MEMS-MARG-BASED DEAD RECKONING FOR AN INDOOR POSITIONING AND TRACKING SYSTEM

MEMS-MARG-BASED DEAD RECKONING FOR

AN INDOOR POSITIONING AND TRACKING SYSTEM

BY

YIQIONG MIAO, M.Sc.

A THESIS

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

Master of Applied Science (2021) McMaster University
(Electrical & Computer Engineering) Hamilton, Ontario, Canada

TITLE: MEMS-MARG-based Dead Reckoning for
an Indoor Positioning and Tracking System

AUTHOR: Yiqiong Miao
M.Sc.
Boston College, Chestnut Hill, MA, U.S.

SUPERVISOR: Dr. Dongmei Zhao

NUMBER OF PAGES: xvi, 122

To my parents

Lay Abstract

With the maturity of microelectromechanical systems (MEMS) technology in recent years, Magnetic, Angular Rate, and Gravity (MARG) sensors are embedded in most smart devices. This research considers the indoor positioning problem and aims to design and develop an infrastructure-free self-contained MEMS MARG inertial sensor-based indoor positioning and tracking system with high precision. The proposed positioning system uses the Pedestrian Dead Reckoning (PDR) approach and includes four main modules at the system level with a dual-mode feature. Specifically, the four main modules are mode detection, step detection and moving distance estimation, heading and orientation estimation, and position estimation. The two modes are static mode and dynamic mode. To address the cumulative error issue of using low-cost inertial sensors, signal processing and sensor fusion techniques are utilized for algorithm design. The detection and estimation algorithms of each module are presented in the system design chapter. Experimental evaluations including trajectory results under five scenarios show that the proposed position estimation algorithm achieves a higher position accuracy than that of conventional estimation methods.

Abstract

Location-based services (LBSs) have become pervasive, and the demand for these systems and services is rising. Indoor Positioning Systems (IPSs) are key to extend location-based services indoors where the Global Positioning System (GPS) is not reliable due to low signal strength and complicated signal propagation environment. Most existing IPSs either require the installation of special hardware devices or build a fingerprint map, which is expensive, time-consuming, and labor-intensive. Developments in microelectromechanical systems (MEMS) have resulted in significant advancements in the low-cost compact MARG inertial sensors, making it possible to achieve low-cost and high-accuracy IPSs.

This research considers the indoor positioning problem and aims to design and develop an infrastructure-free self-contained indoor positioning and tracking system based on Pedestrian Dead Reckoning (PDR) using MEMS MARG inertial sensors. PDR-based systems rely on MARG inertial sensor measurements to estimate the current position of the object by using a previously determined position without external references. Many issues still exist in developing such systems, such as cumulative errors, high-frequency sensor noises, the gyro drift issue, magnetic distortions, etc. As the MARG sensors are inherently error-prone, the most significant challenge is how to design sensor fusion models and algorithms to accurately extract useful location-based information from individual motion and magnetic sensors. The objective of this thesis is to solve these issues and mitigate the challenges. The proposed positioning system is designed with four main modules at the system level and a dual-mode feature. Specifically, the four main modules are mode detection, step detection and moving distance estimation, heading and orientation estimation, and position estimation. To address the cumulative error issue of using low-cost inertial sensors, signal processing and sensor fusion techniques are utilized for algorithm design. Experimental evaluations show that the proposed position estimation algorithm is able to achieve high positioning accuracy at low costs for the indoor environment.

Keywords: Indoor Positioning System (IPS), Inertial Measurement Unit (IMU), Microelectromechanical Systems (MEMS), sensor fusion, Digital Signal Processing (DSP), Pedestrian Dead Reckoning (PDR), object tracking and navigation, robotics, quaternions.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my advisor Prof. Dr. Dongmei Zhao for her dedicated support, encouragement, and valuable guidance throughout this research. It was a great honor to work under her supervision. Without her academic help and immense knowledge this research would not be possible.

I would also like to thank my committee members Prof. Dr. Terry Todd, and Prof. Dr. Jun Chen for reading my thesis and providing insightful comments and suggestions during the process.

Once again, I would like to thank my family, friends, fellow colleagues, and all those who have helped me over the course of completing this research and development work.

Abbreviations

4G	Fourth Generation
5G	Fifth Generation
ADC	Analog-to-Digital Converter
AHDE	Advanced Heuristic Drift Elimination
AHRS	Attitude Heading Reference System
AOA	Angle of Arrival
AP	Access Point
AR	Augmented Reality
ARW	Angular Random Walk
BLE	Bluetooth Low Energy
CAE	Convolutional Autoencoder
CNN	Convolutional Neural Network
DoF	Degrees of Freedom
DR	Dead Reckoning
DSP	Digital Signal Processing
EKF	Extended Kalman Filter
ENU	East-North-Up

GNSS	Global Navigation Satellite System
GPS	Global Positioning System
HDR	Heuristic Drift Reduction
HGB	Histogram-based Gradient Boosting
IC	Integrated Circuit
IMU	Inertial Measurement Unit
IoT	Internet of Things
IPN	Inertial Pedestrian Navigation
IPS	Indoor Positioning System
KF	Kalman Filter
KNNR	K-Nearest Neighbors Regressor
LBS	Location-based Service
LED	Light-Emitting Diode
LOS	Line-of-sight
LSTM	Long Short-Term Memory
MARG	Magnetic, Angular Rate, and Gravity
MEMS	Microelectromechanical Systems
MSE	Mean Squared Error
NED	North-East-Down
NLOS	Non-line-of-sight
PCB	Printed Circuit Board

PDR	Pedestrian Dead Reckoning
RAKF	Robust Adaptive Kalman Filter
RF	Random Forest
RFID	Radio-Frequency Identification
RFR	Random Forest Regressor
RP	Reference Point
RSS	Received Signal Strength
RSSI	Received Signal Strength Indicator
SIFT	Scale Invariant Feature Transform
SLAM	Simultaneous Localization and Mapping
SO(3)	Special Orthogonal Group
SVM	Support Vector Machine
SVR	Support Vector Regressor
SWA	Sliding Window Averaging
TDOA	Time Difference of Arrival
TOA	Time of Arrival
UAV	Unmanned Aerial Vehicle
UWB	Ultra-Wideband
VLC	Visible Light Communication
VRE	Vibration Ratification Error
VRW	Velocity Random Walk

Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
ZARU	Zero Angular Rate Update
ZUPT	Zero Velocity Potential Update

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1. Overview

As an evolving field, indoor positioning enables determining and tracking the position of objects or people in interior spaces where the Global Positioning System (GPS) and other satellite technologies lack precision or fail to provide location-based information. Indoor positioning and tracking technologies have a broad range of applications and have become essential components for deploying services in many industries such as manufacturing, logistics, retail, healthcare, emergency services, public safety industry, etc. However, most of the existing indoor positioning systems and strategies require complex infrastructure or special devices, which can be expensive or difficult to deploy. This research aims to design and develop a practical infrastructure-free Indoor Positioning System (IPS) when deploying network infrastructure using special devices is not a choice. This chapter introduces the research background, motivation, objective, and main contributions.

The demand of location-based services (LBSs) has increased significantly owing to the proliferation of smart devices and Internet of Things (IoT) technologies over the last decade. A considerable number of positioning and navigation systems were developed for detecting the positions of objects or pedestrians in outdoor and indoor environments.

Although GPS is widely used for most of the outdoor navigation systems, it is not reliable for indoor positioning applications due to its low signal strength and the complicated indoor propagation environment, which result in unacceptable positioning accuracy typically ranging from 5 m to 10 m (Nessa *et al.*, 2020). Given the high demand on IPSs, tremendous research efforts have been made on developing such systems with higher accuracy. Various IPSs adopting different technologies have been developed, such as computer-vision-based systems (Fusco and Coughlan, 2018), Wi-Fi-based systems (Wang *et al.*, 2021), Bluetooth-based systems (Yang *et al.*, 2020), ultra-wideband (UWB) communication-based systems (Schroeer, 2018), radio frequency identification (RFID)-based systems (Merenda *et al.*, 2021), visible light communication (VLC)-based systems (Sayed *et al.*, 2019), Inertial Measurement Unit (IMU)-based systems (Hellmers *et al.*, 2013), laser-based systems (Hesch *et al.*, 2010), and ultrasonic-based systems (ultra-sonic sensors) (Qi and Liu, 2017). Locating an object in an indoor environment is challenging because radio signal propagation in the complex unstructured indoor environment is easily affected by obstacles and surroundings. As a result, unpredictable signal propagation problems such as attenuation, scattering, reflection, and diffraction substantially decrease the position accuracy. The disadvantage of using communication technology-based strategies is that the system cannot function in the areas where there is unstable or even no signal coverage. Furthermore, infrastructure-based approaches such as UWB, RFID, and ultra-sonic sensors rely on costly infrastructure and extra devices, which can be difficult to deploy. With the development of microelectromechanical systems (MEMS) technology, inertial sensors can be embedded into smart devices. Adopting the compact, low cost, and low power consumption inertial sensors in developing IPSs is a natural solution that expands the positioning system's capabilities. This research focuses on designing and developing a practical indoor positioning system that solely depends on using smartphone-embedded inertial sensors, including the accelerometer, magnetometer, and gyroscope, which are also known as MARG sensors. The advantage of using a MEMS MARG sensors-based position estimation system is that it is cost-effective, convenient-to-use, and no additional devices or infrastructure deployment is needed. Another advantage of such systems is that computational complexity is relatively low compared to some other systems.

An effective and accurate self-contained IPS is also an indispensable component of Internet of Things (IoT) applications, smart grids, and smart building. As IoT intends to create end-to-end connectivity of all smart devices, its applications will require seamless indoor and outdoor localization services and capabilities. IPSs can also benefit many novel systems and technologies such as augmented reality (AR) and digital twin. These demands will increase especially in areas like healthcare and logistics. During the year 2020, due to the spread of COVID-19, hospitals need to have access to location-based information of patients, medical staff, and medical equipment in real time in order to avoid possible fatal incidents in case of medical emergencies.

Existing indoor positioning approaches fall into two broad categories: infrastructure-based and infrastructure-free solutions. As one of the infrastructure-free solutions, Pedestrian Dead Reckoning (PDR) calculates the current position of a moving object by utilizing the previously calculated position information through continuously detecting gait event and updating current estimations of speed, step length, and heading direction. PDR is commonly decomposed into step detection and estimation, heading estimation, and an integration model (Beauregard and Haas, 2006). Existing PDR-based approaches establish different models for processing sensor data. Also, the sensors used vary in type and the positions placed on the body, such as foot-mounted (Zhang *et al.*, 2020), chest-mounted (Lu *et al.*, 2019), head-mounted (Hou and Bergmann, 2020), etc. Issues still exist in current PDR-based approaches. For example, estimated positions gradually drift away as measurement error accumulates over time. PDR-based methods fulfil the task of position estimation by assuming that the sensor data feeding into the algorithm is highly accurate. Moreover, if the smartphone is handheld by the pedestrian, the position estimation result is affected by the relative position of the smartphone. The pedestrian may hold the phone taking a picture, texting, or swing arms with the phone in his/her hand. In addition, each of the low-cost inertial sensors has its own deficiency and individual raw sensor data are prone to accumulated errors.

Several potential issues have been mentioned in the literature when adopting MEMS inertial sensor-based dead reckoning for indoor positioning and tracking systems (Wu *et al.*, 2016) (Nazemzadeh *et al.*, 2017). As velocity is calculated by taking the integral of measured acceleration with respect to time, noisy accelerometer measurement data leads to linear growth of accumulative errors in velocity estimation. In other words, the mathematical integration process magnifies inherent sensor errors and noises. The low-cost accelerometer is sensitive to all kinds of accelerations and has high frequency noises. Unlike accelerometer, gyroscope measurements will gradually drift away from true values due to a low changing and random process, also known as gyroscope drift. Aiming at diminishing accumulative errors resulting from the gyro drift, (Zhang *et al.*, 2019) proposed a MEMS gyro drift model based on wavelet threshold denoising and improved Elman neural network. The modeling method can be considered as an independent data preprocessing process. The magnetic fields can penetrate building materials without signal propagation errors or multipath effects, which makes magnetometer suitable for indoor positioning solutions. Nevertheless, magnetometer measurements can be distorted under the influence of various internal and external disturbance sources and inherent sensor errors (Soken and Sakai, 2015).

1.2. Motivation and Objective

As the demand for indoor positioning systems and services is rising, it is of great necessity to investigate and develop a compact, cost-effective, energy-efficient, and convenient-to-use indoor positioning solution. With the development of MEMS for inertial sensing, a self-contained MEMS MARG sensor-based indoor positioning system has emerged as a natural solution and has a wide range of industrial applications, such as manufacturing, healthcare, emergency services, etc. Moreover, as a critical component of the IPS, IMU/MARG sensors-based heading and orientation estimation algorithms, also known as Attitude Heading Reference Systems (AHRSs), have become an independent but important research field. AHRS with high precision can be implemented in wearable rehabilitation systems tracking the movement of patients who have joint injuries or stroke patients for telehealth.

This research will contribute to the body of knowledge on MARG sensor-based indoor positioning system and algorithm development, especially for the engineering research fields of sensor fusion, PDR, and quaternion-based heading and orientation estimation. This study will help address several existing problems of adopting the MARG sensing package, most importantly to extract valuable location-based information from error-prone inertial sensors. In addition, this work designs and implements an IPS that can work in an unknown indoor environment without requiring a fingerprint map.

1.3. Contributions

This research will identify key issues and challenges regarding using inertial sensors, design a comprehensive indoor positioning solution, develop algorithms and models for extracting position information from inertial sensor measurements, and finally evaluate the effectiveness of the proposed solution.

The main contributions of the thesis are summarized as follows:

- **We propose a complete and practical IPS based on PDR using MEMS MARG sensors only.**
    - We design the system architecture at both the system level and the module level.
    - The proposed IPS includes four main modules: 1) mode detection, 2) step detection and moving distance estimation, 3) heading and orientation estimation, and 4) position estimation.

- **Magnetometer calibration is incorporated to ensure the reliability of magnetometer data.**
  - We adopt a mathematical hard-and-soft-iron model to eliminate potential magnetic field distortions and disturbances in magnetometer measurements.
- **We propose a dual-mode feature with mode switch capability.**
  - The dual-mode feature includes a static mode and a dynamic mode.
  - We provide the mode detection algorithm.
  - The step detection module is only triggered when the system enters the dynamic mode.
  - For the heading and orientation estimation module, both modes share a *prior* orientation estimate and use different methods to correct the *prior* estimate.
  - The dual-mode design reinforces the robustness and flexibility of the IPS algorithm.
- **We design algorithms of step detection, heading and orientation estimation, and position estimation.**
  - We devise a multi-layer **step detection** algorithm, which consists of the first scenario-based self-adaptive step detection layer with preliminary adjustment, the second step detection refinement layer based on signal pattern matching, and the third step validation layer.
  - We build upon the reference vector idea presented in (Madgwick *et al.,* 2020), and devise a quaternion-based **heading and orientation estimation** sensor fusion algorithm with a dual-mode feature. Specifically, both modes share a

prior orientation estimate generated by gyro integration. The static mode uses reference vectors to correct the *prior* estimate, while the dynamic mode uses a double phase extended Kalman filter to update the *prior* estimate.

- A **position estimation** algorithm is formulated.
- We evaluate the performance of the proposed IPS and compare it with the conventional methods under five scenarios.

1.4. Scope of the Study

The scope of this research and development project is to design and implement an indoor positioning and tracking system using MEMS MARG sensors consisting of accelerometer, magnetometer, and gyroscope. Data acquired from all inertial sensors is syncing to a cloud database in real time. The algorithms and system models are implemented on a computer. After receiving the inertial sensor data stream, the system will convert raw digital signal data to insightful results of detected steps, orientation estimation, position estimation, and moving trajectories. The details of the analog-to-digital converter (ADC) and the conversion process including the method of converting the analog signal to digital data are beyond the scope of this work.

This research focuses on designing and implementing indoor positioning models and algorithms. Smartphone built-in inertial sensors are used for collecting raw sensor data from the accelerometer, magnetometer, and gyroscope. Design, mechanics, and assembly of the sensor integrated circuit (IC) and the printed circuit board (PCB) are out of the scope of this study.

Past research approaches classify the status and calculate the relative position of the smartphone-embedded inertial sensors so that the models and algorithms can be adapted to general smartphone use cases. In this research, the smartphone is handheld, and the relative position of the inertial sensors is fixed to the pedestrian's body. Incorporating machine learning and deep learning algorithms for smartphone status classification and activity recognition is not the emphasis of this research. However, it is considered as future work of this research to extend the functionality and models of our system.

1.5. Organization

The remaining of the thesis is organized as follows. Chapter 2 gives the detailed background of IPSs and discusses core technologies and different approaches explored for solving the problem. A comprehensive literature review of indoor positioning methods is provided as well. Chapter 3 formulates the indoor positioning problem and presents the methodology, system architecture, design models, and algorithms of the proposed IPS. Experimental evaluation of the proposed system is provided in Chapter 4. Chapter 5 summarizes the work and discusses several directions of future work.

Chapter 2

Background and Literature Review

2.1. Introduction

With the development of communication technology, the advancements in sensing technology, and widely adoptions of machine learning and deep learning algorithms in the engineering field, location-based systems and services have experienced substantial growth over the past decade. Indoor positioning has attracted great research attention in recent years. Infrastructure-based and infrastructure-free are the two broad categories of indoor positing solutions. Among various infrastructure-free IPSs, PDR is a major approach and a self-contained navigation technique in which measurements are used to update the orientation and position of the object. Although the position estimation process of PDR is straightforward, PDR is subject to cumulative errors. In addition, developments in microelectromechanical systems (MEMS) have enabled IPS to be built using smartphone-embedded MARG sensors. Moreover, Attitude Heading Reference System (AHRS) is a core research sub-field in terms of developing PDR-based IPSs. This chapter provides a brief background on IPS, PDR, inertial pedestrian navigation, and AHRS, and reviews recent research under these fields. At the end of this chapter, we discuss the main challenges and limitations of existing indoor positioning and tracking methods.

2.2. Indoor Positioning System

An Indoor Positioning System (IPS) is a localization system used within a closed space to provide users and smart objects with detailed location information continuously and in real time through measurements acquired from a network of devices. A wide range of technologies has been considered for implementing indoor localization. Existing solutions mainly fall into three categories: 1) communication technology-based, 2) sensor-based, and 3) computer vision and image-based. A hierarchical classification of the existing IPSs is given in Table 2.1 based on adopted technology.

