techniques are limited to laboratory settings and to relatively simple environments [168]. The map based positioning technique offers three key advantages in terms of locating pedestrian indoors [272, 295, 296].

1) It employs the naturally occurring structure of typical indoor environments to derive pedestrian's position information.
2) It constrains the pedestrians' motion according to the geometry of floor structures.
3) It improves the accuracy of the positioning system by limiting the scope of pedestrian to the nearest feasible point.

The map based positioning technique, is not an independent positioning technology instead it aids positioning by adding constraints [297]. The constraints can be in form of vertical or horizontal lines as illustrated in Figure 6.1. Or these can be physical constraints such as corridor width, obstacles, road connectivity, etc. [298, 299]. There are two primary aspects of map based positioning technique [165]:

1. Building the geospatial data model
2. Map matching

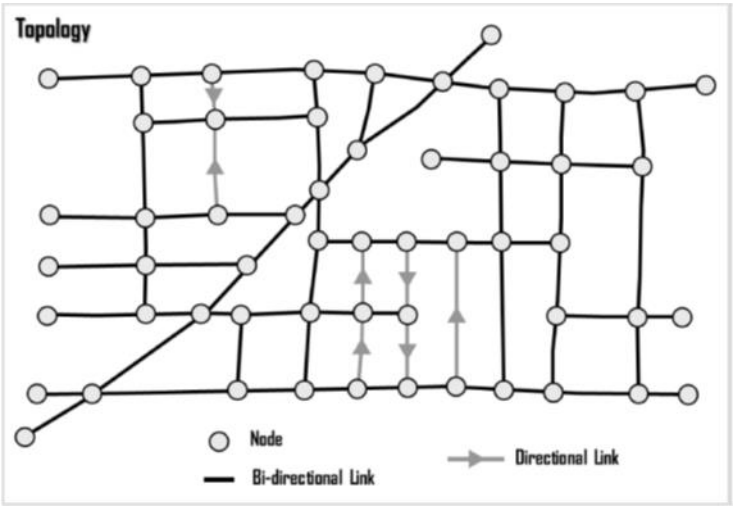


Figure 6.1: An example of a road network model (Source: [300])

6.1.1 Building the geospatial data model

The geospatial data model is a form of Geographic Information System (GIS), with specific attributes moulded for navigation applications [280]. The map presents the pedestrian's location, direction and his/her navigation states in the pictorial form. At its simplest level maps can be in form of a floor plan of a building [28]. However, floor plans do not include any information on semantics or any further alphanumerical data regarding staircases, doors and so on [39]. Therefore, most of the maps are designed in form of spatial graphs. The graphs model all possible passageways, height change access (stairs, elevators, etc.), topological (connectivity between passageways, etc.) and geometrical characteristics of the environment [290, 294].

Broadly the maps are classified into two forms [301] (1) polygon and (2) network. The polygon type maps define the walls and other geometrical features as "obstacles" that cannot be traversed by a person. For instance Walder *et al*. [292] have employed the knowledge of the polygons of the outline of accessible areas of the rooms and transition objects such as doors and stairs to constraint the path of pedestrian. Significantly, increased level of position accuracy was observed from unmatched to the matched trajectory.

In contrast, a network type map defines the path in which pedestrian walks as polylines. It is composed of links, nodes and arcs [302]. According to the Graph theory [303] an arc is a set of curves in the Euclidian space (R^2). An arc can be completely characterized by a finite sequence of points. A node is a point at which an arc terminates or begins or a point at which it is possible to move from one arc to another. A link is a linear element between two adjacent points [296].

Wakuda *et al.* [301] reported a comparison of these two map forms. Some of the key elements of the two map forms are illustrated in Table 6.1.

	Network Map	**Polygon map**
Characteristics	Represents walkable path and steps as polylines.	Represents geometrical structures – wall, walkable space, etc. as polygons.
Strengths	Low calculation cost and easy to prepare.	Open spaces are clearly visible with furniture and other accessories represented as obstacles.
Limitations	Cannot represent open space and simultaneously, extensive knowledge of the map geometries including high quality map data is required.	High calculation cost and cannot represent pathways.

Table 6.1: Key elements of the two map forms.

6.1.2 Map Matching

Map matching is a method to match the estimated position (determined from positioning system) to the coordinates of physical location (samples of data structure) using map data [293, 304]. The

key assumption being that if the positioning system returns a location that is not present on the map then this is likely an error and therefore, identifying the most likely location on the map is a correction. There are currently several approaches to map matching [291, 305], but in regards to indoor environment these can be classified as follows [306]: (1) geometrical map matching (2) topological map matching and (3) probabilistic map matching.

Originally these techniques were used for outdoor road-navigation however they have also proven their use for indoor navigation [303]. If only geometric information is used, the algorithm relies only on the shape of the arcs and not on the way they are connected. When the topological information is used in addition to geometrical information, the connectivity, proximity, and contiguity of the arcs are also considered. Thus the match is done in context and in relationship to the previous established matches. That makes the topological solution more likely to be correct.

One of the commonly used geometric approaches is **point-to-point matching** [307]. In this approach, the position fix from navigation system is snapped to the nearest node in the network as illustrated in Figure 6.2 In practise, the point-to-point matching is easy to implement with low computational load. However, it is very sensitive to the number of shape points building the candidate trajectories. It might be expected for the results to be much more accurate when using digital maps consisting of links with a dense distribution of nodes [308], but where the algorithm fails to match correctly when, for instance, challenged with the scenario shown in Figure 6.2. Assume that the pedestrian is moving on link A, and p_o is the position estimated from navigation system. Since the node b_y is closer to the p_o than any of the nodes belonging to link A, the point-to-point algorithm would falsely match the position fix to b_y.

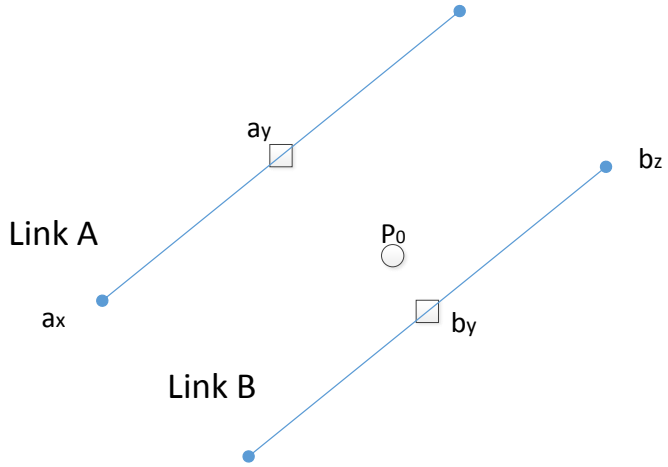


Figure 6.2: point-to-point map matching.

Another commonly used geometric map matching approach is **point-to-curve matching** [308]. In this approach, the position fix obtained from the navigation system is matched onto the closest link in the network as shown in Figure 6.3. The true pedestrian's path is shown by the

thick green line, position fixes and results of map matching are shown by the circles and triangles respectively. In practise, it is more accurate than point-to-point matching however, point-to-curve matching results can be very unstable in dense indoor network because the closest link may not always be the correct link. This approach may fail if a pedestrian is travelling on a nearby parallel links and near intersections as shown in Figure 6.3 by the hollow triangles.

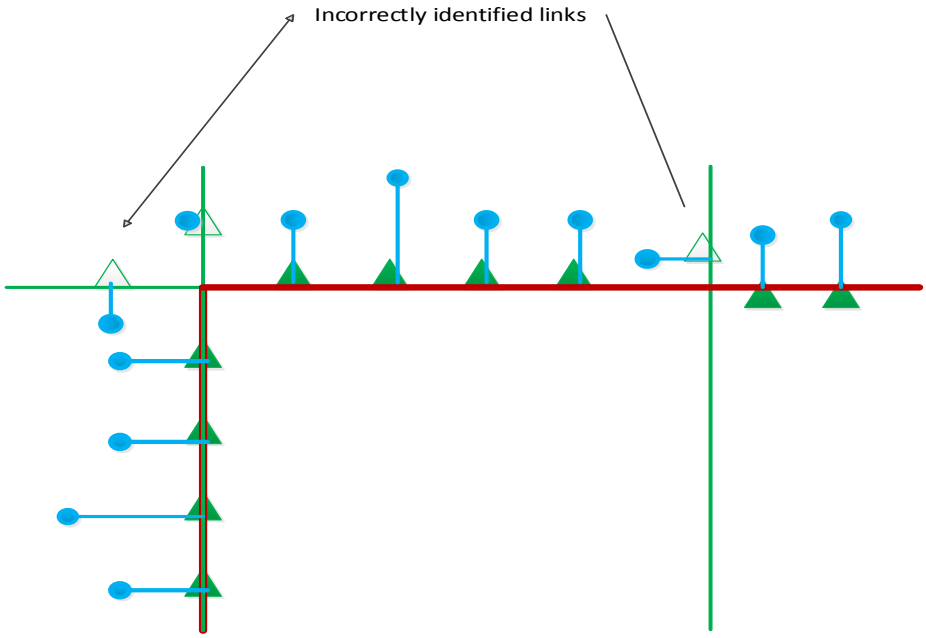


Figure 6.3: Point-to-curve map matching.

An enhancement of the point-to-curve and point-to-point map matching algorithms is generalized in the weighted **topological algorithms** which was first proposed by Greenfeld for road networks. Greenfeld [309] developed the topological map-matching algorithm that involves matching the curve generated by the raw positioning data with the geometry of the network-navigation model segments. However, different versions of the algorithm have been proposed for the indoor applications e.g. [70, 280]. The main purpose of this algorithm is to choose a suitable link where the pedestrian is located on a network map and then location of a pedestrian on this link. This selection is based on the shortest distance between the position fix (the position provided by the navigation system) and the links. The link that yields the shortest distance is selected. In addition, many other constraints can be used with the previous proximity constraint such as the heading of pedestrian. This constraint is implemented by calculating the angle between the bearing of the link (using coordinate information from the spatial database) and the heading of the pedestrian (from navigation sensor output). The link with the smallest angle is selected. The selection of a suitable link is then based on a weighting between the proximity and bearing factors, as shown in Figure 6.4 and Figure 6.5.

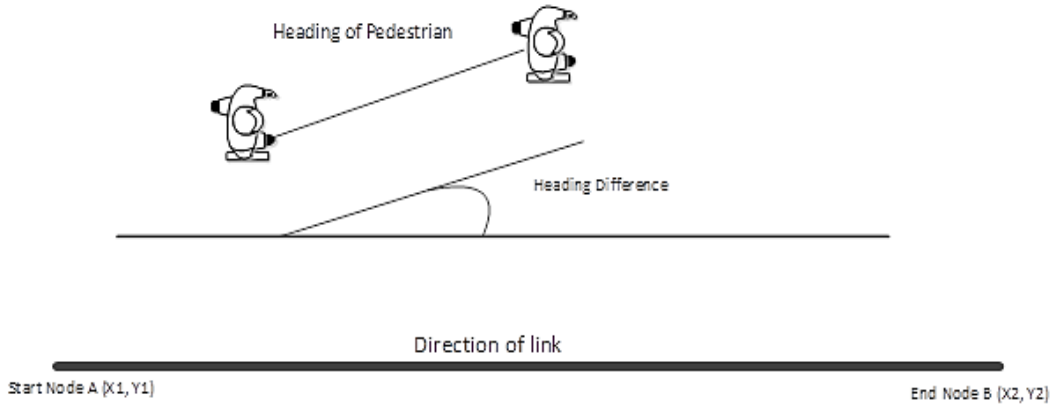


Figure 6.4: Similarity in pedestrian heading and bearing of a link

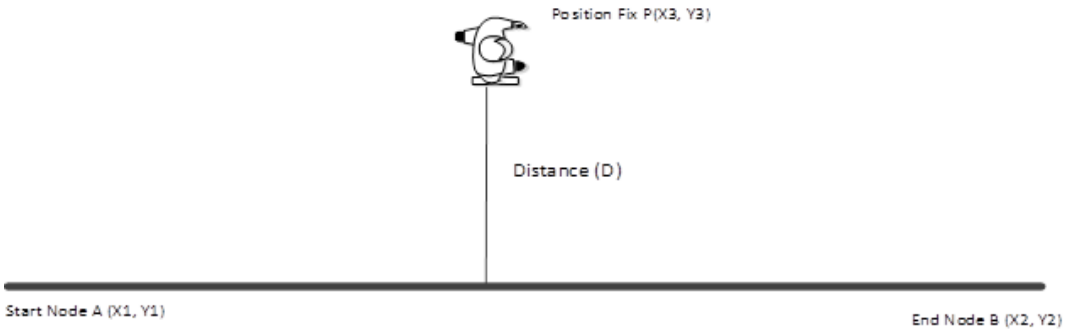


Figure 6.5: Perpendicular distance between position fix and a link.

The **probabilistic approach** gives the most reliable solution compared to other methods. It differs from the geometrical and topological approach in the criteria used for selecting the network link. Rather than using mathematical distances or angles in selecting the link, a statistical test is used [153, 310, 311]. The approach attempts to improve the quality of map matching through better accounting for some of the uncertainty in position estimates. The kind of methods include probabilistic techniques such as Kalman filter, Particle filter, Hidden Marcov models, etc. for positioning and few of these techniques are discussed earlier in Chapter 2, Section 2.3.

In practise, it uses probability inferences to determine the likelihood of a particular location given that prior location vector array has already been detected. The calibration data including initialization is considered as a part of priori conditional probability distribution by the algorithm to determine the likelihood of a particular location occurrence [50, 54].

The MURE map matching algorithm has also been designed based on the principles of probabilistic map matching. In practise, MURE algorithm computes the probability of a pedestrian to be present on a particular link on a map. A full probabilistic treatment of map

matching in indoor localization can result in multiple discrete location distributions at nearby points in the indoor network map provided the indoor map is designed in form of network with nodes and links. A Multiple Uncertain Route Engine (MURE) algorithm is designed to keep track of these.

For a pedestrian moving in a two-dimensional space a one dimensional pedestrian displacement – measured via step counting and stride length estimation algorithm, may be coupled with a measurement of heading to estimate a fully two-dimensional path [312]. However, most indoor environments do not afford pedestrians free movement in a two-dimensional space, rather they are constrained to walk along corridors and into out of rooms. In this case it can be convenient to describe the pedestrian's location as a positon on a network, because paths through a network are inherently one-dimensional. An additional benefit of the network representation is that 'features' of the network may be detected and used to help to locate the pedestrians and limit error growth. For e.g. [62, 313] showed that by detecting turnings as a 'feature' error growth can be bounded. A complication of the network representation arises from the fact that pedestrians are constrained to be on one of a number of discrete edges of the network.

This discretization does not immediately lend itself to the use of the Gaussian distribution and the Kalman filter, which are powerful tools for dynamic localization in continuous space. In particular, if there is an ambiguity over which edge a pedestrian is located on, a single unimodal probability distribution is not suitable to represent the position of the pedestrian.

In this case we propose that the pedestrian's location should be represented by a discrete set of continuous one dimensional distributions, where each distribution has an existential probability. To achieve this we propose an extension to the Kalman filter to keep track of multi-modal discontinuous probability distributions in a network map. We have named this approach the 'multiple uncertain routes engine' (MURE). Complete details of the algorithm is discussed later in Section 6.3.4.2.

6.2 Related work

Indoor version of the probabilistic map matching algorithm using particle filter has been developed by [295, 299, 314, 315]. Nammoon and Youngok [314] introduced a map aided PDR scheme using an INS embedded in smartphone. The step based PDR was used to determine the successive positions of a pedestrian that were later projected onto the network map using a particle filter based map matching. The particles were updated at every step based on estimated step distance and orientation. While those particles that exceeded the reference thresholds of corridor width were degenerated, however if at any position along the way the sum of the total particle weight was lower than a reference value of valid particle, the particles were regenerated at that position.

Kaiser *et al.* [295] proposed the use of an angular probability density function for weighting particles within the particle filter. In this work, wall crossing constraints were not the only constraint. First, the particles were weighted according to their direction with respect to an angular movement model, derived from complete mapping of a building. Second, particles, which crossed walls were naturally de-weighted in this model. An important assumption for the developed system was that the corridors are considered to be narrow Performance of the methods were not considered in scenarios of open areas and when using maps with no internal wall constraints.

In [299, 315] Bao and Wong introduced step counting based dead reckoning algorithm which utilizes map matching. In [315] the tracked path is mapped to a nearby known corridor. It presumes that a pedestrian tends to walk along a quasi-straight line when they are in a corridor. Once this walk pattern is detected and there is a nearby corridor, the estimated location is mapped to the corridor, and the step direction and the sensor's orientation is calibrated by the known corridor's direction. To reduce the dependency on corridors, in [299] an improved particle filter is proposed. The improved PF underlines the uncertainty in the step direction estimation. The particles with the wrong direction estimation are more likely to be the ones with the wrong location estimation and gets eliminated by map constraints. The improved particle filter takes 1/6 of the CPU time compared to the traditional particle filter. Although the improved particle algorithm provides reliable results with less CPU cost, the algorithm at all times need to maintain a minimum number of 100 particles and moreover, the algorithm necessitates a dense map constraint environment with corridors.

In addition to these studies there are several other studies too that have used the probabilistic map matching for positioning via particle filter [28, 316]. The basic idea of probabilistic map matching using particle filter is generally the same. A set of particles are used to represent the posterior density of the unknown position which are distributed over the digital building plan. If a particle collides with a wall, it is excluded from the Monte Carlo simulation [210] where walls represent an impassable obstacles. A significant drawback of this approach is that in any case they require evaluation of particles over feasible map region which is a computationally expansive process [317].

To overcome this, [317] propose a Kalman filter enabled map matching. Map is segmented into polygons with feasible and infeasible areas. If in any case the Kalman filter estimates lie outside feasible areas then map constraints are applied as a measurement to the Kalman filter. By properly choosing the measurement covariance, the Kalman estimate is refined to satisfy the map constraints and the corresponding covariance matrix is updated accordingly. The map matching procedure is effective when the feasible set is smaller than infeasible set. Also, the map matching procedure exhibits a delay of up to few seconds in updating the position estimate.

Attia *et al.* [280] however use geometric map matching to position a pedestrian. A network map is designed in form of node and links. Whenever the pedestrian's dead reckoning trail of estimated locations gets close to points placed on links, all the candidate links close to that estimated locations are selected. The final link is computed by the heading difference between the estimated heading of the pedestrian and the heading of candidate links. The link with the smallest heading difference is considered to be the actual link. After the actual link is determined, the estimated position is projected onto the final link by point to curve map matching. The results showed a remarkable drop of 4.41 m in mean positioning error from unmatched to the matched trajectory however it was not clear as to how the map matching algorithm would select the final link when several candidate links are close together.

6.3 Smartphone based PDR system aided with map

The block diagram of smartphone based PDR positioning system aided with map is illustrated in Figure 6.6. It is composed of five parts: 1) Building a geospatial data model 2) A feature (corner) detection algorithm 3) The MURE algorithm 4) Smartphone based PDR system and 5) Updating stride lengths. The accelerometer and gyroscope samples are again filtered according to equation 4.1 (see Chapter 4, Section 4.1). The filtered accelerometer samples are fed to the smartphone based PDR system. The filtered gyroscope samples are used to detect the corner features. The implicit position information from PDR system, the geospatial data model; represented in a form of topological map and feature (corner) detection module are fused together to estimate the relative position of the pedestrian. In case corner features are detected the stride lengths are updated, and the information is fused via MURE map matching algorithm. The following subsections describe the system architecture in detail.

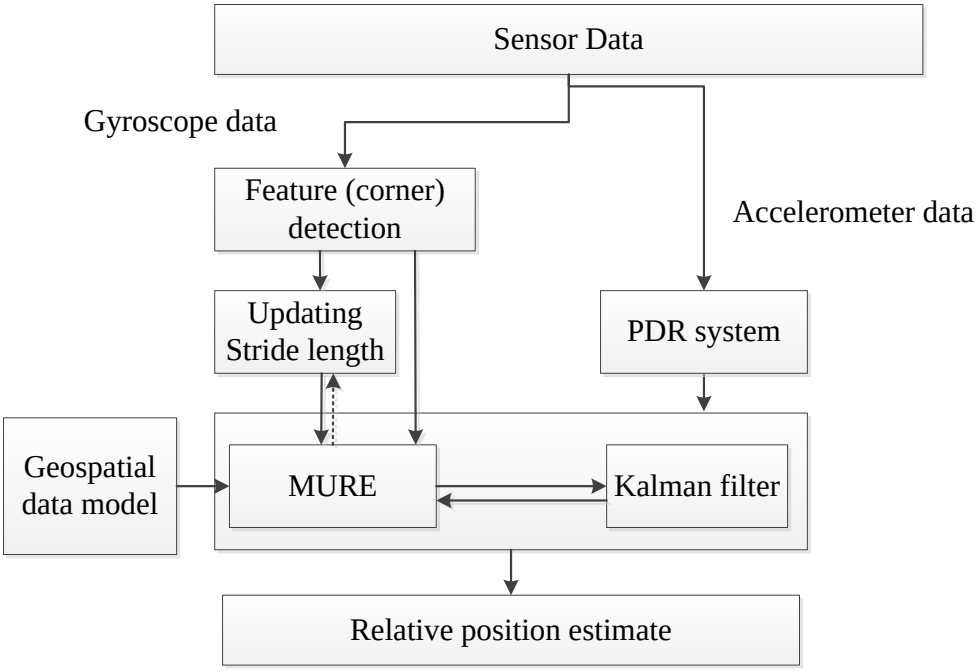


Figure 6.6: Block diagram of the smartphone based PDR positioning system aided with map.

6.3.1 Feature (corner) detection

The process flow diagram of the feature (corner) detection algorithm is illustrated in Figure 6.7. It. utilizes the gyroscope measurements – that have been passed through a low pass filter for sensing the change in heading direction and applies a simple thresholding measurement to detect a turning. As mentioned by El-Gohary *et al.* [318].

"A turn is considered to be detected when a sufficiently high angular velocity is measured by the gyroscope. This high velocity should be visible in the gyroscope output before the walking direction has changed significantly."

A threshold limit of 0.6 rad/s is set on the magnitude of angular velocity measurements to identify a corner turning according to the equation 6.1.

$$|\dot{\theta}_z| = \begin{cases} \geq 0.6 \text{ rad/s}, & \text{pedestrian taking turn} \\ < 0.6 \text{ rad/s}, & \text{pedestrian walking straight} \end{cases} \quad (6.1)$$

where $\dot{\theta}_z$ is the smartphone's angular velocity along the axis perpendicular to the ground. The threshold was set by analysing the angular velocity measurements (from gyroscope) of ten pedestrians from a walk of length 20 m over the straight path. The absolute mean during the straight path varies from 0.04 rad/s to 0.11 rad/s with 95% confidence interval in between –0.3 rad/s to 0.3 rad/s (see Table 6.2). Therefore a threshold value of magnitude much higher than the maximum deviation value was selected to detect turns.

Subjects	Absolute mean (rad/s)	95% confidence level (rad/s)
1	0.04	-0.11 to 0.12
2	0.05	-0.15 to 0.15
3	0.08	-0.22 to 0.22
4	0.05	-0.13 to 0.15
5	0.04	-0.12 to 0.12
6	0.06	-0.18 to 0.18
7	0.09	-0.22 to 0.22
8	0.11	-0.30 to 0.30
9	0.06	-0.14 to 0.14
10	0.08	-0.28 to 0.28

Table 6.2: Angular velocity measurement profile for 10 subjects during the straight walk of length 20 m.

Once a turn is detected the sign of the angular velocity measurements determines whether the turn is right or left turn as illustrated in Figure 6.7. If $\dot{\theta}_z$ is positive, a left turn is detected, otherwise a right turn is detected as illustrated in Figure 6.8. This very simple turning detection algorithm was suitable for this work – where all the turnings in the tested network maps were 90° turns. In maps containing turnings of different angles the authors would recommend to replace this algorithms with a more discriminatory approach, for e.g. as described in [316, 319].

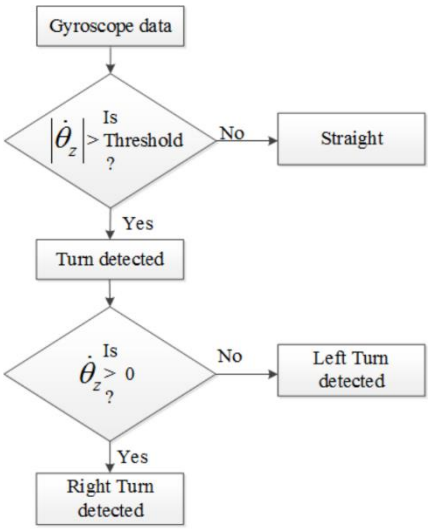


Figure 6.7: Flowchart of feature detection algorithm.

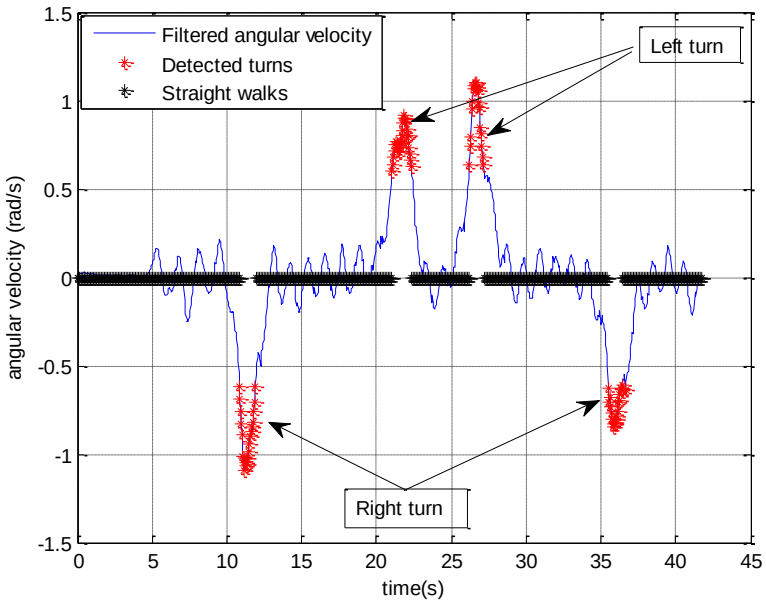


Figure 6.8: Turns and straight walks detected using the gyroscope senor's signal.

6.3.2 PDR system

The system architecture of the smartphone based PDR system is illustrated in Figure 6.9. Basic architecture is similar to the previously developed PDR system (see Chapter 4, Section 4.1), with some modifications. The data from inertial sensors are filtered according to equation 4.1 (see Chapter 4, Section 4.1). The filtered accelerometer signal is forwarded to an activity classification module that recognizes whether the pedestrian is walking or at rest. If the pedestrian is walking, their footstep rate is estimated using the step rate estimation algorithm and this is converted into a position displacement via the stride length estimation. These modules are discussed below.

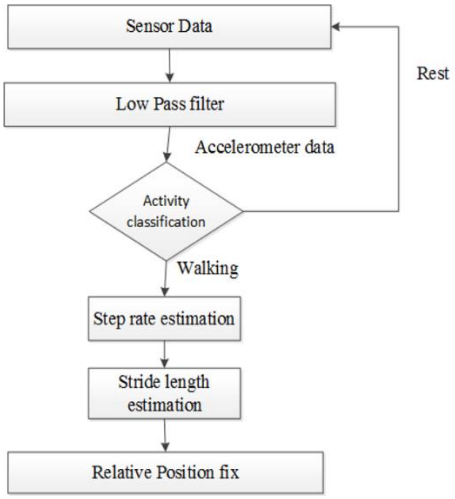


Figure 6.9: Pedestrian dead reckoning (PDR) architecture.

Step Rate estimation

Again, in this chapter we use FFT based step rate estimation technique (see Chapter 4, Section 4.1.1.3 for details) for calculating the step counts taken by a pedestrian over a particular time interval. As mentioned before (see Chapter 4, Section 4.1.1.3), the window length is a critical parameter. It determines the resolution in time and frequency domain. It should neither be too long nor too small. In practise, it should be long enough to achieve a good frequency spectrum estimate, but short enough to capture a local change in the periodicity of the signal. So, in the previous chapter we selected a window size of 1 second (20 samples). This was sufficient to detect the footsteps of selected group of pedestrians; who were asked to walk 'normally' and at 'variable speeds'. However, it was our belief that the performance of algorithm would improve with increase in window size. Also, as mentioned in [266], the performance of FFT algorithm improves by increasing the window size.

So, we designed a FFT window such that its size N increased gradually over the time scale of every 2 s until 10 s and subsequently, it slides over the complete time scale (as illustrated by equation 6.2). The footsteps were counted and updated at every 2 s.

$$N = \begin{cases} t, & \text{if } t <= 10 \text{ s} \\ 10s, & \text{if } t > 10 \text{ s} \end{cases} \quad (6.2)$$

where t is the time scale of 2 s. To investigate this fact, we selected a group of twenty volunteers who were asked to walk freely (without constraint) on a straight pathway of length 25 m. The subjects were provided with smartphone (HTCZ710e) and Smart Glasses simulator (see Figure 5.3, Chapter 5, Section 5.2). Also, they were asked to keep the smartphone flat and stationary (with screen facing towards ceiling) on their hands during walking. While the scenarios do not represent realistic user behaviour in all cases, we decided to limit the complexity of the problem in order to obtain an initial proof of concept. The foot step error statistics is evaluated and compared for the two scenarios (FFT window size of 1 s and 10 s) in Table 6.3. It is determined in terms of the relative mean absolute error (RMAE) percentage via equation 4.11 (see Chapter 4, Section 4.2.1) and 95 % confidence level.

Scenarios	**Window size (s)**	**RMAE**(%)	**95% confidence level (%)**
Smartphone handheld	1	3.82	-2.48 to 7.18
	10	1.15	-2.12 to 5.48
Smart Glasses simulator	1	4.24	-2.39 to 8.01
	10	1.45	-2.95 to 5.85

Table 6.3: Comparison of foot step error characteristics over a FFT window size of 1s and 10s.

The results confirm our belief. This can be observed with the smaller RMAE(%) and lesser spread of 95% confidence level for a sliding window size of 10 seconds (200 samples) with 80% overlap. This can be attributed because of the fact that when a pedestrian moves irregularly their motion is characterised by high values of variance in a short period of time [320]. As a result, shorter window frames are not able to effectively determine the actual step rate of the pedestrian.

Stride length estimation

Based on the findings from previous chapter (see Chapter 4, Section 4.3), we recollected that the Weinberg stride length model [275] performed better in comparison to the other off-the-shelf stride length estimation models. This can be attributed because of the dependence on heuristic parameter 'k', that fitted the selected data set of a group of pedestrians relatively better than others. However, if the group is relatively large one-size-fit-all model will produce large errors for people with more extreme walking characteristics [139]. Also, the position error can continue to grow if the initial conditions are not assumed correctly [243].

To consider this fact and to reduce the dependency on heuristic parameter 'k', the Weinberg stride length model is fused with the linear Kalman filter to estimate the relative position of the pedestrian in 1D using equation 2.6 (see Chapter 2, Section 2.2.4.1). We specifically selected linear Kalman filter because it is computationally light (less number of parameters involved) and can be implemented on different platforms [181] including memory constrained devices. This is detailed in the following section.

6.3.3 Kalman filter (KF)

Figure 6.10 illustrates the process flow diagram of the implemented filter, with centre portion of the filter representing the fundamental structure of the KF algorithm.

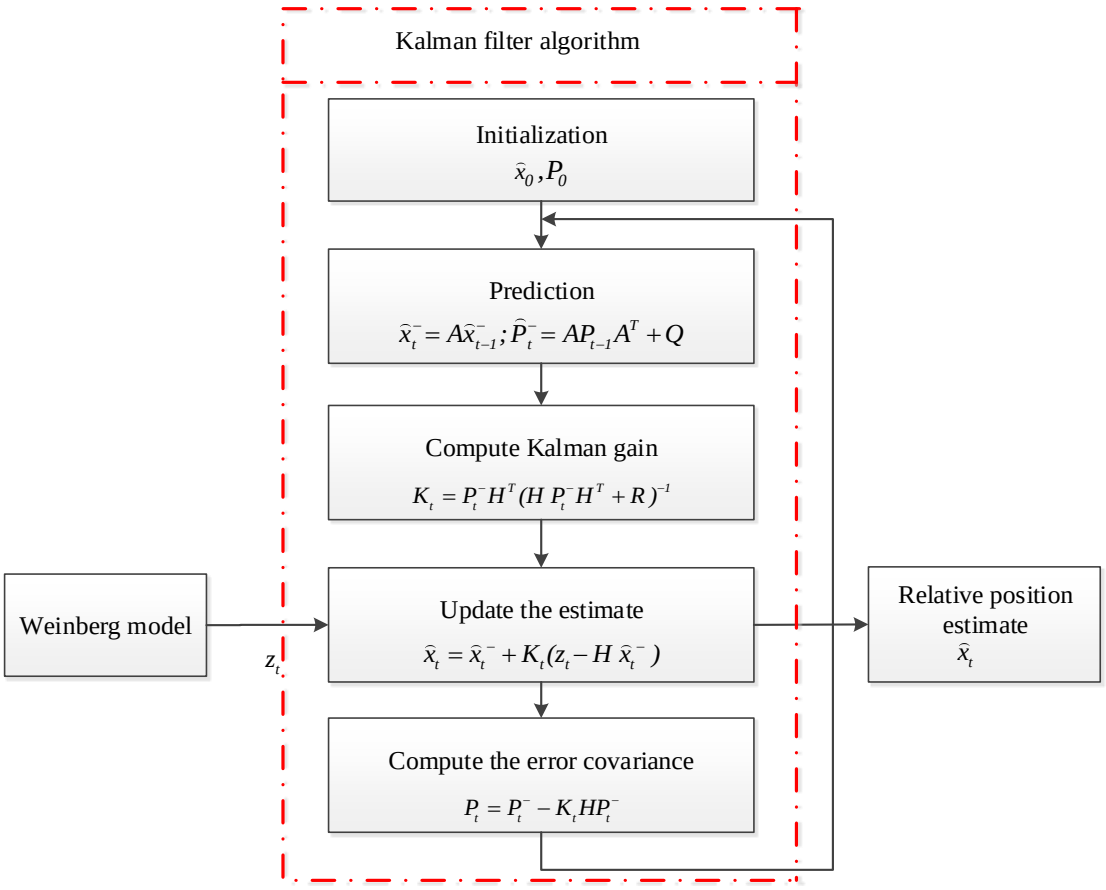


Figure 6.10: Flow diagram of the implemented KF.

The main steps of the integrated KF are as follows; let x be the position of the pedestrian along the link and ẋ be their velocity. Then the state vector x_t is defined according to

$$x_t = \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \tag{6.3}$$

The associated covariance matrix P_t is initialized using equation 6.4.

$$P_t = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix} \tag{6.4}$$

So, the system model of implemented filter at time t is formulated as follows:

$$x_{t+1} = A\, x_t + w_t \tag{6.5}$$

where A is the state transition matrix, defined as follows:

$$A = \begin{bmatrix} 1 & \delta t \\ 0 & 1 \end{bmatrix} \tag{6.6}$$

which simply implied that the pedestrian will continue walking at their constant speed and w_t denotes the system noise with zero mean, white and Gaussian distributed such that

$$w_t \sim N(0, Q_t) \tag{6.7}$$

Q_t is the covariance matrix of system noise and determined experimentally. δt is the time step of 2 seconds. The only measured quantity is the velocity of the pedestrian $\dot{x}$ which is calculated by using equation 6.8.

$$\dot{x} = \mu * SL \tag{6.8}$$

where μ is the step rate, as determined by the step rate estimation algorithm (see Section 6.3.2) and SL is the stride length estimated via Weinberg Stride length model. Therefore the measurement vector reduces to a scalar z_t. This is given as follows:

$$z_t = \mu * SL \tag{6.9}$$

The measurement model relates the measured velocity to the modelled velocity according to equation 6.10.

$$z_t = H\, x_t + v_t \tag{6.10}$$

where H is the state to measurement matrix, computed as follows:

$$H = [0 \quad 1] \tag{6.11}$$

and v_t is the measurement noise which is modelled as zero mean, white and Gaussian distributed. This is defined as follows:

$$v_t \sim N(0, R_t) \tag{6.12}$$

where R_t is the covariance matrix of measurement noise and is determined experimentally. A, H, Q and R and thus applied in the standard linear Kalman filter algorithm to compute the updated state and covariance matrix $\hat{x}_{t+1}$ and $\hat{P}_{t+1}$. The main steps of implemented Kalman filter algorithm (refer Figure 6.10) are detailed as follows:

- Initialization

  The initialization stage starts by estimating the state vectors $\hat{x}_t$ and its corresponding error covariance matrix P_t. In the model, it is assumed that the pedestrian is at rest on start. Therefore, initial state vector $\hat{x}_0$ and its corresponding error covariance matrix P_0 are defined as follows

  $$\hat{x}_0 = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{6.13}$$

  $$P_0 = \begin{bmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{bmatrix} \tag{6.14}$$

  where σ is a constant. Considering the initial state to be close to the actual state, therefore the value of σ was selected to be a small value equal to 0.1.

- Prediction

  The prediction stage, predicts how the estimate $\hat{x}_t$ will vary from the past time $(t-1)$ to the current time t. There are two steps involved in the prediction stage

  $$\hat{x}_t^- = A\hat{x}_{t-1} \tag{6.15}$$

$$P_t^- = A\,P_{t-1}A^T + Q \quad (6.16)$$

Here $\hat{x}_{t-1}$ is the previous best estimate at time $(t-1)$ with the corresponding covariance denoted by P_{t-1}. $\hat{x}_t^-$ is the predicted state at time t with the corresponding covariance denoted by P_t^-.

- Measurement update

  This is final stage of the KF algorithm. It estimates the state $\hat{x}_t$ based on the measurement z_t. There are three steps involved in this stage

$$K_t = P_t^- H^T (H P_t^- H^T + R)^{-1} \quad (6.17)$$

$$\hat{x}_t = \hat{x}_t^- + K_t(z_t - H\hat{x}_t^-) \quad (6.18)$$

$$P_t = P_t^- - K_t H P_t^- \quad (6.19)$$

Here K_t is the Kalman gain at time t. It is used to determine the weighting of the measurement information in updating the state estimates.

After estimating the relative position, the position estimates are refined by applying the map based positioning technique. The detail of map based positioning technique is discussed next.

6.3.4 Map Based positioning technique

It is well known that map based positioning techniques have been used to improve the localization performance of road motor vehicle navigation systems. This is primarily because the motor vehicle navigation is a relatively constrained process that follows set patterns of the road geometry [291]. In contrast, the pedestrian navigation is a less constrained process [139]. Pedestrians can walk forward, backward, move upstairs and downstairs, and can change the direction at any time. Additionally, the road networks are sparsely designed. However, the building floorplans are squeezed to fit several rooms in the available space. Therefore this map based positioning technique is designed keeping in note of these two factors. In particular, the 'less constrained' motion of pedestrian and second, the 'widely' available floorplans. The details of map based positioning technique is as follows, it consist of two parts: (1) Building a geospatial data model and (2) Map matching

6.3.4.1 Building a geospatial data model

The geospatial data model was designed in the form of a network map such that the main characteristic of human made indoor environments namely the parallelism and perpendicularity between the walls were preserved in the estimated features. It was presumed that an access to the building's floorplan (see Figure 6.11) was provided. The features of map can include the following [313]: walkable pathways, corridors, rooms, height change, staircases, point of interests, etc. Many other elements can be added to the list. However, this choice totally depends on the individual's discretion [70]. The designed network map is composed of the following

elements: passageways, corridors, doors, dead ends and point of interests as illustrated in Figure 6.12; represented by a set of nodes and links.

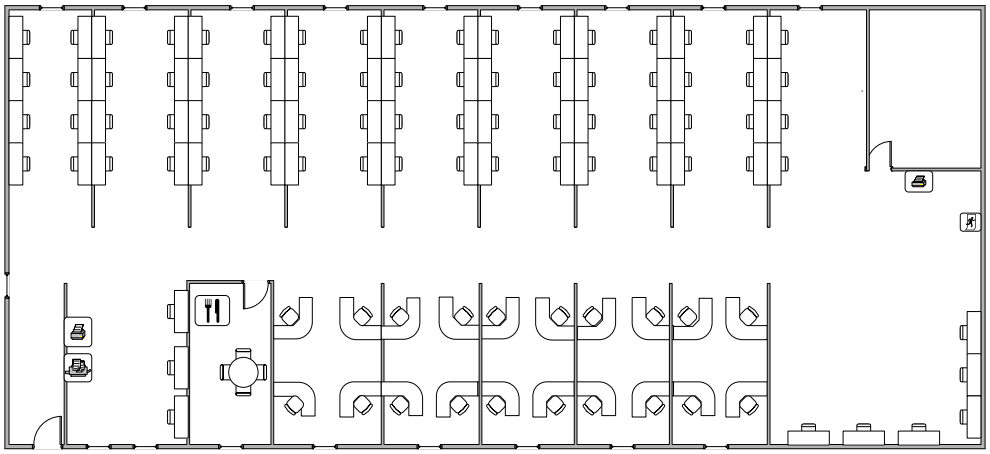


Figure 6.11: Floor plan for the level 3, Building 22 of the University campus.

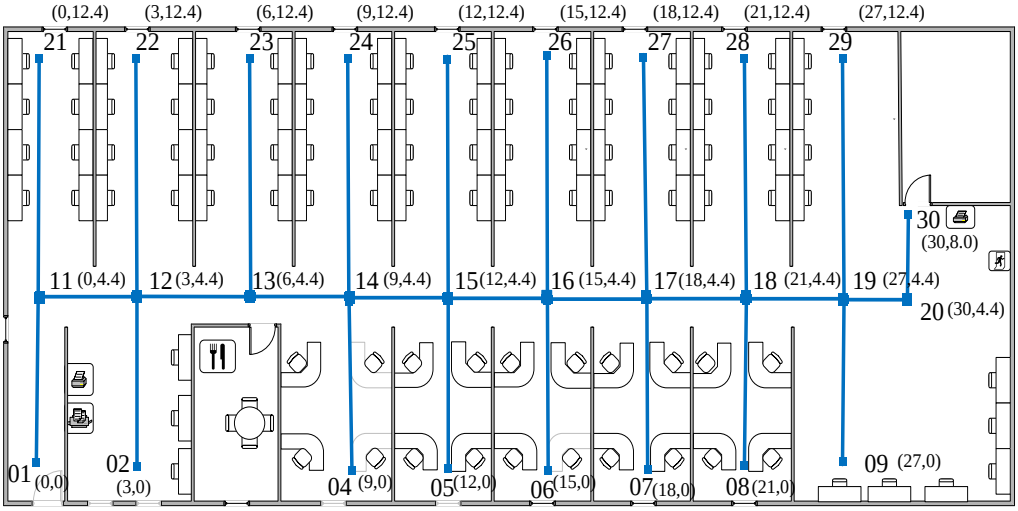


Figure 6.12: Geospatial data model for the level 3, Building 22 of the University campus. The markings (01, 02 ... and 30) indicate the node numbers and the blue lines interconnecting them represent links.

A link is modelled as a straight line, defined by two nodes. The links correspond to the corridors, passageways and walkable pathways for this research work. A node is modelled as a point where a user deviates from a straight path in the network or a point where one or several link ends (see Section 6.1.1). It corresponds to the junctions and to the point of interests in building (e.g. offices, rooms, etc.) for this research work. Each link in the network map is bidirectional. It has two nodes (start node and end node), a unique identification number (IdNumber) and

information concerning the access privileges. The nodes are georeferenced by a set of coordinates and an identifier (IdNumber).

This whole conceptual framework of links and nodes is stored in the form of Extensible Markup Language (XML) Schema as illustrated in Figure 6.13. This conceptual framework of XML Schema is parsed into the MATLAB software. The MATLAB software transforms the XML Schema into the implementation model of Network map.

```
<Link IdNumber="14:04">
      <startNode>14</startNode>
      <endNode>04</endNode>
   <access>1</access>
</Link>
```

```
<Node IdNumber="14">
      <X>9.0</X>
      <Y>4.4</Y>
</Node>
<Node IdNumber="04">
      <X>9.0</X>
      <Y>0.0</Y>
</Node>
```

Figure 6.13: XML Schema for a particular link and its respective nodes.

Although it is not assumed that a pedestrian navigating this map would perfectly follow the paths shown in form of network links (see Figure 6.12), it is however assumed that snapping the pedestrian's position to the network links will be sufficient for indoor localization purposes.

6.3.4.2 Map matching algorithm – MURE

The multiple uncertain routes engine (MURE) algorithm is based on the linear Kalman filter, but extends the representation of a pedestrian's position to support multiple discrete Gaussian probability distributions at different points in the network, where each distribution has an existential probability and all existential probabilities sum to unity.

This framework allows MURE to deal with potential ambiguity over which link a pedestrian has taken from a given node. The corner detection algorithm – described above (see Section 6.3.1) – is very useful for discriminating whether the pedestrian has turned left, right or straight-on at a node. However uncertainty may still arise in scenarios where there are for e.g. two nodes with available left turns close together and it is unclear which turn the pedestrian has taken (see Figure 6.14). In this case it makes sense to *split* the pedestrian's position into discrete distributions one on each candidate link. Furthermore at a later point in the journey one of these distributions may be ruled out – for e.g. by detecting a corner where none is available. In this case the distribution is *merged*, meaning that it is discarded and the *weight* of its existential probability is distributed among the remaining distributions. The policy for *splitting* and *merging* is the fundamental algorithm of MURE algorithm. It is described next.

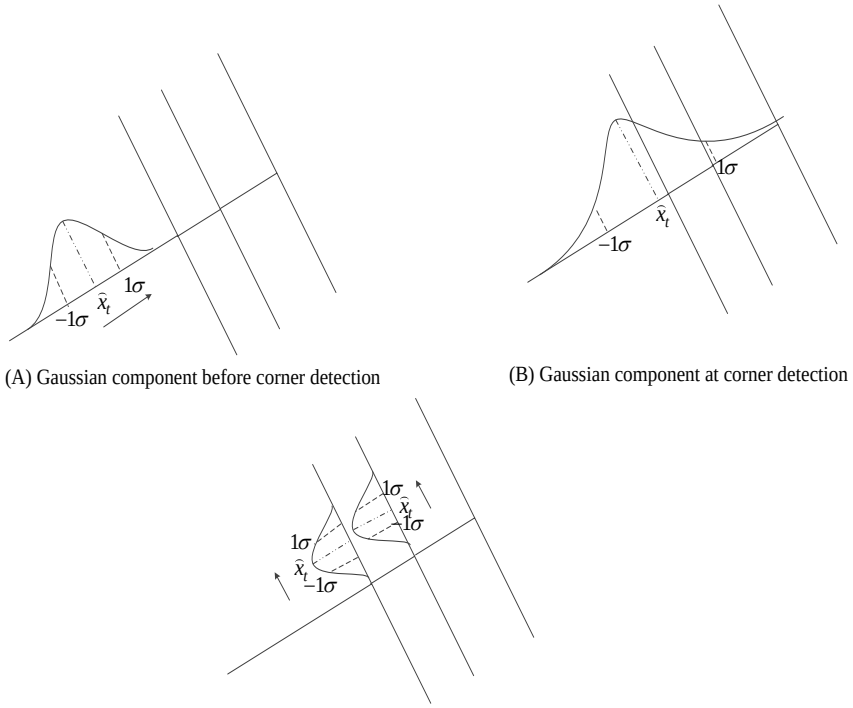


Figure 6.14: Gaussian component gets split into two neighbouring links when a 'left' corner is detected.

In practice each discrete probability distribution is represented by a software agent and each agent runs the MURE algorithm. A process flow diagram for the algorithm is illustrated in Figure 6.15.

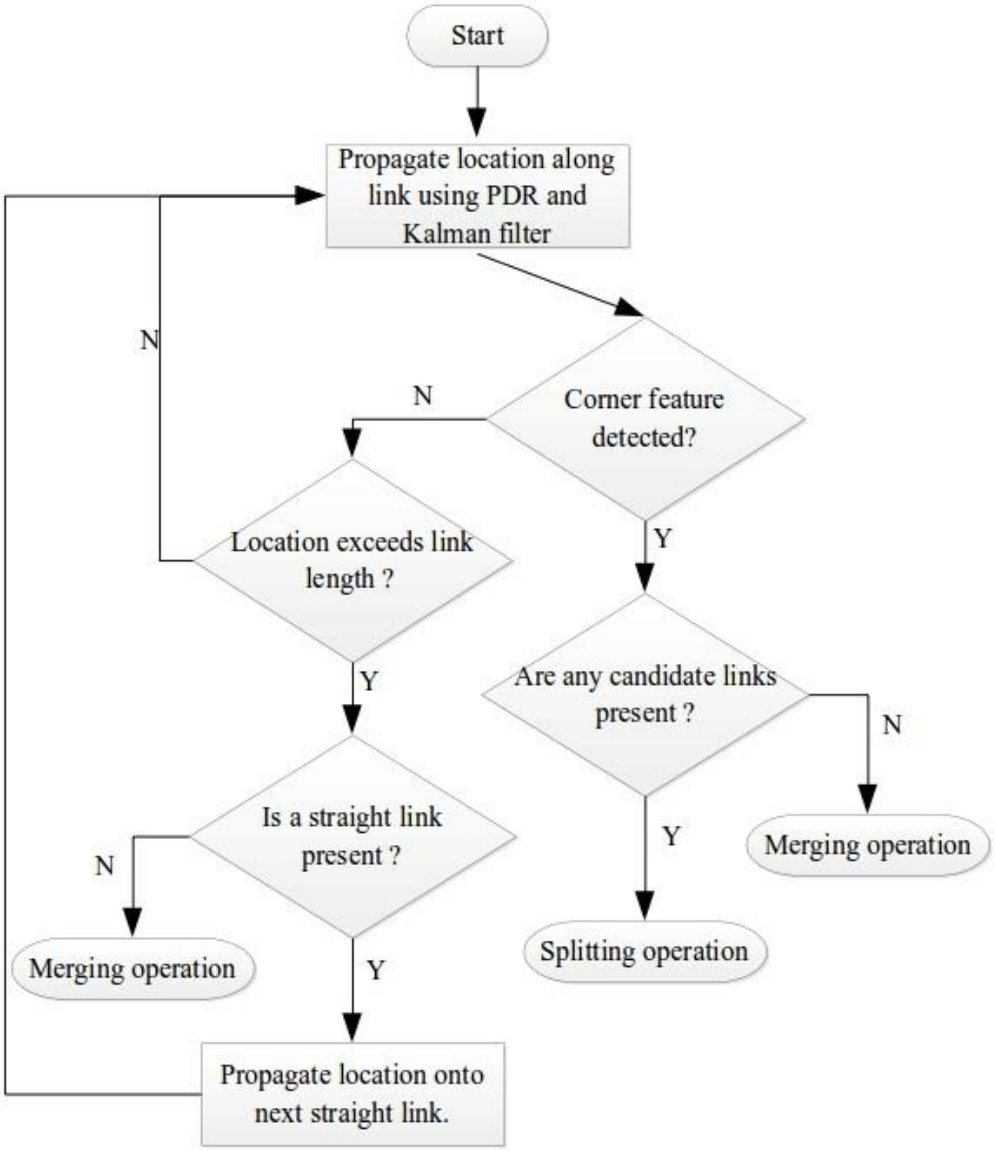


Figure 6.15: Process flow diagram of MURE map matching algorithm

In the event that a corner is detected the MURE algorithm checks all nodes that are within a truncated region of the current position distribution for candidate corners. The size of the truncation is a parameter that may be selected by the designer, in this study we select $\pm 1\sigma$. In the case that more than one candidate link is present a splitting operation occurs. In the case that no candidate links are present a merging operation occurs.

Splitting

Upon splitting, the old (*parent*) position distribution is discarded and new (*child*) position distributions are initialized at the beginning of each of the candidate links. The initial standard deviation of the distribution is a constant parameter that may be selected by the designer according to the response speed of corner detection. In this work an empirically determined initial standard

deviation of $\pm$ 0.7 meters was used. For each child distribution a new agent is created running its own MURE algorithm and Kalman filter (described in the previous Section 6.3.3). This re-initialization of the standard deviations of the child distributions accomplishes two things:

1. It is the 'map-matching' element of the MURE algorithm, reflecting the fact that we know that the pedestrian is near a corner and can reduce the error accordingly.
2. It effectively stops and restarts the Kalman filter at this point thus overcoming the problems of discontinuities in state.

The final step of the splitting process is to assign an existential probability to each of the child distributions. The existential probability is estimated using a heuristic method where the assigned probability is proportional to the probability density value of the old distribution at the location of the node where the corner turning event took place. For each child distribution $i \in m$ the existential probability $P_i^{e\prime}$ is calculated by using equation 6.20.

$$P_i^{e\prime} = \frac{f(d_i)}{\sum_{i=1}^{m} f(d_i)} * P^e \tag{6.20}$$

where P^e is the existential probability of the parent distribution and $f(d_i)$ is the value of Gaussian probability density function at the position (d_i) of the node connecting to the link on which i is located. The $f(d_i)$ is calculated by using equation 6.21.

$$f(d_i) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(\bar{x}-d_i)^2}{2\sigma^2}\right) \tag{6.21}$$

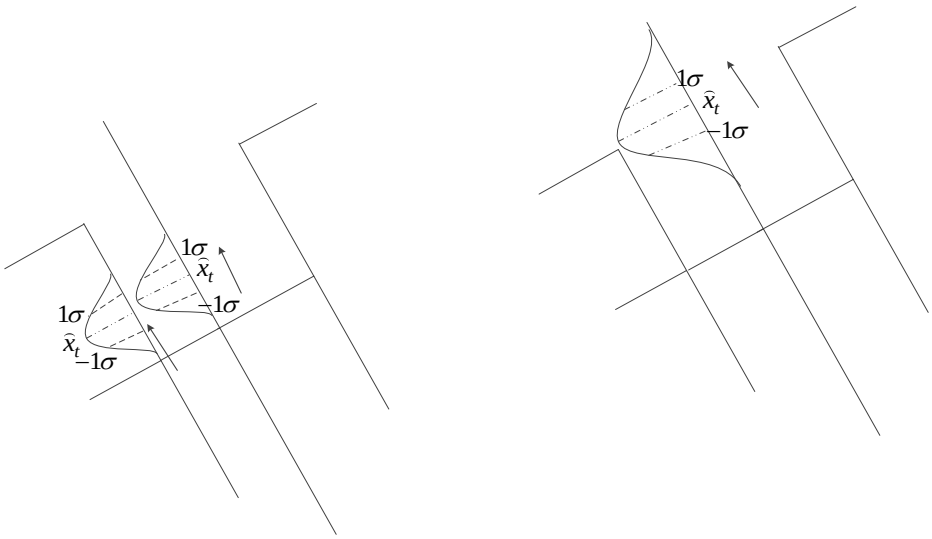
where $\bar{x}$ and σ are, respectively the mean and the standard deviation of the parent distribution. Finally, please note that the case where there is only one corner turning possibility within the truncated parent distribution. In that case it may be considered a degenerate case of splitting and the procedure above holds for this case too, hence it is not represented explicitly in Figure 6.14.

Merging

In the scenario that a corner is detected but there are no corresponding candidate links then a merging operation takes place. The agent and their position distribution are discarded and the existential probabilities of all other agents representing the pedestrian ($i \in n$) are recalculated according to equation 6.22.

$$P_i^{e\prime} = \frac{P_i^e}{\sum_{i=1}^{n} P_i^e} \tag{6.22}$$

A merging operation will also take place in the case where an agent's position exceeds the link length by a margin greater than the size of the truncated position distribution but no straight on link is available and no corner turning movement is detected (see Figure 6.16).



(A) Gaussian component before merging (B) Gaussian component after merging

Figure 6.16: Gaussian component gets merged over the candidate link when next straight link are not observed. (A) Gaussian component before merging and (B) Gaussian component after merging

6.3.5 Updating stride length

The stride length is pseudo-constant – it does not change while the Kalman filter is running but it may be updated at points in the MURE algorithm where the Kalman filter is re-started, for e.g. upon a corner detection.

For a given agent a good estimate of how far they have walked between corner features may be obtained, as this distance is encoded in the map. Combining this distance with the integrated estimates of step count allow us to infer a stride length. However due to the presence of errors in step counting it is pragmatic to update the stride length using a naïve Bayes' filter [147, 254] given by the following equations:

$$\mathrm{d}^{\mathrm{SL}} = \beta * \mathrm{d} + (1 - \beta) * \mathrm{d}' \tag{6.23}$$

$$\mathrm{SL} = \mathrm{d}^{\mathrm{SL}} / \mathrm{SC}' \tag{6.24}$$

Here d is the mapped distance between corner features. d' is the estimated distance travelled by a pedestrian at the point of corner detection. SC' is the estimated number of steps at the point of corner detection. β is a constant and determined experimentally. In this work we used 0.97. That is, heavily weighted towards the mapped distance.

6.4 Experimental setup & results

In order to validate the performance of map aided smartphone based PDR positioning algorithm, a set of experiment was performed over two different indoor settings. The settings were selected in two different buildings of the university campus such that routes followed by pedestrian in one setting was longer than the other as illustrated in Figure 6.17 and Figure 6.18.

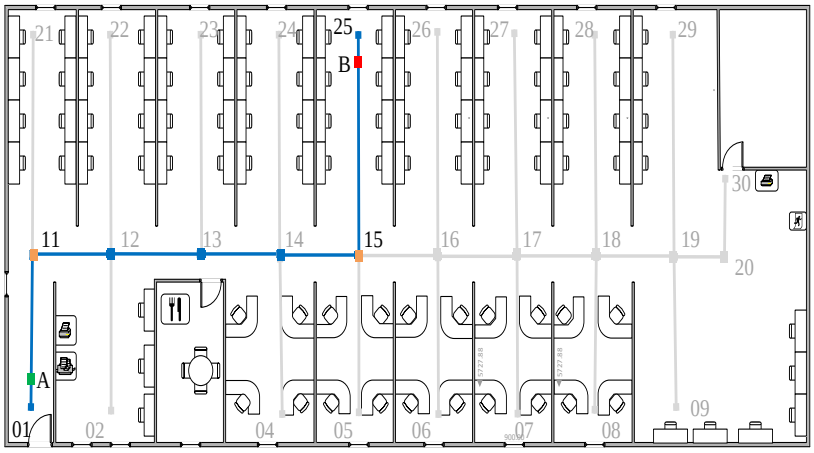


Figure 6.17: Indoor floor plan of the level 3, Building 22 of the university campus. Route followed by the pedestrian is shown in blue line. Pedestrian starts from point A in link 01:11 and stops at point B in link 15:25.The orange squares represent the two corners observed by pedestrian.

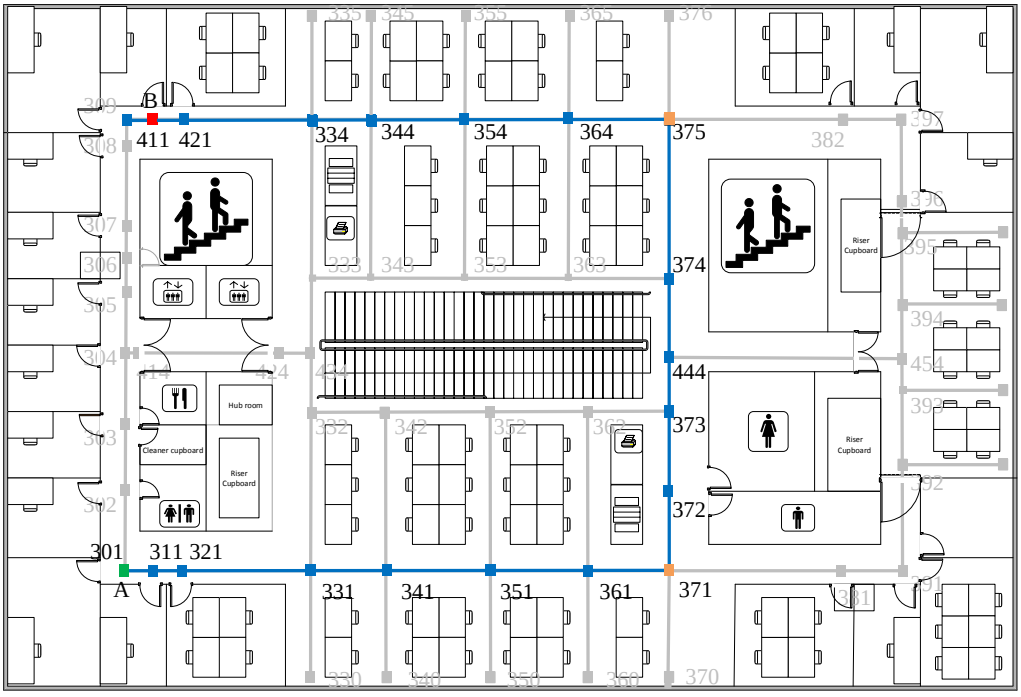


Figure 6.18: Indoor floor plan of the level 4, Building 176 of the university campus. Route followed by the pedestrian is shown in blue line. Pedestrian starts from point A at node 301 and stops on point B at node 411.The orange squares represent the two corners observed by pedestrian.

It was assumed that the starting point was known a-priori. Figure 6.17 represents the indoor floor plan of level 3, Building 22 of the university campus. Pedestrian was asked to start their walk from point A (represented by a green square) along the link 01:11. After walking a length of 3.1 m, pedestrian was further asked to take a right turn towards link 11:12. They walked 13.7 m straight along this pathway. On reaching node 15, the pedestrian was asked to take a left turn towards link 15:25 and to walk along this link until the marked point B (represented by a red square). The measured distance along this link until the marked point B was 7.2 m. Figure 6.18 represents the indoor floor plan of level 4, Building 176 of the university campus. Pedestrian was asked to start their walk from point A (represented by a green square) towards the node 371 along a straight pathway (301-371) of length 21 m. Then at this point pedestrian was asked to take a left turn towards link 371:372. Following this, pedestrian was asked to follow this straight pathway (371-375) until the node 375. The length of this straight pathway (371-375) was measured to be 20.8 m. On reaching the node 375, pedestrian was finally asked to take a left turn towards link 375:364 and to walk until point B (represented by red square) along the straight pathway 375-411. The length of this walked straight pathway was measured to be 19.7 m.

For each setting, volunteer pedestrians were selected (selection criteria and ethics documentation are in Appendix B). Each pedestrian was again given a smartphone – HTCZ710e and Smart Glasses simulator (see Chapter 5, Section 5.2). The experiments were focussed on analysing the two different scenarios as illustrated previously (see Figure 5.3) – smartphone handheld and Smart Glasses simulator. The pedestrians were asked to walk normally and to count the number of steps during the whole walk. The prescribed route was free of obstacles. Table 6.4 contains data about the participants in the experiments. There was no restriction on the type of footwear. Each participant had different footwear. In level 3, Building 22 indoor setting, five of the women wore medium heeled sandals, two of the men wore slippers and the remaining wore sports shoes. While in the other indoor setting, eight of the women wore medium heeled sandals, six of the men wore slippers, three of the men wore boots and the remaining wore sports shoes.

	Level 3, Building 22 indoor setting	**Level 4, Building 176 indoor setting**
Number of subjects	20 (14 men and 6 women)	50 (40 men and 10 women)
Distance	24.0m	61.5m
Age	18 – 40	18 – 40
Weight	45kg – 90 kg	45kg – 100kg
Height	1.55m – 1.81m	1.55m – 1.81m
Repetitions per subject	1	1

Table 6.4: Conditions of the conducted series of experiment.