Table 2.1: Hierarchical Classification of the Existing IPSs

Based on Adopted Technology

Indoor Positioning Systems		
Communication technology-based	**Sensor-based**	**Computer vision and image-based**
• Mobile Telecommunication Networks: 4G, 5G • GNSS • Wireless Local Area Networks (WLAN): Frequency Identification (RFID), Wi-Fi, Bluetooth, UWB • Visible Light Communication (VLC)	• MEMS IMU Pedestrian Dead Reckoning (PDR): Gyroscope, Accelerometer, Magnetometer • Infrared • Ultrasound • Pressure Sensors	• Camera - Computer Vision • Machine Learning • Deep Learning
• Hybrid Methods and Integrated Systems		

Communication Technology-based IPS

Communication-based approaches for indoor positioning include mobile telecommunication networks such as 4G, 5G, Global Navigation Satellite System (GNSS), Wi-Fi, ultra-wideband (UWB), Bluetooth, radio-frequency identification (RFID), visible light communication (VLC), etc.

In early radio frequency-based systems (Wann *et al.*, 2006) (Yang *et al.*, 2008), received signal strength (RSS), time of arrival (TOA), time difference of arrival (TDOA), and angle of arrival (AOA) were used for position estimation. The accuracy of the estimated result is satisfactory only when the pedestrian is in line-of-sight (LOS). A minimum of three base stations are required to be seen from the moving pedestrian. This can be difficult in some manufactory environments. RSS-based positioning methods rely on the trilateration technique and simply use the received signal strength indicator (RSSI) as the main parameter to determine location (Golestanian *et al.*, 2017).

In (Diallo *et al.*, 2019), the authors proposed an indoor localization system which uses RFID tags to locate a moving object by assuming that the position of the object can be derived from the signal strength reflected by RFID tags. The limitation of RFID-based approaches is that the accuracy of the algorithms highly depends on how RFID tags are deployed, the number of tags, and the level of the signal coverage in the object's surrounding area.

Wi-Fi-based IPSs may use fingerprinting, proximity algorithm, and triangulation algorithms for positioning (Ma *et al.*, 2015). Wi-Fi fingerprinting has gained much attention since it does not require line-of-sight measurements from access points (APs) and achieves improved accuracy in complex indoor environment. Wi-Fi fingerprinting associates a position with its detected signal patterns. It is a two-phase process, which has an initial offline survey phase and an online query phase. In the offline phase, a site-wide fingerprints map of all the RSSI vectors of reference points (RPs) is constructed and stored in a database. In the online phase, the server uses certain similarity metric in the signal space to compare the measured RSSI vector with the stored fingerprints and determine the pedestrian's current position (He and Chan, 2016). Constructing and updating the fingerprint map is time-consuming. Recent advances try to introduce autonomous elements into the system (Dai *et al.*, 2020). CHISEL, a deep learning-based framework, proposed in (Wang *et al.*, 2021) further compressed the deep leaning convolutional autoencoder convolutional neural network (CAECNN) model while maintaining highly accurate result.

Bluetooth-based systems (Ji *et al.*, 2015) (Yang *et al.*, 2020) are similar to Wi-Fi-based systems, and these systems add Bluetooth Low Energy (BLE) beacons as a low power consumption alternative of Wi-Fi APs so that signal coverage can be extended to the places where Wi-Fi signals cannot reach.

Given the fact that positioning approaches built on radio frequency signals are affected by electromagnetic interference, VLC-based systems calculate the position through detecting light emitting diodes (LEDs) illumination installed in the indoor space along with TDOA, RSS, AOA, and neural network propagation so that the system can achieve optimized results with greater accuracy and no interference (Othman *et al.*, 2021).

Due to the high precision, no line-of-sight requirement, and the absence of multi path distortion, UWB-based positioning systems have been extensively studied and tested in recent positioning-related research. In contrast to RFID technology, UWB uses multiple bands of frequencies to transmit its signal simultaneously. The drawback of deploying UWB-based systems is that they can cause interference to nearby systems that operate in the ultra-wide spectrum and also to aircraft navigation radio equipment. Moreover, the infrastructure and set of tags for UWB systems are very costly.

For communication technology-based IPSs, the positioning accuracy is affected by the layout and arrangement of APs, base stations, tags, other wireless equipment, and the indoor environment changes.

Sensor-based IPS

There are two major types of sensor-based positioning system. The first type uses the traditional infrastructure-heavy sensors such as infrared, ultrasound, and pressure sensors. Although they can achieve high accuracy, these methods tend to lose popularity because a wide deployment of heavy infrastructures is needed. On the contrary, embeddable microelectromechanical systems (MEMS) IMU sensors are widely used for indoor positioning because of the ease of use of these compact sensors. Various sensor fusion algorithms have been developed for estimating the position of target objects. Using gyroscope, accelerometer, and magnetometer embedded in smartphones is an infrastructure-free and cost-efficient approach for indoor positioning applications. The process of PDR is often used to calculate current position of a target moving object by using the previously determined positions through built-in inertial sensors. However, dead reckoning is subject to cumulative errors. As a result, inertial sensor-based systems need recalibration to maintain the accuracy. PDR is described in Section 2.3. A detailed review of the state-of-the-art research development and a method comparison of IMU-based positioning systems are elaborated in Section 2.4.

Computer Vision and Image-based IPS

Computer vision and machine learning are becoming crucial components in engineering applications. Computer vision and image-based systems extract location information through images and videos captured by 3D cameras or built-in cameras in smart devices. Latest visual-based indoor positioning methods include image retrieval-based methods, visual landmarks-based methods, and learning-based methods (Wang *et al.*, 2019). Visual-based methods rely on image recognition techniques and similarity search. To improve the algorithmic efficiency, image processing algorithms such as Scale Invariant Feature Transform (SIFT) have been adopted for points matching in position estimation. To overcome the time-consuming and labor-intensive data collecting process, robot simultaneous localization and mapping (SLAM) has been introduced to construct a map of an unknown environment while tracking the location of an object at the same time. Machine learning algorithms such as SVM, LSTM, etc. were integrated into the conventional IMU sensor and computer visual hybrid model for human motion detection so as to boost the accuracy of estimated location (Herath *et al.*, 2020) (Yan *et al.*, 2017).

IPS Comparison

Selecting the best model and technical solution depends on the indoor environment and the specific requirements of the user. Table 2.2 summarizes the cost and accuracy of IPSs based on different technologies. Hybrid methods also have been proposed to compensate for the weaknesses of independent systems.

Table 2.2: Indoor Positioning Systems Cost and Accuracy Comparison

Indoor Positioning Technology	**Accuracy**	**Cost**
Mobile Telecommunication Networks	1-2 m	Low
RFID	1 m	Medium / High
Wi-Fi	2-5 m	Low
Bluetooth	1 m	Low
VLC	1 m	Low / Medium
UWB	10 cm	High
IMU	1 m	Low
Infrared, Ultrasound, Pressure Sensors	1-5 m	High
Computer Vision	1-2 m	Low / Medium
SLAM	1 m	Medium / High

2.3. Pedestrian Dead Reckoning

Pedestrian Dead Reckoning (PDR) is the process of calculating current position of a moving object by utilizing the previously calculated position information through continuously detecting gait event and updating current estimations of speed, step length, and heading direction. By leveraging the advantages of microelectromechanical systems (MEMS), most smart devices and smartphones are now embedded with IMU, which includes 3-DoF (Degrees of Freedom) accelerometer, magnetometer, and gyroscope sensors. Inertial sensors are commonly used by unmanned aerial vehicles (UAVs), aircraft, spacecraft, and industrial robots to track the position and heading of the moving object. PDR methods are generally based on path integration (Giorgi *et al.*, 2017), which consists of four major phases: 1) step detection, 2) step length estimation, 3) heading estimation, and 4) position calculation. The position of an object at each moment is determined by combing the distance travelled and the heading directions, which are calculated and estimated by using different sets of sensor measurements. Another PDR method is based on the strapdown algorithm, which simply numerically double integrates the tri-axis accelerometer data to determine the position.

The PDR technique has been the key model component of several navigation projects for various environments where satellite navigation has poor performance (Gartner *et al.*, 2003) (Ott *et al.*, 2005). Recently, the inertial sensor-based Dead Reckoning (DR) algorithm has been integrated with Kalman filter and deep neural networks and applied to vehicle navigation applications (Brossard *et al.*, 2020) (Mikov *et al.*, 2020). For pedestrian indoor positioning, hybrid models incorporating the PDR technique into exiting communication technology-based methods such as Wi-Fi, magnetic matching, iBeacon, UWB have been extensively studied (Li *et al.*, 2016) (Huang *et al.*, 2019) (Lee *et al.*, 2021) (Poulose and Han, 2019) (Hu *et al.*, 2014). Moreover, the design procedure and accuracy of the systems are dependent on the way inertial sensors are placed on the pedestrian's body. A chest-mounted IMU PDR system (Lu *et al.*, 2019), a conventional foot-mounted PDR system (Zhang *et al.*, 2020), a handheld device system (Abid *et al.*, 2017) were investigated over the past five years. A neural network-based system RoNIN was proposed in (Herath *et al.*, 2020), where a mathematical model is used to detect if the inertial sensor device is handheld, in the pocket or in a bag.

Compared with positioning approaches based on wireless signals and computer vision sensors, PDR algorithms can generate relatively accurate position estimations in a short period of time with low power consumption, low computational complexity and fast updating and iteration speed. On the contrary, computer vision methods cost much longer time training the model and processing high dimensional image data and visual contents. However, PDR is subject to cumulative errors of approximation for longer journeys due to taking no account of directional drift. To address the cumulative error issue, a multi-sensor fusion-based method (Qiu *et al.*, 2018) and a robust adaptive Kalman filter (RAKF) algorithm combined with complementary filter (Fan *et al.*, 2019) were devised to improve the positioning accuracy. Also, (Chen *et al.*, 2017) and (Li *et al.*, 2015) proposed to fuse inertial sensor data with UWB or Bluetooth data respectively. For these hybrid methods, inertial positioning compensates for the defects of wireless signal obstacles. In return, UWB and Bluetooth positioning mitigate the cumulative errors resulting from the PDR process.

2.4. Inertial Pedestrian Navigation

An Inertial Measurement Unit (IMU) is an electronic device that delivers navigation parameters, specific forces, and position update information using a combination of accelerometers, magnetometers, and gyroscopes. Specifically, velocity, attitude and orientation, and position of a moving object can be derived from linear acceleration recorded by accelerometers, magnetic field measured by magnetometers, and angular rate of rotational motion reported by gyroscopes.

The accuracy of the inertial sensors is influenced by a number of errors and biases. Short-term measurements of accelerometer fluctuate to a certain degree due to sensor noise, constant bias, and vibration ratification error (VRE). The gravity measured by the accelerometer also acts as a bias. Since it measures all linear acceleration of the body, rotation motion can cause accelerometer estimation errors. Measurement of magnetometer can be corrupted by magnetic field disturbances of the surrounding environment. The magnetic disturbances mainly come from hard-iron sources such as magnet or electric and soft-iron sources such as iron materials. One solution for removing the hard-iron and soft-iron distortion is to recalibrate the magnetometer by finding the offset and transformation matrix. Gyroscope measures the angular rate of the entire system. It produces reliable measurement in the presence of linear acceleration. However, gyroscope data smoothly drifts away after integrating the measured angular rate over time due to its sensor bias and high frequency white noises.

Inertial Pedestrian Navigation Challenges

In practice, there are considerable challenges in obtaining a highly precise positioning solution with MEMS IMU sensors. Double integrating the acceleration to obtain the position results in errors that grow quadratically over time. To eliminate the errors, an alternative position update needs to be integrated into the system. In (Li *et al.*, 2018), the authors proposed a UWB/PDR fusion algorithm. It was shown that the cumulative errors of the heading direction calculation are effectively mitigated in the case of intermittent or continuous UWB ranging noise and signal interruption.

For approaches that solely rely on inertial sensors, various sensor fusion techniques have been applied to achieve optimal results in inertial pedestrian navigation (IPN). In (Giorgi *et al.*, 2017), the authors examined a magnetic field compensation technique with a traditional simple Kalman filter, which detects direction changes by analyzing the KF innovation. The proposed low computational complexity algorithm was shown to be more effective for short paths than long paths. One drawback of this solution is that the estimation of heading direction is based on detecting relative changes of the person with respect to a reference system built during the initialization phase. In this design case, absolute headings cannot be determined, and the system is not sensitive to small rotations.

In (Poulose *et al.*, 2019), the authors presented a sensor fusion-based position estimation algorithm, which consists of two separate Kalman filters. The first Kalman filter combines the accelerometer and gyroscope data to estimate pitch and roll angles, whereas the second Kalman filter utilizes the complementary features of magnetometer and gyroscope to collectively estimate heading direction, which addresses the cumulative error issue in PDR inertial positioning. Although the experimental results demonstrated improved accuracy, the effectiveness of the algorithm for more complicated routes has not been tested.

In (Fan *et al.*, 2019), the authors incorporated wavelet decomposition for signal processing, investigated a dynamic step length model, and explored a double stage filter fusion algorithm for heading estimation. In the first stage, a complementary filter is used to fuse magnetometer, accelerometer, and gyroscope in order to suppress the accumulated errors. In the second stage, a robust adaptive Kalman filter (RAKF) is designed to restrain outliers by introducing a weighting factor based on Huber function. It was shown that combing RAKF with basic complementary filter reduced the influence of outliers on the positioning results. There is still room for improvement regarding the accuracy of heading estimation. Fan's work can be extended by concentrating more on body kinematics models to further improve the positioning algorithm.

Sensor Fusion

Sensor fusion can help improve performance in PDR by taking advantage of the fact that inertial sensors complement each other. More specifically, gyroscope is more accurate over the short term but the gradual accumulation of uncorrected bias instability and angular random walk (ARW), a random high frequency noise, causes gyroscope drift over long term scale. In contrast, accelerometer measurement is relatively stable over the long term because the gravity vector does not change over time, but it is not precise in the short term due to high frequency noises and its high sensitivity to all accelerations.

To tackle the error accumulation and heading drift problem, several techniques including Zero Velocity Potential Update (ZUPT) (Yun *et al.*, 2007), Zero Angular Rate Update (ZARU) (Jiménez *et al.*, 2010), and Heuristic Drift Reduction (HDR) (Borenstein *et al.*, 2009) have been investigated to improve the overall performance of positioning algorithms.

The Heuristic Drift Reduction (HDR) method proposed in (Borenstein *et al.*, 2009) estimates the likelihood that the person is walking along a straight line. Its closed-loop control approach tracks the drift and corrects the gyro measurements when the estimated azimuth angle matches the nearest dominant direction. Built on the basic HDR algorithm, a double low-pass filter and a threshold were applied as enhancements to reduce the impact of real motion conditions of swaying and curving. The limitation of the HDR method is that it introduces errors of its own when the pedestrian walks in various directions and there is always a residual offset. This work was extended in (Ju *et al.*, 2014) where an EKF-based Advanced Heuristic Drift Elimination (AHDE) method is introduced to remove azimuth drift error in indoor environments. The proposed AHDE algorithm follows a two-step model. First, it classifies the motion of the pedestrian into non-straight motion, straight motion, and straight motion along the dominant direction. If the pedestrian is not walking straight forward, the algorithm adopts the ZUPT method. However, if it is determined that the pedestrian is walking straight along the dominant direction, the algorithm corrects the estimated azimuth angle to the closest dominant direction. If the pedestrian is walking straight but not along the dominant direction, the algorithm applies a correction to the gyroscope bias. Both the HDR method and the AHDE method are limited to a more structured indoor environment, which in reality is not always the case as there could be more open space in the indoor environment.

By noticing the periodic nature of motions such as walking and running, (Yun *et al.*, 2007) describes that there are short periods of zero foot velocity when the foot is in full contact with the ground. This pattern is proved to be practical for drift correction when integrating accelerometer data. Zero-velocity detection algorithms have been investigated over the years. Most of the ZUPT-based algorithms use threshold values, which is highly dependent on the motion pattern of the specific pedestrian. For the purpose of generalizing the ZUPT algorithm, (Cho and Park, 2019) explored a threshold-less zero-velocity detection algorithm for PDR. Their method is based on signal processing techniques such as sliding window averaging (SWA), positive peak detection, and negative peak detection. Different from conventional methods, zero velocity is detected by recognizing the shape of the processed signal. However, this algorithm is not robust enough for all kinds of signal forms.

Other than the above-mentioned methods, mathematical modelling also has been applied to remove the drift. In (Kuxdorf-Alkirata *et al.*, 2019), the authors introduced a calibration and training phase and then constructed a linear time-dependent error model for drift reduction followed by a post-processing filter. The error model predicts the trend of the error by fitting a function based on data obtained during the training phase, and then adaptively adjusts the results of the path reconstruction algorithm.

Machine Learning

As machine learning is extensively used in the engineering field in recent years, it has been incorporated into the conventional inertial pedestrian navigation systems. In (Kone *et al.*, 2020), the authors presented two machine learning-based zero-velocity detectors: histogram-based gradient boosting (HGB) and random forest (RF). The design objective is to adapt a single model to different motion types while maintaining a low computational cost. In (Deng *et al.*, 2020), the authors formulated inertial sensor-based position estimation as a supervised regression problem with three regressors: random forest regressor (RFR), k-nearest neighbor regressor (KNNR), and support vector regressor (SVR). The method builds a regression model between IMU sensor data and 2D coordinates and then predicts coordinates with real-time measurement data. Simulation results showed that the drift of long travels is effectively suppressed. However, as supervised learning algorithms, these two Machine Learning-based systems require pre-collected training data and certain training process.

2.5. Attitude Heading Reference Systems (AHRSs)

Unlike the strapdown algorithm of PDR, the step-and-direction approach is largely determined by how accurately heading can be estimated before combing the step detection result and the heading estimation result to obtain the position. The topic of orientation tracking and heading estimation, also known as Attitude and Heading Reference System (AHRS), has been extensively studied over the past decade and is considered as an important branch of inertial navigation system. An AHRS, also referred to as MARG (Magnetic, Angular Rate, and Gravity) system, consists of tri-axis sensors and is able to provide attitude information and measurement of orientation relative to the direction of gravity and the earth's magnetic field. By leveraging the advantages of sensor fusion, reference vectors compensate for the gyroscope drift caused by gyroscope integration.

Inspired by the non-linear complementary filters approach implemented by (Mahony *et al.*, 2008), (Madgwick *et al.*, 2011) formulated the orientation estimation problem as an optimization problem and proposed a mathematical model for orientation estimation using the gradient descent algorithm. To overcome the slow convergence issue and decouple the calculation of roll angle from the calculation of yaw angle, the authors extended their previous work by presenting a new extended complementary filter for full-body orientation estimation. The algorithmic model is established in a way that the orientation in the global reference frame, represented by a quaternion vector, is highly dependent on K the value of gain, especially during the initial convergence phase when the initial orientation is unknown. The K gain controls the weight of the orientation correction terms calculated from accelerometer and magnetometer data. The algorithm can be further improved if the parameters are designed in a self-adaptive way with a shortened initial convergence period (Madgwick *et al.*, 2020).

2.6. Main Challenges and Limitations of Existing Indoor Positioning Methods

The main challenges and limitations of existing indoor positioning methods are summarized as follows.

Limitations of Existing Indoor Positioning Methods

- Most existing IPSs require the installation of special hardware devices, which can be time-consuming and expensive. An approach like fingerprinting is labor-intensive and requires constant updates once the indoor environment changes.
- Hybrid IPSs that combine two or more technologies demonstrate improved accuracy at the sacrifice of computational efficiency.

- Machine learning-based approaches and computer vision methods cost much time training the model and plenty of computational resources processing high dimensional image data and visual contents.
- Most research in the indoor positioning field focuses on either gait analysis or AHRS. There is a lack of research regarding developing a complete and practical IPS.
- Some existing approaches assume the pedestrian is walking at a constant speed. We notice that the signal pattern differs significantly when the pedestrian is moving at varying speeds.
- The design of the step detection algorithm is influenced by how the sensors are fixed to the pedestrian’s body. Most of the existing step detection models only use accelerometer data.

Main Challenges

- The low-cost MARG inertial sensors are inherently error-prone. The magnetometer data is influenced by magnetic distortions and disturbances. The accelerometer data can be corrupted by high-frequency sensor noises and disturbances. On the contrary, the gyroscope data drifts away over time as a result of the gyro drift issue.
- The biggest challenge is how to design sensor fusion models and algorithms to extract accurate location-based information from individual motion and magnetic sensors. The sensor fusion models and algorithms need to be designed in a complementary way that cumulative errors of each of the sensors are mitigated to the greatest extent.

- In practice, as cumulative errors grow linearly in velocity estimation due to noisy accelerometer data, double integration can cause quadratic drift error growth in position estimation over time.
- For step detection, false positive steps are difficult to be detected and removed. Sensor data processing using DSP is challenging.
- For heading and orientation estimation, the tolerable range of the orientation estimation error is low because a minor error in orientation estimate results in completely deviated trajectory coordinates.
- In addition, adopting quaternions for orientation estimation is not as intuitive as using the conventional Euler angles.

Chapter 3

MEMS-MARG-Based Indoor Positioning and Tracking

3.1. Introduction

This chapter considers the indoor localization problem in sensor fusion and signal processing. We explore and implement a complete solution for PDR position estimation with self-contained inertial MARG sensors consisting of a set of smart-phone embedded gyroscope, accelerometer, and magnetometer. The designed positioning and tracking module is the fundamental component of a full-fledged indoor navigation system.

There are several challenges that need to be addressed when utilizing MEMS inertial sensors in smartphones for indoor positioning. Although some low-cost commercial inertial sensors are extensively used in robotics and unmanned aerial vehicles (UAVs), IMU and MARG are not high precision equipment, and they are subject to inherent sensor errors, noises, cross axis and environment sensitivity. Each different type of inertial sensor has its own weaknesses specific to each sensor design mechanism. Magnetometers are normally affected by disturbances of the surrounding magnetic field, which are also known as the hard-iron effect and soft-iron effect. Accelerometer measurement is easily corrupted by high frequency disturbances. Main sources of accumulative accelerometer errors include constant bias, velocity random walk (VRW), scale factor, and vibration rectification error (VRE). Using double integration to calculate positions from acceleration measurement data, the estimation errors tend to increase quadratically over time. Gyroscope is prone to accumulate a slow changing but massive offset, also known as gyroscope drift, due to a bias instability and a high frequency noise named angular random walk (ARW). Sensor fusion algorithms (Poulose *et al.*, 2019) are proved to be effective by utilizing the complementary features of each inertial sensor to collectively estimate heading and position. However, constructing an optimal sensor fusion model and balancing the trade-off between computational complexity and accuracy remain a critical challenge. Furthermore, Euler angles suffer from the gimbal lock problem (Mansur *et al.*, 2020) when they are used to describe the orientation of a rigid body with respect to a fixed coordinate system.

We propose an inertial sensor-based position estimation system and framework for indoor positioning, which solely relies on tri-axis accelerometer, tri-axis magnetometer, and tri-axis gyroscope embedded in smartphones. The system does not require additional infrastructure support and the inertial sensor data stream is collected using a smartphone mobile application. The proposed indoor positioning framework includes four main modules: mode detection, step detection and moving distance estimation, heading and orientation estimation, and position estimation. To remove the internal errors and external disturbances of the magnetometer, magnetometer calibration is included and performed during the calibration phase. To overcome the limitations of individual inertial sensors, our solution is based on sensor fusion and signal processing techniques. To address the gimble lock issue, we adopt quaternions to represent rotations and the heading and orientation of a rigid body at each discrete time instance. Experimental evaluation shows that the proposed system is able to achieve high positioning accuracy with comparatively low computational cost.

The main components of the IPS solution include:

- **Comprehensive IPS solution based on MARG sensing package:**

  The proposed indoor positioning system includes a mode detection module, a step detection and moving distance estimation module, a heading and orientation estimation module, and a position estimation module. The architectures at the system level and module levels are devised.

- **Dual-mode design:**

  The proposed system has two modes: static mode and dynamic mode. The step detection module is only triggered in the dynamic mode. For heading and orientation estimation, both modes share a *prior* orientation estimate and then use different models to update the *prior* estimate. The mode detection module uses a sliding time window technique for the mode detection algorithm.

- **Multi-layer step detection:**

  The proposed step detection module includes a preliminary step adjustment layer, a second step refinement layer, and a third step validation layer. Accelerometer data is used in the first step detection layer. Gyroscope data and magnetometer data are exploited in the step refinement layer to enhance the step detection result.

- **Quaternion-based heading and orientation estimation:**

  To address the gimbal lock issue, a unit quaternion is used as the system state for orientation estimation.

- **Magnetic distortion correction-incorporated system model:**

  The proposed IPS adopts a hard-and-soft -iron model based on the least square fitting of a deviated ellipsoid to correct potential magnetic distortions caused by the surrounding indoor environment.

The rest of this chapter is organized as follows. Section 3.2 describes the methodology used for developing the MARG sensor-based IPS. This section includes a formal statement of the problem, methodology of the solution, and design goals. Section 3.3 discusses the system structure of conventional IMU-based IPSs and devises the system architecture of the proposed indoor positioning system at the system level. Section 3.4 defines the coordinate system and reference frames used for developing the system. Mathematical foundations for orientation representation including special orthogonal group, rotation matrix, Euler angles, gimbal lock, and quaternions are provided in Section 3.5. Section 3.6 focuses on studying magnetic distortion and the influencing factors as well as the external and internal causes. Then to eliminate potential magnetic field distortions, the hard-and-soft-iron model based on least square fitting of a deviated ellipsoid is presented. In Section 3.7, the mode detection algorithm with a sliding time window technique is proposed and explained. Section 3.8 proposes the multi-layer step detection algorithm and provides the structure of the proposed step detection module at the module level. Sensor fusion techniques and modeling of the dual-mode heading and orientation estimation module are elaborated in Section 3.9. Finally, Section 3.10 formulates the position estimation equations for the position estimation module.

3.2. Statement of the Problem

We assume the target moving object is equipped with MARG sensors. The proposed indoor positioning system aims to generate i) a detected step set $S = \{s_j\}_{j=1}^{J}$, ii) the moving distance increment D_i, iii) orientation and heading system states ${}_{w}^{b}q$, and iv) pedestrian position coordinate values $\left(XPose\big(t(i)\big)\,,\ YPose\big(t(i)\big)\,\right)$, where s_j denotes the jth step and J is the total detected steps, ${}_{w}^{b}q$ is a unit quaternion with b and w, respectively, denoting the body frame and the world frame, $t(i)$, $i = 1, 2, \dots, N$, denotes the time instance for the ith measurement, and N is the total number of measurements. The orientation and heading system state can also be represented by the Euler angles set (ϕ, θ, ψ) where ϕ, θ, ψ are defined as the roll angle, the pitch angle, and the yaw angle respectively. This thesis focuses on designing the PDR positioning and tracking estimation algorithm and optimizing the estimation accuracy.

This IPS research is guided by inertial navigation, which is a navigation technique that uses motion sensors and rotation sensors to continuously determine the current velocity, orientation, and position of a target moving object calculated by PDR. To fully exploit the information provided by inertial sensors, we collect data from a full set of self-contained MARG sensors.

Specifically, we consider the scenario that the smartphone is handheld and placed in front of the waist area with its screen facing up. It is different from the scenario that allows the object to handle the IMU-embedded smartphone freely and perform normal activities while measuring, which will not be discussed in this thesis. To recognize various normal phone activities, machine learning-based models and classifiers are used to first identify the position of the phone relative to the pedestrian (Herath *et al.*, 2020). It needs to be noted that for the proposed indoor positioning system the relative position of the smartphone is fixed to the pedestrian's body.

A high-level system block diagram of the methodology is shown in Figure 3.1. The system starts with acquiring data from the accelerometer, magnetometer, and gyroscope sensors. The data is processed to generate intermediate results, which include detecting steps and estimating step length and orientation, and finally the intermediate results are combined to produce position estimations. The system runs on a computer in real time or offline after the smart device sends the inertial sensor data stream to a cloud database. For future work, the system can be executed on the smart device as a mobile application, or the algorithmic model can be directly programmed onto some microcontroller.

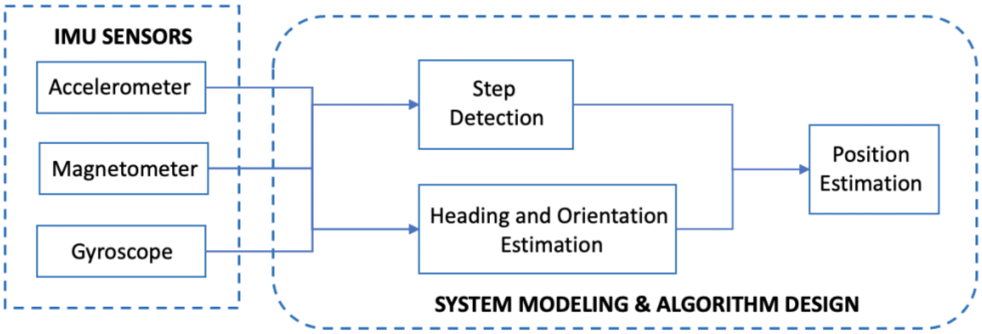

Figure 3.1: System Block Diagram

3.3. System Architecture

Position and orientation in an indoor environment can be determined using different methods and algorithms. The conventional position estimation algorithms filter accelerometer data for step detection and use gyroscope data to generate heading information. The common system structure of a conventional IMU-based indoor positioning system is shown in Figure 3.2. The positioning accuracy is affected by the cumulated errors from both accelerometer and gyroscope sensors. Position estimation results will deviate from the true position after combining unreliable orientation estimation and step count.

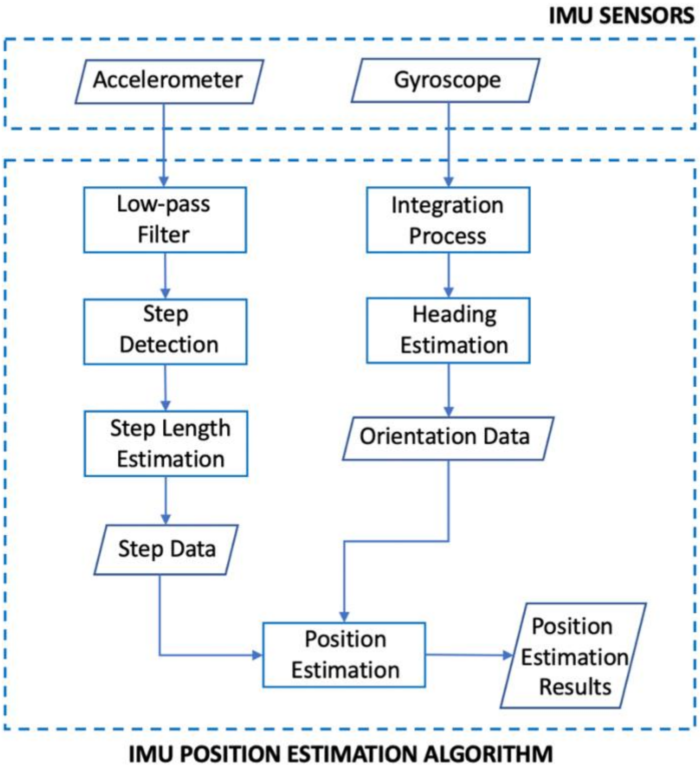

Figure 3.2: Conventional IMU Indoor Positioning System Structure

Directly taking the integral of raw accelerometer measurement data from the sensor stream results in false velocity hence false moving distance estimations. Figure 3.3 shows an incorrect and gradually increased velocity calculated by using raw accelerometer data with cumulative errors. Gyroscope drift also occurs when orientation is estimated by integrating gyroscope measurement data with respect to time. The calculated angle value of each axis slowly drifts away as shown in Figure 3.4.

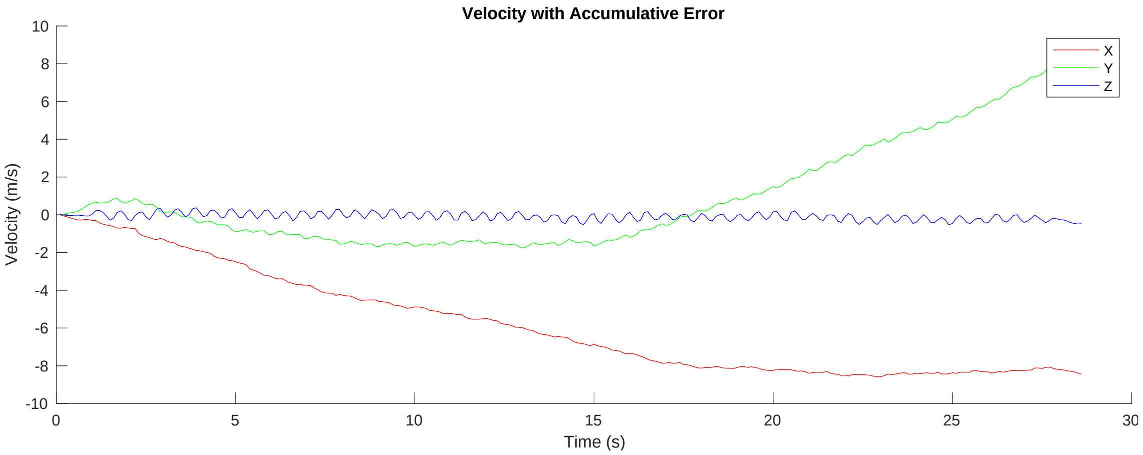

Figure 3.3: Velocity Calculated by Using Raw Accelerometer Measurement Data

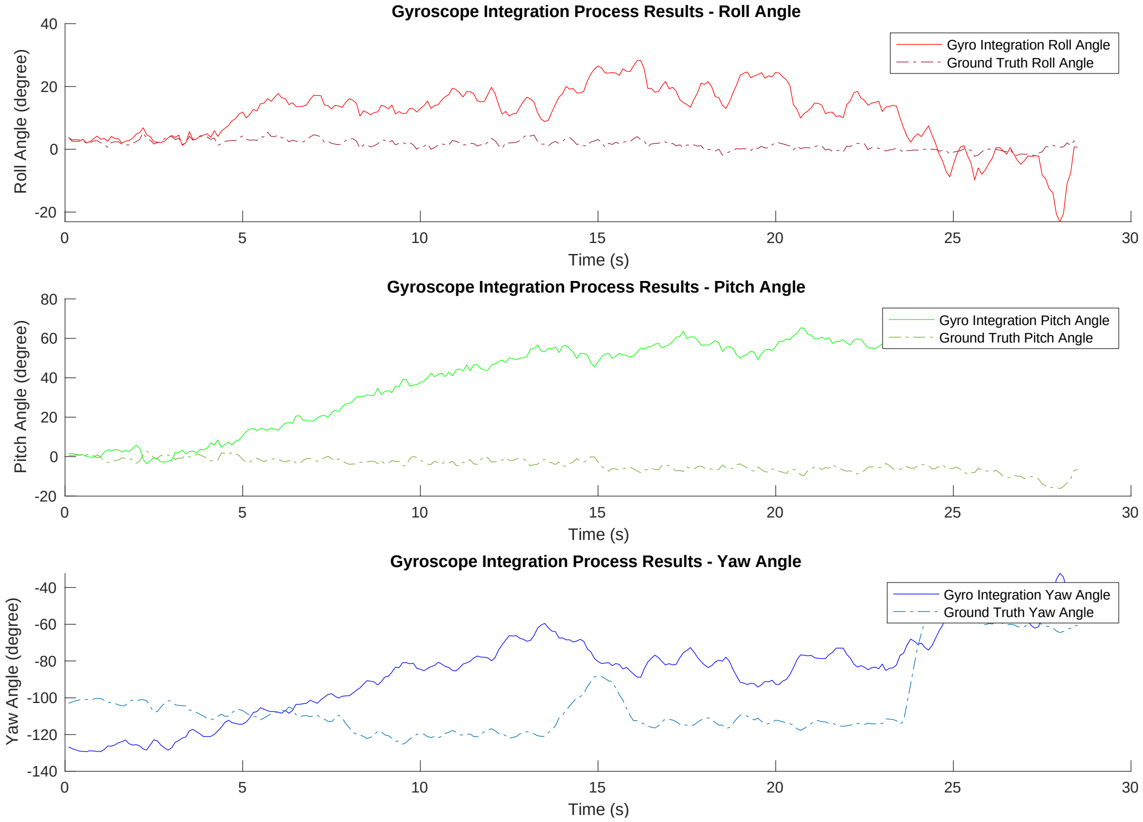

Figure 3.4: Heading and Orientation Estimation Results

through Gyroscope Integration Process

To address the cumulative error issue that exists in simple conventional IPSs and to obtain more accurate position information, we design a more complex IMU MARG-based indoor positioning system with a mode switch feature and an initialization and calibration phase. The overall system architecture of the system is illustrated in Figure 3.5.