6.4.1 Step counter

As discussed earlier (see Section 6.3.2), we use the FFT based step rate estimation technique to count the number of footsteps. The footstep detection error statistics are evaluated in terms of the relative mean absolute error (RMAE) percentage via equation 4.11 (see Chapter 4, Section 4.2.1) and 95% confidence level percentage, as illustrated in Table 6.5 and Table 6.6. Table 6.5 illustrates the error characteristics for first indoor setting i.e. level 3, Building 22 and Table 6.6 illustrates the error characteristics for second indoor setting i.e. level 4, Building 176. The step count error for all test subjects during the complete walk is shown in Figure 6.19 and Figure 6.20. The step count error is evaluated by equation 4.10 (see Chapter 4, Section 4.2.1). Figure 6.19 illustrates the step count error histograms of the pedestrians for the first indoor setting. Figure 6.20 illustrates the step count error histograms of the pedestrians for the second indoor setting.

Test Scenarios	**RMAE** (%)	**95% confidence level (%)**
Smartphone handheld	3.10	-3.2 to 9.4
Smart Glasses Simulator	4.12	-4.92 to 13.16

Table 6.5: Comparison of error characteristics for the two scenarios: (A) smartphone handheld and (B) Smart Glasses simulator in the first indoor setting i.e. level 3, Building 22.

Test Scenarios	**RMAE** (%)	**95% confidence level (%)**
Smartphone handheld	1.44	-0.86 to 3.74
Smart Glasses Simulator	1.95	-1.42 to 5.39

Table 6.6: Comparison of error characteristics for the two scenarios: (A) smartphone handheld and (B) Smart Glasses simulator in the second indoor setting i.e. level 4, Building 176.

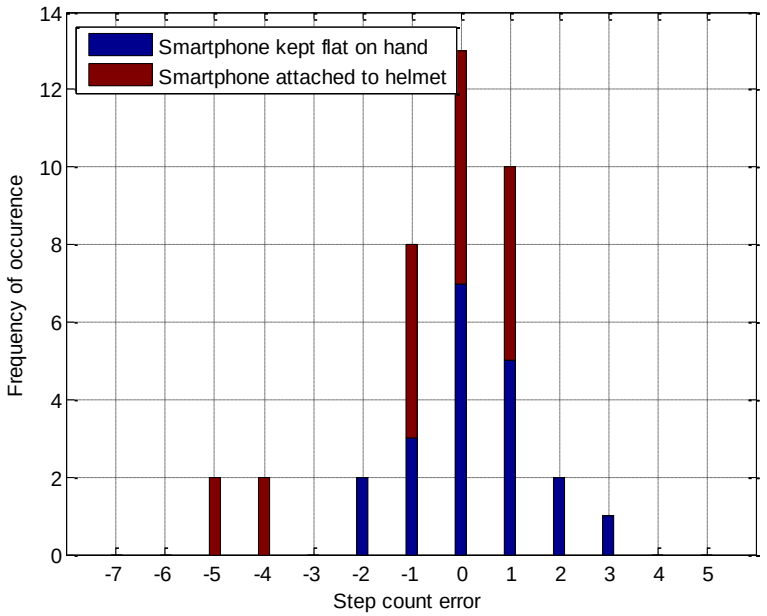


Figure 6.19: Histogram of the step count error for the two scenarios (A) smartphone handheld and (B) Smart Glasses simulator for the first indoor setting i.e. level 3, Building 22.

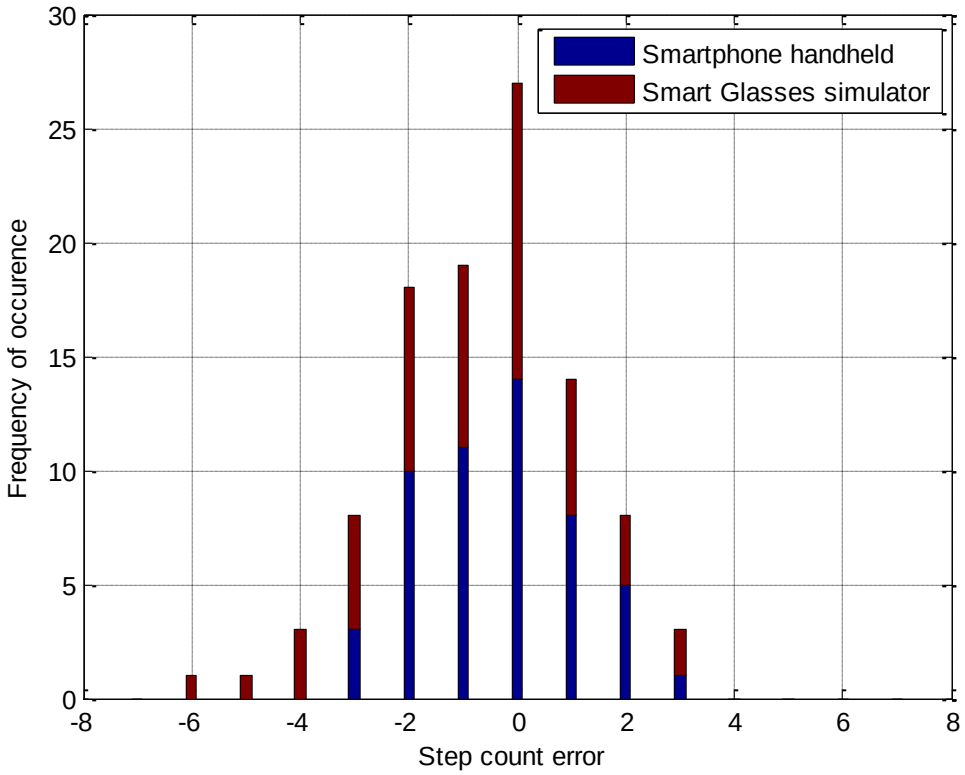


Figure 6.20: Histogram of the step count error for the two scenarios (A) smartphone handheld and (B) Smart Glasses simulator for the second indoor setting i.e. level 4, Building 176.

A similar behavior is observed – as detailed in Chapter 5, Section 5.2.1, the step counter performance is relatively poor for Smart Glasses simulator in comparison to the smartphone handheld. This can be attributed because of the fact that the when a smartphone is placed on hand it is directly attached to the body, and also cautiously as well continuously stabilized by the pedestrian throughout the walk. So, as a result of which the high frequency vibrations and noises get damped. In comparison, the smartphone attached to the Smart Glasses simulator undergoes several events of wobbling throughout the walk. This is likely due to the weight and fit of the helmet that may not be an issue for real smart glasses. As a result, there are several instances which cause the smartphone to move and, to detect false and miss steps. Thus, the step counter accuracy substantially degrades for Smart Glasses simulator.

Furthermore, it is noted from Figure 6.20 that the step count error is greater than 2 steps for a single walk for both scenarios - smartphone handheld and Smart Glasses simulator. This was a special case when a pedestrian was almost doing brisk walking. So, it was expected to have high step count error. As detailed earlier in Chapter 4, Section 4.2.1 when pedestrian moves briskly they shake their body more. As a result more false step detection occurs and subsequently, this causes the performance of the step rate estimation algorithm to be reduced.

The ratio of missing and falsely detecting steps can however vary depending on the different scenarios [139] . Again in this investigation (as earlier, see Chapter 5, Section 5.2.1), the ratio of missing steps is slightly more in comparison to the over estimating of footsteps for Smart Glasses simulator. However the number of pedestrians who have overstepped in the case of Smart Glasses simulator is greater than 20% of the total number of pedestrians in either of the two indoor settings (see Figure 6.19 and Figure 6.20). This behavior is significant. Since, it illustrates the fact that the helmet fit is unstable.

6.4.2 Position error

In this section we present an analysis of the experimental data, looking specifically at the error growth between corner detections. Here we restrict our analysis to the software agents that are on the correct path, regardless of their existential probability. An analysis of existential probability is given in the subsequent Section 6.4.3.

To examine position error growth, the estimated distance travelled was compared to the mapped distance between the starting point of the route and point of interests (POIs), which include each of the corners along the route and the end of the route. In case of the corners, the estimated distance used was the estimation at the point of corner detection but prior to the correction in position being applied by MURE.

Statistics for position error at each POI in both locations were calculated over all participants using equation 5.1 (see Chapter 5, Section 5.2.2). These are visualized in a Box & Whisker Plot [321] in Figure 6.21 and Figure 6.22. The limits of the box indicate the interquartile range with the median marked inside the box. The whiskers indicate the total range, not including outliers, which - if present - are marked with plus symbols. The corresponding numerical data from Figure 6.21 are given in Table 6.7 and the numerical data from Figure 6.22 are given in Table 6.8. The absolute mean error (*ME*) and 95% confidence level taken over all participants and POIs, are given in Table 6.9 and Table 6.10 for the two settings.

The prescribed routes in both settings contain three POIs (see Figure 6.17 and Figure 6.18) – the first corner, the second corner and the end point. In level 3, Building 22 the first corner is at a distance of 3.1 m from starting, the second corner is at a distance of 16.8 m from starting and the end point is at a distance of 24 m from starting. While in level 4, Building 176 the first corner is at a distance of 21.0 m from starting, the second corner is at a distance of 41.8 m from starting and the end point is at a distance of 61.5 m from starting.

It is expected that the position error should gradually reduce with the increase in number of corner features (as opposed to the standard PDR where error increases monotonically), as a result of MURE algorithm. The results depict the same as illustrated by the two figures – Figure 6.21 and Figure 6.22 and the two tables – Table 6.7 and Table 6.8. The results show that position error

drifts gradually from the first corner to the second corner. This behaviour can be observed by the increase in dispersion over 50 percentile region (see Table 6.7 and Table 6.8 for the exact values) from the first corner to the second corner. On the contrary, the drifts in position error decreases with the second corner feature. This behaviour can be observed by the reduction in scatter of 50 percentile region from the second corner to the end point in the two scenarios – smartphone handheld and Smart glasses simulator.

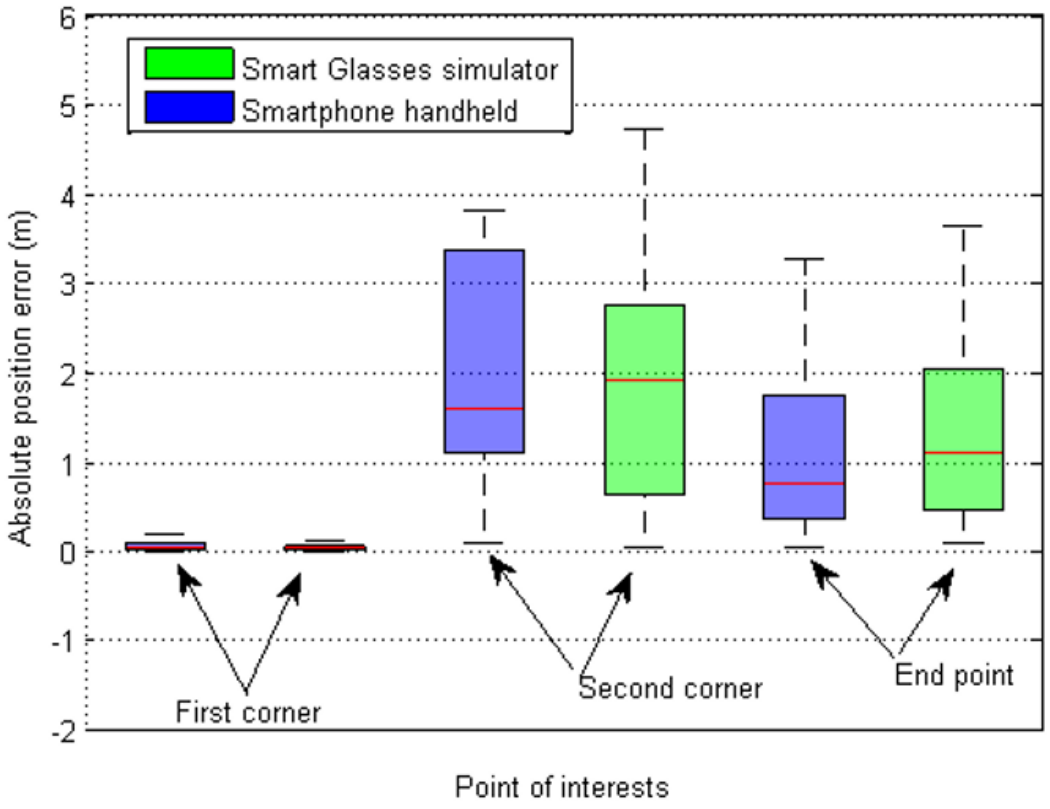


Figure 6.21: Box and whisker plot of the position error at three POIs for the first indoor setting – level 3, Building 22, before Kalman filter is reset.

Values (m)	First corner feature		Second corner feature		End point	
	Smart Glasses simulator	Smartphone hand held	Smart Glasses simulator	Smartphone hand held	Smart Glasses simulator	Smartphone hand held
Maximum	0.13	0.19	4.72	3.81	3.65	3.26
Median	0.05	0.05	1.9	1.6	1.2	0.76
Minimum	0	0	0.06	0.1	0.09	0.04
1st quartile	0.02	0.03	0.65	1.1	0.50	0.36
50th percentile	0.04	0.07	2.51	2.25	1.50	1.38
3rd quartile	0.06	0.1	2.76	3.35	2.0	1.74

Table 6.7: Comparison of the position error for the two scenarios (A) Smartphone handheld scenario and (B) Smart Glasses simulator scenario at three POIs for the first indoor setting – level 3, Building 22, before Kalman filter is reset.

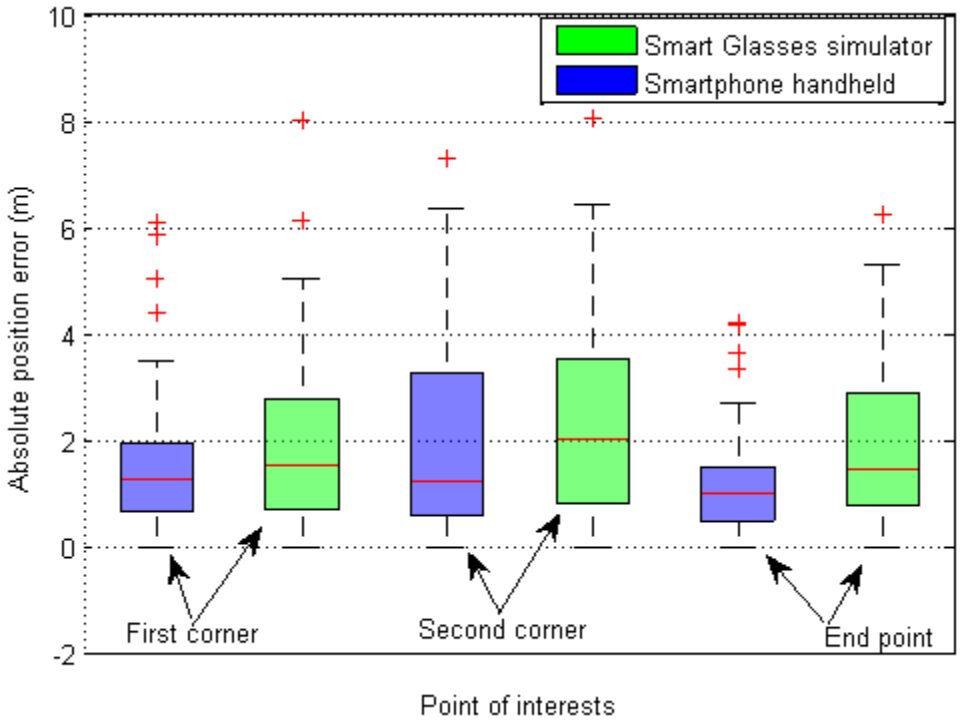


Figure 6.22: Box and whisker plot of the position error at three POIs for the second indoor setting – level 4, Building 176, before Kalman filter is reset.

Values (m)	First corner feature		Second corner feature		End point	
	Smart Glasses simulator	Smartphone hand held	Smart Glasses simulator	Smartphone hand held	Smart Glasses simulator	Smartphone hand held
Maximum	8	6.1	8.04	7.29	6.24	4.2
Median	1.54	1.27	2.03	1.26	1.45	1.0
Minimum	0	0	0	0	0	0
1st quartile	0.7	0.69	0.64	0.6	0.8	0.5
50th percentile	2.1	1.25	2.88	2.66	2.1	1.0
3rd quartile	2.8	1.94	3.52	3.26	2.9	1.5

Table 6.8: Comparison of the position error for the two scenarios (A) Smartphone handheld scenario and (B) Smart Glasses simulator scenario at three POIs for the second indoor setting – level 4, Building 176, before Kalman filter is reset.

Also notice how in Figure 6.21 the size of the error roughly corresponds with the distance between POIs (in Figure 6.22 the distance between POIs is approximately constant), thus the growth in position error is weakly bounded by the maximum distance between corner features in the map. On comparing the d_{err} at end points over the two scenarios, it is observed that the mean absolute position error is less than 2.1 m in either of the two scenarios for the two indoor settings. However, the 95% confidence level is significantly less for the smartphone handheld scenario (see Table 6.9 and Table 6.10). This behaviour is likely to happen, as mentioned earlier (see Chapter 5, Section 5.2.2) due to the fact that, pedestrian continuously tries to stabilize his hand during the walk. As a result, the hand held smartphone actually yield more reliable step length estimates than its head-mounted counterpart. The mean absolute position error (MAE) is computed according to equation 5.2 (see Chapter 5, Section 5.2.2).

Scenarios	MAE **(m)**	**95% confidence level (m)**
Smartphone handheld	1.16	-3.06 to 3.14
Smart Glasses simulator	1.36	-4.1 to 2.6

Table 6.9: Error characteristics at the end point for the two scenarios (A) Smartphone handheld scenario and (B) Smart Glasses simulator scenario for the first indoor setting i.e. level 3, Building 22.

Scenarios	**MAE (m)**	**95% confidence level (m)**
Smartphone handheld	1.25	-3.16 to 3.32
Smart Glasses simulator	2.05	-5.20 to 5.60

Table 6.10: Error characteristics at the end point for the two scenarios (A) Smartphone handheld scenario and (B) Smart Glasses simulator scenario for the second indoor setting i.e. level 4, Building 176.

6.4.3 Map matching evaluation

In this section the *splitting* and *merging* operations of the MURE algorithm are illustrated in Figure 6.23 and Figure 6.24. Each visualize the entire journey for a single experimental run by a participant in – respectively - the building 22 map and the building 176 map. At two second intervals the position of each agent representing the pedestrian is plotted as a point indicating the mean position and an ellipse visualising the 1σ uncertainty in position. In the walk visualised in Figure 6.23 there is a single splitting operation. In the walk visualised in Figure 6.24 there are two splitting operations and two merging operations.

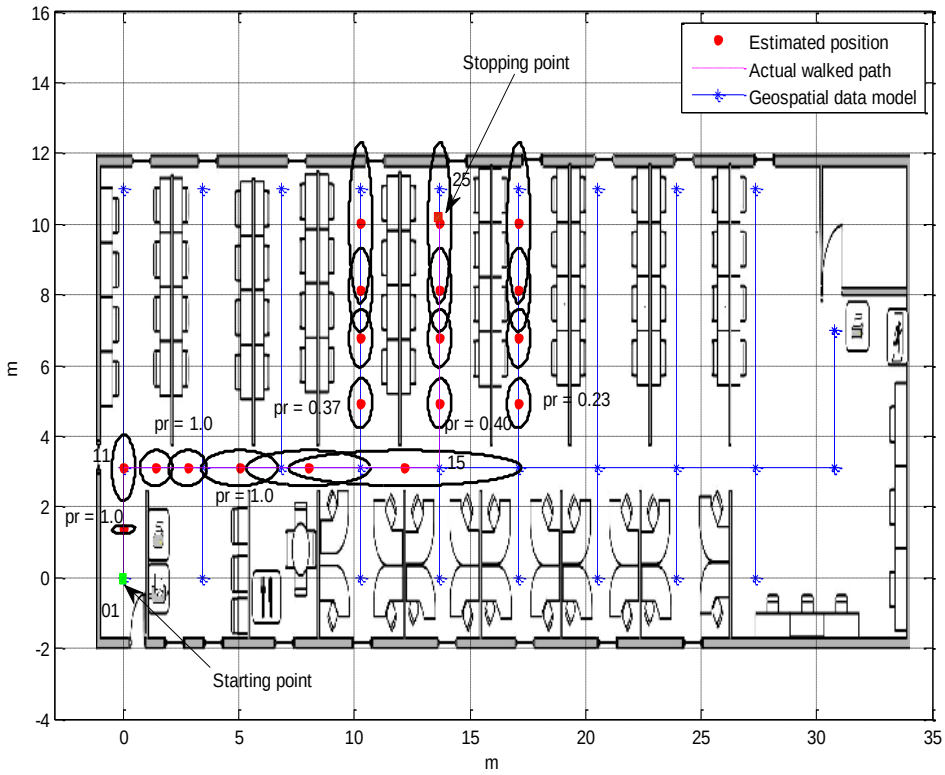
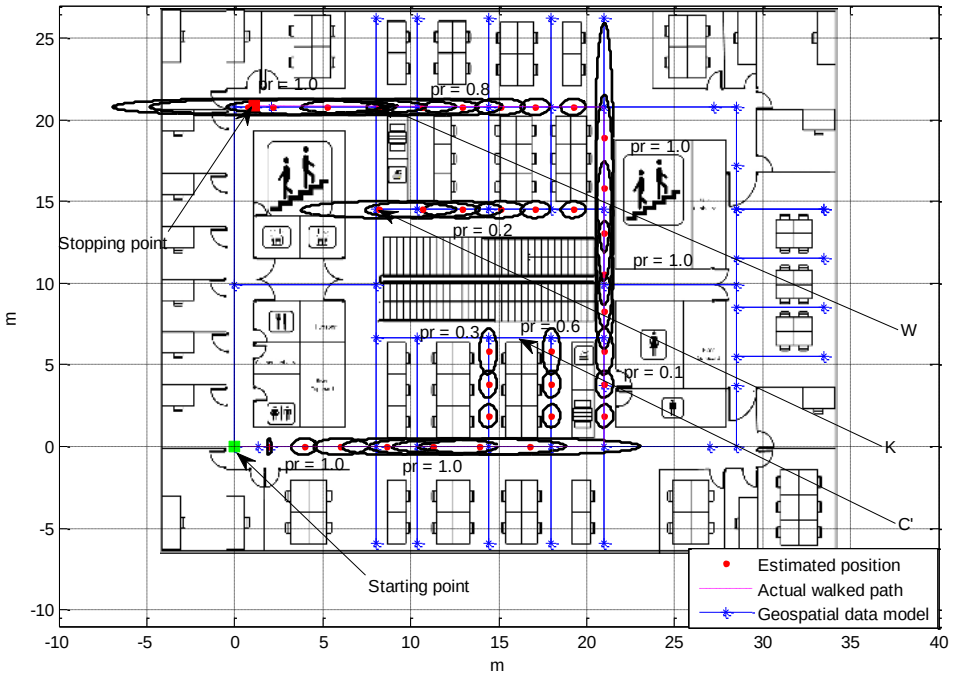
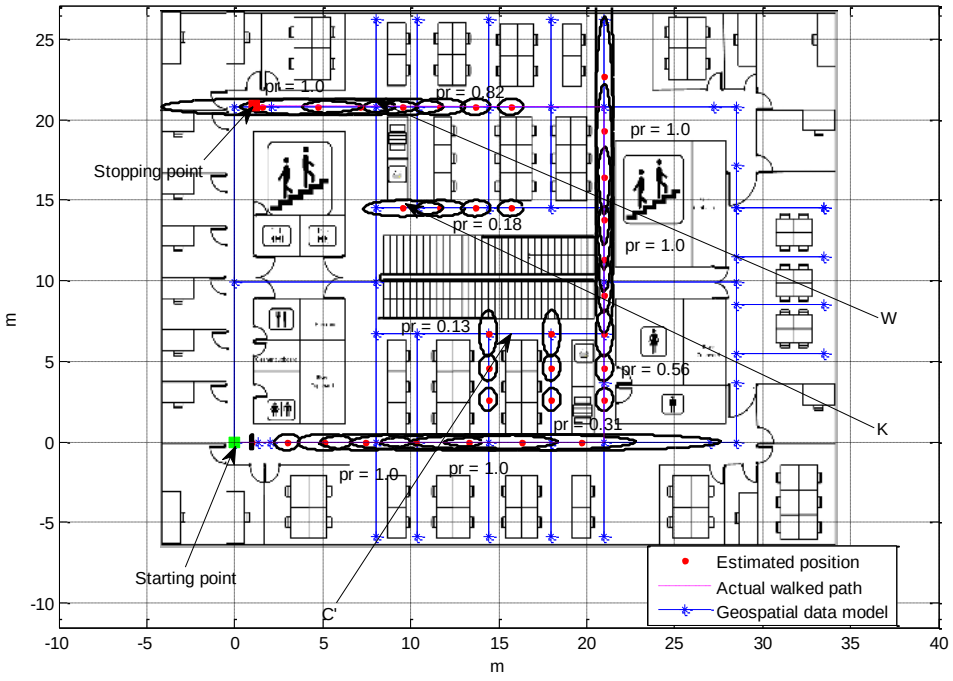


Figure 6.23: Trajectory of a pedestrian when the smartphone was handheld. Pedestrian starts at a point represented by green square along link 01:11 and stops at a point represented by red square in link 15:25. They walk along the path shown in magenta dashed line. The red dots represent the mean positions at lapse of every 2s. The black ellipses around every red dot represent the 1σ uncertainty region associated with the estimated pedestrian's position.



(A)



(B)

Figure 6.24: Trajectory of a pedestrian for the two scenarios: (A) smartphone handheld and (B) Smart Glasses simulator. Pedestrian starts at a point represented by green square and stop at a point represented by red square. They walk along the path shown in magenta dashed line. The

red dots represent the mean positions at lapse of every 2s. The black ellipses around every red dot represent the 1σ uncertainty region associated with the estimated pedestrian's position.

With reference to Figure 6.23 at the start of the walk the pedestrian is represented by a single agent with existential probability $p^e = 1$. Notice how around the first right hand turn the size of the 1σ region of uncertainty shrinks due to the corner detection. However, because there is no ambiguity over which link the pedestrian has taken splitting does not occur. The uncertainty grows as the pedestrian proceeds down the corridor and at node 15 the pedestrian takes a left turn. However due to the size of the region of uncertainty in position it is ambiguous whether the pedestrian turned left at node 14, 15 or 16. Thus there is a splitting operation and three new agents are created located on each of the three candidate links. In this case the existential probability of the true link (15:25) is $p^e = 0.4$. The other two links have $p^e = 0.37$ and $p^e = 0.23$ respectively. At the end of the walk all three agents are still active so the final location of the pedestrian is ambiguous.

The Figure 6.24 depicts the trajectory of a pedestrian in the second indoor setting – level 4, Building 176 for the two scenarios (A) Smartphone handheld and (B) Smart Glasses simulator. It illustrates both splitting and merging operations. Initially the pedestrian is represented by a single agent with existential probability $p^e = 1$. The pedestrian makes a left turn at node 371, but due to ambiguity in position there is a splitting operation resulting in three new software agents. The agent on the true link (371:372) has existential probability $p^e = 0.1$, which is in fact the *lowest* value of all three agents but still the algorithm traces the pedestrian. This emphasizes the importance of the MURE strategy of tracking multiple uncertain routes simultaneously.

At the point in Figure 6.24 labelled C' two of the extant agents encounter a T-junction topology where a corner turning motion must be made to proceed. However as no corner turning is detected – the pedestrian is walking straight on – these two agents are merged. The existential probability of the single remaining agent is now $p^e = 1$.

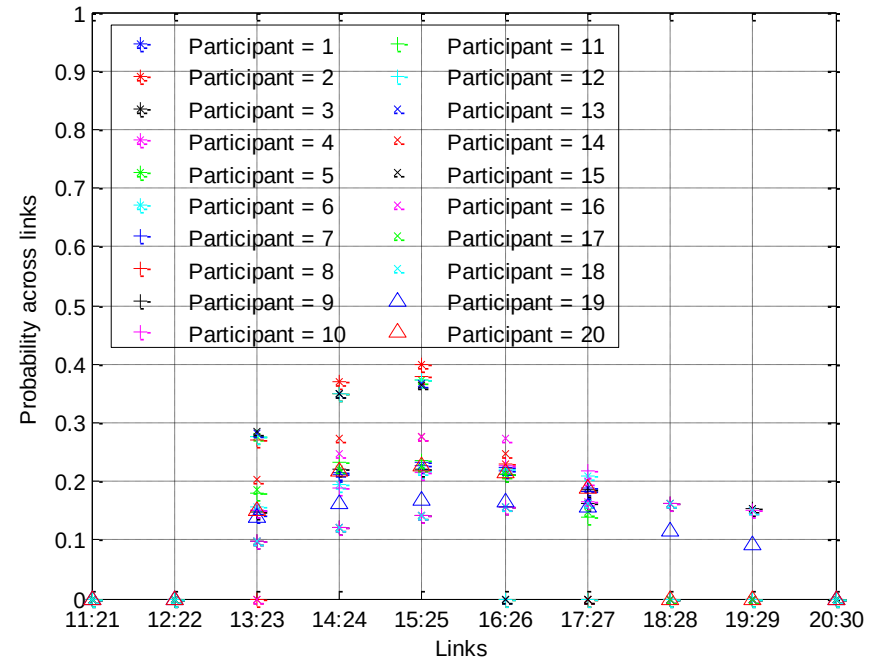
A similar scenario plays out when the pedestrian next makes a left turn at node 375. Here a splitting operation generates two new agents. When one of the agents arrives at the point labelled K in Figure 6.24 there is a T-junction but no turn is detected hence this agent is merged into the remaining single agent at point W. At the end of the walk there is a single agent with $p^e = 1$.

The two journeys described in Figure 6.23 and Figure 6.24 are single runs of a single participant and each of the other runs by different participant had different splitting and merging behaviours. To capture some of this variations Figure 6.25 and Figure 6.26 illustrates the existential probabilities along different links over different participants. Figure 6.25 illustrates sample values of existential probability relating to the left turn made by pedestrians at node 15 in Figure 6.17. The values of existential probability for agents that travelled along the true link

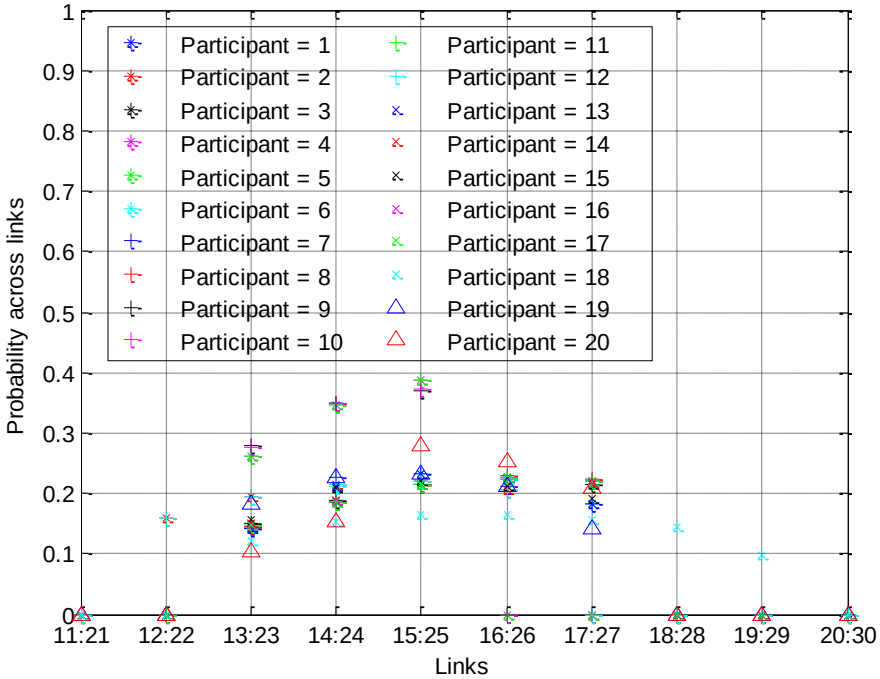
(15:25) are shown in (approximately) the middle of this plot and the values for agents that travelled down other adjacent links (e.g. 16:26) are shown arranged along the x-axis in their topologically relative position.