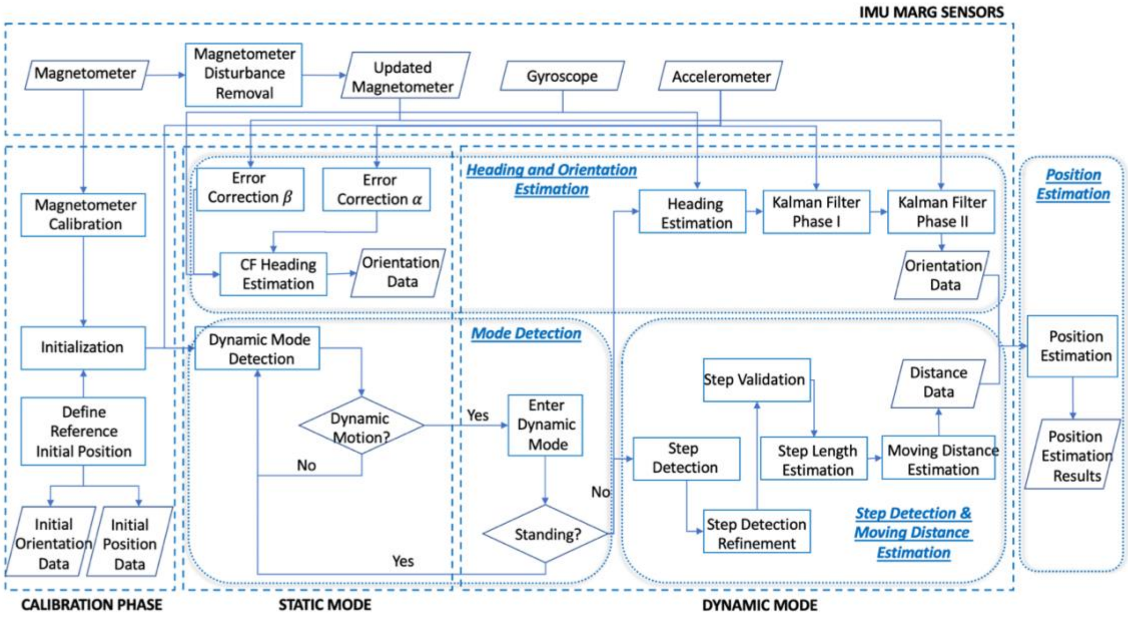

Figure 3.5: System Architecture of the Proposed Indoor Positioning System

We consider a scenario in which a pedestrian is moving inside an unstructured indoor environment while holding a mobile device for motion detection, data collection, and self-navigation. The sampling rates of the inertial sensors are set to 10 Hz initially and can be adjusted if needed.

During the initialization and calibration phase, the reference initial position is also defined. Updated and fully calibrated magnetometer measurements are used for further heading and position estimation.

The presented indoor positioning system runs in two different modes: static mode and dynamic mode. After the initialization and calibration phase, a dynamic mode detection module identifies which mode the pedestrian's motion falls into. The two modes differentiate scenarios in which external accelerations dominate accelerometer measurements from scenarios in which the pedestrian is standing still and only performing turning motions.

Heading and orientation estimation algorithms for both static and dynamic modes are based on sensor fusion but with different fusing mechanisms. In the static mode, reference vectors are used to correct the preliminarily estimated heading calculated by gyroscope integration. In the dynamic mode, a double phase extended Kalman filter is adopted so that a separate Kalman gain controls the fusing process, which can mitigate the influence of using the inaccurate reference gravity vector when the pedestrian is moving at a fast speed. One of the advantages of this design is that the heading and orientation estimation algorithm for the static mode boosts computational efficiency and saves computational resources since the extended Kalman filter is comparatively computationally expensive. Another advantage is that the two modes of the heading and orientation module share the gyroscope integration process calculations while having their own scenario-specific correction elements, which enhances the flexibility of the system design.

Finally, in the position estimation module a pedestrian moving trajectory in the two-dimensional space is generated by combining distance and orientation data obtained from the step detection and moving distance estimation module and the heading and orientation estimation module. The structure of the step detection module is expanded and elaborated in Section 3.8.

3.4. Coordinate System and Reference Frames

Reference Frames

A reference frame is specified by an ordered set of three mutually orthogonal unit-length vectors. A coordinate system specifies a mechanism for determining the position of points within a reference frame (NAIF, 2020). When using the quantities measured by inertial sensors to build system state (position, velocity, etc.) models and describe orientation data, both the reference frame and coordinate system need to be defined. There are two types of reference frames: inertial and non-inertial. The difference between them is that the inertial frame does not undergo acceleration and has constant velocity while the non-inertial frame is accelerated with respect to the assumed inertial frame of reference. Low of inertia only holds in the inertial frame of reference. To represent the attitude and heading of a rigid body in a three-dimensional space using sensor measurements, we consider transforming the data in two frames (Titterton and Weston, 2005):

The body/sensor frame b is the coordinate frame of the IMU. In other words, the body frame is the coordinate system that is aligned with the body of the IMU sensor. The center of this frame is located in the center of the inertial sensors. As the object's own coordinate frame, it is fixed to the body and rotates with the device.

The navigation frame n is the reference coordinate frame in which we want to navigate. In other words, our goal is to express the position and orientation of the body/sensor frame b with respect to the navigation frame n. In this application, the navigation frame n is defined to be stationary relative to the *world/inertial frame w*, which is fixed in the inertial space. The origin is at the center of the earth. A North-East-Down (NED) frame or an East-North-Up (ENU) frame is often used as a local reference frame for orientation estimation.

Inertial Measurement Unit (IMU) Coordinate System

Smartphone inertial sensors use a standard three-axis coordinate system to express data values. Figure 3.6 shows the coordinate system used in iOS devices. The origin of the coordinate system is the center of the IMU sensor. When the device is held facing up, the positive x-axis is horizontal and points to the right, the positive y-axis is vertical and points to the front, and the positive z-axis is perpendicular to the xy-plane and points up. Inertial sensor values may be positive or negative depending on the direction of the motion.

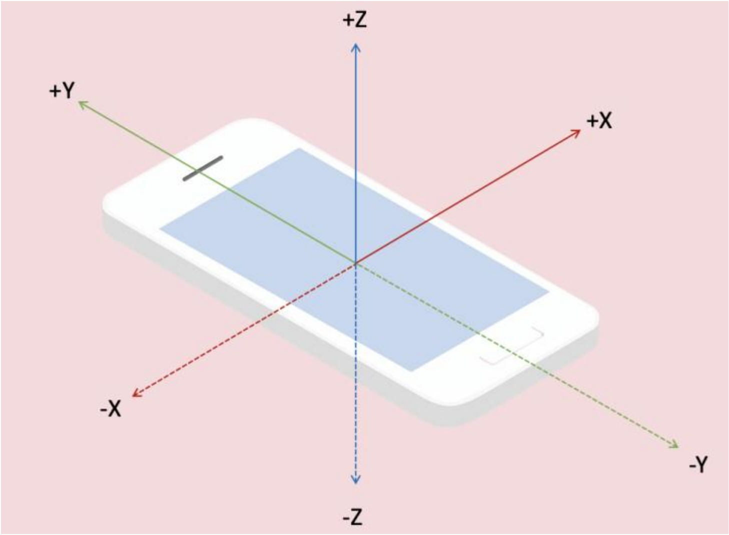

Figure 3.6: iOS Device Three-axis Coordinate System

Position and orientation in space are relative and as described above they are expressed using specific coordinate frames. Orientation tracked with IMU models relative rotation of the body/sensor frame b in the external reference coordinate frame. Specifically, to define an orientation, we need to choose the reference frame which the orientation is described against and also specify the rotation of the sensor with respect to the reference frame. There are three main mathematical constructs to represent a three-dimensional rotation between two coordinate frames: Euler angles, the quaternion, and the Direction Cosine Matrix (DCM) (Premerlani and Bizard, 2009).

3.5. Mathematical Foundations for Orientation Representation

The special orthogonal group denoted by $SO(3)$, also known as the three-dimensional rotation group (Jacobson, 2009) is the group of all rotations in $\mathbb{R}^3$. Rotating a vector in $\mathbb{R}^3$

only changes the direction of the vector while preserving its magnitude. In this section, we introduce how to represent orientation in $\mathbb{R}^3$ using rotation matrices, Euler angles, and quaternions (Diebel, 2006) (Kim and Chae, 2013) (Kok *et al.*, 2017). In addition, an example is provided to demonstrate the gimbal lock problem.

Rotation Matrix

A rotation matrix R as defined in (3.5.1) describes the rotation of an object in three-dimensional space,

$$R = [r_1 \quad r_2 \quad r_3] \in SO(3) \tag{3.5.1}$$

where $R \in \mathbb{R}^{3\times3}$, $\|r_1\| = \|r_2\| = \|r_3\| = 1$, and ${r_1}^T r_2 = {r_1}^T r_3 = {r_2}^T r_3 = 0$. That is, all the column vectors have unit size and are orthogonal to each other.

Multiplying the rotation matrix by a vector rotates the vector along one of the three orthogonal axes. When performing coordinate transformations, pre-multiplying the rotation matrix R by a vector $p \in \mathbb{R}^3$ expressed in the reference coordinate frame yields the same vector $p' \in \mathbb{R}^3$ expressed in the body-fixed coordinate frame. The equations of coordinate transformations are given by

$$p' = Rp \tag{3.5.2}$$

$$p = R^T p' \tag{3.5.3}$$

Euler Angles

By using Euler angles, any rotations can be represented by three composed elemental rotations in sequence, which are the rotations about each of the axes of a coordinate system. The three Euler angles are roll angle denoted as ϕ, pitch angle denoted as θ, and yaw angle denoted as ψ. As shown in Figure 3.7, the yaw angle represents a rotation about the z-axis by an angle ψ, the pitch angle represents a rotation about the y-axis by an angle θ, and the roll angle represents a rotation about the x-axis by an angle ϕ.

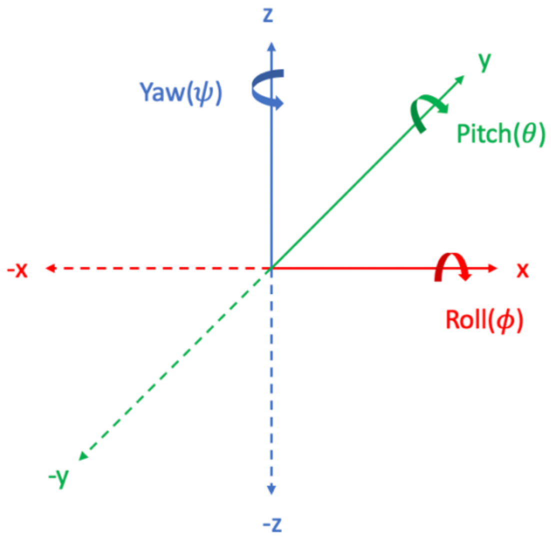

Figure 3.7: Roll ϕ, Pitch θ, and Yaw ψ Angles

A coordinate rotation, also referred to as elemental rotation, is a rotation about a single coordinate axis. The measurement data is transformed between the body/sensor frame and world/inertial frame by multiplying a sequence of coordinate rotation matrices.

The complete rotation matrix (Murray *et al.*, 1994) from the world/inertial frame to the body/sensor frame is given by

$$R_w^b(\phi,\theta,\psi) = R_{n2}^b(\phi)R_{n1}^{n2}(\theta)R_w^{n1}(\psi) \tag{3.5.4}$$

where b, w denote the body frame and the world frame respectively, and $n1$, $n2$ denote the transitional frames.

Similarly, the complete rotation matrix (Murray *et al.*, 1994) from the body/sensor frame to the world/inertial frame is given by

$$R_b^w(\phi,\theta,\psi) = R_w^{n1}(-\psi)R_{n1}^{n2}(-\theta)R_{n2}^b(-\phi) \tag{3.5.5}$$

With Euler angles, the individual coordinate rotations about each of the column unit vectors are defined as follows. $R_x(\phi)$ is the rotation purely about the x-axis, $R_y(\theta)$ is the rotation purely about the y-axis, and $R_z(\psi)$ is the rotation purely about the z-axis. These coordinate rotations are all represented by rotation matrices, therefore $R_x(\phi), R_y(\theta)$, $R_z(\psi) \in SO(3)$. The row and column vectors associated with the rotating primary axis stay unchanged. $R_x(\phi)$, $R_y(\theta)$, and $R_z(\psi)$ can be written as (Murray *et al.*, 1994)

$$R_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \tag{3.5.6}$$

$$R_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{3.5.7}$$

$$R_z(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.5.8}$$

Figure 3.8 illustrates the three elementary coordinate rotations defined above.

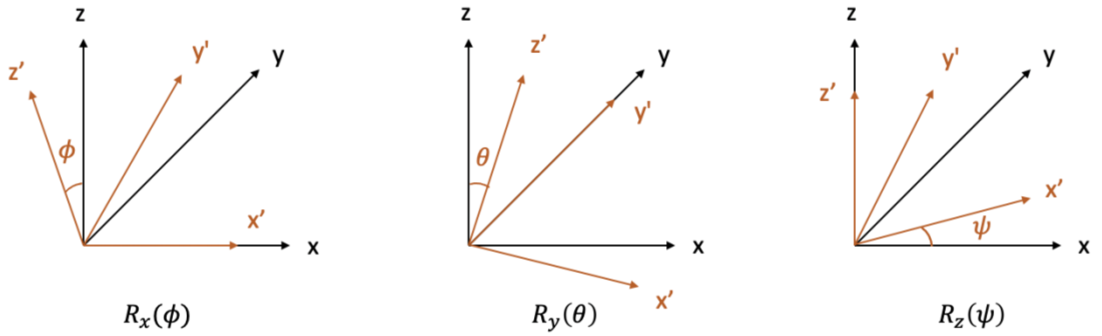

Figure 3.8: Coordinate Rotations

As matrices are not commutative, a different sequence of rotations generates a different final orientation. That is, $R_x(\phi)R_y(\theta)\ R_z(\psi) \neq R_z(\psi)R_y(\theta)R_x(\phi)$. Since the same axis cannot be in succession, there are 12 combinations of elementary rotations in total, which are xyx, xyz, xzx, xzy, yxy, yxz, yzx, yzy, zxy, zxz, zyx, and zyz.

Formed by multiplying a sequence of elemental rotation matrices, suppose the complete rotation matrix $R_w^b(\phi, \theta, \psi) \in \mathbb{R}^{3\times3}$ is given by

$$R_w^b(\phi, \theta, \psi) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{3.5.9}$$

then the roll ϕ, pitch θ, and yaw ψ angles can be calculated using the following equations (Diebel, 2006).

$$\phi = atan2(r_{32}, r_{33}) \tag{3.5.10}$$

$$\psi = atan2(r_{21}, r_{11}) \tag{3.5.11}$$

$$\theta = \begin{cases} atan2\left(-r_{31}, \frac{r_{21}}{\sin(\psi)}\right) & if \cos{(\psi)} = 0 \\ atan2\left(-r_{31}, \frac{r_{11}}{\cos(\psi)}\right) & otherwise \end{cases} \tag{3.5.12}$$

As influenced by the order of rotation operations, Euler angles can experience the phenomenon of gimbal lock in trajectory simulations.

Gimbal Lock

Gimbal lock (Hanson, 2006) is a phenomenon that occurs when the axes of two of the three gimbals become parallel and the system is forced to be locked into rotations in a two-dimensional space. In other words, in a three-dimensional space, gimbal lock results in the system losing one degree of freedom. When the gimbal lock occurs, there is no unique solution to extract all three angles, roll ϕ, pitch θ, and yaw ψ, from the formed rotation matrix. Here we provide an example of the gimbal lock singularity phenomenon. We consider the case when we rotate about the y-axis by 90 degrees. That is, the pitch angle θ is $\frac{\pi}{2}$, and the complete rotation matrix is calculated as follows.

$$R_x(\phi)R_y(\theta)R_z(\psi) = R_x(\phi)R_y\left(\frac{\pi}{2}\right)R_z(\psi) \tag{3.5.13}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} \cos\left(\frac{\pi}{2}\right) & 0 & -\sin\left(\frac{\pi}{2}\right) \\ 0 & 1 & 0 \\ \sin\left(\frac{\pi}{2}\right) & 0 & \cos\left(\frac{\pi}{2}\right) \end{bmatrix} \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix} \begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & -1 \\ \sin(\phi)\cos(\psi) - \cos(\phi)\sin(\psi) & \sin(\phi)\sin(\psi) + \cos(\phi)\cos(\psi) & 0 \\ \cos(\phi)\cos(\psi) + \sin(\phi)\sin(\psi) & \cos(\phi)\sin(\psi) - \sin(\phi)\cos(\psi) & 0 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & -1 \\ \sin(\phi-\psi) & \cos(\phi-\psi) & 0 \\ \cos(\phi-\psi) & -\sin(\phi-\psi) & 0 \end{bmatrix}$$

The result shows that there is no unique pair of solution for the roll angle and the yaw angle. Hence the orientation of the system cannot be uniquely represented when the pitch angle approaches 90 degrees. To generalize the gimbal lock problem, all Euler angle sequences that contain all three individual axes of rotation have singularities when pitch angle $\theta = \frac{\pi}{2} + n\pi$.

Quaternions

Using quaternions (Hanson, 2006) to represent the orientation solves the gimbal lock problem. Quaternions are the four-dimensional extensions of complex numbers and are used in mathematics, computer graphics, and computer vision applications. In the proposed MARG-based indoor positioning system, the quaternion representation of orientation (Murray *et al.*, 1994) is adopted as part of the system state. The relative orientation obtained by integrating the angular velocity is represented using a quaternion vector q. The column vector notation of quaternion q is given by (Graf, 2008)

$$q = [q_0 \quad q_1 \quad q_2 \quad q_3]^T \tag{3.5.14}$$

A quaternion is generally represented in the form of a set of four elements, where a, b, c, and d are real numbers, and i, j, and k are imaginary units. The imaginary dimensions are perpendicular to the real number line and perpendicular to each other. A quaternion q may be written as (Graf, 2008)

$$q = q_0 + q_1 i + q_2 j + q_3 k \tag{3.5.15}$$

The basic multiplication rules of imaginary units i, j, and k (Graf, 2008) are

$$i^2 = j^2 = k^2 = -1 \tag{3.5.16}$$

$$ij = -ji = k \tag{3.5.17}$$

$$ki = -ik = j \tag{3.5.18}$$

$$jk = -kj = i \tag{3.5.19}$$

As shown in Equation (3.5.20), the quaternion q can also be written as a pair of a scalar q_0 and a vector $\vec{q} = [q_1 \quad q_2 \quad q_3]^T$ (Graf, 2008).

$$q = (q_0, \overrightarrow{q}) \tag{3.5.20}$$

Quaternion multiplication (Diebel, 2006) is used to perform a rotation operation in a three-dimensional space. Suppose an original point, denoted as vector $p = [x_1 \quad y_1 \quad z_1]^T \in \mathbb{R}^3$, is rotated by the rotation quaternion q to a new point denoted as vector $p' \in \mathbb{R}^3$, the rotation operation is expressed by

$$p' = q \cdot p \cdot q^{-1} \tag{3.5.21}$$

$$= (w_0 + x_0 i + y_0 j + z_0 k)(x_1 i + y_1 j + z_1 k)(w_0 - x_0 i - y_0 j - z_0 k)$$

where · is quaternion multiplication.