Figure 6.26 shows exactly the same thing for the left turn that occurs at node 371 in Figure 6.18 (plot A: smartphone handheld and plot B: Smart Glasses simulator) and the left turn that occurs at node 375 in Figure 6.18 (plot C: smartphone handheld and plot D: Smart Glasses simulator). Over all these tests, there was always an agent with some existential probability on the correct link, that is, tracking was never lost. The existential probability is highest along the actual walked link (after the splitting takes place on detection of a corner feature) in more than 70% of pedestrian walks (runs) whether the smartphone was handheld or attached to a Smart Glasses simulator i.e. link 15:25 for the first indoor setting and link 371:372 after splitting takes place at the first corner for the second indoor setting and link 375:364 after splitting takes place at the second corner for the second indoor setting.

There is some evidence of systematic error in the PDR algorithm because in 95% of cases the estimated distance travelled was underestimated at a corner detection. This implies that either the stride length algorithm or the step rate algorithms (or both) are slightly underestimating. Also, it is observed that although the splitting of Gaussian component takes place at 1σ the existential probabilities can potentially cover up all the nearby links including the actual walked link. This behaviour is significant. Since it ascertain the fact that 1σ cut off is sufficient enough for selecting the actual walked link after the detection of corner feature and projecting the pedestrian's position onto multiple candidate links.



(A)



(B)

Figure 6.25: Variation of existential probabilities along different links after splitting takes place on detection of a corner feature (second i.e. node '15', see Figure 6.17 for details) for 20 pedestrial walks (runs) in the first indoor setting i.e. level 3, Building 22 for two scenarios (A) Smart Glasses simulator and (B) smartphone handheld.

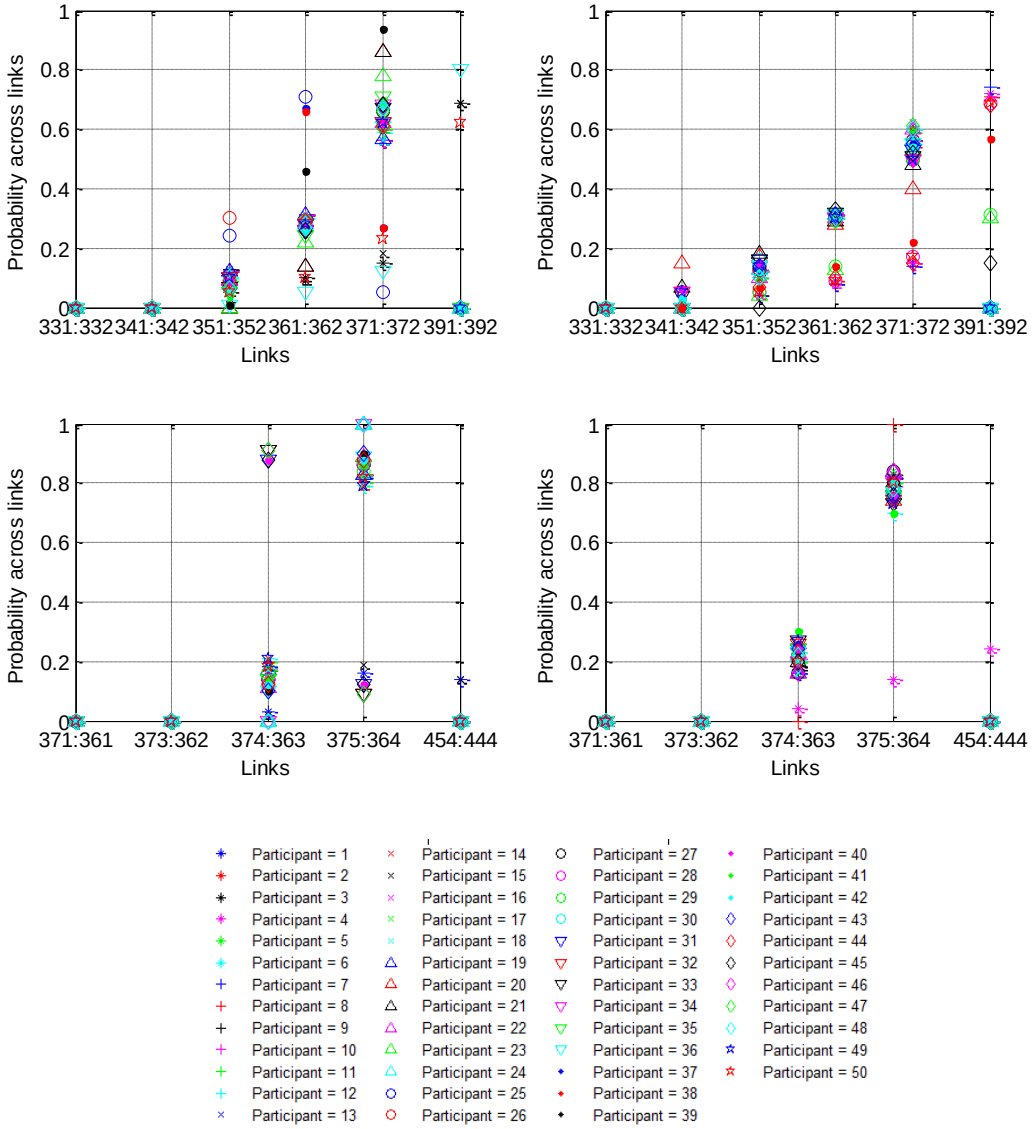


Figure 6.26: Variation of existential probabilities along different links after splitting takes place on detection of a corner feature – node '371' and node '375' (see Figure 6.18 for details) for 50 pedestrial walks (runs) in the second indoor setting i.e. level 4, Building 176. The headings (A) and (B) represent the variation of existential probabilities for the two scenarios – (A) smartphone handheld and (B) Smart Glasses simulator along the first corner – node '371'. While the headings (C) and (D) represent the variation of existential probabilities for the two scenarios – (C) smartphone handheld and (D) Smart Glasses simulator along the second corner – node '375'.

We can extract an overall metric for the performance of the MURE algorithm by calculating the average existential probability over the sampled positions of agents that are located on the

correct path for a given run. This metric – the *map matching ratio* – was calculated for each experimental run according to the following equation.

$$\text{Map matching ratio} = \frac{\sum_{i=1}^{M} pr_i}{M} \quad (6.25)$$

where pr_i is the estimated existential probability of the pedestrian on the Mth walked pathway. The minimum and maximum values recorded in each setting are given in Table 6.11.

Values	**Smartphone handheld**		**Smart Glasses simulator**	
	1st indoor setting	2nd indoor setting	1st indoor setting	2nd indoor setting
Minimum map matching ratio	0.73	0.85	0.72	0.80
Maximum map matching ratio	0.81	0.94	0.8	0.92

Table 6.11: Map matching ratio for the two scenarios – smartphone handheld and Smart Glasses simulator in the two indoor settings – first indoor setting (level 3, Building 22) and second indoor setting (level 4, Building 176).

It can be observed that the map matching ratio is almost similar in both the scenarios – smartphone handheld or Smart Glasses simulator, as expected. However, the map matching ratio is slightly better in the second indoor setting in comparison to the first indoor setting even though this involved a longer walk in a more complex map. This can be attributed because of the occurrence of two key map matching characteristics – splitting and merging in the second indoor setting. The results confirm the efficacy of MURE map matching algorithm i.e. it can provide a logical solution to successfully estimate the position of a pedestrian and locate the pathways taken by a pedestrian, provided the pathways are described as links and nodes.

6.5 Summary

In this chapter, we focussed on the usage of 'maps' to counter the positioning drifts (reported in the earlier chapter, see Chapter 4). A map matching technique – MURE, has been designed in this context (based on our knowledge and understanding) and tested indoors. It is different from the previous map matching techniques that snap the estimated position to the nearest pathways based on the proximity of the trajectory to the elements of the network map [280, 307]. The basic idea of MURE map matching algorithm is based on the linear Kalman filter but extends the representation of a pedestrian's position to support multiple discrete Gaussian probability distributions at different points in the network map, where each distribution has an existential probability and all existential probabilities sum to unity.

Based on nodes and links defined in a network map, MURE map matching initially searches for the turning corner feature. On detection of a turning corner feature, the Gaussian

distribution gets split into candidate links, provided all nodes are within truncated region of the current position distribution. Following this at a later point in the journey one of these distributions may be ruled out – for e.g. by detecting a corner where none is available. In this case the distribution is merged, meaning that it is discarded and the weight of its existential probability is distributed among the remaining distributions. This process of splitting and merging continues throughout the journey.

An important shortcoming reported in the previous chapter (see Chapter 4) was the dependence on the heuristic parameter 'k' for the estimation of the stride lengths. So we formulated this problem in a linear perspective on the data fusion part. Specifically, we integrated the information from Weinberg stride length model and information from geospatial data model (see Section 6.3.4.1) using Kalman filter (see Section 6.3.3 and Section 6.3.4.2). We have again investigated two scenarios (as earlier detailed in the previous chapter, see Chapter 5) – smartphone handheld and Smart Glasses simulator over the two indoor settings – level 3, Building 22 and level 4, Building 176. The main findings of this chapter are as follows:

- As earlier (see Chapter 5, Section 5.2.1), the step count error for a Smart Glasses simulator is relatively high in comparison to the smartphone handheld scenario (see Figure 6.19 and Figure 6.20) for the two indoor settings. This can be attributed because of the fact that they don't experience the damping effect means they experience high frequency vibration and noises. This can make signal processing activity like step detection more difficult. As a result performance of step counter algorithm degrades.
- The mean absolute position error is less than 1.26 m for smartphone handheld scenario where as it is less than 2.1 m in the smart glasses simulator scenario for the two indoor settings. However, 95% confidence level is relatively less when the smartphone is handheld. This behaviour is likely to happen because the pedestrian continuously tries to stabilize his hand during the walk. As a result, the hand held smartphone actually yield more reliable step length estimates than its helmet mounted counterpart in a Smart Glasses simulator.
- The minimum map matching ratio is greater than 0.70 for both the scenarios – smartphone handheld or Smart Glasses simulator in the two indoor settings. This behaviour is significant. Since, it ascertains that the path estimated by MURE map matching algorithm is actually the true path in more than 70% of the pedestrial walks. Moreover, in the second indoor setting the map matching ratio is slightly higher in comparison to the first indoor setting. This can be attributed because of the occurrence of two key map matching characteristics – splitting and merging in the second indoor setting.

These results have a substantial practical significance since the estimated level of positioning performance would certainly be useful in situations like emergency operations, search and rescue operations, indoor tracking, etc. However, there are some limitations. These are as follows:

- Path has less complexities: The selected path in the two indoor settings have less complexities. Neither the selected path have obstacles (such as pedestrians) nor are staircases considered. Moreover, the selected paths have only 90^0 turns. Considering the fact that the indoor geometry is complex, it is possible that the turns are less than 90^0. Or there are several intersecting turns.
- Gaussian component is arbitrarily cut off at 1σ: The Gaussian distribution associated with pedestrian's position is arbitrarily truncated at $\pm 1\sigma$ on detection of the corner feature in MURE map matching algorithm. Although, it is assumed that truncation at $\pm 1\sigma$ can suffice the purpose, considering the fact that existential probabilities may grow as pedestrian traverses over the network. However, there is a trade-off between this truncation and how aggressively modelled error grows (governed by Q and R in the Kalman filter).

Additionally, it is also possible that there are larger gaps between corner features. This can potentially result in the growth of position errors. Therefore, in order to reduce the growth of these errors we have investigated the performance of our MURE map matching algorithm by placing the artificial landmarks at some distances. The will be discussed in the next chapter.

7. Chapter 7: Landmark aided PDR

The previous chapters have given us the building blocks for a low cost positioning system suitable for use in an infrastructure free indoor environment by a pedestrian. In particular, the Chapter 3 investigated whether it is possible to position an individual (toy vehicle) on an infrastructure free environment by means of a case study, carried on an artificial test bed – Scalextric. The Chapter 4 focussed on analysing the inertial sensor measurements recorded from smartphone sensors to detect the occurrence of footsteps and indirectly estimating stride lengths of pedestrians. A significant drawback of the proposed stride length models was the presence of drift errors and the dependence of heuristic parameter 'k'. Also, the position of smartphone was dictated to the pedestrians throughout the experiments to keep it straight and flat such that the orientation of pedestrian relative to the smartphone is static. However, at times this was not true because of sudden obstacles or glitches. Therefore, a more realistic scenario was investigated by means of Smart Glasses simulator in the Chapter 5 to consider this assumption valid. The other shortcomings were dealt in Chapter 6. Particularly, the developed smartphone based PDR positioning system was aided with maps in Chapter 6 to reduce the drift errors and dependence on heuristic parameter 'k'.

Experimental results showed interesting performance (For the smartphone handheld scenario, the mean absolute position error was 1.25 m with 95% confidence level in the region -3.16 to 3.32 m at end points. On the contrary, when the smartphone was attached to the Smart Glasses simulator, mean absolute position error was 2.05 m with 95% confidence level in the region -5.20 to 5.60 m at end point), considering the corners are separated by a distance of approximately 21 m. However, it is still possible that the corners are further far apart. This may cause the positioning errors to further grow.

Therefore this chapter provides a reference implementation of the artificial landmarks – Quick response (QR) codes placed at fixed distance apart (approximately 20 m) to reduce these positioning errors by means of proximity detection method (see Chapter 2, Section 2.2.1.1); while utilising the previous indoor setting for this investigation (level 4, Building 176). It is to be noted that although QR codes are not actually desirable in an actual application, it is here used as a proof of concept i.e. growth in position errors can substantially be reduced by utilising the landmarks (for e.g. QR codes) commonly present in the environment in addition to the corner features provided the features (QR code landmarks and corner features) are present at close distances (approximately 10 m). From the literature review (see Chapter 2, Section 2.2.5) it is clear that [177], *"Landmarks are stationary, distinct, and salient objects or places, which serve as cues for structuring and building a mental representation of the surrounding area. Any object can be perceived as a landmark if it is unique enough in comparison to the adjacent items."*

Two key advantages of the landmark based positioning in perspective of the pedestrians are as follows [322, 323]:

1) It makes the pedestrian sure that they are on the correct route, since they can see the very landmark which may be a part of their navigational instruction.
2) It provides an attractive means of self-orientation. Particularly at decision points, these landmarks play a vital role in orienting the pedestrian to the right direction.

The basic principles of map aided PDR also apply to the landmark aided PDR [165]. However, it differs from the map aided PDR in the sense that maps can only serve to constraint the pedestrian geometrically whereas landmarks can potentially serve as a reference point in case when estimating the relative position of pedestrian [324]. To begin with, the chapter details about the designed smartphone based PDR positioning system aided with the maps and landmarks in Section 7.1. Section 7.2 presents the experimental setup and results obtained from the smartphone based PDR system aided with map and landmarks. Finally, Section 7.3 summarizes the key findings of this chapter.

7.1 Smartphone based PDR system aided with map and landmarks

The block diagram of smartphone based PDR positioning system aided with map and landmark database is detailed in Figure 7.1. It is composed of seven parts: 1) Building a geospatial data model 2) A feature (corner) detection algorithm 3) The MURE algorithm 4) Smartphone based PDR system 5) Updating stride lengths 6) Landmark database and 7) Landmark matching. The accelerometer and gyroscope samples are again filtered according to equation 4.1 (see Chapter 4, Section 4.1). The filtered accelerometer samples are fed to the smartphone based PDR system. The filtered gyroscope samples are used to detect the corner features. The implicit position information from PDR system, the geospatial data model; represented in a form of topological map and feature (corner) detection module are fused together via MURE algorithm to estimate the relative position of the pedestrian. Further corrections are then applied by the stored landmark database using landmark matching algorithm to get the refined position estimate. In case corner features are detected the stride lengths are updated, and the information is fused by the MURE algorithm.

As illustrated above, the basic working of modules associated with the smartphone based PDR positioning system aided with map and landmark database is similar to the previous chapter (see Chapter 6, Section 6.3) except addition of two new modules namely landmark database and landmark matching algorithm. Therefore, the following subsections will describe the working of these two new modules in detail. For the working of other modules refer Chapter 6, Section 6.3.

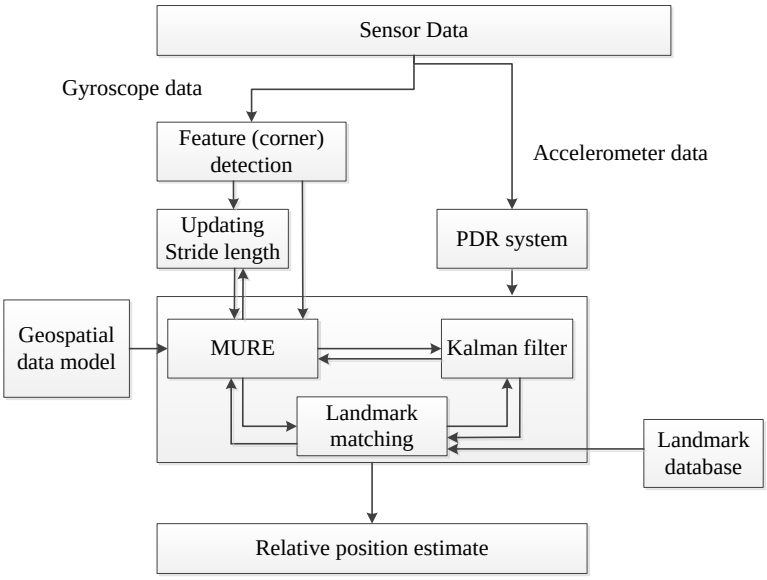


Figure 7.1: Block diagram of the smartphone based PDR positioning system aided with map and landmark.

7.1.1 Landmark based positioning

Over the past few decades landmark based positioning has been significantly used by pedestrians to self-orient themselves and to navigate through different (indoor and outdoor) environments. However, most of the prior works have primarily focussed on using landmarks as direction cues [325, 326]. This landmark based positioning technique uses landmarks, particularly artificial landmarks in the form of QR codes to not only assure the pedestrian about the correct pathway to the destination but also improve the positioning accuracy of the positioning system by calibrating the positioning system on sensing proximity to landmarks. The novelty of this approach is that it utilizes the human as a sensor to detect proximity to the landmarks. There are two main paradigms of the landmark based positioning. These are as follows:

1. Landmark selection and building landmark database
2. Landmark matching algorithm

7.1.1.1 Landmark selection and building landmark database

The basic criteria of visualising a landmark is that, in principle it should ensure that it is able to distinguish itself from nearby objects [177]. In general, a landmark's salience is considered to result from visual, semantic and structural properties, accompanied by its advance visibility [327, 328]. However, it is still an open question, which features of an object are the most essential to assess its suitability for serving as a landmark [329]. Our landmark is selected and designed in form of 2D codes – Quick response (QR) codes (see Figure 7.2) based on these four features [166, 179]:

- Visual clarity: The landmark is designed in form of a 2D code such that it appears visually distinct to the passer-by from the nearby objects. Special consideration is given to its façade area (contrast to nearby surroundings) and shape.
- Human friendliness: The data is encoded in form of 2D code following ISO/IEC 18004 standard [330]. It is one of the most widely used 2D codes in the world that can be read even by ordinary mobile phones.
- Robustness: The QR codes are robust. Primarily, these can sustain damage and can continue to function even when a part of the QR code image is obscured.
- High productivity: The QR codes can be effectively generated by numerous software or services available freely (no cost incurred) online. These can be printed even by the lowest quality printers. So potentially anyone can make this landmark code.

Figure 7.2: An example of the landmark code – QR code. It consist the following information: (a) link in which it is placed: 331:341 (provided the building floor plan is represented in form of network map) (b) previous interconnecting link: 321:331 (c) next interconnecting link: 341:351 (d) relative position of the QR code from the starting point: 9.0 m and (e) a unique QR code identifier number: LW3001

A QR code is a two-dimensional machine readable symbol consisting of an array of black and white squares (see Figure 7.2). It was invented in 1994 by Denso Wave Corporation [331]. The primary advantage of QR codes in comparison to the other linear symbols is that it can store high density of data (approximately 100 times more) while requiring less processing (reading) time [331].

These designed QR codes are pasted onto the wall and pillar of the building at measured distances at a height of 1.2 m from the ground. The whole conceptual framework of QR codes is stored in form of Extensible Markup Language (XML) schema as illustrated in Figure 7.3, which

can be represented diagrammatically in Figure 7.4. Here the QR code is placed on link 331:341, the next interconnecting link is 341:351 and the previous interconnecting link is 321:331.

Each QR code consists of the following information: a unique identification number (IdNumber), link in which it is placed (mLink), previous interconnecting link (pLink), next interconnecting link (nLink) and the relative position of QR code from the starting point (relPos).

```
<QRFeature number="LW3001">
        <pLink>321:331</pLink>
        <mLink>331:341</mLink>
        <nLink>341:351</nLink>
        <relPos>9.0</relPos>
</QRFeature>
```

Figure 7.3: XML Schema for a particular QR code

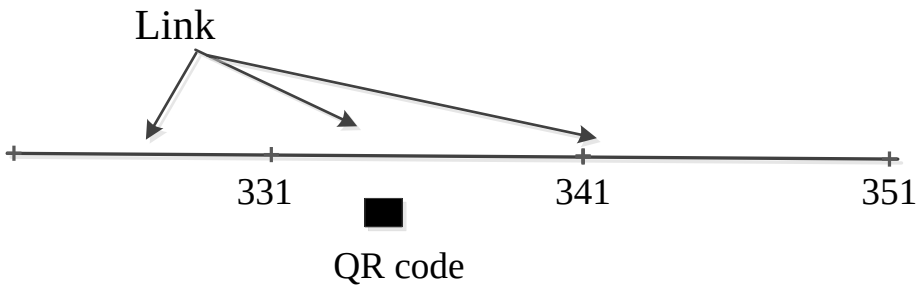


Figure 7.4: Diagrammatic representation of the setup of a particular QR code – LW30001, refer Figure 7.2.

7.1.1.2 Landmark matching algorithm

The landmark matching algorithm is an extension to the MURE algorithm. The MURE framework in principle, detect corner features to deal with potential ambiguity amongst the links a pedestrian has taken from a given node. Uncertainty may arise in scenarios where there are, for e.g. two nodes with available left or right turns close together and in this case it makes sense to split the pedestrian's position into discrete distributions one on each candidate link. Furthermore at a later point in the journey one of these distributions are ruled out – for e.g. by detecting a corner feature where none is available. In this case the distribution is merged. However, if the corners are potentially absent or present over the long distances the ambiguity in distributions may persist. In that case it makes sense to use the landmarks e.g. QR code to rule out the distributions i.e. enable merging.

A process flow diagram for the algorithm is illustrated in Figure 7.5. A map of landmarks is stored in a database. So when a pedestrian passes through these QR codes (placed at requisite distances from the starting point) the entries are updated by clicking the TimeStamp Button and

the following QR code identifier number button (see Figure 7.6). This registers the point of time a pedestrian observes a particular code. Once the entries are updated, the time of occurrence of these landmarks is compared with the parallel time running in the smartphone. If it matches, the position error is reset by updating the stride lengths and existential probabilities are merged.

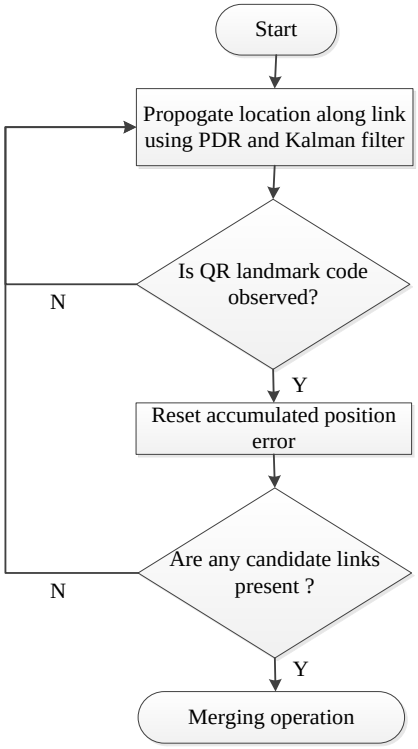


Figure 7.5: Process flow diagram for the landmark matching algorithm.

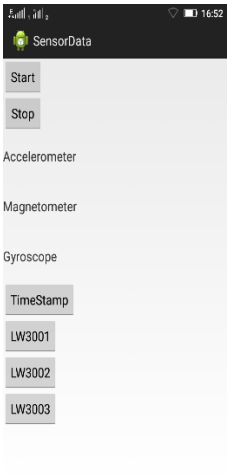


Figure 7.6: Screenshot of the modified SensorData application

Reset the accumulated position error

The accumulated position error is reset by updating stride lengths in addition the position distribution is truncated to ± 0.7 m on observation of a QR code. Again this parameter is designer

specific and depends on the response speed of the pedestrian. This re-initialization of the standard deviation accomplishes two things:

- It is the 'map-matching' element of the MURE algorithm, reflecting the fact that we know that the pedestrian is near a landmark and can reduce the error accordingly.
- It serves as a direction cue to the pedestrian, confirming the pedestrian that they are on the true path

In practise, it is equally possible that when a pedestrian clicks a timestamp button on observation of a particular QR code they may be slightly ahead, behind or just opposite to the QR code. For a given agent a good estimate of how far they have walked between landmarks may be obtained, as this distance is encoded in the landmark database. Combining this distance with the integrated estimates of step count allow us to infer a stride length. However due to the presence of errors in step counting it is pragmatic to update the stride length using a naïve Bayes filter given by the following equations [254]:

$$\mathrm{d}^{\mathrm{SL}} = \beta * \mathrm{d} + (1-\beta) * \mathrm{d}' \tag{7.1}$$

$$\mathrm{SL} = \mathrm{d}^{\mathrm{SL}}/\mathrm{SC}' \tag{7.2}$$

Here d is the mapped distance from origin of journey to landmarks. d' is the estimated distance travelled by a pedestrian at the point of landmark observation. $\mathrm{SC}'$is the estimated number of steps at the point of landmark observation. β is a constant and equal to 0.97. That is, heavily weighted towards the mapped distance.

Merging

In the case when landmarks are observed, the agent and their position distribution are discarded and the existential probabilities of all other agents representing the pedestrian ($i \in n$) are recalculated according to the equation 7.3.

$$\mathrm{P}_{\mathrm{i}}^{\mathrm{e}\prime} = \frac{\mathrm{P}_{\mathrm{i}}^{\mathrm{e}}}{\sum_{\mathrm{i}=1}^{\mathrm{n}} \mathrm{P}_{\mathrm{i}}^{\mathrm{e}}} \tag{7.3}$$

where $\mathrm{P}_{\mathrm{i}}^{\mathrm{e}}$ is the existential probability of the parent distribution.

7.2 Experimental setup & results

In order to validate the performance of map aided, and artificial landmark aided smartphone based PDR positioning algorithm a set of experiment was performed. We selected the same indoor setting (particularly second indoor setting, considering the corner features were distantly separated – approximately 21 m) and the same volunteer participants were approached as in the previous chapter (see Chapter 6, Section 6.4). Refer Figure 7.7 for details about the indoor setting. The path was however inlaid with QR codes this time. Participants were again asked to walk on a typical route – 301-371-375-411. Pedestrian started walking from point A (represented by a green square) towards the node 371 along a straight pathway (301-371) of length 21 m. Then at

this point pedestrian was asked to take a left turn towards link 371:372. Following this, pedestrian was asked to follow this straight pathway (371-375) until the node 375. The length of this straight pathway (371-375) was measured to be 20.8 m. On reaching the node 375, pedestrian was finally asked to take a left turn towards link 375:364 and to walk until point B (represented by red square) along the straight pathway 375-411. The length of this walked straight pathway was measured to be 19.7 m approximately. The landmark (QR) codes were placed at requisite distances from the starting point. In total, three QR codes were placed along the complete route at strategic locations.

The pedestrians were further asked to click the TimeStamp button and unique code identifier button on SensorData app (see Figure 7.6) when passing through any of these QR codes. This registered the point of time when a pedestrian observed a particular QR code. Information about the QR code and the timestamp was logged in a csv file which was stored in smartphone's external memory (SD card). It was later analysed by the developed Java framework.

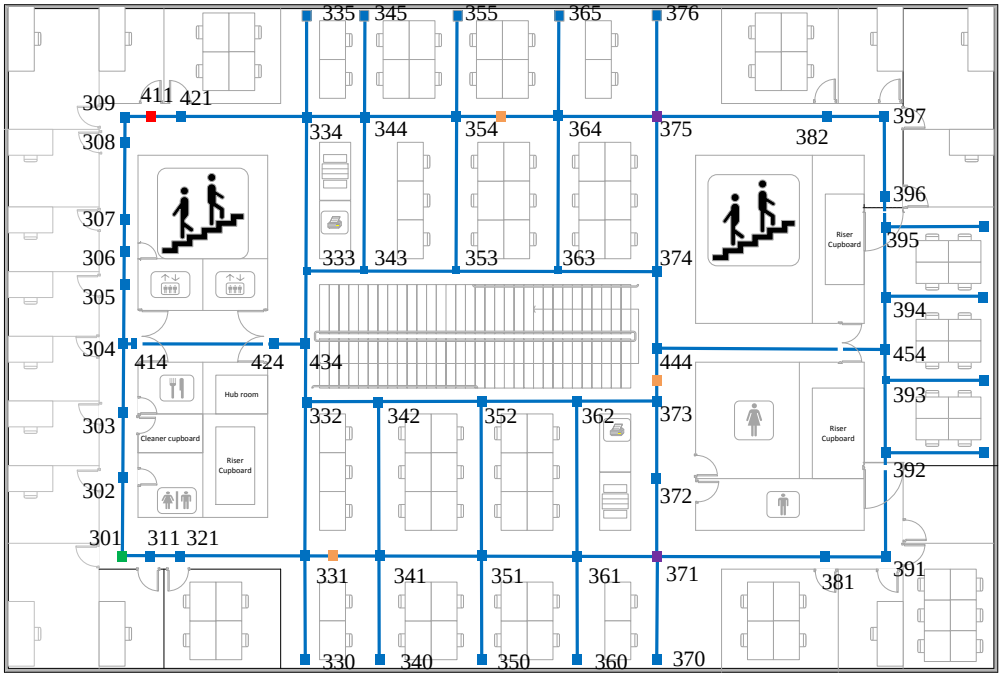


Figure 7.7: Geospatial model for the fourth level of Building 176 in the University campus. The markings (301, 311 ... and 411) indicate the node numbers. The green square depicts the starting point of the walk, and the red square depicts the end point of the walk. The orange square depicts the 3 QR codes observed by the pedestrian along the walk. The violet square depicts the 2 corners observed by the pedestrian along the walk.

Again this time (as referred in the previous chapter, see Chapter 5, Section 5.2) the experiment was focussed on analysing two different scenarios (see Figure 5.3) – smartphone handheld and Smart Glasses simulator. Each pedestrian was given a smartphone (HTC Z710e) to handheld and a Smart Glasses simulator for the complete walk. The pedestrians were asked to

walk normally. It was presumed that there was no obstacle during the whole walk and secondly, the initial conditions were known a-priori. The subjects were similar to the previous chapter (see Chapter 6, Section 6.4), therefore other conditions of the experiment were also similar i.e. eight of the women wore medium heeled sandals, six of the men wore slippers, three of the men wore boots and the remaining wore sports shoes. Table 7.1 summarizes the conditions of conducted series of experiment.

Conditions	**Values**
Number of subjects	50 (40 men and 10 women)
Distance	61.5 m
Weight	45kg – 100kg
Height	1.55 m – 1.81 m
Repetitions per subject	1

Table 7.1: Conditions of the conducted series of experiment.

7.2.1 Position error

The distance travelled is calculated by summing up the estimated step length of every counted footstep (estimated from the step counter algorithm see Chapter 6, Section 6.3.2; the performance of step counter algorithm was similar as in the previous chapter since the participants were same). This estimated distance is then compared against the actual distance at known point of interests (POIs) to compute the position error d_{err} (according to equation 5.1, see Chapter 5, Section 5.2.2).

The variation of position error at different POIs is visualised as a Box & Whisker Plot (see Chapter 6, Section 6.4.2 for complete details). There are six POIs in the selected route (see Figure 7.7) – two corners, end point and three QR codes. The distance to these POIs is measured offline – first corner is at a distance of 21.0 m from starting, the second corner is at a distance of 41.8 m from starting and the end point is at a distance of 61.5 m from starting. The QR codes are placed at distances 9 m, 30 m and 46 m from the starting point.

Figure 7.8 illustrates the variation of position error at six POIs. A similar behaviour is observed as seen in the previous chapter (see Chapter 6, Section 6.4.2). In particular, the drift in positioning error appears to gradually reduce with the increase in number of corner features and QR codes for the two scenarios – smartphone handheld and Smart Glasses simulator (see Table 7.2 for the exact values). This can also be confirmed by observing the decrease in dispersion over the 50 percentile region from Figure 7.8. Another interesting characteristic observed from Figure 7.8 is that the performance of smartphone handheld scenario is relatively better in comparison to the Smart Glasses simulator. This behaviour is similar to the previous chapter (see Chapter 6,

Section 6.4.2). It can be explained with the same reasoning as detailed earlier (see Chapter 6, Section 6.4.2) that is pedestrian continuously tries to stabilize his hand during the walk. As a result, the hand held smartphone actually yields more reliable step length estimates than its head mounted counterpart in a Smart Glasses simulator.

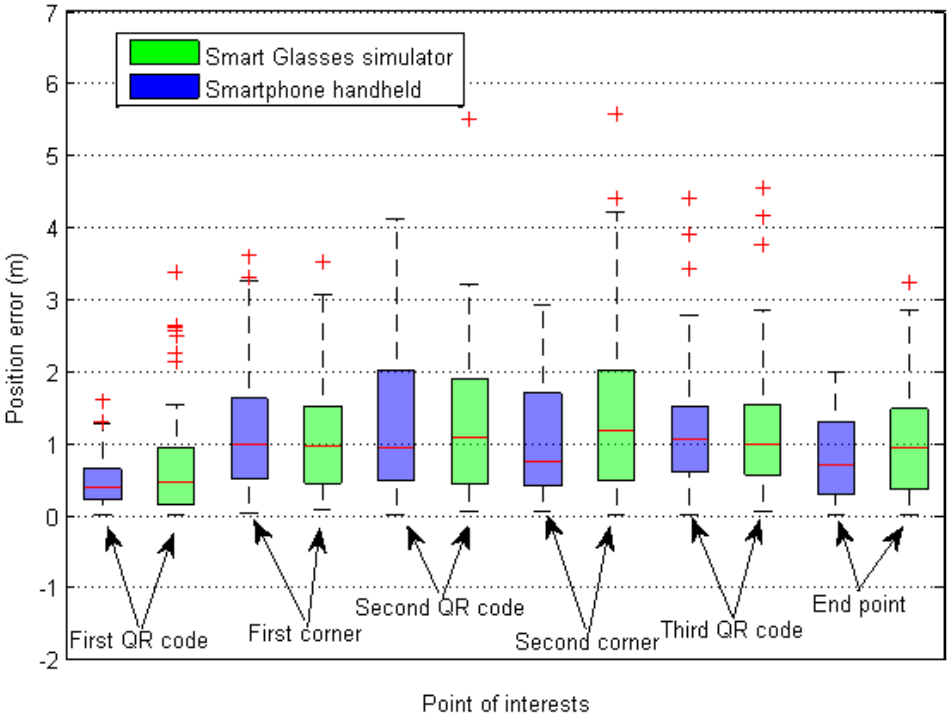


Figure 7.8: Box and whisker plot of the position error using corner features and QR codes at six POIs – first corner, first QR code, second corner, second QR code, third corner, third QR code and end point, before Kalman filter is reset.

Values (m)	**1st QR code**		**1st corner feature**		**2nd QR code**		**2nd corner feature**		**3rd QR code**		**End point**	
	Smart Glasses simulator	Smartphone handheld	Smart Glasses simulator	Smartphone handheld	Smart Glasses simulator	Smartphone handheld	Smart Glasses simulator	Smartphone handheld	Smart Glasses simulator	Smartphone handheld	Smart Glasses simulator	Smartphone handheld
Maximum	3.38	1.6	3.52	3.6	5.5	4.1	5.56	2.92	4.54	4.4	3.23	2
Median	0.5	0.4	1	1	1.1	0.9	1.2	0.8	1	1.1	0.9	0.7
Minimum	0	0	0.08	0.04	0.05	0.01	0	0.05	0.05	0	0	0
1st quartile	0.16	0.2	0.45	0.51	0.44	0.5	0.48	0.41	0.57	0.6	0.36	0.3
50th percentile	0.79	0.45	1.07	1.11	1.46	1.52	1.54	1.29	0.97	0.91	1.12	1
3rd quartile	0.95	0.65	1.52	1.62	1.9	2.02	2.02	1.7	1.54	1.51	1.48	1.30

Table 7.2: Comparison of position error using corner features and QR codes at six POIs – first corner, first QR code, second corner, second QR code, third corner, third QR code and end point for the two scenarios (A) smartphone handheld and (B) Smart Glasses simulator, before Kalman filter is reset.

On comparing the position error with the earlier test case (i.e. considering only the corner enabled map matching see Chapter 6, Section 6.4.2., Figure 6.22), it is observed that the performance of smartphone aided with both QR codes and corner features is more balanced and robust in comparison to its counterpart. This is illustrated by the presence of lesser number of outliers and the smaller dispersion over the 50 percentile region in Figure 7.8 at three POIs – first corner, second corner and end point in comparison to the similar points in Figure 6.22. The exact numbers can be observed from Table 6.8 and Table 7.2 respectively.

It is also observed that the mean absolute position error is less than 1 m for this test case in the two scenarios (see Table 7.3) – smartphone handheld and Smart Glasses simulator. Even the 95% confidence level is comparatively less for this test case in either of the two scenarios (see Table 7.3). This behaviour is as expected. Since the corner features, in corner enable map matching test case are distantly separated. Therefore, the positioning error gets accumulated overtime. In comparison, on the second test case the positioning error is significantly reduced by updating SLs (according to equations 7.1 and 7.2, see Section 7.1.1.2) derived from the measured ground truth at shorter intervals on the observation of either QR codes or corner features and the uncertainty (1σ) associated with pedestrian's relative position $\hat{x}_t$ is significantly reduced to ± 0.7 m on observation of these features. This results in resetting the Kalman filter at shorter intervals (see Section 7.1.1.2 for details). This restricts the growth of position error, and improves the overall accuracy of positioning system. The mean absolute error (MAE) is derived according to equation 5.2 (see Chapter 5, Section 5.2.2 for the exact formula).

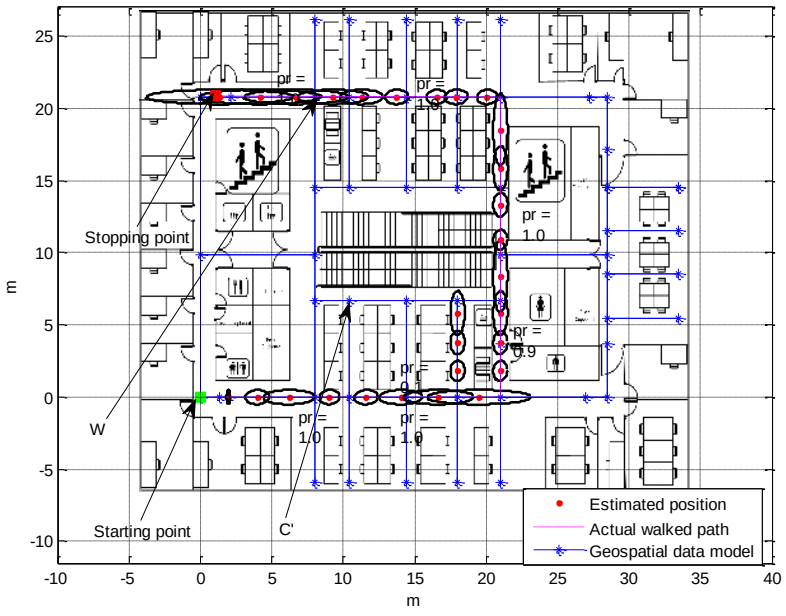
Values (m)	**Smart Glasses simulator**		**Smartphone handheld**	
	Corner only	Corner & QR code	Corner only	Corner & QR code
MAE	2.05	0.98	1.25	0.80
95% confidence level	-5.20 to 5.60	-2.69 to 2.47	-3.16 to 3.32	-2.0 to 2.0

Table 7.3: Comparison of position error at end point for the two test cases: (A) only the corner enabled map matching (readings referred in Table 6.10 are again illustrated here for the sake of completeness and readability) and (B) both QR codes and corner features included, in the two scenarios – smartphone handheld and Smart Glasses simulator.

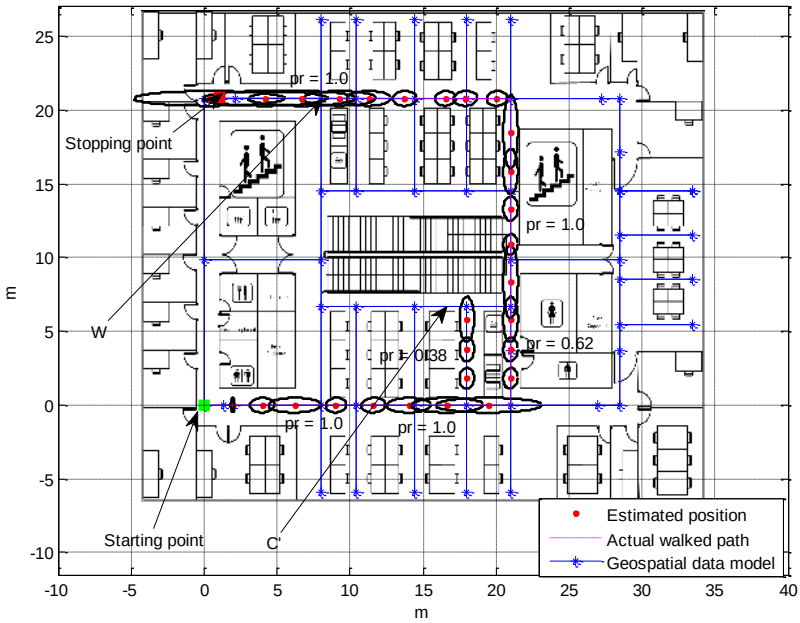
7.2.2 Map and landmark matching evaluation

The performance of map aided and, artificial landmark aided smartphone based PDR positioning algorithm is also analysed graphically. The key characteristic of MURE map matching algorithm is 'splitting' and 'merging' (see Chapter 6, Section 6.3.4.2). While the key characteristic of map

aided and, artificial landmark aided smartphone based PDR positioning algorithm is resetting of position error at shorter intervals in addition to splitting and merging of the Gaussian component associated with pedestrian's position. Figure 7.9 illustrates the trajectory of a pedestrian for the two scenarios (A) Smartphone handheld and (B) Smart Glasses simulator considering both QR codes and corner features are included in the indoor setting.



(A)



(B)

Figure 7.9: Trajectory of a pedestrian for the two scenarios: (A) smartphone handheld and (B) Smart Glasses simulator considering both QR codes and corner features are included in the indoor setting. Pedestrian starts at a point represented by green square and stop at a point represented by red square. They walk along the path shown by magenta dashed line. The red dots represent the estimated positions at lapse of every 2s. The black ellipses around every red dot represent the 1σ uncertainty region associated with the estimated pedestrian's position.

It is observed that the graphical performance of map aided and, landmark aided algorithm is quiet similar to the earlier test case discussed in the last chapter (using only the corner enable map matching, see Figure 6.24). Initially the pedestrian is represented by a single agent with existential probability $p^e = 1$. The pedestrian makes a left turn at node 371, but due to ambiguity in position there is a splitting operation resulting in two new software agents. The agent on the true link (371:372) has highest existential probability $p^e = 0.9$ and $p^e = 0.82$ for the two scenarios – smartphone handheld and smart Glasses simulator respectively.

At the point in Figure 7.9 labelled C' two of the extant agents encounter a T-junction topology where a corner turning motion must be made to proceed. However as no corner turning is detected – the pedestrian is walking straight on –these two agents get merged. The existential probability of the single remaining agent is now $p^e = 1$ and remains 1 throughout.

On the contrary, in the earlier test case (see Figure 6.24), the splitting operation again takes place at the second turning corner resulting in two new software agents. These agents got merged when one of the agent arrives at labelled point K where there is a T junction and no turn is detected hence the agents get merged into a single remaining agent at point W. This is unlikely for this case. Considering the fact that there are more number of features (QR codes and corner features) along the route of a pedestrian. So as a result of this, the pedestrian's position uncertainty (1σ) does not grow as fast as in the former case. Hence, the area to which splitting takes place tentatively remains smaller for this test case. Therefore, as a result of this the software agent ideally has higher p^ealong the actual walked route and even does not *splits* after the pedestrian encounters a turning at node 375.

The map matching ratio is computed according to equation 6.25 (see Chapter 6, Section 6.4.3). It is observed that the map matching ratio for even the worst case is greater than 0.84 for the two scenarios (see Table 7.4) – smartphone handheld and Smart Glasses simulator. This behaviour can primarily be attributed due to the occurrence of the two key phenomena in MURE map matching algorithm – splitting and merging (as detailed earlier in the Chapter 6, Section 6.3.4.2). This again confirms the aforementioned fact (see Chapter 6, Section 6.4.3) i.e. the MURE map matching algorithm is sufficiently enough to provide a logical solution to estimate the

position a pedestrian and the path taken by a pedestrian, provided the map is described as links and nodes.

Values	**Smartphone handheld**		**Smart Glasses simulator**	
	Only corners	Corner & QR codes	Only corners	Corner & QR codes
Minimum map matching ratio	0.85	0.89	0.80	0.84
Maximum map matching ratio	0.94	0.99	0.92	1

Table 7.4: Map matching ratio for the two test cases (A) only the corner enabled map matching (readings referred in Table 6.11 are again illustrated here for the sake of completeness and readability) and (B) both QR codes and corner features included, in the two scenarios – smartphone handheld and Smart Glasses simulator.

Nevertheless, on comparing the map matching ratio with the earlier test case – only the corner enabled map matching (see Table 7.4), it is observed that that the map matching ratio is slightly higher for the this test case. This behaviour can be explained with the same reasoning as above i.e. because of the presence of QR codes in mid of the corner features, the 1σ standard deviation associated with pedestrian's position does not grow as fast as in the former case. As a result of which the area to which splitting occurs on observation of a corner feature practically remains smaller. Subsequently, this improves the overall existential probability (as illustrated in Figure 7.10) of an agent on a particular link. Also, the presence of QR codes serve as an additional landmark to initiate the map matching process of merging.

The two journeys described in Figure 7.9 are single runs of a single participant and each of the other runs by different participant had different splitting and merging behaviours. To capture some of this variation Figure 7.10 illustrates the existential probabilities along different links over different participants.

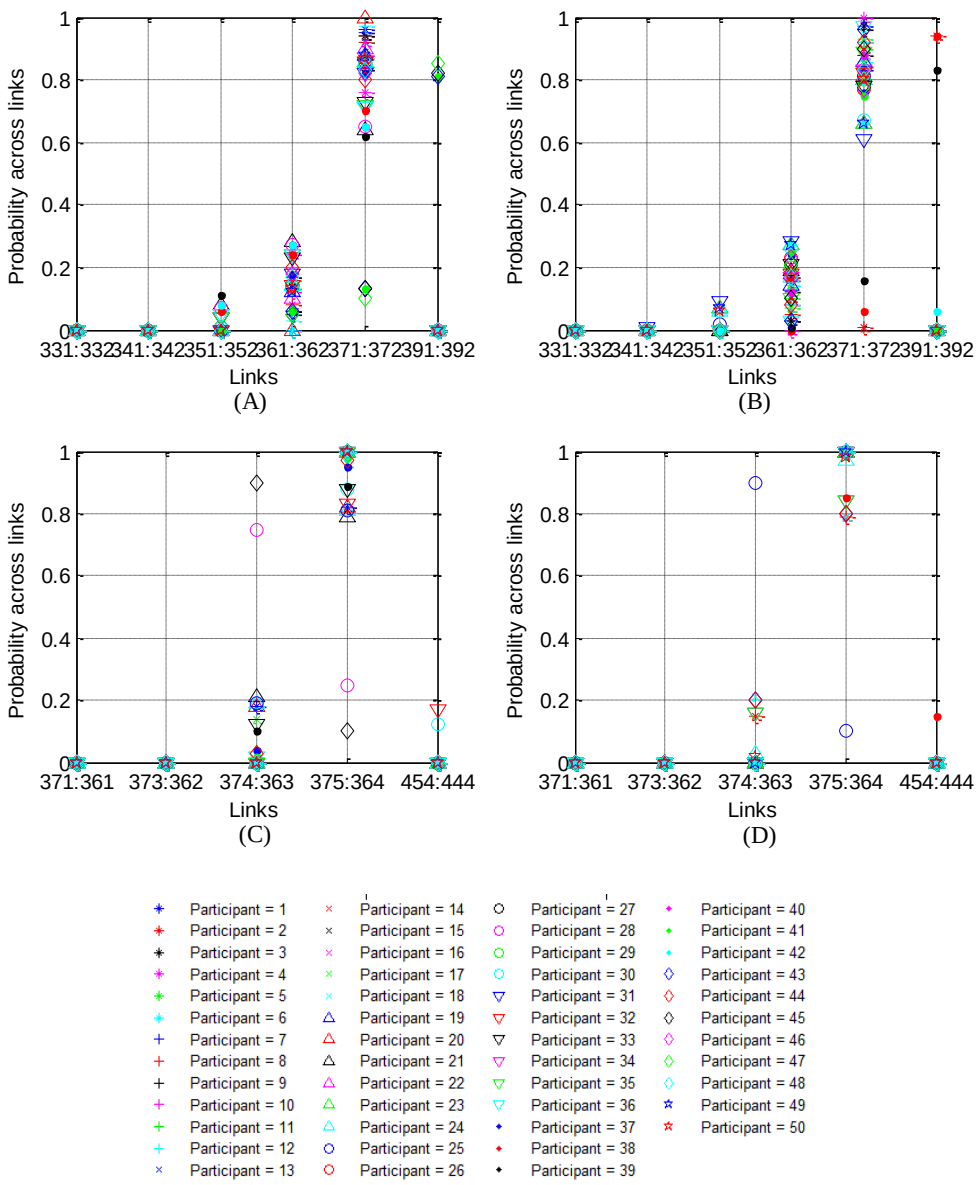


Figure 7.10: Existential probabilities of agents along different links after splitting takes place on detection of a corner feature – node '371' and node '375' (see Figure 66 for details) for 50 pedestrial walks using both corner features and QR codes. The headings (A) and (B) represent the existential probabilities of agents for the two scenarios – (A) smartphone handheld and (B) Smart Glasses simulator after the first turning at node 371. While the headings (C) and (D) represent the existential probabilities of agents for the two scenarios – (C) smartphone handheld and (D) Smart Glasses simulator along the second turning at node 375.

Figure 7.10 illustrates sample values of existential probability relating to the left turn made by pedestrians at node 371 and node 375 in Figure 7.7 for the two scenarios – smartphone handheld and Smart Glasses simulator. The values of existential probability for agents that travelled along the true link 371:372 after left turn occurs at node 371 are shown in

(approximately) the middle of the plot (A) and plot (B) for the two scenarios respectively and the values for agents that travelled down other adjacent links (e.g. 361:362, 391:392, etc.) are shown arranged along the x-axis in their topologically relative position. Plot (C) and plot (D) represent the variation of existential probability for agents after left turn occurs at node 375 for the two scenarios respectively. The link 375:364 is the true link taken up by an agent after the left turn occurs at node 375 whereas the other links – 374:363, 373:362 and 371:361 are perpendicularly separated links from the actual link. The link 397:382 is the link following the actual link 375:364, while link 454:444 is the link parallel to the link 397:382.

As expected the values of existential probability for agents that travelled along extreme links (i.e. link 331:332 for the first turning corner and link 371:361 for the second turning corner) is least, approximately zero and the values of existential probability for agents that travelled along links closer to the true links is higher with the highest along the true links taken up by an agent. However, in comparison to the earlier test case (see Figure 6.26) the existential probability for agents is significantly higher (closer to 1) in the true link over 50 pedestrial walks. This can be attributed because of the same reasoning as detailed above i.e. the area to which splitting of Gaussian component takes place on detection of the corner feature is significantly less for this test case.

Also, it is to be noted that there is always an agent with some existential probability on the correct link, that is, tracking is never lost. In addition it is noted that the least existential probability for an agent on the true link is greater than 0.84 in 95% of the pedestrial walks for the two scenarios.

7.3 Summary

In this chapter, we focussed on the usage of artificial landmarks (QR codes) for positioning. These landmarks were used in addition to the corner features, occurring naturally for positioning. A landmark based positioning technique has been designed in this context. The uniqueness of this approach lies in the fact that it utilizes the human as a sensor to detect proximity to landmarks. The basic idea of landmark based positioning involved two key paradigms. First it involved building a geospatial database i.e. augmenting the physical space with landmarks and selecting the landmark. Second it involved landmark matching i.e. resetting the accumulated positioning error to minimum on sensing proximity to landmarks by humans. The key findings of this chapter are as follows:

- The mean position error is relatively less when the smartphone is handheld in comparison to the Smart Glasses simulator (see Table 7.3). This behaviour is likely to happen because of the fact that pedestrian continuously tries to stabilize his hand during the walk. As a

result, the hand held smartphone actually yield more reliable step length estimates than its helmet mounted counterpart in a Smart Glasses simulator.

- The performance of smartphone aided with both QR codes and corner features is more balanced and robust, in comparison to smartphone aided with corner features only (see Figure 6.22 and Figure 7.8).This behaviour is likely to happen since, the QR codes ascertains the pedestrian of its position and moreover, serve as an additional cues to its destination.
- The existential probability of an agent along the true link is greater than 0.84 in more than 95% of pedestrial walks for the two scenarios (see Table 7.4, considering both splitting and merging phenomena takes place in the MURE map matching algorithm) – smartphone handheld and Smart Glasses simulator. This behaviour is significant. Since, it ascertains that the likelihood of the path estimated by the algorithm is the actual path followed by pedestrian is greater than 0.84.
- The map matching ratio is significantly higher when the smartphone aided with both QR code and corner features are considered in comparison to the smartphone aided with corner features only (see Table 7.4). This behaviour can be attributed because of the fact that the presence of QR codes in mid of the corner features limits the rate of growth of uncertainty (1σ) associated with pedestrian's position. Consecutively, the area to which splitting occurs on observation of corner feature practically remains smaller. Hence, this results in the improvement of the existential probability of a pedestrian on a particular link.

These results confirm the consistency of positioning algorithm (as previously observed in the last chapter, see Chapter 6, Section 6.4) to locate pedestrians and moreover, provide an alternative means to reduce the growth of position errors if corner features are distantly separated. However, there are some limitations. These are as follows:

- Choice of sensors: The precision of employing humans as a sensor to log in the time at which they observe the QR code is not very good. Since, many a times they forgot to click the TimeStamp button on the SensorData application (see Figure 7.6) when they passed across landmarks and, also at times they reported of clicking the TimeStamp button far off from the actual landmark.
- Dependence on landmarks: The landmarks (QR codes) are artificially designed and selected, based on their saliency and visual characteristics. This is subsequently attached to the scene at strategic locations. This requires time and manual effort.

Also, QR code based landmarks affect the 'user experience'. So ideal to minimize their usage i.e. use corner features as much as possible. However, QR codes are an effective tool to solve the

problem of many routes in MURE map matching algorithm. Methods to address these limitations and inaccuracies will be discussed in the chapter ‘Conclusion and Future Work’. The next chapter however discusses how this study complements the existing body of research.

8. Chapter 8: Discussion

In this chapter we firstly compare the performance of developed systems with the evidence available in the literature and secondly we describe the suitability of the developed algorithms for practical implications by industry.

8.1 Performance comparison with the other systems

Chapter 3 and Chapter 6 presented evidence on the performance of various map and landmark aided PDR positioning systems using smartphones or an independent IMUs as a sensing unit. The most significant capability of our system is to offer accurate logical positioning and tracking performance. A comparative summary of the existing map aided PDR systems and landmark aided PDR systems along with our system developed in this research study is presented in Table 8.1 and Table 8.2.

System	Accuracy ($\bar{x}$)	Precision ($\bar{x} \pm 2\sigma$)	Placement
Attia *et al.* [280]	7.12 m	Unspecified	Hand
Nammoon and Youngok [314]	4.19 m	0.19 m to 8.19 m	Hand
Bao *and Wong* [299]	0.75 m	0.4 m to 1.2 m	Trouser pocket
Li *et al.* [316]	1.5 m	-7.1 m to 10.1 m	Hand and pocket
Xiao *et al.* [28]	2 m – 4 m	97% of the cases, converges within an error less than 3 m for distances less	Hand
Our system	1.25 m	-3.16 m to 3.32 m	Hand
	2.05 m	-5.20 m to 5.6 m	helmet mounted

Table 8.1: Performance comparison for the map aided PDR systems using smartphones as a sensing unit.

System	Accuracy ($\bar{x}$)	Precision ($\bar{x} \pm 2\sigma$)	Placement	Cost
Mulloni *et al.* [180]	Unspecified, Depends on the marker density.	Unspecified	Hand	Prior mapping cost of less than a minute for a single marker.
Fallah *et al.* [189]	1.85 m	-3.63 m to 7.33 m	Hand	2000 USD + prior mapping cost of 3 days.
Wang *et al.* [62]	1.69 m	Unspecified	Hand and pocket	Cost of Wi-Fi infrastructure + prior training needed.
Our system	0.80 m	-2.0 m to 2.0 m	handheld	Prior mapping cost of less than 30 s, for a single QR code.
	0.98 m	-2.69 m to 2.47 m	Helmet mounted	

Table 8.2: Performance comparison for the landmark aided PDR systems using smartphones as a sensing unit. Mapping cost is related to the time needed to augment the environment with landmarks.

The system developed by Bao and Wong [299] localized pedestrian accurately to 0.75 m with precision ranging from 0.4 m to 1.2 m; this was based on calibrating location as well as step direction and sensor's orientation. In comparison, when they calibrated only the locations the average error over twenty trials was 0.95 m which was slightly ahead of our approach.

The performance of map aided PDR system developed in this study is better than most of the existing map aided PDR systems (with respect to accuracy and precision) reported in the literature and equals to systems using costly infrastructures and multimodal fingerprinting [25, 97]. This performance can be attributed because of the developed MURE map matching algorithm and step length estimation technique. The landmark aided PDR system developed in this study is an extension to the developed map aided system. Its performance is much superior in comparison to existing landmark aided PDR systems (with respect to accuracy, precision and cost).

The MURE algorithm including its extension landmark matching algorithm is lightweight in both running time and memory usage because it neither stores transition matrices nor performs expensive matrix operations as in [332]. For a normal pedestrian tracking application, the processing time for each step should be less than 800 ms. Meanwhile, the memory limit for a single application on an Android platform is 20 MB. Any algorithm with requirements exceeding these limits is not suitable for pedestrian tracking [28, 255]. Although, the developed algorithms in this study are currently offline, therefore we are not sure about the RAM usage however

practical size of the application (Java software stack and smartphone sensor application) is 3.0 MB. This is lesser in comparison to the other commercial software applications [333, 334].

The minimum map matching ratio is 0.73 that is a measure to identify the correct link over which pedestrian has walked. This ratio can be even equal to 0.99 when aided with landmarks – QR codes. This demonstrates the applicability of our algorithm for tracking applications. This practically outperforms the existing map matching algorithms [280, 296].

8.2 Suitability of the developed algorithms

The research has developed a map aided smartphone based PDR positioning system and, a map and landmark aided smartphone based PDR positioning system. The positioning capability of these systems relied on map-matching algorithm – MURE and its extension – a landmark matching algorithm that are the key contributions of this research study. To estimate the pedestrian's position continually, the data was collected from inertial sensors embedded in a smartphone at every 50 ms. The validation of the developed algorithms were carried out on two different floors of two different buildings by more than 70 different subjects. This is one of the very few comprehensive investigations that have ever been performed. Marschollek *et al.* [335] did a detailed study with over 200 test subjects however their major focus was to analyse the performance of pedometer. This provides evidence of the reliability and transferability of the algorithms.

The developed algorithms are potentially lightweight, therefore they can be easily implemented on memory constraint devices. Further position computation can directly be carried on devices itself thus privacy of users can also be maintained. The developed algorithms have high potential to be implemented by industry for tracking applications as these both algorithms are easy to implement, reliable, transferable and lightweight.

The key benefit of this research is mainly enhancement of indoor positioning systems to support a range of location-based services including indoor pedestrian tracking.

9. Chapter 9: Conclusion and future works

This thesis concludes that it is possible to logically localize a pedestrian using the information from inertial sensors embedded in a smartphone with minimum usage of infrastructure or none at all. As per the objective of this research work to investigate how the information from smartphone sensors can be used to provide reliable, accurate, low cost logical localization. This research work provides a reference implementation for developing such a positioning solution. Several objectives were formulated in Chapter 1. Table 9.1 gives a brief description of each of the objectives and the corresponding chapters in which they were achieved or addressed.

Objective	**Method/Description**	**Chapter**
To critically asses different existing indoor positioning technologies, systems, and combinations of those.	Literature review: (1) Positioning technologies and methods (2) Commercial products (3) Sensor fusion	2
Analyse the performance characteristics of smartphone embedded sensors. The focus was be primarily towards inertial sensors.	Characterised the systematic and non-systematic errors.	3
Understand whether the positioning is possible using smartphone embedded INS.	A case study was performed on an artificial test bed	3
Understanding the movement pattern of pedestrians, since the unconstrained movement of pedestrians is often varied.	(1) Multimodal data from inertial sensors was investigated in time and frequency domain. (2) Various stride estimation techniques were applied to detect footsteps and estimate stride lengths (3) Performance evaluation	4
To develop a map database for indoor localization.	(1) Indoor floor plan was designed in form of geospatial data model. (2) The data model composed of the following elements: passageways, corridors, doors, dead ends and point of interests; represented by a set of nodes and links.	6
To develop an intelligent sensor fusion technique for merging the multimodal data from inertial sensors.	Using Kalman filter, multimodal data from inertial sensors was integrated	6

To develop a map matching algorithm for projecting the position fix from sensor fusion technique into the map database of studied region.	(1) MURE algorithm (2) Performance evaluation	6
To develop a landmark matching algorithm for projecting the position fix from sensor fusion technique into the map database of studied region while integrating the knowledge of artificial landmarks present in the route.	(1) Build landmark database (2) Landmark matching algorithm (2) Performance evaluation	7
Compare the developed map matching algorithm and landmark matching algorithm.	(2) Performance revaluation	7

Table 9.1: Research objectives

Chapter 5 was not listed in the Table 9.1; it described a helmet mounted PDR system which was subsequently used in later chapters for evaluation of the performance of various algorithms.

At the start of this thesis, we noted that amongst all the indoor localization research studies and commercial products there was none of the positioning solution that had deep penetration into the society (e.g. GPS for outdoor positioning). On finding the reasons, we concluded that there were several limitations: (1) most of the available positioning solutions needed some or the form of infrastructure that required investments from owner or builder, (2) reliability, (3) privacy, (4) logical localization and (5) importantly, accuracy and cost trade off.

To overcome these limitations, we performed several experiments and carried out different investigations.

We used low energy inertial sensors, readily available maps and QR codes to keep the cost as low as possible. In addition we used algorithms e.g. Kalman filter that can be easily ported into memory constraint devices which have low computation cost. To ensure user privacy, we used secured systems to store the multi-modal data collected from inertial sensors during pedestrial walking. Also, prior verbal consent was taken for storing the pedestrian's sensory data from the smartphones.