The rotation quaternion q is constructed by the axis of rotation $\vec{q} = [q_1 \quad q_2 \quad q_3]^T \in \mathbb{R}^3$ a unit vector and the angle of rotation α around the rotation axis $\vec{q}$. By this definition, we have

$$q = \cos\left(\frac{\alpha}{2}\right) + \sin\left(\frac{\alpha}{2}\right)\vec{q} \tag{3.5.22}$$

Unit quaternions (Diebel, 2006) are used in the calculations of rotation and orientation. A unit quaternion is a quaternion of norm one. Dividing a quaternion q by its norm $\|q\|$ yields a unit quaternion U_q. The equations of calculating the quaternion norm and unit quaternion are

$$\|q\,\| = \sqrt{{q_0}^2 + {q_1}^2 + {q_2}^2 + {q_3}^2} \tag{3.5.23}$$

$$U_q = \frac{q}{\|q\|} \tag{3.5.24}$$

As two different representations of spatial rotations in a three-dimensional space, Euler angles and unit quaternions are interchangeable (Diebel, 2006). To convert the rotation representation from the quaternion to Euler angles in terms of using the xyz sequence of Euler angles, the roll ϕ, pitch θ, and yaw ψ angles are given by

$$\phi = atan2(2q_0q_1 + 2q_2q_3, {q_0}^2 - {q_1}^2 - {q_2}^2 + {q_3}^2) \tag{3.5.25}$$

$$\theta = -\mathrm{asin}\,(2q_1q_3 - 2q_0q_2) \tag{3.5.26}$$

$$\psi = atan2(2q_0q_3 + 2q_1q_2, {q_0}^2 + {q_1}^2 - {q_2}^2 - {q_3}^2) \tag{3.5.27}$$

The inverse conversion from the unit quaternion to Euler angles can be calculated using

$$q_{xyz}(\phi,\theta,\psi)=\begin{bmatrix}\cos\frac{\phi}{2}\cos\frac{\theta}{2}\cos\frac{\psi}{2}+\sin\frac{\phi}{2}\sin\frac{\theta}{2}\sin\frac{\psi}{2}\\ \cos\frac{\theta}{2}\cos\frac{\psi}{2}\sin\frac{\phi}{2}-\cos\frac{\phi}{2}\sin\frac{\theta}{2}\sin\frac{\psi}{2}\\ \cos\frac{\phi}{2}\cos\frac{\psi}{2}\sin\frac{\theta}{2}+\sin\frac{\phi}{2}\cos\frac{\theta}{2}\sin\frac{\psi}{2}\\ \cos\frac{\phi}{2}\cos\frac{\theta}{2}\sin\frac{\psi}{2}-\sin\frac{\phi}{2}\cos\frac{\psi}{2}\sin\frac{\theta}{2}\end{bmatrix} \tag{3.5.28}$$

The pedestrian's orientation at the current system state described by a unit quaternion q is obtained by taking the integral of the first derivative of quaternion with respect to time, which is given by

$$q=\int\dot{q}\ dt \tag{3.5.29}$$

where the first derivative of the unit quaternion $\dot{q}$, also known as quaternion rates, is the rate of change of the orientation and can be calculated from the angular velocity and the previous estimated unit quaternion $\hat{q}$ (Graf, 2008). We denote the angular velocity in the world/inertial frame as $\vec{w}\in\mathbb{R}^3$, and the angular velocity in the body/sensor frame as $\overrightarrow{w'}\in\mathbb{R}^3$. The equations to calculate $\dot{q}$ can therefore be written as

$$\dot{q_w}=\frac{1}{2}\begin{bmatrix}0\\ \vec{w}\end{bmatrix}\hat{q} \tag{3.5.30}$$

$$\dot{q_{w'}}=\frac{1}{2}\hat{q}\begin{bmatrix}0\\ \overrightarrow{w'}\end{bmatrix} \tag{3.5.31}$$

3.6. Magnetic Distortion and Magnetometer Calibration

As one of the embedded inertial sensors, the magnetometer is used to detect heading and orientation by measuring the Earth's magnetic field. It is a useful measurement data source to compensate the gyro drift because errors do not accumulate in magnetometer data. However, the compact tri-axis magnetometer is subject to internal errors and external magnetic and electric disturbances. When implementing the system, the magnetic field measured is a combination of both Earth's magnetic field and magnetic fields created by nearby magnetic disturbances. Misalignment of the sensor also exists in the integrated MEMS magnetometer. Since accurate magnetometer measurements are essential components of the sensor fusion algorithm, magnetometer calibration is necessary to be performed during the initialization and calibration phase of the proposed indoor positioning system.

All magnetic interferences can be divided into two categories: hard-iron interferences and soft-iron interferences (Silhavy *et al.*, 2017). The hard-iron interferences are created by objects that generate their own magnetic field, while the soft-iron interferences deflect and bend the magnetic field when the magnetic object passes through the existing magnetic field. In an ideal environment without hard-iron or soft-iron interferences, the magnetometer measurements form a sphere centered at the origin. Hard-iron interferences create a permanent bias (Fang *et al.*, 2011), which pulls the center of the sphere away from the origin to the coordinate that equals the hard-iron offset. Soft-iron distortions stretch the magnetometer output to an ellipsoid depending on the relative direction of the interference. Generally, the hard-iron effect has a larger contribution to the magnetic distortions than the soft-iron effect.

The hard-and-soft-iron model using least squares fitting (Ozyagcilar, 2015) is adopted to mitigate magnetic interferences. After all the parameters of the fitted mathematical model are determined, the contaminated magnetometer measurements from the surface of the ellipsoid can be transformed back to the surface of a sphere centered at the origin. The hard-and-soft-iron model is established with three parameters: the geomagnetic field strength scaler B, the hard-iron vector M_{hard}, and the soft-iron matrix M_{soft}. In the following section, the magnetometer calibration model and its least squares solution are elaborated.

The Magnetometer Calibration Model

According to (Ozyagcilar, 2013), any orientation of the smartphone can be modeled as resulting from three elemental rotations in sequence applied to a starting orientation with the smartphone facing up. Suppose the magnetometer reading of the starting orientation is B_0, the geomagnetic field strength is B, and the inclination angle is α. Magnetic inclination is the angle that the geomagnetic field is tilted with respect to the surface of the earth. Then, we have

$$B_0 = B \begin{bmatrix} \cos\alpha \\ 0 \\ \sin\alpha \end{bmatrix} \tag{3.6.1}$$

The hard-iron offset is defined as a vector $M_{hard} \in \mathbb{R}^3$, and the soft-iron interference is defined as a matrix $M_{soft} \in \mathbb{R}^{3\times3}$. Then, the magnetometer reading B_r measured after three arbitrary rotations in yaw ψ, pitch θ and roll ϕ by the rotation matrices $R_z(\psi)$, $R_y(\theta)$, and $R_x(\phi)$ can be written as

$$B_r = M_{soft}R_x(\phi)R_y(\theta)R_z(\psi)B\begin{bmatrix}\cos\alpha\\0\\\sin\alpha\end{bmatrix} + M_{hard} \tag{3.6.2}$$

To obtain a unique solution of the hard-iron vector M_{hard} and the soft-iron matrix M_{soft}, a symmetric constraint needs to be imposed onto the inverse soft-iron matrix ${M_{soft}}^{-1}$. For most cases when the hard-iron offset dominates the interference, the soft-iron matrix M_{soft} is assumed to be the identity matrix $I \in \mathbb{R}^{3\times3}$. Hence, only the hard-iron vector M_{hard} and the geomagnetic field strength *B* need to be determined. If we move M_{hard} and M_{soft} to the left side of the above rotation equation, we have

$${M_{soft}}^{-1}(B_r - M_{hard}) = R_x(\phi)R_y(\theta)R_z(\psi)B\begin{bmatrix}\cos\alpha\\0\\\sin\alpha\end{bmatrix} \tag{3.6.3}$$

Then we square both sides of the equation, the rotation matrices and the inclination angle are cancelled out after expanding the transpose of the right side of the equation. The calculation process is shown as

$$\left(M_{soft}{}^{-1}(B_r - M_{hard})\right)^T M_{soft}{}^{-1}(B_r - M_{hard}) = \left(R_x(\phi)R_y(\theta)R_z(\psi)B\begin{bmatrix}\cos\alpha\\0\\\sin\alpha\end{bmatrix}\right)^T R_x(\phi)R_y(\theta)R_z(\psi)B\begin{bmatrix}\cos\alpha\\0\\\sin\alpha\end{bmatrix}$$

(3.6.4)

With the soft-iron matrix assumption, the equation can be simplified to

$$(B_r - M_{hard})^T(B_r - M_{hard}) = B^2 \quad (3.6.5)$$

$$B_r{}^T B_r - 2B_r{}^T M_{hard} + M_{hard}{}^T M_{hard} - B^2 = 0 \quad (3.6.6)$$

We define the components of the hard-iron vector M_{hard} and the components of the magnetometer measurement B_r as $M_{hard} = [M_{hard-x} \quad M_{hard-y} \quad M_{hard-z}]^T$, $B_r = [B_{rx} \quad B_{ry} \quad B_{rz}]^T$ respectively. In practice, there is always a residual $r(i)$ for the ith measurement, which can be expressed as

$$\begin{aligned} r(i) = {} & B_{rx}(i)^2 + B_{ry}(i)^2 + B_{rz}(i)^2 \\ & -2B_{rx}(i)M_{hard-x} - 2B_{ry}(i)M_{hard-y} - 2B_{rz}(i)M_{hard-z} \\ & +M_{hard-x}{}^2 + M_{hard-y}{}^2 + M_{hard-z}{}^2 - B^2 \end{aligned} \quad (3.6.7)$$

The above equation can also be written in its matrix form as

$$r(i) = \left(B_{rx}(i)^2 + B_{ry}(i)^2 + B_{rz}(i)^2\right) - \begin{bmatrix}B_{rx}(i)\\B_{ry}(i)\\B_{rz}(i)\\1\end{bmatrix}^T \begin{bmatrix}2M_{hard-x}\\2M_{hard-y}\\2M_{hard-z}\\B^2 - M_{hard-x}{}^2 - M_{hard-y}{}^2 - M_{hard-z}{}^2\end{bmatrix}$$

(3.6.8)

Now, we expand the equation with N number of measurements used to perform the magnetometer calibration. We have

$$\begin{bmatrix} r(1) \\ r(2) \\ \cdots \\ r(N) \end{bmatrix} = \begin{bmatrix} B_{rx}(1)^2 + B_{ry}(1)^2 + B_{rz}(1)^2 \\ B_{rx}(2)^2 + B_{ry}(2)^2 + B_{rz}(2)^2 \\ \cdots \\ B_{rx}(N)^2 + B_{ry}(N)^2 + B_{rz}(N)^2 \end{bmatrix}$$

$$- \begin{bmatrix} B_{rx}(1) & B_{ry}(1) & B_{rz}(1) & 1 \\ B_{rx}(2) & B_{ry}(2) & B_{rz}(2) & 1 \\ \cdots & \cdots & \cdots & 1 \\ B_{rx}(N) & B_{ry}(N) & B_{rz}(N) & 1 \end{bmatrix} \begin{bmatrix} 2M_{hard-x} \\ 2M_{hard-y} \\ 2M_{hard-z} \\ B^2 - {M_{hard-x}}^2 - {M_{hard-y}}^2 - {M_{hard-z}}^2 \end{bmatrix} \tag{3.6.9}$$

We define

$$r' = \begin{bmatrix} r(1) \\ r(2) \\ \cdots \\ r(N) \end{bmatrix} \in \mathbb{R}^{N\times 1} \tag{3.6.10}$$

$$b' = \begin{bmatrix} B_{rx}(1)^2 + B_{ry}(1)^2 + B_{rz}(1)^2 \\ B_{rx}(2)^2 + B_{ry}(2)^2 + B_{rz}(2)^2 \\ \cdots \\ B_{rx}(N)^2 + B_{ry}(N)^2 + B_{rz}(N)^2 \end{bmatrix} \in \mathbb{R}^{N\times 1} \tag{3.6.11}$$

$$A' = \begin{bmatrix} B_{rx}(1) & B_{ry}(1) & B_{rz}(1) & 1 \\ B_{rx}(2) & B_{ry}(2) & B_{rz}(2) & 1 \\ \cdots & \cdots & \cdots & 1 \\ B_{rx}(N) & B_{ry}(N) & B_{rz}(N) & 1 \end{bmatrix} \in \mathbb{R}^{N\times 4} \tag{3.6.12}$$

$$X = \begin{bmatrix} 2M_{hard-x} \\ 2M_{hard-y} \\ 2M_{hard-z} \\ B^2 - {M_{hard-x}}^2 - {M_{hard-y}}^2 - {M_{hard-z}}^2 \end{bmatrix} \in \mathbb{R}^{4\times 1} \tag{3.6.13}$$

The equation can be further expressed in a simple form as

$$r' = b' - A'X \tag{3.6.14}$$

From the above equation, we can conclude that the objective of the magnetometer calibration problem is to fit the model parameters M_{hard} and B using N number of magnetometer measurements so that the residual r' is minimized in a least squares sense. Thus, the magnetometer calibration problem can be expressed in the form of an optimization problem as shown in Equation (3.6.15).

$$X_{LS} = \underset{X}{\operatorname{argmin}} \|b' - A'X\|_2^2 \tag{3.6.15}$$

The Least Squares Solution

The least squares solution of the magnetometer calibration problem based on the hard-and-soft-iron model is calculated by

$$X = ({A'}^T A')^{-1} {A'}^T b' \tag{3.6.16}$$

Since both the matrix A' and the vector b' can be obtained from the magnetometer measurements vector B_r, $X = [X_1 \quad X_2 \quad X_3 \quad X_4]^T$ can be determined through the least squares equation. Once vector X is determined, M_{hard} and B are given by

$$M_{hard} = \begin{bmatrix} M_{hard-x} \\ M_{hard-y} \\ M_{hard-z} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} X_1 \\ X_2 \\ X_3 \end{bmatrix} \tag{3.6.17}$$

$$B = \sqrt{X_4 + {M_{hard-x}}^2 + {M_{hard-y}}^2 + {M_{hard-z}}^2} \tag{3.6.18}$$

Figure 3.9 shows the raw magnetometer data with both hard-iron and soft-iron interferences and the corrected data by using the calibration method discussed above. Figure 3.10 is the comparison of the results before and after the calibration process for the case when hard-iron interference dominates. Possible sources of the hard and soft-iron distortions are electromagnetic components on the smartphone Printed Circuit Board (PCB). Also, the magnetometer is very sensitive to the surrounding magnetic fields. Any local magnetic interferences caused by ferromagnetic materials, magnetic objects, fields generated by current flows, motors, batteries will corrupt the output of the smartphone electronic magnetometer.

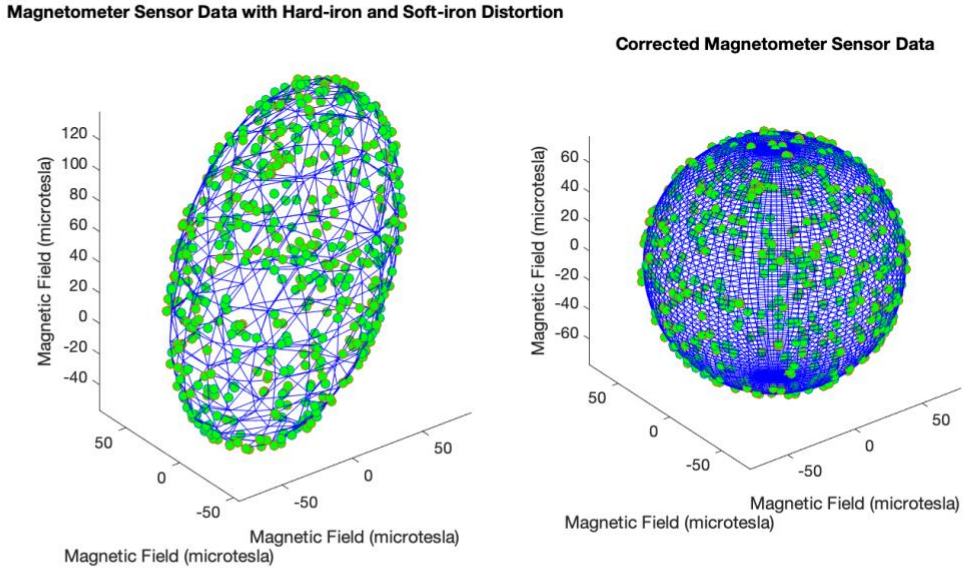

Figure 3.9: Magnetometer Calibration for Hard-iron and Soft-iron Distortion

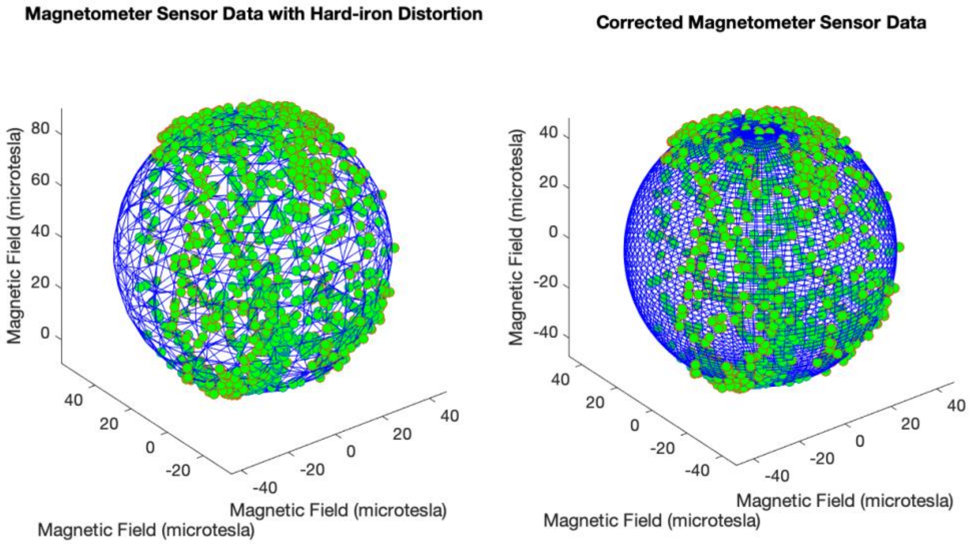

Figure 3.10: Magnetometer Calibration When Hard-iron Distortion Dominates

3.7. Mode Detection

After the calibration and initialization phase, the system enters the mode detection module, which identifies if the system is in the static or dynamic mode. In the dynamic mode, the system is undergoing dynamic motions and the accelerometer is influenced by both continuous linear accelerations and the gravitational acceleration instead of only the gravitational acceleration. In the static mode, the pedestrian is perceived as standing still and only changing moving directions.