The first investigation, we performed was a case study on an artificial testbed – Scalextric track. The principal behind this work was to show how the 'low cost smartphone embedded INS' system can be used for positioning in an infrastructure free environment. Results show that positioning is possible, that also, using only the single inertial sensor. Position error does not grow with time, unlike other traditional inertial positioning techniques [140]. The case study was performed in a control environment – Scalextric track. While in real scenario, the building floor

plan is not as detailed as an artificial test bed and moreover, the gait of pedestrian is far different from the locomotion pattern of a toy vehicle.

To understand the movement of pedestrians, a detailed investigation was performed in time and frequency domain over the multimodal data from inertial sensors. Various off-the-shelf stride estimation techniques (footstep detection techniques and stride length estimation models) and traditional heading estimation algorithm were applied onto the data from inertial sensors embedded in a smartphone. Results showed that FFT based step rate detection algorithm outperformed the other footstep detection techniques whereas Weinberg stride length model performed better in comparison to the other off-the-shelf stride length estimation models. Average percentage heading error was less than 11% in corners.

A major focus of this research was to identify the logical location of a pedestrian. We proposed a novel probabilistic map matching technique – MURE to identify the logical location of a pedestrian. Like any other map matching technique, it counters the positioning drifts introduced because of the pedestrian dead reckoning however uniqueness lies in the fact that it utilises a basic Linear Kalman filter but extends the representation of a pedestrian's position to support multiple discrete Gaussian probability distributions at different points in the network. The network is designed in form of a geospatial data model with specific attributes moulded for navigation and tracking applications. The fundamental idea behind MURE map matching algorithm is based on detection of a corner feature that initiates the process of merging and splitting.

The algorithm is tested on two different smartphone orientations – smartphone handheld and smartphone helmet mounted, on two different floors of two different buildings. Results indicate an average absolute error less than 2.1 m in either of the two smartphone orientations with 95% confidence level lesser for the smartphone handheld scenario over the two indoor settings. We extracted an overall metric – map matching ratio, for evaluating the performance of MURE algorithm by calculating the average existential probability over the sampled positions of agents that are located on the correct path for a given run.

We found that minimum map matching ratio was greater than 0.70 for both the scenarios in the two indoor settings. Also, in the second indoor setting the map matching ratio was slightly greater in comparison to the first indoor setting. To improve the accuracy and more importantly, to reduce the dependency on corner features we proposed a novel landmark matching algorithm.

The algorithm extends MURE algorithm. It resets the position error and, initiates the process of merging. Only when a pedestrian observes a landmark that is stored in the landmark database. In addition, the time of occurrence of the landmark matches with the corresponding time running in the smartphone. To validate the performance of algorithm experiments were

performed and results were compared with the earlier test case i.e. map aided smartphone based PDR positioning system. Results showed that performance of smartphone aided with both landmarks (QR codes) and corner features was more balanced and robust. In addition the average absolute error and 95% confidence level was marginally less using this technique. This also indicates that it is possible to achieve the similar accuracy levels using only the corner features provided corners are not distantly separated. We further found that map matching ratio was significantly greater using this approach in comparison to the previous map matching approach in either of the two scenarios.

It is to be noted that although QR codes have been used for this research work to add redundancy and make experiments more robust however in actual application this would rarely be used. Since, it affects the user experience. Section 9.1 highlights some of the advantages and disadvantages of the essential technologies developed in this research work. Section 9.2 gives directions for the future work. Finally, Section 9.3 presents some of the potential applications of this research work.

9.1 Advantages and disadvantages of the key technologies

During this research work, we have identified and developed technologies that can be helpful in locating the pedestrians accurately and reliably under variable conditions. The advantages and disadvantages of some of the technologies developed during this research work are as follows:

i. Feature (corner) detection: The gyroscope sensor was primarily used to identify the 90^0 turnings (see Chapter 6, Section 6.3.1). These turnings were considered as corner features for this research work to limit the growth of position errors. A significant advantage of this technology is that it can bound the error growth in INS. Another advantage of this technique is that it works with the existing sensors while prohibiting the user from any other additional costs of building the database or clicking a TimeStamp button on observing QR codes. On the downside, the technology works best in parallel with map matching. Therefore, it necessitates the area to be properly mapped. Furthermore, the technology has been tested only in simple environments so its robustness is still required to be proved in complex building structures.

ii. Smart Glasses simulator: A scenario similar to the smart glasses has been emulated by means of Smart Glasses simulator (see Figure 5.3), and tested indoors on different settings. The results were particularly negative. It has however highlighted the fact that handheld devices naturally damp high frequency vibrations which is helpful for signal processing. On the contrary, poor fit of the helmet leads to a poor simulation of smart glasses. This further resulted in under performance of the Smart Glasses simulator.

iii. Multiple uncertain routes engine (MURE): The MURE (see Chapter 6, Section 6.3.4.2) is designed based on a probabilistic framework to estimate and predict the position of a pedestrian on a particular link in a map. In particular, it extends the Kalman filter to enable multiple discrete Gaussian probability distributions to model pedestrian's location in a map. The two essential characteristics of MURE technology is 'splitting' and 'merging'. These together can manage link ambiguity in networks where position accuracy is low in relation to the length of links. As demonstrated knowledge of the network structure helps to reduce ambiguity through merging (see Chapter 6, Section 6.3.4.2) as the journey proceeds. However, it is not clear how multiple discrete Gaussian distributions may be presented to a 'layman user' on a smartphone without causing confusion.

iv. Landmark (QR code) based positioning: The landmark based positioning technique (see Chapter 7, Section 7.1.1) is designed such that it uses landmarks to not only assure the pedestrian about the correct pathway to the destination but also it improves the positioning accuracy of the positioning system (especially, by calibrating the positioning system on sensing the proximity to landmarks). The key advantage of this technology is that it utilizes the human as a sensor rather than just as an actuator. Another advantage of this technology is that it instantly removes the ambiguity in position generated due to splitting of the Gaussian component (caused as a result of MURE). However, it puts a burden on the user in clicking the TimeStamp button on sensing the proximity to landmarks and additionally, it affects the user experience on observing the QR codes at some intervals.

9.2 Future work

Evidence gathered throughout this research work has shown that, in principle it is possible to envision a positioning system that can provide accurate, reliable, low cost logical indoor localization of the pedestrians with minimal usage of infrastructures. Nevertheless, the research findings led to new questions and requirements for system improvement. These emergent issues are considered as future research directions.

- Sensors choice

  The research was limited to use of INS embedded in smartphone. The perennial problem of using inertial sensors is that it employs a DR approach and therefore growth in errors is unbounded. Moreover, it needs an initial starting point, otherwise results are often unreliable [139]. By employing corner enabled map matching and alternatively landmarks in form of QR codes, we managed to achieve positioning with a bounded error (no drift). However, it was presumed throughout this research work that the starting point

was known. This may not be always valid e.g. when a pedestrian is in mid of a crowded art gallery or a museum. Therefore, supplying the starting position would be a problem.

In such a case it is recommended to use built in light sensor and pressure sensor embedded in smartphone to determine the position of a pedestrian in respective floor level. The pressure sensor can determine the floor level as previously done by Xia *et al.* [336]. While light sensor could determine the position of a pedestrian based on the light intensity of surrounding [28]. Another constraint reported in Chapter 7 was that many a times pedestrian forgot to click the TimeStamp button on the SensorData application when they passed across landmarks. So it is recommended to use built in cameras [177] in the smartphone to sense the proximity to landmarks.

- Orientation

  Throughout this research work, it has been presumed that orientation of the smartphone relative to pedestrian is fixed either handheld or as Smart Glasses simulator. Although it is viably true that a large portion of the pedestrians hold their smartphone in hands [274], particularly in wrists, yet the position of wrists significantly changes from rest to swinging of hands or hand upheld to an arbitrary position. As a result the orientation of smartphone also changes. This principally means that the reported position is different from the actual position because of the change in smartphone's coordinate system relative to global coordinate system (see Figure 3.2) [337]. It is therefore recommended to use the smartphone's built in orientation sensor or rotation matrix [338] to compute the actual orientation of device in global coordinate system.

- Features and landmarks

  As illustrated in the Chapter 6 and Chapter 7, the positioning solution is highly reliant on recognising corner features and landmarks. In Chapter 6, map aided PDR positioning solution recognises corner features present in the indoor environment via gyroscope sensor that are subsequently matched to the corners in network map. Similarly in Chapter 7, landmarks (QR codes) are recognised by pedestrians that are subsequently matched to the landmark database. As mentioned before (see Chapter 7, Section 7.1.1.1) these landmarks are easy to design, inexpensive and offer easiness to the pedestrians in recognising them yet attaching to the scene is a labour intensive process.

  As a result their usage should be minimised, instead naturally occurring features (or landmarks) in the environment should be used. As stated by Wang *et al.* [62] corners are not only the features that may exist in the indoor environment. Other features may include stairways, elevators and closed doorways which in principle may be detected by the accelerometer or magnetic sensor that can be used to position a pedestrian. Subbu *et al.*

[339] have identified the magnetic signatures of ferromagnetic materials by magnetic sensor embedded in smartphone to position a pedestrian in the map corresponding to those materials in the physical world.

- Importing the developed Java application into smartphone framework

  At present positioning is done offline, pedestrian walks on a typical route indoors. The smartphone based inertial application collects and stores data in SD card. This is later analysed in the Java framework to estimate the pedestrian's position. The whole process involves significant amount of time. Instead it would be better if the whole Java based framework is present in the smartphone itself, as previously done in some of the research studies [226, 316]. Importantly, a pedestrian loads the map of a building from application's database stored in his/her smartphone. Then following this, the application locates pedestrian's position on the go.

  As a matter of fact the developed Java based application is designed in perspective of memory constraint devices, involving usage of INS, map and human centric approach to sense proximity to landmarks. The usage of radio based signals is avoided that significantly drain the smartphone's battery rapidly [340].

- Comparison with the Smart Glasses

  In this research work, we investigated a scenario to emulate the smart glasses (e.g. Google glass) via Smart Glass simulator. The smartphone was attached to a bicycle helmet in a way such that it can emulate the true scenario of smart glasses (see Figure 5.2). Although results appear to be negative in comparison to the smartphone handheld scenario, yet it would be interesting to investigate that how the sensors embedded in actual smart glasses position the pedestrian accurately and reliably in varied indoor settings.

  Due to the limited resources and time constraints, it is not tested however it can be useful to take up in the future.

9.3 Potential applications

Few of the potential applications that require accuracy less than 1.5 m and can be supported by the developed algorithms are as follows:

- Room/warehouse/event planning and indoor tracking

  If real-time solutions help mobile users navigate and find objects indoors, then post-processing the movements can be studied to improve indoor environments and workflow. This is very useful in large warehouses where workers have to get goods from large buildings and load them to trucks. Placement of goods can be organized in a more efficient way by indoor moment processing and analysing.

Analysing movements of people in big festivals, concerts, fairs and conferences can help in better organizing these events. In case people get lost in these events then the designed algorithms can be of potential help in tracking their position.

- Tourism

  It often happens that when a tourist (especially children and old people) travels into a museum or art galleries, they get involved in gazing souvenirs and paintings kept across different rooms in a building at different floor levels. As a result they often get lost and moreover forget the path from which they came in. Although signposts are often available at the end of corridor yet those are significantly tricky and boring to understand [341]. In such a scenario the designed algorithms can be a potential help in recognising their location and planning their path backwards.

- Search and rescue

  In a search and rescue mission, it is essential for a rescuer to know their position and importantly know the position of their hostage. Significantly, the geometries of building can be complex and it is a possibility that they may get lost in their mission. This can potentially risk their lives and the lives of hostages too. In such a scenario the designed algorithms can be a potential help to them in recognising their location and planning their path backwards to bring hostages along with them.

Appendix A

Papers published

[1] S. K. Gupta, S. Box, and R. E. Wilson, "Low cost infrastructure free form of positioning using low cost sensors". In, 5th IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), Busan, South Korea, 27 – 30 October 2014.

[2] S. K. Gupta, S. Box, and R. E. Wilson, "Vehicle localization based on heading data using low cost sensors". In, 46th Annual Conference of the Universities' Transport Study Group (UTSG), Newcastle University, GB, 06 – 08 January 2014.

[3] S. Box, J. L. Miller, J. Snowdon, J. Hammond, A. Hamilton, S.K. Gupta, R.E. Wilson and B. Waterson, "Lessons from proving ground experiments to investigate junction control". In, 16th IEEE International Conference on Intelligent Transport Systems, The Hague, 06 - 09 October 2013.

Papers in process

[1] S.K. Gupta, S. Box and R. E. Wilson, "Multiple Uncertain Routes Engine (MURE): A map matching algorithm for indoor localization". In Preparation.

[2] S.K. Gupta and S. Box, R. E. Wilson and B. Waterson, "Technologies for Vehicle and Pedestrian Localization – A Technical Review". In Preparation.

[3] S.K. Gupta and S. Box and R. E. Wilson, "Feature detection in inertial measurements for error-bounded indoor localization". In Preparation.

Poster

[1] S. K. Gupta, S. Box, and R. E. Wilson, "Low cost infrastructure free form of indoor localization". In Transportation Theme Conference, University of Southampton, Southampton, United Kingdom, 29 May 2014

Appendix B

ERGO application form – Ethics form

All mandatory fields are marked (M*). Applications without mandatory fields completed are likely to be rejected by reviewers. Other fields are marked "if applicable". Help text is provided, where appropriate, in italics after each question.

1. APPLICANT DETAILS

1.1 (**M*) Applicant name:**	Shashank Kumar Gupta
1.2 **Supervisor (if applicable):**	Dr. Simon Box & Dr. R.E. Wilson
1.3 **Other researchers/collaborators (if applicable): Name, address, email, telephone**	

2. STUDY DETAILS

2.1 (M*) Title of study:	Investigate the techniques and algorithms that can be ubiquitously employed, for low cost indoor infrastructure free localization.
2.2 (M*) Type of study (*e.g. Undergraduate, Doctorate, Masters, Staff*):	Doctorate
2.3 i) (M*) Proposed start date:	04/01/2013
2.3 ii) (M*) Proposed end date:	03/03/2016

2.4 (M*) What are the aims and objectives of this study?
1. Investigating the data fusion techniques to pin point the location of the pedestrian based on the contextual information collected by several sensors in the smartphone. 2. Understanding the movement of the pedestrian. 3. Development of an algorithm for continuous localization of the pedestrian.

2.5 (M*) Background to study (*a **brief** rationale for conducting the study*)**:**

Location based services are becoming an indispensable part of our life. The wide adoption of satellite based positioning - Global Positioning System (GPS) has solved the problem of outdoor localization for a wide range of scenarios. Unfortunately, satellite based positioning is not possible indoors because of weak radio signals and loss of the direct line of sight from the satellites. Therefore, significant efforts have been motivated towards finding a practical solution for indoor localization.

Despite significant efforts, results vary in characteristics, performance and cost. They either need an expensive infrastructure deployment or have specialised hardware needs, or have low accuracy. Moreover, there is always an accuracy and cost trade-off. As a result of this, there is not a single ubiquitous indoor positioning solution available.

In regards to this, the research would investigate the techniques and algorithms that can be ubiquitously employed, for low cost indoor infrastructure free localization.

2.6 (M*) Key research question (*Specify hypothesis if applicable*):

This research study would try to answer to key questions:

The majority of available indoor positioning solutions are dependent on some form of infrastructure or hardware needs. This can possibly be reduced by developing innovative algorithms that utilize the natural features present in the environment to position the pedestrian, instead of relying on any form of infrastructure or specialized hardware needs. These natural features can be such as stair case, elevators, corners etc. So one of the main research question that this study would answer is that, Is it possible to position a pedestrian indoors using only the natural features present in environment, with minimal or no hardware usage?.

Recently smartphones have redefined the notion of mobile computing platforms. These phones are not just constrained to voice communication. Most of these phones are embedded with rich set of sensors such as accelerometers, microphones, magnetometers, camera etc. that can support localization and moreover they are ubiquitous and affordable by all. As a result the second key question that this study would address is that, What is the variability in the performance of proposed navigation technology across a representative population of smartphone users?.

2.7 (M*) Study design (*Give a **brief** outline of basic study design*) *Outline what approach is being used, why certain methods have been chosen.*

In order to localize the pedestrian indoors, it is required that localization technology is infrastructure free, accurate, reliable, ubiquitous and cost effective. Based on these needs, a detailed review of the state-of-art of localization techniques and available products that can support localization is done. It is noted that recently smartphones have redefined the notion of mobile computing platforms. These phones are not just constrained to voice communication. Most of these phones are embedded with rich set of sensors such as accelerometers, microphones, magnetometers, camera etc., and have cellular connectivity including Wi-Fi and GPS that can provide localization. Also, these are affordable by all. As a result this research utilises smartphones for positioning.

Technologies such as Wi-Fi have been effectively used for positioning pedestrians indoors in the recent past [1]. However, they need the deployment of Wi-Fi beacons and moreover the radio map varies with the change in physical conditions. This makes it non-preferable for the general purpose indoor localization. The other technology that has been effectively used for positioning pedestrian indoors is dead reckoning [2]. It involves the usage of inertial sensors - accelerometers and gyroscopes. A key advantage of this technology is that it offers 'autonomy' to pedestrian positioning. On contrary, the inertial sensors suffer from drift errors when integrated over time. As a result, this research focusses on designing innovative positioning solution using the inertial sensors embedded in smartphone while reducing the drifts to minimum. The key idea is that to recognise the pedestrian's stride using the accelerometer signal and their heading via gyroscope sensor embedded in smartphone. Methods such as peak detection [3], zero crossing [2], Fourier transform [4], etc. would be investigated to recognise the pedestrian's footsteps. The stride length would be estimated using methods such as Weinberg stride length model [5], Scarlett model [6], etc. and the pedestrian's heading shall be determined using gyroscope. To reduce the drifts methods such as map matching [7] would be employed and moreover, the data from various sensors shall be fused by techniques like Kalman filter [8].

[1] O. Costilla-Reyes and K. Namuduri, "Dynamic Wi-Fi Fingerprinting Indoor Positioning System," in International Conference on Indoor Positioning and Indoor Navigation, 2014.

[2] S. Beauregard and H. Haas, "Pedestrian dead reckoning: A basis for personal positioning," in Proceedings of the Third Workshop on Positioning, Navigation and Communication, 2006, pp. 27-35.

[3] A. Ali and N. El-Sheimy, "Low-Cost MEMS-Based Pedestrian Navigation Technique for GPS-Denied Areas," Journal of Sensors, 2013.

[4] L. Tan and J. Jiang, Digital signal processing: fundamentals and applications: Academic Press, 2013.

[5] H. Weinberg, "Using the ADXL202 in pedometer and personal navigation applications," Analog Devices AN-602 application note, 2002.

[6] J. Scarlett, "Enhancing the performance of pedometers using a single accelerometer," Application Note, Analog Devices, 2007.

[7] M. A. Quddus, W. Y. Ochieng, and R. B. Noland, "Current map-matching algorithms for transport applications: State-of-the art and future research directions," Transportation Research Part C: Emerging Technologies, vol. 15, No. 5, pp. 312-328, 2007.

[8] M. S. Grewal and A. P. Andrews, Kalman filtering: theory and practice using MATLAB: Wiley. com, 2011.

3. SAMPLE AND SETTING

3.1 (M*) How are participants to be *approached*? *Give details of what you will do if recruitment is insufficient. If participants will be accessed through a third party (e.g. children accessed via a school) state if you have permission to contact them and* ***upload any letters of agreement to your submission in ERGO****.*

Any person who is capable to walk is the participant of this research. In order to carry out this research, we will advertise in the faculty for participants (particularly aged greater than 18 years) familiar with using a smartphone. Since, it is an exploratory research involving predictive data analysis we will restrict the number of participants to 20. If in case we feel the participants are less or the collected data is insufficient we will re-advertise in the faculty.

3.2 (M*) Who are the proposed sample and where are they from (e.g. fellow students, club members)? *List inclusion/exclusion criteria if applicable. NB The University does not condone the use of 'blanket emails' for contacting potential participants (i.e. fellow staff and/or students).It is usually advised to ensure groups of students/staff have given prior permission to be contacted in this way, or to use of a third party to pass on these requests. This is because there is a potential to take advantage of the access to 'group emails' and the relationship with colleagues and subordinates; we therefore generally do not support this method of approach. If this is the only way to access a chosen cohort, a reasonable compromise is to obtain explicit approval from the Faculty Ethics Committee (FEC) and also from a senior member of the Faculty in case of complaint.*

Any person who is capable to walk is the participant of this research. In order to carry out this research, we will advertise in the faculty for participants (particularly aged greater than 18 years) familiar with using a smartphone. The advertisement will be done by sticking leaflets in the faculty. It is expected that the participants would generally be students and staff members.

3.3 (M*) Describe the relationship between researcher and sample (*Describe any relationship e.g. teacher, friend, boss, clinician, etc.*)
None

3.4 (M*) Describe how you will ensure that fully informed consent is being given: (*include how long participants have to decide whether to take part*)
Based on the participant information i.e. verbal consent, it can be said that they fully approve of carrying on the data collection work. In total, the data collection would be taking less than 10 minutes.

4. RESEARCH PROCEDURES, INTERVENTIONS AND MEASUREMENTS

4.1 (M*) Give a brief account of the procedure as experienced by the participant *(Make clear who does what, how many times and in what order. Make clear the role of all assistants and collaborators. Make clear total demands made on participants, including time and travel).* ***Upload any copies of questionnaires and interview schedules to your submission in ERGO***.
Initial contact to surveyors about data collection. During the data collection, participant walks on the designated route indoors (selected such that it has least number of obstacles) two times, while carrying smartphone in their hands. In total, the experiment takes less than 10 minutes. The participant is asked to follow the instructions shown in the attached 'Instruction set form'

5. STUDY MANAGEMENT

5.1 (M*) State any potential for psychological or physical discomfort and/or distress?
None

5.2 (M*) Explain how you intend to alleviate any psychological or physical discomfort and/or distress that may arise? (if applicable)
N/A

5.3 Explain how you will care for any participants in 'special groups' *(i.e. those in a dependent relationship, vulnerable or lacking in mental capacity)* **(if applicable)?**
N/A

5.4 Please give details of any payments or incentives being used to recruit participants (if applicable)?
N/A

5.5 i) How will participant anonymity and/or data anonymity be maintained (if applicable)? *Two definitions of anonymity exist: i)* Unlinked anonymity **-** *Complete anonymity can only be promised if questionnaires or other requests for information are not targeted to, or received from, individuals using their name or address or any other identifiable characteristics. For example if questionnaires are sent out with no possible identifiers when returned, or if they are picked up by respondents in a public place, then anonymity can be claimed. Research methods using interviews cannot usually claim anonymity – unless using telephone interviews when participants dial in. ii)* Linked anonymity **-** *Using this method, complete anonymity cannot be promised because participants can be identified; their data may be coded so that participants are not identified by researchers, but the information provided to participants should indicate that they could be linked to their data.*
Unlinked anonymity - the data collected from participants is stored in form of codes, that are randomly chosen.

5.5 ii) How will participant confidentiality be maintained (if applicable)? *Confidentiality is defined as the non-disclosure of research information except to another authorised person. Confidential information can be shared with those who are already party to it, and may also be disclosed where the person providing the information provides explicit consent.*
The study is an investigative research that will use exploratory data analysis and predictive data analysis on the collected data, no inference of any form or to any individual is intended. Moreover, the data collected from the participants is is stored in the form of random codes. Therefore, looking at the codes it cannot be interpreted whoes data it is. This would further assure the confidentiality to the participants.

5.6 (M*) How will personal data and study results be stored securely during and after the study? *Researchers should be aware of, and compliant with, the Data Protection policy of the University. You must be able to demonstrate this in respect of handling, storage and retention of data.*
The research complies with the Data Protection Act. All data collected by pedestrians will be stored in password secured computers. It will be treated confidentially and will remain anonymous in all research output.

5.7 (M*) Who will have access to these data?
Researchers directly involved in this study only.

N.B. – Before you upload this document to your ERGO submission remember to:

1. Complete ALL mandatory sections in this form
2. Upload any letters of agreement referred to in question 3.1 to your ERGO submission
3. Upload any interview schedules and copies of questionnaires referred to in question 4.1

Selection criteria – Participant Information Sheet

Study Title:

Investigate the techniques and algorithms that can be ubiquitously employed, for low cost indoor infrastructure free localization.

Researcher: Shashank Kumar Gupta
Ethics number: 16756

Please read this information carefully before deciding to take part in this research. If you are happy to participate then only we may proceed.

What is the research about?
As per the survey conducted by National Human Activity Pattern Survey [1], almost 80% of the time we spend indoors, either in offices, museums or shopping complexes, etc. But still there is not a single viable commercial indoor positioning solution that is reliable, fast, accurate and robust such as GPS for outdoors.

Therefore, this research project as a part of my Ph.D., would investigate the techniques and algorithms that can be ubiquitously employed for localization of the pedestrians indoors. This may help in developing a viable commercial indoor positioning solution in near future.

[1] N. E. Klepeis, W. C. Nelson, W. R. Ott, J. P. Robinson, A. M. Tsang, P. Switzer, *et al.*, "The National Human Activity Pattern Survey (NHAPS): a resource for assessing exposure to environmental pollutants," Journal of exposure analysis and environmental epidemiology, vol. 11, pp. 231-252, 2001.

Why have I been chosen?
We are looking for participants who spend time indoors, in essence everyone. At the current moment this research is focussed towards pedestrians only.

The data collected from pedestrian's walks will help me in developing a robust positioning algorithm; that can be a useful asset for my research.

What will happen to me if I take part?
If a person takes part, he/she will be explained the instructions in less than 2 minutes. Following this, the data collection would be taking less than 10 minutes.

It will be an independent choice of theirs whether they want to get involved or not, and no follow up would be needed.

Are there any benefits in my taking part?
The results of this study will help to provide knowledge about innovative techniques that can be utilised by industries in developing technologies for indoor positioning. Or alternatively, it can also be utilised in designing a commercial viable indoor positioning application.

Are there any risks involved?
'Privacy', however private information about the pedestrian walking characteristics, derived from the collected data will not be shared to anyone. Moreover, it will be stored in the password secured computers.

Will my participation be confidential?

Yes the participation of the pedestrian will remain be confidential, and will be anonymised in all outputs from this project.

What happens if I change my mind?
In any case, and at any point of the time he/she can withdraw from this survey and the data that they have provided shall be deleted.

What happens if something goes wrong?
In the unlikely case of concern or complaint about participating in this study, you can contact, Research Governance officer (c/o: Miss Trudi Bartlett) (Phone: 02380598848, Id: rgoinfo@soton.ac.uk)

Where can I get more information?
For more details about this project or concerning any queries relevant to the project, please contact me (Shashank Gupta) by email s.k.gupta@soton.ac.uk.

References

[1] Jonas Callmer, "Autonomous Localization in Unknown Environments," M.S. thesis, Linköping Studies Science Technology, Linköping, Sweden, 2013.

[2] Tony Shu-Hsien Wang, Dian Tjondronegoro, Michael Docherty, Wei Song, and Joshua Fuglsang, "A recommendation for designing mobile pedestrian navigation system in university campuses," in Proceedings of the 25th Australian Computer-Human Interaction Conference: Augmentation, Application, Innovation, Collaboration, 2013, pp. 3-12.

[3] Paul D Groves, Principles of GNSS, inertial, and multisensor integrated navigation systems, Second ed.: Artech House, 2013.

[4] Roy Want, Andy Hopper, Veronica Falcao, and Jonathan Gibbons, "The active badge location system." ACM Transactions on Information Systems (TOIS), vol. 10, No. 1, pp. 91-102, 1992.

[5] Hui Liu, Houshang Darabi, Pat Banerjee, and Jing Liu, "Survey of wireless indoor positioning techniques and systems." IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol. 37, No. 6, pp. 1067-1080, 2007.

[6] Jack Steenstra, Alexander Gantman, Kirk Taylor, and Liren Chen, "Location based service (LBS) system and method for targeted advertising," U.S Patent No. CA2588241 A1, 2004.

[7] foursquare Inc., "foursquare." Available: https://foursquare.com/. Accessed on: 31 September 2016.

[8] BrightKite Inc., "Unlimited Group Texting Free. Everywhere." Available: http://brightkite.com/. Accessed on: 30 September 2016.

[9] RallyUp Inc., "I am pretty private, if I don't want to be on the radar for whatever reason, i don't have to." Available: http://www.getupandrally.com/. Accessed on: 30 September 2016.

[10] Kathryn Zickuhr, "Location-Based Services," Pew Research Center's Internet & American Life Project, U.S, 2013.

[11] Jim Martin, "Best sat nav 2016 UK: the best sat navs and GPS units you can buy in the UK." Available: http://www.pcadvisor.co.uk/buying-advice/gadget/best-sat-nav-2016-uk-3645029/. Accessed on: 30 September 2016,2016.

[12] European Global NAvigation Satellite Systems Agency, "GSA Releases 2012 GNSS Market Report." Available: https://www.gsa.europa.eu/news/gsa-releases-2012-gnss-market-report. Accessed on: 30 September 2016.

[13] Michael Brown, James Pinchin, and Chris Hide, "Opening Indoors: The Advent of Indoor Positioning," in Proceedings of the International Conference on Ergonomics & Human Factors, 2013.

[14] Dean Takahashi, "Finnish engineers use earths magnetic field for indoor map navigations." Available: http://venturebeat.com/2012/07/15/finnish-engineers-used-earths-magnetic-field-for-indoor-map-navigation/. Accessed on: 30 September 2016.

[15] Lingling Zhu, Aolei Yang, Dingbing Wu, and Li Liu, "Survey of Indoor Positioning Technologies and Systems," in Life System Modeling and Simulation, Springer Berlin Heidelberg, 2014, pp. 400-409.

[16] Jarice Hanson, 24/7: how cell phones and the Internet change the way we live, work, and play: Greenwood Publishing Group, 2007.

[17] Navizon, "Navizon Accurate Positioning Anywhere." Available: https://www.navizon.com/indoor-positioning-accuware. Accessed on: 31 August 2016.

[18] Skyhook Wireless Inc., "Skyhook." Available: http://www.skyhookwireless.com. Accessed on: 18 August,2015.

[19] Apple Inc., "About iBeacon on your iPhone, iPad, and iPod touch." Available: https://support.apple.com/en-gb/HT202880. Accessed on: 31 August 2016

[20] Ye Tian, Bruce Denby, Iness Ahriz, Pierre Roussel, and Gérard Dreyfus, "Robust indoor localization and tracking using GSM fingerprints." EURASIP Journal on Wireless Communications and Networking, vol. 2015, No. 1, pp. 1-12, 2015.

[21] Aare Puussaar, "Indoor Positioning Using WLAN Fingerprinting with Post-Processing Scheme," Master thesis, Faculty of Science and Technology, Tartu Ülikool, 2014.

[22] Nissanka B Priyantha, Anit Chakraborty, and Hari Balakrishnan, "The cricket location-support system," in Proceedings of the 6th annual international conference on Mobile computing and networking, 2000, pp. 32-43.

[23] Kimmo Kalliola, "High accuracy indoor positioning based on BLE." Nokia research center presentation, Available: https://www.tekes.fi/globalassets/global/ohjelmat-ja-palvelut/ohjelmat/ubicom/aineistot/tilaisuuksien-ja-seminaarien-materiaalit-1/20110411_bluetoothlowenergy/seminaari_20110411-haip-indoor-positioning-kimmokalliola-nokia.pdf. Accessed on: 30 September 2016.

[24] Navisens Inc., "Navisens." Available: http://www.navisens.com/. Accessed on: 30 August 2016.

[25] Thomas King, Stephan Kopf, Thomas Haenselmann, Christian Lubberger, and Wolfgang Effelsberg, "COMPASS: A probabilistic indoor positioning system based on 802.11 and digital compasses," in Proceedings of the 1st international workshop on Wireless network testbeds, experimental evaluation & characterization, 2006, pp. 34-40.