Whether the system is in the static or dynamic mode affects the corresponding step detection and heading and orientation estimation. For the step detection, a valid step detection process is activated after the system enters the dynamic mode. If the system is considered static, only orientation of the system needs to be updated. For the heading and orientation estimation, the static mode allows the system to use reference vectors to correct the preliminary estimated heading calculated by angular velocity integration, while in the dynamic mode, a double phase extended Kalman filter is adopted so that more reliable orientation estimation results are produced when the accelerometer measures both gravitational acceleration and other accelerations. This dual-mode design reinforces the strength of each heading and orientation estimation algorithm in its corresponding mode and improves the overall computational efficiency of the entire system.

To detect the start of the dynamic mode, we calculate the sample variance $S^2$of the magnitude a_i of accelerometer readings A_r over a time window $w(n)$ of length L. Given N as the total number of measurements, there are $N - L + 1$ continuous time windows. The ith accelerometer measurement is denoted by

$$A_r\big(t(i)\big) = \big[A_{rx}\big(t(i)\big) \quad A_{ry}\big(t(i)\big) \quad A_{rz}\big(t(i)\big)\big]^T, \quad i = 1, 2, \dots, N \tag{3.7.1}$$

where $t(i)$ denotes the time instance for the ith measurement. The magnitude a_i of each accelerometer reading A_r is calculated by

$$a_i = \sqrt{A_{rx}\big(t(i)\big)^2 + A_{ry}\big(t(i)\big)^2 + A_{rz}\big(t(i)\big)^2} - g, \quad i = 1, 2, \dots, N \tag{3.7.2}$$

where g is the gravitational acceleration. The mean value $\mu(w(n))$ of the L observations in a time window $w(n)$ is given by

$$\mu\big(w(n)\big) = \frac{1}{L}\sum_{i=n}^{L-1+n} a_i\,, \quad n = 1, 2, \cdots, N - L + 1 \tag{3.7.3}$$

Hence, each sample variance $S^2$of the magnitude a_i of accelerometer readings A_r over a time window $w(n)$ is calculated by

$$S^2\big(w(n)\big) = \frac{1}{L-1}\sum_{i=n}^{L-1+n} \Big(a_i - \mu\big(w(n)\big)\Big)^2, n = 1, 2, \cdots, N - L + 1 \tag{3.7.4}$$

Once the sample variance $S^2\big(w(n)\big)$ of a certain time window exceeds the threshold value ${S_d}^2$, the system enters the dynamic mode and activates the step detection module.

3.8. Step Detection

As one of the main modules of the proposed indoor positioning system, step detection is a crucial task that affects the accuracy of the ultimate positioning results. Research of both motion recognition (Jabbar *et al.*, 2020) and gait analysis (Sant'Anna and Wickström, 2010) offer insights into the design of step detection techniques. The step detection module not only analyzes gait event of each individual step but also produces step count for the distance travelled. The proposed step detection technique does not require a classifier or any pre-defined templates. When the pedestrian's moving speed is fast, the stance phase of a gait may be diminished, in which case the threshold-based detection will fail in step detection. For this reason, the step detection module is capable of switching between threshold-based detection method and peak/valley-based detection method depending on the pedestrian's real-time moving velocity and acceleration. Accelerometer is the most exploited IMU measurements for step detection algorithms. Traditional signal processing techniques (Forsman *et al.*, 2009) (Díez *et al.*, 2015) and an innovative threshold-less signal pattern matching approach (Cho and Park, 2019) have been explored. To develop a more robust step detection module, we also leverage the information extracted from gyroscope and magnetometer measurements, which further refines the preliminary step detection results. A multi-layer step validation approach is used to prevent false step counts. The structure of the proposed step detection module is provided in Figure 3.11 and described as follows.

- Signal processing and filtering is performed on raw accelerometer data to reduce and smooth out high frequency noises.
- A scenario-based self-adaptive step detection serves as the first layer of step detection. Real-time velocity and acceleration information is used to decide if the system goes into the threshold-based detection branch or the peak/valley-based detection branch. Also, preliminary adjustments and false step rejection are included.
- Gyroscope and magnetometer measurements are incorporated as step detection refinement, which serves as the second layer of step detection.
- A multi-layer step validation process enhances the performance of the step detection module.

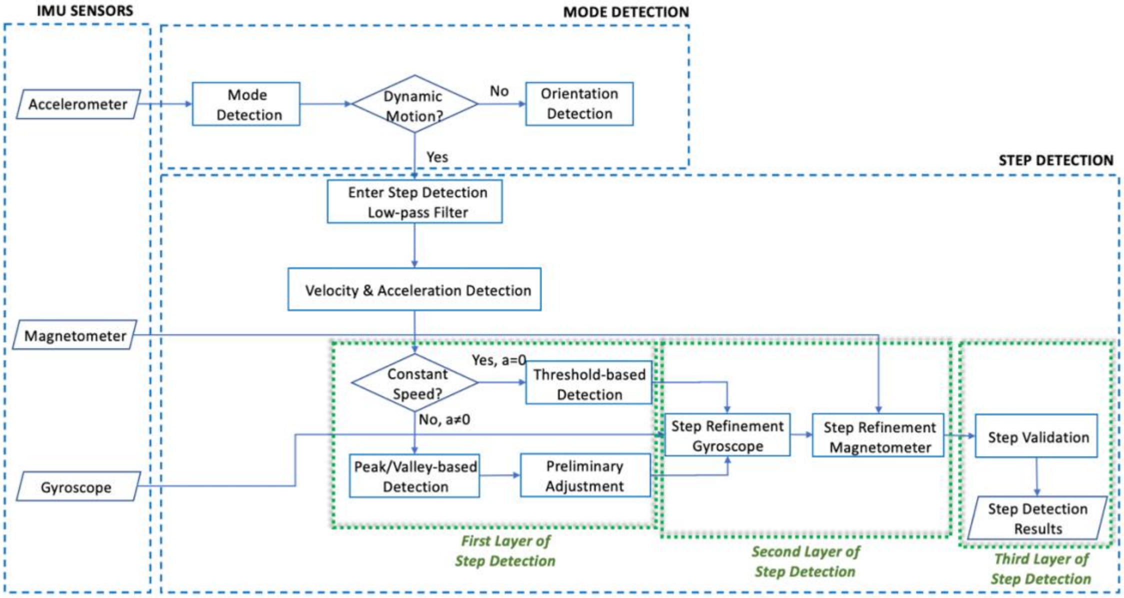
IMU SENSORS
MODE DETECTION
Accelerometer
Mode Detection
Dynamic Motion?
No
Orientation Detection
Yes
STEP DETECTION
Enter Step Detection Low-pass Filter
Velocity & Acceleration Detection
Magnetometer
Constant Speed?
Yes, a=0
Threshold-based Detection
No, a≠0
Gyroscope
Peak/Valley-based Detection
Preliminary Adjustment
Step Refinement Gyroscope
Step Refinement Magnetometer
Step Validation
Step Detection Results
First Layer of Step Detection
Second Layer of Step Detection
Third Layer of Step Detection

Figure 3.11: Proposed Step Detection Module Structure

Digital Signal Filtering

IMU sensors exhibit a variety of deterministic errors and stochastic high frequency noises. Deterministic errors can be eliminated by analyzing the digital signal and building an error model. The stochastic high frequency white noise of the accelerometer can be smoothed out through digital signal filtering techniques. As a data preprocessing method, a low-pass filter is used for removing those high frequency noises shown in the frequency domain of the accelerometer signal. This step serves as a preliminary digital signal processing step to filter out sensor noises. The cut-off frequency of the filter is actively adjusted based on the pedestrian's moving speed. Using a fixed cut-off frequency for filtering the signal may result in over-filtering or under-filtering the signal.

Scenario-based Self-adaptive Step Detection

We consider that the smartphone is handheld and placed in front of the waist area with its screen facing up, mimicking the circumstance of foot-mounted devices. The first advantage of this device setup is that change of orientation mainly affects the z-axis instead of all three axes. Another advantage is that the mid stance phase of a human gait cycle is detectable although the static period is usually assumed for foot-mounted step detection systems. Identifying the mid stance phase of a gait cycle is of importance because raw accelerometer data lead to cumulative errors as shown in Figure 3.3. Once a stance phase is detected, the corresponding velocity can be corrected as zero for the gait cycle. The acceleration norm without gravity compensation is used for detecting the steps. Figure 3.12 illustrates the signal pattern of acceleration norm when the moving speed is constant. The mid stance phase can be determined when the acceleration norm is less than a threshold δ, which can be written as

$$\left\|A_r\big(t(i)\big)\right\| < \delta, \quad i = 1, 2, \dots, N \tag{3.8.1}$$

However, the acceleration fluctuates greatly in magnitude when moving with varying speeds as shown in Figure 3.13, in which case, threshold-based detection may leave out the periods under slower speed motion. We use each peak/valley pair to represent a single step when the pedestrian is not moving with a constant speed.

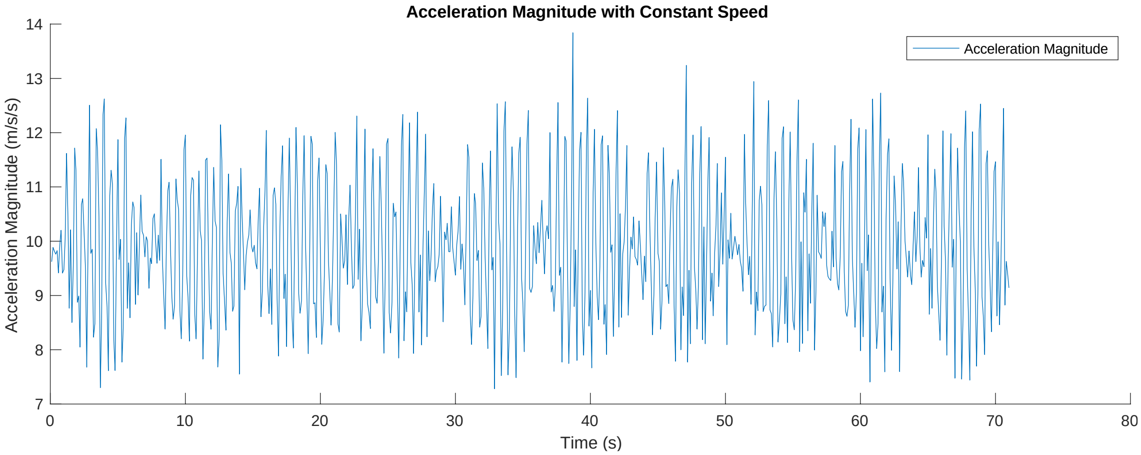

Figure 3.12: Signal Pattern under Constant Speed Motion

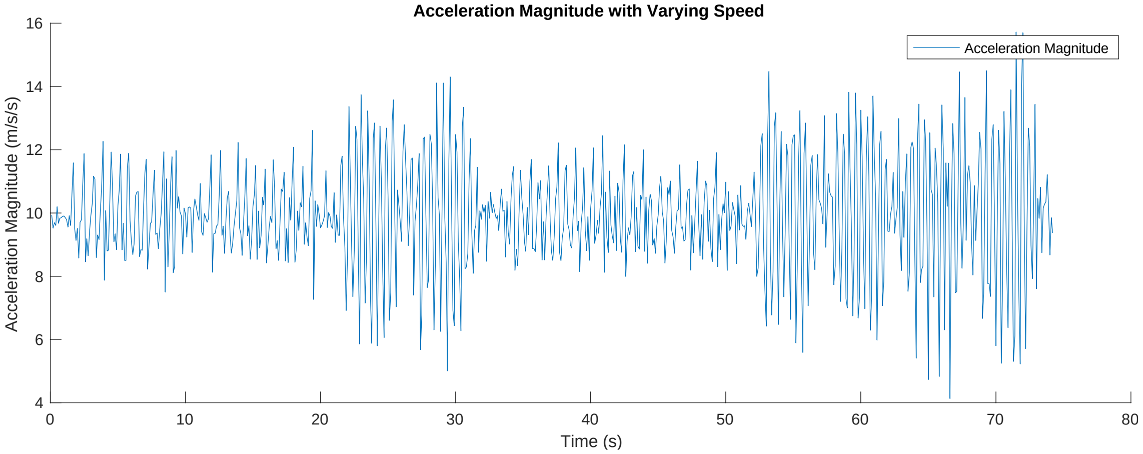

Figure 3.13: Signal Pattern under Varying Speeds Motion

For peak-based detection, a step is detected at the time instance $t(i-1)$ when $\left\|A_r\big(t(i-1)\big)\right\|$ is a local maximum. In addition to this condition, more constraints are needed to eliminate the undesired false positive peaks due to noises in the signal. The conditions combined with all constraints are summarized as follows.

Algorithm 1a Peak/Valley-based Step Detection: Peak/Valley Detection

Input: mth peak time instance $t_p(m)$, kth valley time instance $t_v(k)$, all peaks time instance set T_p, all valleys time instance set T_v, accelerometer data A_r, time instance $t(i), i = 1,2, \dots$.

```
$m = 0, k = 0$
**for all** $i = 2, \dots$ **do**
    **if** $\|A_r(t(i-1))\| > \|A_r(t(i-2))\|$ and $\|A_r(t(i-1))\| > \|A_r(t(i))\|$ **then**
        $m = m + 1$, $t_p(m) = t(i-1)$, append $t_p(m)$ to set $T_p$
    **end if**
    **if** $\|A_r(t(i-1))\| < \|A_r(t(i-2))\|$ and $\|A_r(t(i-1))\| < \|A_r(t(i))\|$ **then**
        $k = k + 1$, $t_v(k) = t(i-1)$, append $t_v(k)$ to set $T_v$
    **end if**
**end for**
```

Algorithm 1b Peak/Valley-based Step Detection: Preliminary Adjustments, Acceptance Regions, and Sudden Change in Speed and Device Motion

Input: mth peak time instance $t_p(m)$, mth valley time instance $t_v(m)$, all peaks time instance set T_p, all valleys time instance set T_v, accelerometer data A_r, minimum time interval $\Delta t_p{}'$,time instance $t(i), i = 1,2, \ldots$.

```
 1: for all $m = 3,4, \ldots$ do
 2:     if $\left\| A_r\left(t_p(m)\right) \right\| < \frac{1}{2}\left(\left\| A_r\left(t_p(m-1)\right) \right\| + \left\| A_r\left(t_p(m-2)\right) \right\|\right)$ then
 3:         if $t_p(m) - t_p(m-1) < \Delta t_p{}'$ then
 4:             Remove $t_p(m)$ from set $T_p$
 5:         end if
 6:         if $t_p(m) < t_v(m) + \frac{1}{2}\left(t_v(m) - t_p(m-1)\right)$ then
 7:             Remove $t_p(m)$ from set $T_p$
 8:         end if
 9:         if $t_p(m) > t_v(m) + \frac{4}{5}\left(t_v(m) - t_v(m-1)\right)$ then
10:             Remove $t_p(m)$ from set $T_p$
11:         end if
12:     end if
13:     if $\left\| A_r\left(t_v(m)\right) \right\| < \frac{1}{2}\left(\left\| A_r\left(t_v(m-1)\right) \right\| + \left\| A_r\left(t_v(m-2)\right) \right\|\right)$ then
14:         if $t_v(m) < t_p(m-1) + \frac{1}{2}\left(t_p(m-1) - t_v(m-1)\right)$ then
15:             Remove $t_v(k)$ from $T_v$
16:         end if
17:         if $t_v(m) > t_p(m-1) + \frac{4}{5}\left(t_p(m-1) - t_p(m-2)\right)$ then
18:             Remove $t_v(k)$ from $T_v$
19:         end if
20:     end if
21: end for
```

As shown in Algorithm 1a, a peak or valley is detected if the value at the time instance is a local maximum or a local minimum. The minimum time difference between two consecutive detected peaks is constrained by a threshold $\Delta t_p{}'$ as shown in line 3 of Algorithm 1b. The peak-based detection also makes preliminary adjustments based on peak/valley acceptance and rejection regions. Any spikes that fall into the rejection regions are removed. Lines 6-11 of Algorithm 1b define the peak rejection region. Lines 14-19 of Algorithm 1b define the valley rejection region. The starting points of the acceptance regions are based on the previous detected peak and valley. Specifically, the peak-based step detection algorithm will start to search for the next potential peak/valley only after a time period that equals half of the time interval between the most recently detected peak and valley. Figure 3.14 illustrates the acceptance and rejection regions for peaks and valleys. The magnitude difference of the acceleration norm signal represents change in moving speed. A peak detected in the rejection region with sufficient magnitude may indicate sudden change in speed, so it is converted back to a valid peak value (the outer if statements of Algorithm 1b). Figure 3.15 shows the detected peaks based on the conditions and constraints of the peak-based step detection method.

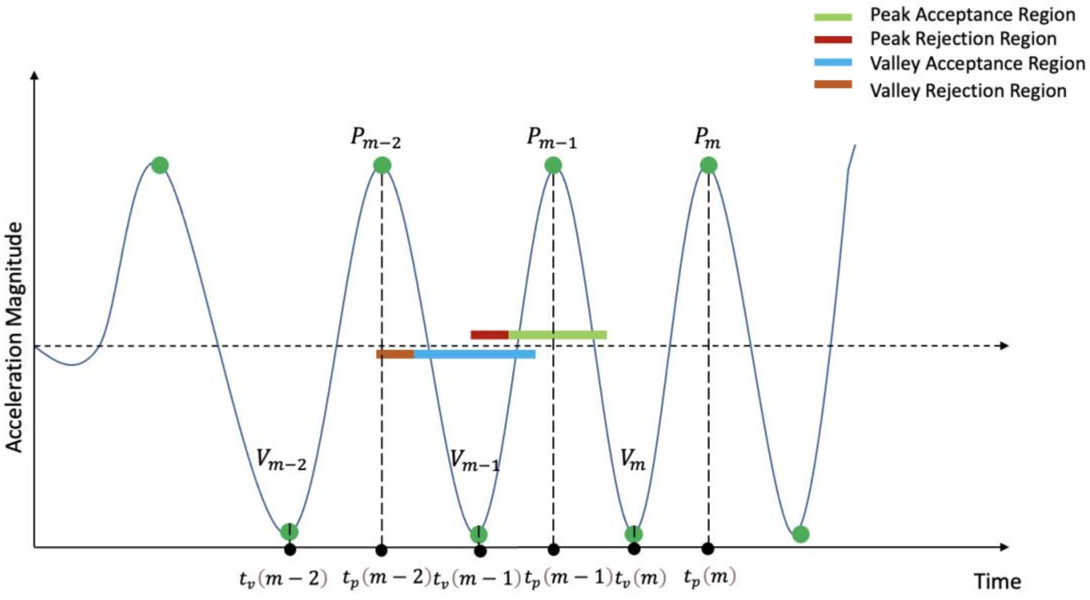

Figure 3.14: Acceptance and Rejection Regions of Peak-based Approach

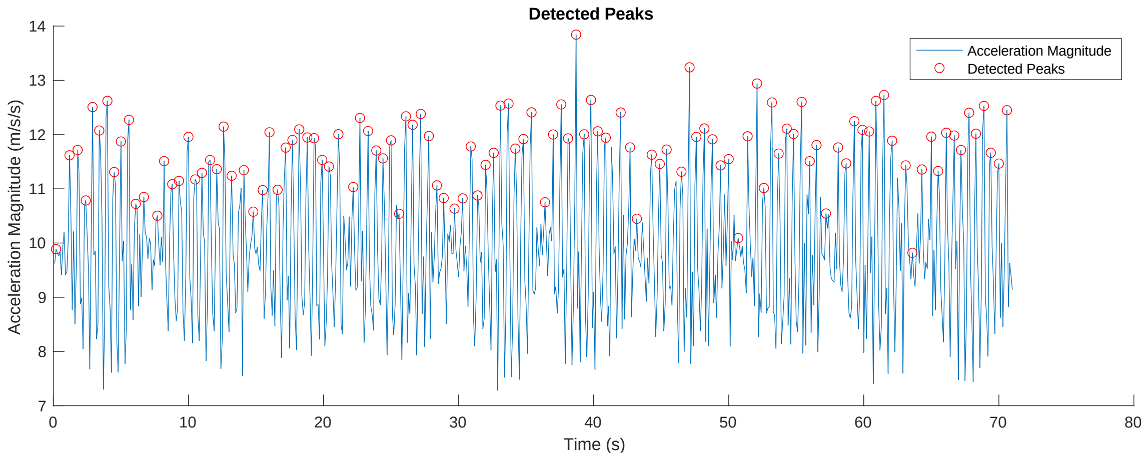

Figure 3.15: Step Detection Result of Peak-based Approach

Step Detection Refinement

As the second layer of step detection of the proposed positioning system, step detection refinement extracts useful information from both measured angular velocity and magnetic field to further optimize the step detection results. These two separate data sources are used to not only verify the detected steps generated by the first layer based on acceleration norm but also eliminate some undesired peaks.