[26] Ewa Niewiadomska-Szynkiewicz, "Localization in wireless sensor networks: Classification and evaluation of techniques." International Journal of Applied Mathematics and Computer Science, vol. 22, No. 2, pp. 281-297, 2012.

[27] Anshul Rai, Krishna Kant Chintalapudi, Venkata N Padmanabhan, and Rijurekha Sen, "Zee: zero-effort crowdsourcing for indoor localization," in Proceedings of the 18th annual international conference on Mobile computing and networking, 2012, pp. 293-304.

[28] Zhuoling Xiao, Hongkai Wen, Andrew Markham, and Niki Trigoni, "Lightweight map matching for indoor localisation using conditional random fields," in Proceedings of the 13th international symposium on Information processing in sensor networks, 2014, pp. 131-142.

[29] Nattapong Swangmuang, A location fingerprint framework towards efficient wireless indoor positioning systems: ProQuest, 2008.

[30] Kentaro Nishi, Kota Tsubouchi, and Masamichi Shimosaka, "Hourly pedestrian population trends estimation using location data from smartphones dealing with temporal and spatial sparsity," in Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, 2014, pp. 281-290.

[31] Yves-Alexandre De Montjoye, César A Hidalgo, Michel Verleysen, and Vincent D Blondel, "Unique in the crowd: The privacy bounds of human mobility." Scientific reports, vol. 3, No. __, 2013.

[32] Veljo Otsason, "Accurate indoor localization using wide GSM fingerprinting," Master's thesis, Faculty of Mathematics and Computer Science, University of Tartu, Estonia, 2005.

[33] Maurice F Fallon, Hordur Johannsson, and John J Leonard, "Efficient scene simulation for robust Monte Carlo localization using an RGB-D camera," in IEEE International Conference on Robotics and Automation (ICRA), 2012, pp. 1663-1670.

[34] Andrew R Golding and Neal Lesh, "Indoor navigation using a diverse set of cheap, wearable sensors," in The Third International Symposium on Wearable Computers, 1999, pp. 29-36.

[35] Lionel M Ni, Yunhao Liu, Yiu Cho Lau, and Abhishek P Patil, "LANDMARC: indoor location sensing using active RFID." Wireless networks, vol. 10, No. 6, pp. 701-710, 2004.

[36] Martin Azizyan, Ionut Constandache, and Romit Roy Choudhury, "SurroundSense: mobile phone localization via ambience fingerprinting," in ACM Proceedings of the 15th

Annual International Conference on Mobile Computing and Networking, 2009, pp. 261-272.

[37] Ahmed Sherif, Hanan Sabry, Rasha Arafa, and Ayman Wagdy, "Energy efficient hospital patient room design: effect of room shape on window-to-wall ratio in a desert climate," in 30th International plea conference, CEPT University, Ahmedabad, India, 2014.

[38] Division of the University Architect, "Design Guidance: Learning Environments," University of Cincinnati, U.S, 2003.

[39] Gerald Glanzer, Thomas Bernoulli, Thomas Wiessflecker, and Ulrich Walder, "Semi-autonomous indoor positioning using MEMS-based inertial measurement units and building information," in 6th Workshop on Positioning, Navigation and Communication, 2009, pp. 135-139.

[40] Nicholas D Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T Campbell, "A survey of mobile phone sensing." IEEE Communications Magazine, vol. 48, No. 9, pp. 140-150, 2010.

[41] Zheng Yang, Chenshu Wu, Zimu Zhou, Xinglin Zhang, Xu Wang, and Yunhao Liu, "Mobility increases localizability: a survey on wireless indoor localization using inertial sensors." ACM Computing Surveys (CSUR), vol. 47, No. 3, 2015.

[42] Shashank Kumar Gupta, Simon Box, and RE Wilson, "Low cost infrastructure free form of indoor positioning," in IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2014.

[43] Google Inc., "Google Glass Help." 2015, Available: https://support.google.com/glass/answer/3064128?hl=en. Accessed on: 18 August,2015.

[44] VR Oculus, "Oculus rift-virtual reality headset for 3d gaming." Available: https://www.oculus.com/. Accessed on: 30 September 2016.

[45] Philipp Lukas Bolliger, "Robust indoor positioning through adaptive collaborative labeling of location fingerprints," Ph.D. thesis, ETH ZURICH, 2011.

[46] Jeffrey Hightower, Barry Brumitt, and Gaetano Borriello, "The location stack: A layered model for location in ubiquitous computing," in IEEE Proceedings Fourth Workshop on Mobile Computing Systems and Applications 2002, pp. 22-28.

[47] Jeffrey Hightower and Gaetano Borriello, "Location systems for ubiquitous computing." Computer, vol. 34, No. 8, pp. 57-66, 2001.

[48] Jeffrey Hightower and Gaetano Borriello, "A Survey and Taxonomy of Location Systems for Ubiquitous Computing." IEEE computer, vol. 34, No. 8, pp. 57-66, 2001.

[49] Jiang Xiao, Zimu Zhou, Youwen Yi, and Lionel M Ni, "A Survey on Wireless Indoor Localization from the Device Perspective." ACM Computing Surveys (CSUR), vol. 49, No. 2, 2016.

[50] Björn Bittins, "Indoor Navigation Based on a Multimodal Positioning System," MSc. thesis, Economic Sciences II, Applied Sciences Berlin, Germany, 2012.

[51] Zahid Farid, Rosdiadee Nordin, and Mahamod Ismail, "Recent Advances in Wireless Indoor Localization Techniques and System." Journal of Computer Networks and Communications, vol. 2013, No. _, pp. 1-10, 2013.

[52] Ruizhi Chen, Ubiquitous Positioning and Mobile Location-Based Services in Smart Phones: IGI Global, 2012.

[53] Gu Yanying, A. Lo, and I. Niemegeers, "A survey of indoor positioning systems for wireless personal networks." IEEE Communications Surveys & Tutorials, vol. 11, No. 1, pp. 13-32, 2009.

[54] Cisco Systems Inc., "Wi-Fi Location-Based Services 4.1 Design Guide ", Available: http://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Mobility/WiFiLBS-DG.pdf. Accessed on: 18 August,2015.

[55] Axel Küpper, Location-based services: fundamentals and operation: John Wiley & Sons, 2005.

[56] Leslie W Barclay, Propagation of radiowaves, Second ed.: Institute of Engineering and Technology, 2003.

[57] ITU-R P.1546 Recommendation, "Method for point-to-area predictions for terrestrial services in the frequency range 30 MHz to 3000 MHz," Technical Report, International Telecommunication Union, 2001.

[58] Andrei Popleteev, "Indoor positioning using FM radio signals," Ph.D thesis, University of Trento, 2011.

[59] Homayoun Hashemi, "The indoor radio propagation channel." Proceedings of the IEEE, vol. 81, No. 7, pp. 943-968, 1993.

[60] Theofilos Chrysikos, Giannis Georgopoulos, and Stavros Kotsopoulos, "Site-specific validation of ITU indoor path loss model at 2.4 GHz," in IEEE Workshop on Advanced Experimental Activities on Wireless Networks and Systems, 2009, pp. 1-6.

[61] Christos Laoudias, "Π3: Overview of location techniques." Available: http://www2.ucy.ac.cy/~laoudias/pages/penek/deliverables/D3.pdf. Accessed on: 18 August,2015.

[62] 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, 2012, pp. 197-210.

[63] Jorge Torres-Solis, Tiago H Falk, and Tom Chau, A review of indoor localization technologies: towards navigational assistance for topographical disorientation: INTECH Open Access Publisher, 2010.

[64] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins, GPS-Global Positioning System. Theory and practice.: Springer, 1997.

[65] Simon Banville, Paul Collins, and François Lahaye, "Concepts for Undifferenced GLONASS Ambiguity Resolution," in Proceedings of the 26th International Technical Meeting of the Satellite Division of The Institute of Navigation, 2013, pp. 1186-1197.

[66] Vyasaraj Guru Rao, Gérard Lachapelle, and SB Vijaykumar, "Analysis of IRNSS over Indian subcontinent," in Proceedings of the 2011 International Technical Meeting of The Institute of Navigation, 2011, pp. 1150-1162.

[67] Elliott Kaplan and Christopher Hegarty, Understanding GPS: principles and applications: Artech house, 2005.

[68] Shun Miura, Shoma Hisaka, and Shunsuke Kamijo, "GPS multipath detection and rectification using 3D maps," in 16th IEEE International Conference on Intelligent Transportation Systems-(ITSC), 2013, pp. 1528-1534.

[69] Wei Chen, Ruizhi Chen, Yuwei Chen, Heidi Kuusniemi, and Jianyu Wang, "An effective pedestrian dead reckoning algorithm using a unified heading error model," in Proceedings of the IEEE/ION PLANS, 2010, pp. 340-347.

[70] Ivan Spassov, "Algorithms for map-aided autonomous indoor pedestrian positioning and navigation," DSc. thesis, Faculty Of the Natural, Architectural and Constructed Environment, École Polytechnique Fédérale De Lausanne, France, 2007.

[71] Oleg A Mezentsev, Sensor aiding of HSGPS pedestrian navigation: Citeseer, 2006.

[72] Gérard Lachapelle, H Kuusniemi, Diep TH Dao, G MacGougan, and ME Cannon, "HSGPS signal analysis and performance under various indoor conditions." Navigation, vol. 51, No. 1, pp. 29-43, 2004.

[73] Gérard Lachapelle, "Pedestrian navigation with high sensitivity GPS receivers and MEMS." Personal and Ubiquitous Computing, vol. 11, No. 6, pp. 481-488, 2007.

[74] Joel Barnes, Chris Rizos, Jinling Wang, David Small, Gavin Voigt, and Nunzio Gambale, "Locata: A new positioning technology for high precision indoor and outdoor positioning," in Proceedings of the International Symposium on GPS/GNSS, 2003, pp. 9-18.

[75] Rui Xu, Wu Chen, Ying Xu, and Shengyue Ji, "A New Indoor Positioning System Architecture Using GPS Signals." Sensors, vol. 15, No. 5, pp. 10074-10087, 2015.

[76] Vasileios Zeimpekis, George M Giaglis, and George Lekakos, "A taxonomy of indoor and outdoor positioning techniques for mobile location services." ACM SIGecom Exchanges, vol. 3, No. 4, pp. 19-27, 2002.
[77] H.A. Karimi, Indoor Wayfinding and Navigation: CRC Press, 2015.
[78] Alan Bensky, Wireless Positioning Technologies and Applications, Artech House, 2007.
[79] Balaram Singh, Soumya Pallai, and Susil Kumar Rath, "A Survey of Cellular Positioning Techniques in GSM Networks," in National Conference in Mobile Computing-NCMC'12, 2012.
[80] Tomislav Kos, Mislav Grgic, and Gordan Sisul, "Mobile user positioning in GSM/UMTS cellular networks," in 48th International Symposium on Multimedia Signal Processing and Communications, 2006, pp. 185-188.
[81] Veljo Otsason, Alex Varshavsky, Anthony LaMarca, and Eyal De Lara, "Accurate gsm indoor localization," in UbiComp 2005: Ubiquitous Computing, Springer, 2005, pp. 141-158.
[82] Eyal de Lara and Stefan Saroiu, "CILoS: a CDMA indoor localization system," in Proceedings of the 10th international conference on Ubiquitous computing, 2008, pp. 104-113.
[83] Md Arafin Mahamud and Mahfuzulhoq Chowdhury, "Indoor Location System with Wi-Fi and Alternative Cellular Network Signal." International Journal of Multimedia and Ubiquitous Engineering, vol. 10, No. 3, pp. 59-70, 2015.
[84] Krzysztof W Kolodziej and Johan Hjelm, Local positioning systems: LBS applications and services: CRC press, 2010.
[85] Mikkel Baun Kjærgaard, Mads Vering Krarup, Allan Stisen, Thor Siiger Prentow, Henrik Blunck, Kaj Grønbæk, and Christian S Jensen, "Indoor positioning using wi-fi–how well is the problem understood?," in IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2013.
[86] Anthea Wain Sy Au, "RSS-Based WLAN Indoor Positioning and Tracking System Using Compressive Sensing and Its Implementation on Mobile Devices," Master's thesis, University of Toronto, 2010.
[87] Octavian Manu, "A Study of Indoor Localization Techniques." DOCT-US, vol. 1, No. 2, 2009.
[88] Kevin P Murphy, Machine learning: a probabilistic perspective: MIT press, 2012.
[89] Moustafa Youssef and Ashok Agrawala, "The Horus WLAN location determination system," in Proceedings of the 3rd international conference on Mobile systems, applications, and services, 2005, pp. 205-218.
[90] Chenshu Wu, Zheng Yang, Yunhao Liu, and Wei Xi, "WILL: Wireless indoor localization without site survey." IEEE Transactions on Parallel and Distributed Systems, vol. 24, No. 4, pp. 839-848, 2013.
[91] Paramvir Bahl and Venkata N Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in IEEE Proceedings on Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, 2000, pp. 775-784.
[92] Padraig Cunningham and Sarah Jane Delany, "k-Nearest neighbour classifiers," Multiple Classifier Systems, 2007.
[93] Sebastian Fudickar and Markus Valentin, "Most accurate algorithms for RSS-based Wi-Fi indoor localisation," in 5th International Conference on Indoor Positioning and Indoor Navigation, 2014.
[94] Filip Lemic, Arash Behboodi, Vlado Handziski, and Adam Wolisz, "Experimental Decomposition of the Performance of Fingerprinting-based Localization Algorithms," in International conference on Indoor Positioning and Navigation (IPIN), 2014.
[95] Omar Costilla-Reyes and Kamesh Namuduri, "Dynamic Wi-Fi Fingerprinting Indoor Positioning System," in International Conference on Indoor Positioning and Indoor Navigation, 2014.

[96] Paolo Baronti, Prashant Pillai, Vince WC Chook, Stefano Chessa, Alberto Gotta, and Y Fun Hu, "Wireless sensor networks: A survey on the state of the art and the 802.15. 4 and ZigBee standards." Computer communications, vol. 30, No. 7, pp. 1655-1695, 2007.

[97] Xin Hu, Lianglun Cheng, and Guangchi Zhang, "A Zigbee-based localization algorithm for indoor environments," in International Conference on Computer Science and Network Technology (ICCSNT), 2011, pp. 1776-1781.

[98] Luca Mainetti, Luigi Patrono, and Ilaria Sergi, "A survey on indoor positioning systems," in 22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM), 2014, pp. 111-120.

[99] Roy Want, "An introduction to RFID technology." IEEE Pervasive Computing, vol. 5, No. 1, pp. 25-33, 2006.

[100] Takeshi Tomioka and Takanao Ochiai, "RFID tag and antenna arranging method," U.S Patent No. 20060232419 A1, 2005.

[101] Zhongliang Deng, Yanpei Yu, Xie Yuan, Neng Wan, and Lei Yang, "Situation and development tendency of indoor positioning." Communications, China, vol. 10, No. 3, pp. 42-55, 2013.

[102] Mathieu Bouet and Aldri L dos Santos, "RFID tags: Positioning principles and localization techniques," in First IFIP Wireless Days 2008, pp. 1-5.

[103] Ricardo Tesoriero, J Gallud, Manuel Lozano, and V Penichet, "Using active and passive RFID technology to support indoor location-aware systems." IEEE Transactions on Consumer Electronics, vol. 54, No. 2, pp. 578-583, 2008.

[104] Martijn Kiers, W Bischof, M Dornhofer, and E Krajnc, "A New Approach for an RFID Indoor Positioning System Without Fixed Coordinates for Visually Impaired and Blind People," in IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2010, pp. 317-318.

[105] Martijn Kiers, "Evaluation and improvements of an rfid based indoor navigation system for visually impaired and blind people," in IEEE International Conference on Indoor Positioning and Indoor Navigation, 2011.

[106] Jeffrey Hightower, "SpotON: An indoor 3D location sensing technology based on RF signal strength." UW CSE 00-02-02, University of Washington, Department of Computer Science and Engineering, Seattle, WA, No. 2000.

[107] Yuntian Brian Bai, Suqin Wu, Hongren Wu, and Kefei Zhang, "Overview of RFID-based indoor positioning technology," in Proceedings of the Geospatial Science Research Symposium. RMIT University, 2012.

[108] Giovanni Bellusci, "Ultra-wideband ranging for low-complexity indoor positioning applications," Doctor thesis, TU Delft, Delft University of Technology, 2011.

[109] Gabriel Deak, Kevin Curran, and Joan Condell, "A survey of active and passive indoor localisation systems." Computer Communications, vol. 35, No. 16, pp. 1939-1954, 2012.

[110] Sinan Gezici and H Vincent Poor, "Position estimation via ultra-wide-band signals." Proceedings of the IEEE, vol. 97, No. 2, pp. 386-403, 2009.

[111] Mohammad Ghavami, Lachlan Michael, and Ryuji Kohno, Ultra wideband signals and systems in communication engineering: John Wiley & Sons, 2007.

[112] Suheer Alhadhrami, AbdulMalik Al-Salman, Hend Al-Khalifa, Abdulrahman Alarifi, Ahmad Alnafessah, Mansour Alsaleh, and Mai Al-Ammar, "Ultra Wideband Positioning:An Analytical Study of Emerging Technologies," in 8th International Conference on Sensor Technologies and Applications, 2014, pp. 66- 74.

[113] Esmond Mok, Linyuan Xia, Guenther Retscher, and Hui Tian, "A case study on the feasibility and performance of an UWB-AoA real time location system for resources management of civil construction projects." Journal of Applied Geodesy, vol. 4, No. 1, pp. 23-32, 2010.

[114] Suheer Alhadhrami, A Al-Salman, H Al-Khalifa, Abdulrahman Alarifi, Ahmad Alnafessah, Mansour Alsaleh, and M Al-Ammar, "Ultra wideband positioning: An

analytical study of emerging technologies," in Proceedings of the Eighth International Conference on Sensor Technologies and Applications, SENSORCOMM, 2014, pp. 1-9.

[115] Zhi Ning Chen, "UWB antennas: Design and application," in 6th International Conference on Information, Communications & Signal Processing, 2007, pp. 1-5.

[116] Ubisense Inc., "Ubisense Research Package." Available: http://pdf.directindustry.com/pdf/ubisense/research-package/124957-508091.html. Accessed on: 18 August,2015.

[117] Lei Chen and Ouyang Shan, "Through-wall Surveillance using Ultra-wideband Short Pulse Radar: Numerical Simulation," in 2nd IEEE Conference on Industrial Electronics and Applications 2007, pp. 1551-1554.

[118] D Schneider, "You are here." Spectrum, IEEE, vol. 50, No. 12, pp. 34-39, 2013.

[119] Anja Bekkelien, Michel Deriaz, and Stéphane Marchand-Maillet, "Bluetooth indoor positioning," Master's thesis, University of Geneva, 2012.

[120] R. P. Minch, "Privacy issues in location-aware mobile devices," in IEEE Proceedings of the 37th Annual Hawaii International Conference on System Sciences, 2004.

[121] Mortaza S Bargh and Robert de Groote, "Indoor localization based on response rate of bluetooth inquiries," in Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments, 2008, pp. 49-54.

[122] Javier JM Diaz, Rodrigo de A Maues, Rodrigo B Soares, Eduardo F Nakamura, and Carlos MS Figueiredo, "Bluepass: An indoor bluetooth-based localization system for mobile applications," in IEEE Symposium on Computers and Communications (ISCC), 2010, pp. 778-783.

[123] David Madigan, E Einahrawy, Richard P Martin, Wen-Hua Ju, Parameshwaran Krishnan, and AS Krishnakumar, "Bayesian indoor positioning systems," in Proceedings of the 24th Annual Joint Conference of IEEE Computer and Communications Societies, 2005, pp. 1217-1227.

[124] Binghao Li, Thomas Gallagher, Andrew G Dempster, and Chris Rizos, "How feasible is the use of magnetic field alone for indoor positioning?," in IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2012, pp. 1-9.

[125] Rainer Mautz, "Indoor positioning technologies," Habilitation Thesis, ETH Zürich, 2012.

[126] Jörg Blankenbach and Abdelmoumen Norrdine, "Position estimation using artificial generated magnetic fields," in IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2010, pp. 1-5.

[127] Janne Haverinen and Anssi Kemppainen, "A global self-localization technique utilizing local anomalies of the ambient magnetic field," in IEEE International Conference on Robotics and Automation, 2009, pp. 3142-3147.

[128] Yang Liu, Marzieh Dashti, and Jie Zhang, "Indoor localization on mobile phone platforms using embedded inertial sensors," in 10th Workshop on Positioning Navigation and Communication (WPNC), 2013, pp. 1-5.

[129] IndoorAtlas Ltd., "Ambient magnetic field-based indoor location technology: Bringing the compass to the next level." IndoorAtlas Ltd, Available: http://web.indooratlas.com/web/WhitePaper.pdf. Accessed on: 18 August,2015.

[130] Jaewoo Chung, Matt Donahoe, Chris Schmandt, Ig-Jae Kim, Pedram Razavai, and Micaela Wiseman, "Indoor location sensing using geo-magnetism," in Proceedings of the 9th international conference on Mobile systems, applications, and services, 2011, pp. 141-154.

[131] Brandon Gozick, Kalyan Pathapati Subbu, Ram Dantu, and Tomyo Maeshiro, "Magnetic maps for indoor navigation." IEEE Transactions on Instrumentation and Measurement, vol. 60, No. 12, pp. 3883-3891, 2011.

[132] Muhammad Haris Afzal, Valerie Renaudin, and Gerard Lachapelle, "Multi-magnetometer based perturbation mitigation for indoor orientation estimation." Navigation, vol. 58, No. 4, pp. 279-292, 2012.

[133] Oliver J Woodman, "An introduction to inertial navigation," Technical Report 696, University of Cambridge, 2007.
[134] David Titterton, Strapdown inertial navigation technology vol. 17: IET, 2004.
[135] Mohinder S Grewal, Lawrence R Weill, and Angus P Andrews, Global positioning systems, inertial navigation, and integration: John Wiley & Sons, 2007.
[136] Khairi Abdul Rahim, "Heading drift mitigation for low-cost inertial pedestrian navigation," Ph.D. thesis, University of Nottingham, 2012.
[137] Vladislav Apostolyuk, "Theory and Design of Micromechanical Vibratory Gyroscopes," in MEMS/NEMS: Handbook Techniques and Applications, C. T. Leondes, Boston, MA, Springer US, 2006, pp. 173-195.
[138] Robin Mannings, Ubiquitous positioning: Artech House, 2008.
[139] Robert Harle, "A survey of indoor inertial positioning systems for pedestrians." IEEE Communications Surveys & Tutorials, vol. 15, No. 3, pp. 1281-1293, 2013.
[140] Oliver Woodman and Robert Harle, "Pedestrian localisation for indoor environments," in Proceedings of the 10th international conference on Ubiquitous computing, 2008, pp. 114-123.
[141] Michael Cramer, "GPS/INS Integration," in Photogrammetric Week '97. Proceedings, Wichmann Verlag, Heidelberg, 1997.
[142] Yang Liu, "An indoor pedestrian localisation system with self-calibration capability," Ph.D. thesis, University of Sheffield, 2016.
[143] Jessica Rose, James Gibson Gamble, and Janet M Adams, Human walking: Lippincott Williams & Wilkins Philadelphia, 2006.
[144] Ross Grote Stirling, "Development of a pedestrian navigation system using shoe mounted sensors," Citeseer, 2004.
[145] Azkario Rizky Pratama and R Hidayat, "Smartphone-based Pedestrian Dead Reckoning as an indoor positioning system," in International Conference on System Engineering and Technology (ICSET), 2012, pp. 1-6.
[146] Moustafa Alzantot and Moustafa Youssef, "UPTIME: Ubiquitous pedestrian tracking using mobile phones," in IEEE Wireless Communications and Networking Conference (WCNC), 2012, pp. 3204-3209.
[147] Quentin Ladetto, Vincent Gabaglio, and Josephus Van Seeters, "Pedestrian navigation method and apparatus operative in a dead reckoning mode," U.S Patent No. EP1253404 A2, 2004.
[148] Thomas Judd and Robert W Levi, "Dead reckoning navigational system using accelerometer to measure foot impacts," U.S Patent No. 5583776, 1996.
[149] R.W. Levi and R.R. Marshall, "Navigation device for personnel on foot," U.S. Patent No. 6813582, 2004.
[150] L Sher, "Personal inertial navigation system (PINS)," DARPA, 1996.
[151] Eric Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors." Computer Graphics and Applications, vol. 25, No. 6, pp. 38-46, 2005.
[152] Mohinder S Grewal and Angus P Andrews, Kalman filtering: theory and practice using MATLAB: Wiley. com, 2011.
[153] Khairi Abdulrahim, C Hide, T Moore, and C Hill, "Integrating low cost IMU with building heading in indoor pedestrian navigation." Journal of Global Positioning Systems, vol. 10, No. 1, pp. 30-38, 2011.
[154] Michael Angermann and Patrick Robertson, "Footslam: Pedestrian simultaneous localization and mapping without exteroceptive sensors—hitchhiking on human perception and cognition." Proceedings of the IEEE, vol. 100, No. Special Centennial Issue, pp. 1840-1848, 2012.
[155] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard, "Improved techniques for grid mapping with rao-blackwellized particle filters." IEEE transactions on Robotics, vol. 23, No. 1, pp. 34-46, 2007.

[156] Slawomir Grzonka, Giorgio Grisetti, and Wolfram Burgard, "Towards a navigation system for autonomous indoor flying," in IEEE International Conference on Robotics and Automation, 2009, pp. 2878-2883.
[157] Oxford Technical Solutions Inc., "Rt3000 inertial and gps navigation system." Available: http://www.oxts.com/products/rt3000-family/. Accessed on: 18 August 2015.
[158] Yunye Jin, Hong-Song Toh, Wee-Seng Soh, and Wai-Choong Wong, "A robust dead-reckoning pedestrian tracking system with low cost sensors," in IEEE International Conference on Pervasive Computing and Communications (PerCom), 2011, pp. 222-230.
[159] Shahid Ayub, Xiaowei Zhou, Soroush Honary, Alireza Bahraminasab, and Bahram Honary, "Indoor pedestrian displacement estimation using Smart phone inertial sensors." International Journal of Innovative Computing and Applications, vol. 4, No. 1, pp. 35-42, 2012.
[160] Juan Carlos Alvarez, Diego Alvarez, Antonio López, and Rafael C González, "Pedestrian navigation based on a waist-worn inertial sensor." Sensors, vol. 12, No. 8, pp. 10536-10549, 2012.
[161] Juan C Alvarez, Antonio M López, Rafael C Gonzalez, and Diego Álvarez, "Pedestrian dead reckoning with waist-worn inertial sensors," in 12th IEEE International Conference on Instrumentation and Measurement Technology Conference (I2MTC), 2012, pp. 24-27.
[162] Stéphane Beauregard, "A helmet-mounted pedestrian dead reckoning system," in Third International Forum on Applied Wearable Computing (IFAWC), 2006, pp. 1-11.
[163] CliffRandell ChrisDjiallis HenkMuller, "Personal position measurement using dead reckoning," in Proceedings of the Seventh IEEE International Symposium on Wearable Computers (ISWC'03), 2003.
[164] Estefania Munoz Diaz and Ana Luz Mendiguchia Gonzalez, "Step Detector and Step Length Estimator for an Inertial Pocket Navigation System," in IEEE International Conference on Indoor Positioning and Indoor Navigation, 2014, pp. 105-110.
[165] Johann Borenstein, HR Everett, and Liqiang Feng, "Where am I? Sensors and methods for mobile robot positioning." University of Michigan, vol. 119, No. 120, 1996.
[166] Martin Raubal and Stephan Winter, Enriching wayfinding instructions with local landmarks: Springer, 2002.
[167] Hwan Hur and Hyo-Sung Ahn, "Hybrid-style Wireless Localization Network for Indoor Mobile Robot Applications," in US-Korea Conference on Science, Technology, and Entrepreneurship, 2009, pp. 1-5.
[168] Johann Borenstein, Hobart R Everett, Liqiang Feng, and David Wehe, "Mobile robot positioning-sensors and techniques." Journal of Robotic Systems, vol. 14, No. 4, pp. 231-249, 1997.
[169] Sylvie Treuillet and Eric Royer, "Outdoor/indoor vision-based localization for blind pedestrian navigation assistance." International Journal of Image and Graphics, vol. 10, No. 04, pp. 481-496, 2010.
[170] Sooyong Lee and Jae-Bok Song, "Mobile robot localization using infrared light reflecting landmarks," in International Conference on Control, Automation and Systems, 2007, pp. 674-677.
[171] Kuang-Chin Tai and Chien-Hsu Chen, "Symbol Detection in Low-Resolution Images using a Novel Method." International Journal of Control & Automation, vol. 7, No. 12, pp. 143-154, 2014.
[172] Stephen P Tarzia, Peter A Dinda, Robert P Dick, and Gokhan Memik, "Indoor localization without infrastructure using the acoustic background spectrum," in Proceedings of the 9th international conference on Mobile systems, applications, and services, 2011, pp. 155-168.
[173] Mark Nixon, Feature extraction & image processing: Academic Press, 2008.
[174] Richard Hartley and Andrew Zisserman, Multiple view geometry in computer vision: Cambridge university press, 2003.

[175] Stefan Seer, Norbert Brändle, and Carlo Ratti, "Kinects and human kinetics: A new approach for studying pedestrian behavior." Transportation research part C: emerging technologies, vol. 48, No. pp. 212-228, 2014.
[176] Danica Kragic, "Object search and localization for an indoor mobile robot." CIT. Journal of Computing and Information Technology, vol. 17, No. 1, pp. 67-80, 2009.
[177] Anahid Basiri, Pouria Amirian, and Adam Winstanley, "The Use of Quick Response (QR) Codes in Landmark-Based Pedestrian Navigation." International Journal of Navigation and Observation, vol. 2014, No. __, 2014.
[178] Yutaka Yamamoto, Paolo Pirjanian, Mario Munich, Enrico Dibernardo, Luis Goncalves, Jim Ostrowski, and Niklas Karlsson, "Optical sensing for robot perception and localization," in IEEE Workshop on Advanced Robotics and its Social Impacts, 2005, pp. 14-17.
[179] Hiroyuki Kobayashi, "A new proposal for self-localization of mobile robot by self-contained 2d barcode landmark," in IEEE Proceedings of SICE Annual Conference, 2012, pp. 2080-2083.
[180] Alessandro Mulloni, Daniel Wagner, Istvan Barakonyi, and Dieter Schmalstieg, "Indoor positioning and navigation with camera phones." IEEE Pervasive Computing, vol. 8, No. 2, pp. 22-31, 2009.
[181] Zhenghua Chen, Han Zou, Hao Jiang, Qingchang Zhu, Yeng Chai Soh, and Lihua Xie, "Fusion of WiFi, Smartphone Sensors and Landmarks Using the Kalman Filter for Indoor Localization." Sensors, vol. 15, No. 1, pp. 715-732, 2015.
[182] Niranjini Rajagopal, Patrick Lazik, and Anthony Rowe, "Visual light landmarks for mobile devices," in Proceedings of the 13th international symposium on Information processing in sensor networks, 2014, pp. 249-260.
[183] Xun Li, Jinling Wang, and Tao Li, "Seamless positioning and navigation by using Geo-referenced images and multi-sensor data." Sensors, vol. 13, No. 7, pp. 9047-9069, 2013.
[184] Rui Wang, Fang Zhao, Haiyong Luo, Bo Lu, and Tao Lu, "Fusion of wi-fi and bluetooth for indoor localization," in Proceedings of the 1st international workshop on Mobile location-based service, 2011, pp. 63-66.
[185] Valérie Renaudin, Okan Yalak, Phillip Tomé, and Bertrand Merminod, "Indoor navigation of emergency agents." European Journal of Navigation, vol. 5, No. 3, pp. 36-45, 2007.
[186] Xiaodong Xu, Yuan Luo, and Honggang Hao, "Vision-based mobile robot localization using natural landmarks," in IEEE International Conference on Systems and Informatics (ICSAI), 2012.
[187] Barry Brumitt, Brian Meyers, John Krumm, Amanda Kern, and Steven Shafer, "Easyliving: Technologies for intelligent environments," in Handheld and ubiquitous computing, 2000, pp. 12-29.
[188] Junichi Ido, Yoshinao Shimizu, Yoshio Matsumoto, and Tsukasa Ogasawara, "Indoor navigation for a humanoid robot using a view sequence." The International Journal of Robotics Research, vol. 28, No. 2, pp. 315-325, 2009.
[189] Navid Fallah, Ilias Apostolopoulos, Kostas Bekris, and Eelke Folmer, "The user as a sensor: navigating users with visual impairments in indoor spaces using tactile landmarks," in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 2012, pp. 425-432.
[190] Etienne E Kerre and Mike Nachtegael, Fuzzy techniques in image processing vol. 52: Physica-Verlag, 2013.
[191] Nikos Petrellis, Nikos Konofaos, and George Ph Alexiou, "Target localization utilizing the success rate in infrared pattern recognition." Sensors, vol. 6, No. 5, pp. 1355-1364, 2006.
[192] Carl Fischer and Hans Gellersen, "Location and navigation support for emergency responders: A survey." IEEE Pervasive Computing, vol. 9, No. 1, pp. 38-47, 2010.