The angular velocity measured by the gyroscope sensor demonstrates repetitive signal patterns in z-axis under certain types of motions. As shown in Figure 3.16, through calculating the variance of each axis of the signal, the z-axis is detected as the dominant axis influenced by rotational motions. If the pattern with high angular velocity occurs in the signal periodically, the gyroscope signal data in the z-axis will be compared and analyzed with the acceleration norm. Figure 3.17 overlies the angular velocity along the z-axis on top of the parallel shifted acceleration norm. The high angular velocity moments represent a body rotation motion, while the normal stable angular velocity periods are considered as motions of moving in a straight line at a speed determined by the acceleration. Any steps that are detected by either the threshold-based method or the peak-based method and also fall into the momentary body rotation periods are eliminated, because of the abnormally high angular rates.

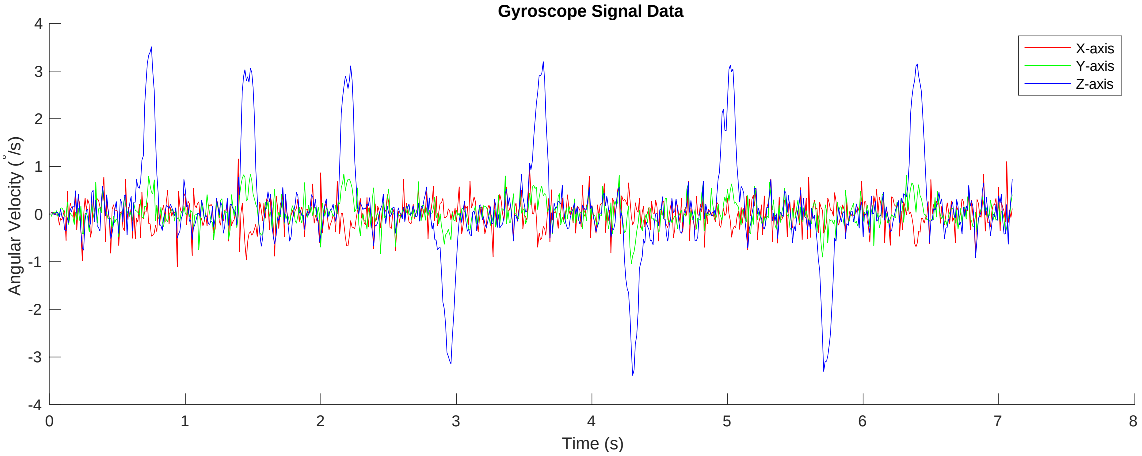

Figure 3.16: Tri-axis Gyroscope Angular Velocity Signal Data

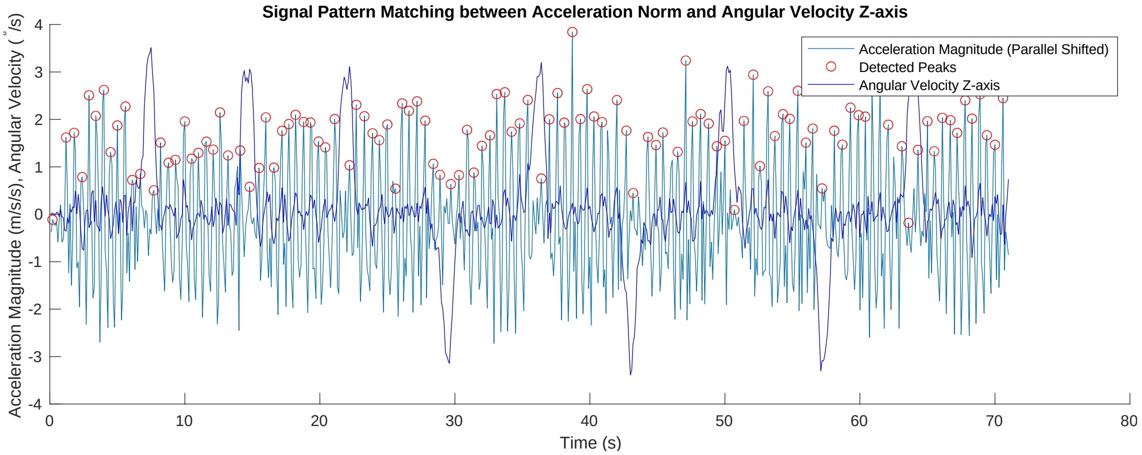

Figure 3.17: Signal Pattern Matching between

Acceleration Norm and Angular Velocity along the *z*-axis

The magnetic field intensity should be fixed if the pedestrian stays in the same position with the same orientation in the indoor environment. Any abrupt changes in measured magnetic field intensity are the results of either surrounding magnetic interferences or change of device orientation. Similar to the gyroscope signals, magnetometer signals also demonstrate periodical signal patterns. The difference between the two is that the repetitive pattern in the magnetometer signal shows in more than one axis. The gyroscope signal is affected by certain rotation motions with respect to one axis, while all three orthogonal components of the magnetometer signal are influenced by the specific position and orientation of the sensor. As shown in Figure 3.18, the periods experiencing significant magnetic intensity changes (the x-axis and y-axis magnetic field components are plotted) coincide with those high angular velocity moments, which further validates the body rotation motions detected by the gyroscope. When there is no active magnetic interference, the sudden changes of the magnetic intensity are solely caused by changes in the orientation of the sensor frame as expected. Magnetic perturbations are detected when a sudden change in magnetic intensity occurs with a stable near zero angular velocity. The magnetometer should be recalibrated to remove the magnetic distortion if any strong magnetic interference is detected. Moreover, repeating motions can be recognized by identifying highly repetitive magnetic field signal patterns.

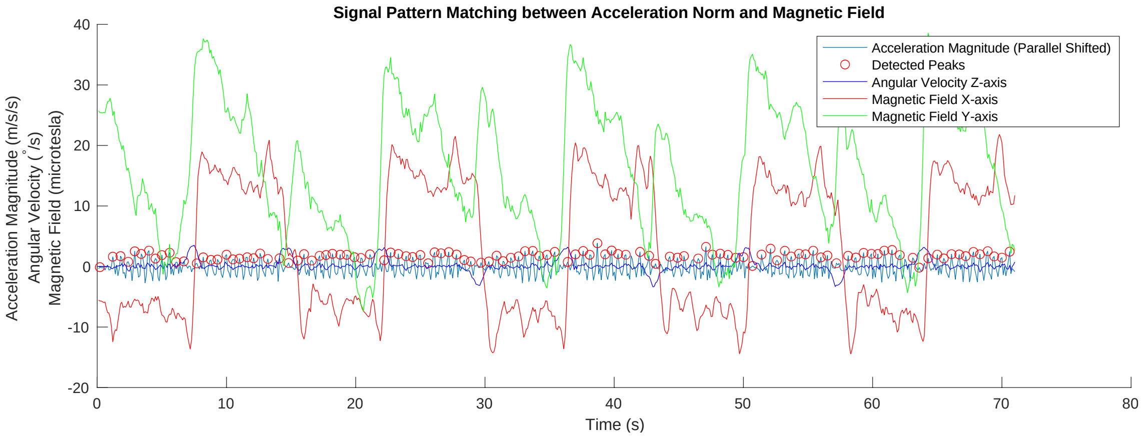

Figure 3.18: Signal Pattern Matching

between Acceleration Norm and Magnetic Field

Step Validation

After two layers of step detection, a final layer of step validation is implemented as a validity check of the step detection results. The step validation includes a velocity validity check and a distance validity check. First, we take the integral of the acceleration signal combined with the detected step sequence pattern with respect to time. It is considered valid as long as the calculated velocity is under a threshold value Vel_{normal} for walking or $Vel_{running}$ for running. Second, a distance validity check is performed by calculating the displacement through taking the integral of the velocity. Similar to the velocity validity check, the step is considered a true step if the calculated displacement is under a threshold value d_{normal} for walking or $d_{running}$ for running.

The proposed step detection module relies on the three layers to detect steps using MARG sensor data stream and verify the estimated step sequence pattern.

3.9. Dead Reckoning-based Heading and Orientation Estimation

Position is estimated by combing the detected steps and the heading and orientation estimation results. Heading estimation is the process of determining the direction of the pedestrian's movement. As discussed in the step detection section, the smartphone is handheld and placed in front of the waist area with its screen facing up and the relative position of the smartphone is fixed to the pedestrian's body, so coordinate transformations between the body and the sensor frame are unnecessary. For both the static mode and the dynamic mode, the heading and orientation estimation module first generates a preliminary estimated gyroscope-based heading calculated by angular velocity integration. Adopting the reference vectors idea proposed in (Madgwick *et al.*, 2020), then the system uses reference vectors obtained from accelerometer and magnetometer sensors to correct the preliminary estimated heading if it is in the static mode. Sensor fusion corrections are necessary because gyroscope measurements drift away over time, which leads to errors in the calculated heading and orientation results. If the system is in the dynamic mode, a double phase extended Kalman filter is adopted to pull back the gyroscope-based heading estimation and produce a more accurate orientation estimation, because the accelerometer under the dynamic mode measures both the gravitational acceleration and external accelerations. Since a gyroscope provides relative heading, the initial orientation is calculated using accelerometer and magnetometer measurements. This dual-mode design not only reinforces the strength of each heading and orientation estimation algorithm in its more suitable mode but also increases the overall computational efficiency.

Initial Orientation Calculation

The initial orientation can be calculated based on accelerometer and magnetometer measurements. Any orientation of the smartphone can be modeled as resulting from three elemental rotations in sequence applied to a starting orientation with the smartphone facing up and not undergoing any linear acceleration. Suppose the accelerometer readings in the starting position and measured after three rotations are A_0 and A_r respectively. We have

$$A_0 = [0 \quad 0 \quad g]^T \tag{3.9.1}$$

$$A_r = R_x(\phi_0)R_y(\theta_0)R_z(\psi_0)A_0 \tag{3.9.2}$$

According to (Ozyagcilar, 2015), the initial roll angle ϕ_0 and the initial pitch angle θ_0 are calculated from the accelerometer measurements by pre-multiplying A_r by inverse rotation matrices, which are expressed as

$$R_y(-\theta_0)R_x(-\phi_0)A_r = R_z(\psi_0)A_0 = \begin{bmatrix} \cos(\psi_0) & \sin(\psi_0) & 0 \\ -\sin(\psi_0) & \cos(\psi_0) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.9.3}$$

$$R_y(-\theta_0)R_x(-\phi_0)A_r = R_z(\psi_0)A_0 = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{3.9.4}$$

The three components of A_r are defined as $A_r = [A_{rx} \quad A_{ry} \quad A_{rz}]^T$. By substituting A_r with its column vector form, we have

$$R_y(-\theta_0)R_x(-\phi_0)\begin{bmatrix}A_{rx}\\A_{ry}\\A_{rz}\end{bmatrix}=\begin{bmatrix}0\\0\\g\end{bmatrix} \tag{3.9.5}$$

By substituting Equations (3.5.7) and (3.5.6) into Equation (3.9.5) and rearranging, we have

$$\begin{bmatrix}A_{rx}\cos(\theta_0)+A_{ry}\sin(\theta_0)\sin(\phi_0)+A_{rz}\sin(\theta_0)\cos(\phi_0)\\A_{ry}\cos(\phi_0)-A_{rz}\sin(\phi_0)\\-A_{rx}\sin(\theta_0)+A_{ry}\cos(\theta_0)\sin(\phi_0)+A_{rz}\cos(\theta_0)\cos(\phi_0)\end{bmatrix}=\begin{bmatrix}0\\0\\g\end{bmatrix} \tag{3.9.6}$$

The initial roll angle ϕ_0 and the initial pitch angle θ_0 are given by

$$\phi_0=\tan^{-1}\left(\frac{A_{ry}}{A_{rz}}\right) \tag{3.9.7}$$

$$\theta_0=\tan^{-1}\left(\frac{-A_{rx}}{A_{ry}\sin(\phi_0)+A_{rz}\cos(\phi_0)}\right) \tag{3.9.8}$$

The initial yaw angle ψ_0 can be obtained by using the magnetometer measurements. As discussed in Section 3.6, the magnetometer reading B_r measured after three rotations is given by

$$B_r = R_x(\phi_0)R_y(\theta_0)R_z(\psi_0)B\begin{bmatrix}\cos\alpha\\0\\\sin\alpha\end{bmatrix} + M_{hard} \tag{3.9.9}$$

By rearranging and pre-multiplying the term $B_r - M_{hard}$ by $R_x(-\phi_0)$ and $R_y(-\theta_0)$, we have

$$R_y(-\theta_0)R_x(-\phi_0)(B_r - M_{hard}) = R_z(\psi_0)B\begin{bmatrix}\cos\alpha\\0\\\sin\alpha\end{bmatrix} \tag{3.9.10}$$

After we substitute M_{hard} and B_r with their column vector forms, substitute Equations (3.5.6), (3.5.7), and (3.5.8) into Equation (3.9.10), and rearrange the equation, we have

$$\begin{bmatrix}(B_{r-x} - M_{hard-x})\cos(\theta_0) + \left(B_{r-y} - M_{hard-y}\right)\sin(\theta_0)\sin(\phi_0) + (B_{r-z} - M_{hard-z})\sin(\theta_0)\cos(\phi_0)\\ \left(B_{r-y} - M_{hard-y}\right)\cos(\phi_0) - (B_{r-z} - M_{hard-z})\sin(\phi_0)\\ -(B_{r-x} - M_{hard-x})\sin(\theta_0) + \left(B_{r-y} - M_{hard-y}\right)\cos(\theta_0)\sin(\phi_0) + (B_{r-z} - M_{hard-z})\cos(\theta_0)\cos(\phi_0)\end{bmatrix} = \begin{bmatrix}\mathrm{B}\cos(\psi_0)\cos\alpha\\ -\mathrm{B}\sin(\psi_0)\cos\alpha\\ B\sin\alpha\end{bmatrix} \tag{3.9.11}$$

As Equation (3.9.11) is in the form of matrix equality, we have three equations by comparing each element of the two matrices. If we divide the second equation by the first equation from both sides, the initial yaw angle ψ_0 can be calculated as

$$\psi_0 = \tan^{-1}\left(\frac{(B_{r-z} - M_{hard-z})\sin(\phi_0) - (B_{r-y} - M_{hard-y})\cos(\phi_0)}{(B_{r-x} - M_{hard-x})\cos(\theta_0) + \left(B_{r-y} - M_{hard-y}\right)\sin(\theta_0)\sin(\phi_0) + (B_{r-z} - M_{hard-z})\sin(\theta_0)\cos(\phi_0)}\right) \tag{3.9.12}$$

Static Mode Heading and Orientation Estimation

In the static mode, the indoor positioning system uses a unit quaternion ${}^{b}_{w}q$ to represent the orientation in the world reference frame relative to the sensor/body frame. Since the relative position of the IMU sensors is fixed to the pedestrian's body, the unit quaternion ${}^{b}_{w}q$ also represents the pedestrian's orientation at the current system state and can be obtained by taking the integral of the first derivative of the quaternion with respect to time. ${}^{b}_{w}q$ is then given by

$${}^{b}_{w}q = \int \dot{q}_{w'}\ dt \tag{3.9.13}$$

The first derivative of the unit quaternion $\dot{q}_{w'}$, also known as quaternion rates, is the rate of change of the orientation. $\dot{q}_{w'}$ can be calculated based on gyroscope measurements $C_r = [C_{rx} \quad C_{ry} \quad C_{rz}]^T$ and the previous orientation. All gyroscope measurements are defined as

$$C_r\big(t(i)\big) = \big[C_{rx}\big(t(i)\big) \quad C_{ry}\big(t(i)\big) \quad C_{rz}\big(t(i)\big)\big]^T, \quad i = 1, 2, \dots, N \tag{3.9.14}$$

where $t(i)$ denotes each time instance. Hence, the first derivative of the unit quaternion $\dot{q_{w'}}$ is estimated by

$$(\dot{q_{w'}})_i = \frac{1}{2}({}_{w}^{b}q)_{i-1} \cdot \left[0 \quad C_{rx}\big(t(i)\big) - k_1 e \quad C_{ry}\big(t(i)\big) - k_1 e \quad C_{rz}\big(t(i)\big) - k_1 e\right]^T$$

(3.9.15)

where · is quaternion multiplication.

In Equation (3.9.15), a model gain value k_1 and a control variable e are introduced based on (Madgwick *et al.*, 2020) to prevent the gyroscope measurements from drifting away and to correct the preliminarily estimated heading calculated by pure gyroscope integration. The model gain value k_1 serves as the weight given to the orientation estimation correction term. The normal value of k_1 should be comparatively small when the system stays in a stable state without any significant turbulences. The reason of applying a small value of k_1 is that the control variable e is used to compensate for low frequency errors resulting from the gyroscopic drift. However, the value of k_1 could be moderately increased during the early stages of the system because the initial calculated orientation always deviates significantly from the true orientation. Moreover, when erroneous accelerometer and magnetometer measurements are collected, the value of k_1 should be decreased so that the estimated orientation mainly depends on the result of gyroscope integration. The orientation estimation correction control variable e is given by $e = e_a + e_m$, where e_a and e_m are calculated using the measured and reference vectors for gravity and magnetic field respectively. That is, the angle difference between the measured vector and the rotated reference vector determines the orientation correction values.