[193] Jouni Rantakokko, Joakim Rydell, Peter Stromback, Peter Handel, Jonas Callmer, David Tornqvist, Fredrik Gustafsson, Magnus Jobs, and Mathias Gruden, "Accurate and reliable soldier and first responder indoor positioning: multisensor systems and cooperative localization." IEEE Wireless Communications, vol. 18, No. 2, pp. 10-18, 2011.

[194] Andy Ward, Alan Jones, and Andy Hopper, "A new location technique for the active office." Personal Communications, vol. 4, No. 5, pp. 42-47, 1997.

[195] Andy Harter, Andy Hopper, Pete Steggles, Andy Ward, and Paul Webster, "The anatomy of a context-aware application." Wireless Networks, vol. 8, No. 2/3, pp. 187-197, 2002.

[196] Sonitor Inc., "Sonitor RTLS Technologies." Available: http://www.sonitor.com/. Accessed on: 18 August,2015.

[197] Eric A Wan and Anindya S Paul, "A tag-free solution to unobtrusive indoor tracking using wall-mounted ultrasonic transducers," in International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2010.

[198] Mike Hazas and Andy Hopper, "Broadband ultrasonic location systems for improved indoor positioning." IEEE Transactions on Mobile Computing, vol. 5, No. 5, pp. 536-547, 2006.

[199] Roberto Casas and David Cuartielles, "Hidden issues in deploying an indoor location system." IEEE Pervasive Computing, vol. 6, No. 2, pp. 62-69, 2007.

[200] Andrew Rice and Robert Harle, "Evaluating lateration-based positioning algorithms for fine-grained tracking," in Proceedings of the joint workshop on Foundations of mobile computing, 2005, pp. 54-61.

[201] Paul Hii and Arkady Zaslavsky, "Improving location accuracy by combining WLAN positioning and sensor technology," in Proceedings of the Workshop on Real-World Wireless Sensor Networks, 2005.

[202] Thomas Strang and Deutsches Zentrum für Luft-und Raumfahrt, Lokalisierungsverfahren: Shaker, 2008.

[203] Jan Wendel, Integrierte Navigationssysteme: Sensordatenfusion, GPS und Inertiale Navigation: Walter de Gruyter, 2011.

[204] P. K. Bora, "Statistical Signal Processing." Available: http://www.iitg.ernet.in/scifac/qip/public_html/cd_cell/chapters/Statistical%20Signal%20Processing.pdf. Accessed on: 23 August 2016.

[205] Sebastian Thrun, Wolfram Burgard, and Dieter Fox, Probabilistic robotics: MIT press, 2005.

[206] Anthony Hayter, Probability and statistics for engineers and scientists: Nelson Education, 2012.

[207] Fredrik Gustafsson, "Particle filter theory and practice with positioning applications." Aerospace and Electronic Systems Magazine, vol. 25, No. 7, pp. 53-82, 2010.

[208] Zhe Chen, "Bayesian filtering: From Kalman filters to particle filters, and beyond." Statistics, vol. 182, No. 1, pp. 1-69, 2003.

[209] Dieter Fox, Jeffrey Hightower, Lin Liao, Dirk Schulz, and Gaetano Borriello, "Bayesian filtering for location estimation." IEEE pervasive computing, vol. 2, No. 3, pp. 24-33, 2003.

[210] Babak Naderi, "Monte carlo localization for pedestrian indoor navigation using a map aided movement model," Master thesis, Technische Universität Berlin, 2012.

[211] YC Ho and Robert Lee, "A Bayesian approach to problems in stochastic estimation and control." IEEE Transactions on Automatic Control, vol. 9, No. 4, pp. 333-339, 1964.

[212] Neil J Gordon, David J Salmond, and Adrian FM Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," in IEEE Proceedings F-Radar and Signal Processing, 1993, pp. 107-113.

[213] Chih-Hao Chao, Chun-Yuan Chu, and An-Yeu Wu, "Location-constrained particle filter human positioning and tracking system," in IEEE Workshop on Signal Processing Systems, 2008, pp. 73-76.

[214] Artem Vassilyev, "Indoor Navigation on Smartphones," MSc. thesis, Tampere University of Technology, 2014.
[215] Quantitec Inc., "Quantitec." Available: https://intranav.com/. Accessed on: 03 Sepember 2016.
[216] Navigine Inc., "Navigine." Available: https://navigine.com/. Accessed on: 31 August, 2016.
[217] infsoft inc., "infsoft smart connected locations." Available: https://www.infsoft.com/. Accessed on: 31 August 2016.
[218] Dragan Stojanović and Natalija Stojanović, "Indoor localization and tracking: Methods, technologies and research challenges." Facta Universitatis, Series: Automatic Control and Robotics, vol. 13, No. 1, pp. 57-72, 2014.
[219] IndoorLBS. Awesome Inc., "Ericsson Indoor Maps and Positioning Platform ", Available: http://indoorlbs.blogspot.in/2011/10/ericsson-indoor-maps-and-positioning.html. Accessed on: 31 Augusr 2016.
[220] Shaun Vann, "Indoor Google Maps." Available: http://www.aci-na.org/sites/default/files/session-4.pdf. Accessed on: 03 September 2016.
[221] Martijn van der Ham, "Real Time Localization of Assets in Hospitals using Quuppa Indoor Positioning Technology," MSc thesis, TU Delft, Netherlands, 2015.
[222] Senion Inc., "StepInside Indoor Positioning When location matters." Available: https://senion.com/. Accessed on: 09 September 2016.
[223] GeniusMatcher Inc., "GeniusMatcher." Available: http://www.geniusmatcher.com/. Accessed on: 03 Septemeber 2016.
[224] Navvis Inc., "Navvis." Available: http://www.navvis.com/. Accessed on: 18 August 2016.
[225] Sourav Bhattacharya, Teemu Pulkkinen, Petteri Nurmi, and Antti Salovaara, "Monstre: A mobile navigation system for retail environments," in International Workshop on Smart Mobile Applications (SmartApps), 2011.
[226] Gennady Berkovich, "Accurate and reliable real-time indoor positioning on commercial smartphones," in IEEE International conference on indoor positioning and indoor navigation (IPIN), 2014, pp. 27-30.
[227] Nisarg Kothari, Balajee Kannan, Evan D Glasgwow, and M Bernardine Dias, "Robust Indoor localization on a commercial smart phone." Procedia Computer Science, vol. 10, No. _, pp. 1114-1120, 2012.
[228] Shashank Kumar Gupta, Simon Box, and RE Wilson, "Vehicle localization based on heading data using low cost sensors," presented at the 46th Annual Conference of the Universities' Transport Study Group (UTSG), Newcastle, UK, 2014.
[229] Android Inc., "Open handset alliance." Available: http://www.openhandsetalliance.com/android_overview.html. Accessed on: 18 August,2015.
[230] Android Inc., "Ice Cream Sandwich." Available: http://developer.android.com/about/versions/android-4.0-highlights.html. Accessed on: 18 August,2015.
[231] Greg Milette and Adam Stroud, Professional Android sensor programming: Wiley. com, 2012.
[232] Reto Meier, Professional Android 4 application development: John Wiley & Sons, 2012.
[233] Devin Coldewey, "Review: Motorola Atrix 2." Available: https://techcrunch.com/2011/11/09/review-motorola-atrix-2/. Accessed on: 30 September 2016,2011.
[234] SpecDevice, "Android Device Specification Database: HTC Sensation Z710e." Available: http://specdevice.com/showspec.php?id=0453-de1b-0085-88be5dd78f42. Accessed on: 18 August,2015.

[235] Edward Chester, "Samsung Galaxy S2 review." Available: http://www.trustedreviews.com/samsung-galaxy-s-ii-i9100-review. Accessed on: 30 September 2016.
[236] GSMARENA Inc., "Motorola Razr XT910." Available: http://www.gsmarena.com/motorola_razr_xt910-4273.php. Accessed on: 30 September 2016.
[237] Gsmarena Inc., "Samsung Galaxy Nexus 19250." Available: http://www.gsmarena.com/samsung_galaxy_nexus_i9250-4219.php. Accessed on: 30 September 2016.
[238] PhoneArena Inc., "SONY ERICSSON Xperia Arc X12 SO-01C LT15i." Available: http://www.imei.info/phonedatabase/8932-sony-ericsson-xperia-arc-x12-so-01c-lt15i/. Accessed on: 30 September 2016.
[239] Yashodhar Rajnikant Patel, "Comparing performance of applications written in smartphone development environments," Master's thesis, San Diego State University, 2011.
[240] HTC Inc., "Smartphones." Available: http://www.htc.com/us/smartphones/htc-sensation/. Accessed on: 20 September,2014.
[241] Anton Hansson and Linus Tufvesson, "Using Sensor Equipped Smartphones to Localize WiFi Access Points," MSc. thesis, Lund University, 2011.
[242] Roger Gillham, Scalextric: A Race Through Time–The Official 50th Anniversary Book: JH Haynes & Co. Ltd, 2007.
[243] Arvind Thiagarajan, Lenin Ravindranath, Katrina LaCurts, Samuel Madden, Hari Balakrishnan, Sivan Toledo, and Jakob Eriksson, "VTrack: accurate, energy-aware road traffic delay estimation using mobile phones," in Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, 2009, pp. 85-98.
[244] Benoit Mariani, Mayté Castro Jiménez, François JG Vingerhoets, and Kamiar Aminian, "On-shoe wearable sensors for gait and turning assessment of patients with Parkinson's disease." IEEE transactions on biomedical engineering, vol. 60, No. 1, pp. 155-158, 2013.
[245] Karsten Groÿekatthöfer and Zizung Yoon, "Introduction into quaternions for spacecraft attitude representation," TU Berlin, Germany 2012.
[246] British Standard Tester, "PSD 30/3B - High Performance Regulated DC Power Supply (Digital Display)." Available: http://www.bstcaltek.com/catalog2/index.php/high-performance-regulated-dc-power-supply-psd30-3b.html. Accessed on: 18 August,2015.
[247] Android Inc., "Sensors Overview." Available: http://developer.android.com/guide/topics/sensors/sensors_overview.html. Accessed on: 18 August,2015.
[248] Vincent Gabaglio, "GPS/INS Integration for Pedestrian Navigation," DSc. thesis, Institute of Geomatics, Swiss Federal Institute of Technologye Lausanne, 2003.
[249] Guang Tan, Mingming Lu, Fangsheng Jiang, Kongyang Chen, Xiaoxia Huang, and Jie Wu, "Bumping: A bump-aided inertial navigation method for indoor vehicles using smartphones." IEEE Transactions on Parallel and Distributed Systems, vol. 25, No. 7, pp. 1670-1680, 2014.
[250] Xsens Technologies B.V., "MTi-G User Manual and Technical Documentation." Available: https://www.xsens.com/wp-content/uploads/2013/11/MTi-G_User_Manual_and_Technical_Documentation.pdf. Accessed on: 30 September 2016.
[251] Kimon Voutsis, Paul D Groves, Mark Holbrow, and Colin Ford, "The Importance of Human Motion for Simulation Testing of GNSS," in Proceedings of the 27th International Technical Meeting of The Satellite Division of the Institute of Navigation, 2014.
[252] Damien Douxchamps, "A small list of IMU / INS / INU." Available: https://damien.douxchamps.net/research/imu/. Accessed on: 15 August 2016.
[253] Invensense Inc., "MPU-9150 Product Specification Revision 4.3." Available: https://www.invensense.com/wp-content/uploads/2015/02/MPU-9150-Datasheet.pdf. Accessed on: 20 August 2016.

[254] Phil Kim and Lynn Huh, Kalman filter for beginners: with MATLAB examples vol. 4: CreateSpace, 2011.
[255] Mike Schlaich. (2005). Guidelines for the design of footbridges.
[256] Jingbin Liu, Ruizhi Chen, Ling Pei, Robert Guinness, and Heidi Kuusniemi, "A hybrid smartphone indoor positioning solution for mobile lbs." Sensors, vol. 12, No. 12, pp. 17208-17233, 2012.
[257] Zuolei Sun, Xuchu Mao, Weifeng Tian, and Xiangfen Zhang, "Activity classification and dead reckoning for pedestrian navigation with wearable sensors." Measurement Science and Technology, vol. 20, No. 1, 2009.
[258] S. H. Shin, C. G. Park, J. W. Kim, H. S. Hong, and J. M. Lee, "Adaptive Step Length Estimation Algorithm Using Low-Cost MEMS Inertial Sensors," in Sensors Applications Symposium, 2007, pp. 1-5.
[259] Thomas Gallagher, Elyse Wise, Binghao Li, Andrew G Dempster, Chris Rizos, and Euan Ramsey-Stewart, "Indoor Positioning System based on Sensor Fusion for the Blind and Visually Impaired," in IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2012.
[260] Je-Nam Kim, Mun-Ho Ryu, Yoon-Seok Yang, and Jun-Yong Hong, "Estimation of Walking Direction Estimation using a Shoe-mounted Acceleration Sensor." International Journal of Multimedia and Ubiquitous Engineering, vol. 9, No. 5, pp. 215-222, 2014.
[261] Stephane Beauregard and Harald Haas, "Pedestrian dead reckoning: A basis for personal positioning," in Proceedings of the Third Workshop on Positioning, Navigation and Communication, 2006, pp. 27-35.
[262] K Saito, M Zecca, S Sessa, Z Lin, L Bartolomeo, S Cosentino, K Petersen, H Ishii, T Ikai, and A Takanishi, "Assessment of walking quality by using Inertial Measurement Units," in First International Conference on Innovative Engineering Systems (ICIES), 2012, pp. 13-18.
[263] Abdelrahman Ali and Naser El-Sheimy, "Low-Cost MEMS-Based Pedestrian Navigation Technique for GPS-Denied Areas." Journal of Sensors, vol. 2013, No. 2013.
[264] Li Tan and Jean Jiang, Digital signal processing: fundamentals and applications: Academic Press, 2013.
[265] Walt Kester, Mixed-signal and DSP design techniques: Newnes, 2003.
[266] S. Salivahanan and A. Vallavaraj, Digital Signal Processing: McGraw-Hill Education (India) Pvt Limited, 2001.
[267] Jesse Hansen, "FFT Tutorial." Available: http://www.ele.uri.edu/~hansenj/projects/ele436/fft.pdf. Accessed on: 30 August 2016.
[268] Alan V Oppenheim, Alan S Willsky, and Syed Hamid Nawab, Signals and systems vol. 2: Prentice-Hall Englewood Cliffs, 1983.
[269] Sebastijan Sprager and Matjaz B Juric, "Inertial sensor-based gait recognition: a review." Sensors, vol. 15, No. 9, pp. 22089-22127, 2015.
[270] Tâm Huynh and Bernt Schiele, "Analyzing features for activity recognition," in Proceedings of the 2005 joint conference on Smart objects and ambient intelligence: innovative context-aware services: usages and technologies, 2005, pp. 159-163.
[271] J Won Kim, Han Jin Jang, Dong-Hwan Hwang, and Chansik Park, "A step, stride and heading determination for the pedestrian navigation system." Journal of Global Positioning Systems, vol. 3, No. 1-2, pp. 273-279, 2004.
[272] Stéphane Beauregard and Martin Klepal, "Indoor PDR performance enhancement using minimal map information and particle filters," in IEEE/ION Symposium on Position, Location and Navigation, 2008, pp. 141-147.
[273] S Godha and G Lachapelle, "Foot mounted inertial system for pedestrian navigation." Measurement Science and Technology, vol. 19, No. 7, pp. 1-9, 2008.
[274] foreseegame.com and Microsec Research, "The Smartphone Saga - an online survey report." Opininon SQUARE, Available:

http://www.foreseegame.com/OpinionSquare/The_Smartphone_Saga.pdf. Accessed on: 18 August,2015.

[275] Harvey Weinberg, "Using the ADXL202 in pedometer and personal navigation applications." Analog Devices AN-602 application note, vol. 2, No. 2, pp. 1-6, 2002.

[276] Jim Scarlett. (2007) Enhancing the performance of pedometers using a single accelerometer. Analog Dialogue. Available: http://www.analog.com/media/en/technical-documentation/application-notes/47076299220991AN_900.pdf

[277] Yi-Ting Li, Guaning Chen, and Min-Te Sun, "An Indoor Collaborative Pedestrian Dead Reckoning System," in 42nd IEEE International Conference on Parallel Processing (ICPP), 2013, pp. 923-930.

[278] Wasiq Waqar, Andrew Vardy, and Yuanzhu Chen, "Motion Modelling using Smartphones for Indoor MobilePhone Positioning," in 20th Newfoundland Electrical and Computer Engineering Conference, Newfoundland, Canada, 2011.

[279] Wenjie Li and Pablo Esteban Quiroga Garcia, "On indoor positioning for mobile devices," Signals and Systems, Chalmers University of Technology, Goteborg, Sverige, 2011.

[280] Mohamed Attia, Adel Moussa, and Naser El-Sheimy, "Map Aided Pedestrian Dead Reckoning Using Buildings Information for Indoor Navigation Applications." Positioning, vol. 4, No. 3, pp. 227-239, 2013.

[281] Fang Lei, P. J. Antsaklis, L. A. Montestruque, M. B. McMickell, M. Lemmon, Sun Yashan, Fang Hui, I. Koutroulis, M. Haenggi, Xie Min, and Xie Xiaojuan, "Design of a wireless assisted pedestrian dead reckoning system - the NavMote experience." IEEE Transactions on Instrumentation and Measurement, vol. 54, No. 6, pp. 2342-2358, 2005.

[282] Frank Ebner, Frank Deinzer, Lukas Köping, and Marcin Grzegorzek, "Robust Self-Localization using Wi-Fi, Step/Turn-Detection and Recursive Density Estimation," in International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2014.

[283] Ionut Constandache, Romit Roy Choudhury, and Injong Rhee, "Towards mobile phone localization without war-driving," in Proceedings of INFOCOM 2010, pp. 1-9.

[284] Xu Wei, Liu Shou India, Huangxian Li, and Lu Serra, "Method for pedestrian step size estimation and dead reckoning," U.S Patent No. CN103411607 A, 2013.

[285] Guobin Shen, Zhuo Chen, Peichao Zhang, Thomas Moscibroda, and Yongguang Zhang, "Walkie-markie: indoor pathway mapping made easy," in Proceedings of the 10th USENIX conference on Networked Systems Design and Implementation, 2013, pp. 85-98.

[286] Rísin McNaney, John Vines, Daniel Roggen, Madeline Balaam, Pengfei Zhang, Ivan Poliakov, and Patrick Olivier, "Exploring the acceptability of google glass as an everyday assistive device for people with Parkinson's," in Proceedings of the 32nd annual ACM conference on Human factors in computing systems, 2014, pp. 2551-2554.

[287] Max J Egenhofer and Werner Kuhn, "Beyond desktop GIS," in GIS PlaNET, Lisbon, Portugal, 1998.

[288] Jeffrey R Blum, Daniel G Greencorn, and Jeremy R Cooperstock, "Smartphone sensor reliability for augmented reality applications," in Mobile and Ubiquitous Systems: Computing, Networking, and Services, Springer, 2013, pp. 127-138.

[289] Wen-Yuah Shih, Liang-Yu Chen, and Kun-Chan Lan, "Estimating walking distance with a smart phone," in IEEE International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), 2012, pp. 166-171.

[290] Mohamed Attia, Adel Moussa, Xing Zhao, and Naser El-Sheimy, "Assisting personal positioning in indoor environments using map matching." Archives of Photogrammetry, Cartography and Remote Sensing, vol. 22, No. _, pp. 39-49, 2011.

[291] Mohammed A. Quddus, Washington Y. Ochieng, and Robert B. Noland, "Current map-matching algorithms for transport applications: State-of-the art and future research directions." Transportation Research Part C: Emerging Technologies, vol. 15, No. 5, pp. 312-328, 2007.

[292] Ulrich Walder and T Bernoulli, "Context-adaptive algorithms to improve indoor positioning with inertial sensors," in IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), 2010, pp. 1-6.
[293] Christopher E White, David Bernstein, and Alain L Kornhauser, "Some map matching algorithms for personal navigation assistants." Transportation Research Part C: Emerging Technologies, vol. 8, No. 1, pp. 91-108, 2000.
[294] Yuan Li and Zizhang He, "3D indoor navigation: a framework of combining BIM with 3D GIS," in 44th ISOCARP congress, 2008.
[295] Susanna Kaiser, Mohammed Khider, and Patrick Robertson, "A human motion model based on maps for navigation systems." EURASIP Journal on Wireless Communications and Networking, vol. 2011, No. 1, pp. 1-14, 2011.
[296] Pierre-Yves Gilliéron, Daniela Büchel, Ivan Spassov, and Bertrand Merminod, "Indoor navigation performance analysis," in Proceedings of the 8th European Navigation Conference GNSS, 2004.
[297] Frederic Chausse, Vincent Voisin, Jean Laneurit, and Roland Chapuis, "Centimetric localization of a vehicle combining vision and low cost GPS," in IAPR workshop on machine vision applications, 2002, pp. 538-541.
[298] Mohammed Khider, Susanna Kaiser, Patrick Robertson, and Michael Angermann, "The Effect of Maps-Enhanced Novel Movement Models on Pedestrian Navigation Performance," in Proceedings of The 12th annual European Navigation Conference (ENC 2008), Toulouse, France, 2008.
[299] Haitao Bao and Wai-Choong Wong, "A novel map-based dead-reckoning algorithm for indoor localization." Journal of Sensor and Actuator Networks, vol. 3, No. 1, pp. 44-63, 2014.
[300] Jean-Paul Rodrigue, "The Geography of transport systems." Available: http://people.hofstra.edu/geotrans/eng/methods/nettopology.html. Accessed on: 18 August,2015.
[301] Yuki Wakuda, Satoshi Asano, Noboru Koshizuka, and Ken Sakamura, "An adaptive map-matching based on Dynamic Time Warping for pedestrian positioning using network map," in Position Location and Navigation Symposium (PLANS), 2012, pp. 590-597.
[302] Yunyoung Nam, "Map-based indoor people localization using an inertial measurement unit." Journal of Information Science and Engineering, vol. 27, No. 4, pp. 1233-1248, 2011.
[303] Tim Nelen, Maarten Weyn, and Martin Klepal, "Indoor Navigation for Complex Environments," Papers of the e-lab Master's theses, 2010.
[304] Beomju Shin, Jung Ho Lee, Hyunho Lee, Eungyeong Kim, Jeahun Kim, Seok Lee, Young-su Cho, Sangjoon Park, and Taikjin Lee, "Indoor 3D pedestrian tracking algorithm based on PDR using smarthphone," in 12th International Conference on Control, Automation and Systems (ICCAS), 2012, pp. 1442-1445.
[305] Mohammed A Quddus, "Map Matching Algorithms for Intelligent Transport Systems," in Handbook of Research on Geoinformatics, Information Science Reference, 2009.
[306] Saurabh Taneja, Burcu Akinci, James H Garrett, Lucio Soibelman Jr, and Hassan A Karimi, "Effects of Positioning Data Quality and Navigation Models on Map-Matching of Indoor Positioning Data." Journal of Computing in Civil Engineering, vol. 30, No. 1, 2014.
[307] Pavel Davidson, "Algorithms for Autonomous Personal Navigation Systems," Ph.D. thesis, Tampere University of Technology. Publication; Vol. 1171, 2013.
[308] Pierre-Yves Gilliéron, Ivan Spassov, and Bertrand Merminod, "Indoor navigation enhanced by map-matching." European Journal of navigation, vol. 3, No. 3, pp. 6-13, 2005.
[309] Joshua S Greenfeld, "Matching GPS observations to locations on a digital map," in Transportation Research Board 81st Annual Meeting, 2002.

[310] Pavel Davidson, Jussi Collin, and Jarmo Takala, "Application of particle filters for indoor positioning using floor plans," in Ubiquitous Positioning Indoor Navigation and Location Based Service (UPINLBS), 2010, pp. 1-4.

[311] Frédéric Evennou, François Marx, and Emil Novakov, "Map-aided indoor mobile positioning system using particle filter," in Wireless Communications and Networking Conference, 2005, pp. 2490-2494.

[312] K. C. Lan and W. Y. Shih, "Using smart-phones and floor plans for indoor location tracking - Withdrawn." IEEE Transactions on Human-Machine Systems, vol. 44, No. 2, pp. 211-221, 2014.

[313] Pierre-Yves Gilliéron and Bertrand Merminod, "Personal navigation system for indoor applications," in 11th IAIN World Congress, 2003, pp. 21-24.

[314] Nammoon Kim and Youngok Kim, "Indoor Positioning System with IMU, Map Matching and Particle Filter," presented at the Recent Advances in Electrical Engineering and Computer Science, Michigan, US, 2015.

[315] Haitao Bao and Wai-Choong Wong, "An indoor dead-reckoning algorithm with map matching," in 9th International Wireless Communications and Mobile Computing Conference (IWCMC), 2013, pp. 1534-1539.

[316] 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 ACM Conference on Ubiquitous Computing, 2012, pp. 421-430.

[317] Mohamed F Mansour and Deric W Waters, "Map-assisted Kalman filtering," in IEEE International Conference on Acoustics, Speech and Signal Processing, 2013, pp. 3208-3212.

[318] Mahmoud El-Gohary, Sean Pearson, James McNames, Martina Mancini, Fay Horak, Sabato Mellone, and Lorenzo Chiari, "Continuous monitoring of turning in patients with movement disability." Sensors, vol. 14, No. 1, pp. 356-369, 2013.

[319] Kun-Chan Lan and Wen-Yuah Shih, "On Calibrating the Sensor Errors of a PDR-Based Indoor Localization System." Sensors, vol. 13, No. 4, pp. 4781-4810, 2013.

[320] Valérie Renaudin, Melania Susi, and Gérard Lachapelle, "Step length estimation using handheld inertial sensors." Sensors, vol. 12, No. 7, pp. 8507-8525, 2012.

[321] Alessio De Angelis, Satyam Dwivedi, P Handel, Antonio Moschitta, and Paolo Carbone, "Ranging results using a UWB platform in an indoor environment," in IEEE International Conference on Localization and GNSS (ICL-GNSS), 2013, pp. 1-5.

[322] Anahid Basiri, Adam Winstanley, and Pouria Amirian, "Landmark-based pedestrian navigation," in Proceedings of the 21st GIS Research (GISRUK) Conference, 2013.

[323] H Zahit Selvi, İ Öztuğ Bildirici, and Georg Gartner, "Landmark orientation for pedestrians: a case study at Selcuk University," in 4th International Conference on Cartography and GIS, Albena, Bulgarien, 2012.

[324] R González, F Rodrıguez, JL Guzmán, and M Berenguel, "Comparative study of localization techniques for mobile robots based on indirect kalman filter," in International Symposium on Robotics, Barcelona, Spain, 2009.

[325] Alberto Serra, Davide Carboni, and Valentina Marotto, "Indoor pedestrian navigation system using a modern smartphone," in Proceedings of the 12th international conference on Human computer interaction with mobile devices and services, 2010, pp. 397-398.

[326] Kristin L Lovelace, Mary Hegarty, and Daniel R Montello, "Elements of good route directions in familiar and unfamiliar environments," in Spatial information theory, cognitive and computational foundations of geographic information science, Springer, 1999, pp. 65-82.

[327] Stephan Winter, Martin Raubal, and Clemens Nothegger, "Focalizing measures of salience for wayfinding," in Map-based mobile services, Springer, 2005, pp. 125-139.

[328] Molly E Sorrows and Stephen C Hirtle, "The nature of landmarks for real and electronic spaces," in Spatial information theory, cognitive and computational foundations of geographic information science, Springer, 1999, pp. 37-50.

[329] Christina Ohm, Manuel Müller, Bernd Ludwig, and Stefan Bienk, "Where is the Landmark? Eye Tracking Studies in Large-Scale Indoor Environments," presented at the Second International Workshop on Eye Tracking for Spatial Research co-located with the eightth International Conference on Geographic Information Science (GIScience 2014), Vienna, Austria, 2014.

[330] A. Sankara Narayanan, "QR Codes and Security Solutions." International Journal of Computer Science and Telecommunications, vol. 3, No. 7, 2012.

[331] Denso Wave Incorporated, "QR Code.com." Available: http://www.qrcode.com/en/. Accessed on: 18 August 2015.

[332] Hongkai Wen, Zhuoling Xiao, Niki Trigoni, and Phil Blunsom, "On assessing the accuracy of positioning systems in indoor environments," in European Conference on Wireless Sensor Networks, 2013, pp. 1-17.

[333] Seversion Software, "Room+ Indoor WiFi Positioning APK." Available: https://apkpure.com/room-indoor-wifi-positioning/com.seversion.roomplus. Accessed on: 30 August 2016

[334] Indoo.rs Inc., "indoo.rs - Navigation APK." Available: https://apkpure.com/indoo-rs-navigation/com.customlbs.android. Accessed on: 30 August 2016.

[335] M. Marschollek, M. Goevercin, K. H. Wolf, B. Song, M. Gietzelt, R. Haux, and E. Steinhagen-Thiessen, "A performance comparison of accelerometry-based step detection algorithms on a large, non-laboratory sample of healthy and mobility-impaired persons," in 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, 2008, pp. 1319-1322.

[336] Hao Xia, Xiaogang Wang, Yanyou Qiao, Jun Jian, and Yuanfei Chang, "Using multiple barometers to detect the floor location of smart phones with built-in barometric sensors for indoor positioning." Sensors, vol. 15, No. 4, pp. 7857-7877, 2015.

[337] Xiaojun Zhu, Qun Li, and Guihai Chen, "APT: Accurate outdoor pedestrian tracking with smartphones," in Proceedings of IEEE INFOCOM, 2013, pp. 2508-2516.

[338] Marco D Tundo, Edward Lemaire, and Natalie Baddour, "Correcting Smartphone orientation for accelerometer-based analysis," in Proceedings of IEEE International Symposium on Medical Measurements and Applications (MeMeA), 2013, pp. 58-62.

[339] Kalyan Pathapati Subbu, Brandon Gozick, and Ram Dantu, "LocateMe: Magnetic-fields-based indoor localization using smartphones." ACM Transactions on Intelligent Systems and Technology (TIST), vol. 4, No. 4, 2013.

[340] Ling Pei, Ruizhi Chen, Jingbin Liu, Heidi Kuusniemi, Yuwei Chen, and Tomi Tenhunen, "Using motion-awareness for the 3D indoor personal navigation on a Smartphone," in Proceedings of the 24th International Technical Meeting of the Satellite Division of the Institute of Navigation, 2011, pp. 2906-2912.

[341] Irene Rubino, Jetmir Xhembulla, Andrea Martina, Andrea Bottino, and Giovanni Malnati, "MusA: Using indoor positioning and navigation to enhance cultural experiences in a museum." Sensors, vol. 13, No. 12, pp. 17445-17471, 2013.