The control variable e_a at current system state is equal to the cross product of $\hat{A}_r\big(t(i)\big)$ the normalized accelerometer measurement vector at current time instance and $p_g{}'$ the reference vector for gravity. So e_a is given by

$$e_a = \hat{A}_r\big(t(i)\big) \times p_g{}' \tag{3.9.16}$$

For measured accelerometer data, the normalized vector is calculated by

$$\hat{A}_r = \frac{A_r}{\|A_r\|} \tag{3.9.17}$$

The reference vector $p_g{}'$ is derived from quaternion multiplications of the rotation quaternion $({}_w^bq)_i$, the gravity vector p_g, and the inverse of the rotation quaternion $({}_w^bq)_i{}^{-1}$. $p_g{}'$ can be calculated by

$$p_g{}' = ({}_w^bq)_i \cdot p_g \cdot ({}_w^bq)_i{}^{-1} \tag{3.9.18}$$

where $\cdot$ is quaternion multiplication. $({}_w^bq)_i$ is the current estimation of the pedestrian's orientation, which can be expressed as

$$({}_{w}^{b}q)_i = [q_{i0} \quad q_{i1} \quad q_{i2} \quad q_{i3}]^T \tag{3.9.19}$$

p_g is the gravity vector, which is given by

$$p_g = [0 \quad 0 \quad -1]^T \tag{3.9.20}$$

Based on the relationship between unit quaternion and rotation matrix described in (Diebel, 2006), ${p_g}'$ is further calculated as the result of matrix multiplication of rotation matrix R_q and the gravity vector p_g. ${p_g}'$ is given by

$$p_g{}' = R_q p_g \tag{3.9.21}$$

$$R_q = \begin{bmatrix} {q_{i0}}^2 + {q_{i1}}^2 - {q_{i2}}^2 - {q_{i3}}^2 & 2q_{i1}q_{i2} + 2q_{i0}q_{i3} & 2q_{i1}q_{i3} - 2q_{i0}q_{i2} \\ 2q_{i1}q_{i2} - 2q_{i0}q_{i3} & {q_{i0}}^2 - {q_{i1}}^2 + {q_{i2}}^2 - {q_{i3}}^2 & 2q_{i2}q_{i3} + 2q_{i0}q_{i1} \\ 2q_{i1}q_{i3} + 2q_{i0}q_{i2} & 2q_{i2}q_{i3} - 2q_{i0}q_{i1} & {q_{i0}}^2 - {q_{i1}}^2 - {q_{i2}}^2 + {q_{i3}}^2 \end{bmatrix} \tag{3.9.22}$$

By simple matrix multiplication, the reference vector ${p_g}'$ at current system state is

$$p_g{}' = [2(q_{i0}q_{i2} - q_{i1}q_{i3}) \quad -2(q_{i2}q_{i3} + q_{i0}q_{i1}) \quad -{q_{i0}}^2 + {q_{i1}}^2 + {q_{i2}}^2 - {q_{i3}}^2]^T \tag{3.9.23}$$

Similarly, the control variable e_m at current system state is equal to the cross product of $\hat{B}_r{}'\big(t(i)\big)$ the converted and normalized magnetometer measurement vector at current time instance and $p_m{}'$ the reference vector for the magnetic field. So e_m is given by

$$e_m = \hat{B}_r{}'\big(t(i)\big) \times p_m{}' \qquad (3.9.24)$$

By taking the cross product of the measured gravity vector and the measured magnetic vector, the measured magnetic vector $\hat{B}_r\big(t(i)\big)$ is converted to magnetic east, which is perpendicular to gravity. The $\hat{B}_r{}'\big(t(i)\big)$ denotes the converted $\hat{B}_r\big(t(i)\big)$ and is given by

$$\hat{B}_r{}'\big(t(i)\big) = \hat{A}_r\big(t(i)\big) \times \hat{B}_r\big(t(i)\big) \qquad (3.9.25)$$

For measured magnetometer data, the normalized vector is calculated by

$$\hat{B}_r = \frac{B_r}{||B_r||} \qquad (3.9.26)$$

The reference vector $p_m{}'$ is derived from quaternion multiplications of the rotation quaternion $({}_w^b q)_i$, the magnetic east vector p_m, and the inverse of the rotation quaternion $({}_w^b q)_i{}^{-1}$. $p_m{}'$ can be calculated by

$$p_m{}' = ({}^b_w q)_i \cdot p_m \cdot ({}^b_w q)_i{}^{-1} \tag{3.9.27}$$

where · is quaternion multiplication. p_m is the original reference vector for the magnetic field, which is given by

$$p_m = [0 \quad -1 \quad 0]^T \tag{3.9.28}$$

Based on the same matrix multiplication method used for calculating $p_g{}'$, the reference vector $p_m{}'$ at current system state can be written in the same way as

$$p_m{}' = R_q p_m \tag{3.9.29}$$

$$p_m{}' = [-2(q_{i1}q_{i2} + q_{i0}q_{i3}) \quad -q_{i0}{}^2 + q_{i1}{}^2 - q_{i2}{}^2 + q_{i3}{}^2 \quad 2(q_{i0}q_{i1} - q_{i2}q_{i3})]^T \tag{3.9.30}$$

Dynamic Mode Heading and Orientation Estimation

In the dynamic mode, a double phase extended Kalman filter is adopted so that more reliable orientation estimation results are produced when the accelerometer measures both the gravitational acceleration and other accelerations. Similar to the Kalman filter (Kalman, 1960), a typical extended Kalman filter algorithm consists of two stages: prediction and update. The proposed indoor positioning system uses the gyroscope data to estimate the pedestrian's orientation in the prediction stage and uses the accelerometer data and magnetometer data to correct the angular velocity-based orientation estimation errors in the update stage.

The prediction stage includes two critical equations: predicted state estimate and predicted error covariance. The prediction stage equations are given by

Predicted state estimate:

$$\hat{x}_i{}^{-} = F\hat{x}_{i-1}{}^{+} + Bu_{i-1} \tag{3.9.31}$$

Predicted error covariance:

$$P_i{}^{-} = FP_{i-1}{}^{+}F^T + Q \tag{3.9.32}$$

In the extended Kalman filter equations, the hat operator denotes an estimate of the variable. The superscripts $-$ and $+$ denote *prior* and *posterior* estimates respectively.

The predicted state estimate is evolved from the previous updated state estimate. The predicted state estimate equation defines a discrete time state model that describes the evolution of the current system state $\hat{x}_i{}^{-}$ from the previous updated system state estimation $\hat{x}_{i-1}{}^{+}$. F is the state transition matrix from the previous state to the current state. B is the input control matrix associated with the system input u_{i-1}. Since any prediction has noises and uncertainties, keeping track of the error covariance is necessary. In the predicted error covariance equation, the *prior* state error covariance $P_i{}^{-}$ representing the error in the state transition and estimation process is calculated using the same state transition matrix F, the previous updated *posterior* error covariance matrix ${P_{i-1}}^+$ and the process noise covariance matrix Q.

Similar to the static mode model, a unit quaternion ${}^b_w q'$representing the orientation is used as system state in the predicted state estimate equation for the dynamic mode model. Quaternion ${}^b_w q'$ can be obtained by performing direct gyroscope integration. The gyroscope integration process defines the system state transition process, and there is no external system input u_{i-1} and input control matrix B for the dynamic mode model. Given gyroscope measurements $C_r = [C_{rx} \quad C_{ry} \quad C_{rz}]^T$, ${}^b_w q'$ is calculated by

$$ {}^b_w q' = \int \dot{q_{w'}}' \ dt \tag{3.9.33}$$

$$ (\dot{q_{w'}}')_i = \frac{1}{2} ({}^b_w q')_{i-1} \cdot \left[0 \quad C_{rx}\big(t(i)\big) \quad C_{ry}\big(t(i)\big) \quad C_{rz}\big(t(i)\big)\right]^T \tag{3.9.34}$$

where · is quaternion multiplication. ${}^b_w q'$ is the current prediction of the pedestrian's orientation in the dynamic mode, which is expressed as

$$ ({}^b_w q')_i = [q_{i0}' \quad q_{i1}' \quad q_{i2}' \quad q_{i3}']^T \tag{3.9.35}$$

As described above, the proposed indoor positioning system uses gyroscope data only to estimate the pedestrian's orientation in the prediction stage. In the update stage of the extended Kalman filter, the accelerometer data and magnetometer data serve as new observations to further correct the *prior* estimates ${\hat{x}_i}^-$ and ${P_i}^-$ (Sabatelli *et al.*, 2013). The update stage of the extended Kalman filter algorithm includes three main equations: update state estimate, update error covariance, and Kalman gain.

Update state estimate:

$$\hat{x}_i{}^{+} = \hat{x}_i{}^{-} + K_i\big(z_i - h(\hat{x}_i{}^{-})\big) \tag{3.9.36}$$

Kalman gain:

$$K_i = P_i{}^{-} H_i{}^{T} (H_i P_i{}^{-} H_i{}^{T} + R)^{-1} \tag{3.9.37}$$

Update error covariance:

$$P_i{}^{+} = (I - K_i H)\, P_i{}^{-} \tag{3.9.38}$$

The update state estimate equation generates a new *posterior* estimate $\hat{x}_i{}^{+}$ by combining the *prior* estimate $\hat{x}_i{}^{-}$obtained from the prediction stage with accelerometer measurements and magnetometer measurements. The proposed orientation algorithm is designed to have two phases in the extended Kalman filter update stage. The first phase uses accelerometer data to correct the orientation estimate and the second phase uses magnetometer data to further adjust the result. In the update state estimate equation, z_i is the actual accelerometer or magnetometer measurement data. The nonlinear function $h(\hat{x}_i{}^{-})$is the expected measurement calculated from the prediction stage. In this system, $h(\hat{x}_i{}^{-})$ is predicted gravity for Phase I and magnetic field for Phase II. The term $z_i - h(\hat{x}_i{}^{-})$, which is the difference between the actual and expected measurements, is known as the measurement residual or innovation. Then the measurement residual is weighted with the Kalman gain K_i to calculate the *posterior* estimate $\hat{x}_i{}^{+}$. Adopting the Kalman gain is most effective for the system dynamic mode when the accelerometer measures both gravity and other accelerations because under this circumstance the measurement residual is large.

The Kalman gain K_i determines the weight given to the estimation correction derived from measurement data. The Kalman gain K_i is obtained by comparing the *prior* state error covariance $P_i{}^{-}$ and the measurement noise covariance matrix R. Measurement noise includes accelerometer and magnetometer measurement noise and other system noises. In the Kalman gain equation, H_i is the Jacobian matrix that collects the partial derivatives of the predicted measurement $h({}_{w}^{b}q')$ obtained using the current orientation estimation quaternion with respect to the quaternion. Since the predicted measurement $h({}_{w}^{b}q')$ for Phase I and Phase II are calculated differently, two Kalman gains are used in this model.

In the update stage of the extended Kalman filter algorithm, the *posterior* error covariance matrix $P_i{}^{+}$ is calculated using the *prior* state error covariance $P_i{}^{-}$ and the Kalman gain K_i, as shown in the update error covariance equation.

Phase I of the extended Kalman filter Update stage uses accelerometer data to correct the orientation estimation, which is also the current system state $\hat{x}_i{}^{-}$. The predicted accelerometer measurement vector $h_a({}_{w}^{b}q')$ for Phase I is calculated by multiplying the gravity vector with the rotation matrix $R_q{}'$:

$$h_a({}_w^bq') = {R_q}'[0 \quad 0 \quad -1]^T \tag{3.9.39}$$

$$h_a({}_w^bq') = \left[2({q_{i0}}'{q_{i2}}' - {q_{i1}}'{q_{i3}}') \quad -2({q_{i2}}'{q_{i3}}' + {q_{i0}}'{q_{i1}}') \quad -{q_{i0}}'^2 + {q_{i1}}'^2 + {q_{i2}}'^2 - {q_{i3}}'^2\right]^T \tag{3.9.40}$$

where

$${R_q}' = \begin{bmatrix} {q_{i0}}'^2 + {q_{i1}}'^2 - {q_{i2}}'^2 - {q_{i3}}'^2 & 2{q_{i1}}'{q_{i2}}' + 2{q_{i0}}'{q_{i3}}' & 2{q_{i1}}'{q_{i3}}' - 2{q_{i0}}'{q_{i2}}' \\ 2{q_{i1}}'{q_{i2}}' - 2{q_{i0}}'{q_{i3}}' & {q_{i0}}'^2 - {q_{i1}}'^2 + {q_{i2}}'^2 - {q_{i3}}'^2 & 2{q_{i2}}'{q_{i3}}' + 2{q_{i0}}'{q_{i1}}' \\ 2{q_{i1}}'{q_{i3}}' + 2{q_{i0}}'{q_{i2}}' & 2{q_{i2}}'{q_{i3}}' - 2{q_{i0}}'{q_{i1}}' & {q_{i0}}'^2 - {q_{i1}}'^2 - {q_{i2}}'^2 + {q_{i3}}'^2 \end{bmatrix} \tag{3.9.41}$$

The Jacobian matrix $(H_i)_a$ for Phase I is then given by

$$(H_i)_a = \left[\frac{\partial h_a}{\partial {q_{i0}}'} \quad \frac{\partial h_a}{\partial {q_{i1}}'} \quad \frac{\partial h_a}{\partial {q_{i2}}'} \quad \frac{\partial h_a}{\partial {q_{i3}}'}\right] \tag{3.9.42}$$

When updating the *posterior* estimate ${\hat{x}_i}^+$ using accelerometer data to calculate the measurement residual in the update stage, the fourth element of the system state quaternion vector will not be updated so that Phase I only corrects the pitch and roll angles.

Similar to Phase I of the update stage, Phase II of the extended Kalman filter update stage uses magnetometer data to further adjust the orientation estimation ${\hat{x}_i}^-$. The predicted magnetometer measurement vector $h_m({}_w^bq')$ for Phase II is calculated by multiplying the magnetic field vector with the rotation matrix ${R_q}'$:

$$h_m({}_w^b q') = R_q{}'[0 \quad -1 \quad 0]^T \tag{3.9.43}$$

$$h_m({}_w^b q') = \left[-2(q_{i1}{}'q_{i2}{}' + q_{i0}{}'q_{i3}{}') \quad -q_{i0}{}'^2 + q_{i1}{}'^2 - q_{i2}{}'^2 + q_{i3}{}'^2 \quad 2(q_{i0}{}'q_{i1}{}' - q_{i2}{}'q_{i3}{}')\right]^T \tag{3.9.44}$$

The Jacobian matrix $(H_i)_m$ for Phase II is then given by

$$(H_i)_m = \left[\frac{\partial h_m}{\partial q_{i0}{}'} \quad \frac{\partial h_m}{\partial q_{i1}{}'} \quad \frac{\partial h_m}{\partial q_{i2}{}'} \quad \frac{\partial h_m}{\partial q_{i3}{}'}\right] \tag{3.9.45}$$

For Phase II corrections that update the *posterior* estimate $\hat{x}_i{}^+$ using magnetometer data to calculate the measurement residual in the update stage, the second and third elements of the system state quaternion vector will not be changed. Therefore, Phase II corrections only adjust the yaw angle and both pitch and roll angles are not affected.

3.10. Position Estimation

The last module of the proposed indoor positioning system is the position estimation module. Finally, the pedestrian's position is estimated using the moving distance data generated in the step detection module and the orientation/heading data estimated in the heading and orientation estimation module. The current pedestrian's position is calculated from the previous position with the addition of the directional moving distance increment at the current time instance. The angle used in the position estimation equations is the angular difference of the current and the initial yaw angles. The calculation and derivation of the initial yaw angle ψ_0 are provided in Section 3.9. The pedestrian's position estimation is calculated by

$$XPose\big(t(i)\big) = XPose\big(t(i-1)\big) + D_i \times \sin(\,\psi_i - \,\psi_0)\,,\;\; i = 1, 2, \dots, N \qquad (3.10.1)$$

$$YPose\big(t(i)\big) = YPose\big(t(i-1)\big) + D_i \times \cos(\,\psi_i - \,\psi_0)\,,\;\; i = 1, 2, \dots, N \qquad (3.10.2)$$

In the above position estimation equations, $XPose\big(t(i)\big)$ and $YPose\big(t(i)\big)$ are the current position coordinates and $XPose\big(t(i-1)\big)$ and $YPose\big(t(i-1)\big)$ are the previous position coordinates. The moving distance increment is denoted as D_i. ψ_i and ψ_0 denote the current and the initial yaw angles respectively.

Chapter 4

Experimental Evaluation

4.1. Introduction

In this chapter, we evaluate the performance and accuracy of the proposed indoor positioning system in a residential building for an unstructured indoor environment. The performance of the proposed positioning system model and algorithms are evaluated through carrying out extensive real-world experiments for different scenarios and analyzing the results. The experimental evaluation process includes investigating the performance of mode detection, step detection, the heading and orientation estimation algorithms, and finally the position estimation results. The main points we seek to validate are: (i) how effective can the proposed mode detection model separate dynamic mode from static mode, (ii) how accurately can the proposed step detection generate a detected step set $S = \{s_j\}_{j=1}^{J}$, (iii) how well do the proposed heading and orientation estimation algorithms work in real-world scenarios, and (iv) what is the overall position estimation accuracy and computational efficiency of the proposed indoor positioning system under different scenarios.

4.2. Experimental Setup

The system model and algorithms are developed and implemented using MathWorks MATLAB and tested with real acquired sensor data. The system does not require additional infrastructure support and the inertial sensor data sets are collected using a smartphone mobile application and then sent to a data cloud. During the data collection process, the pedestrian walks in the reference path and the smartphone is handheld by the pedestrian and placed in front of the waist area with its screen facing up. The relative position of the smartphone is fixed to the pedestrian's body. As mentioned in Section 3.9, coordinate transformations between the body and the sensor frame are unnecessary. The system has three types of input data: the accelerometer data measuring acceleration in meters per second squared (m/s^2), the magnetometer data measuring the magnetic field in microtesla (μT), and gyroscope data measuring angular velocity in radians per second (rad/s). The default sample rate is 10 Hz. All experiments were executed on a machine with a 3.3 GHz Dual-Core Intel Core i7 processor and 16 GB 2133 MHz memory, running macOS Catalina 10.15.7.

To evaluate the performance of the proposed positioning model and algorithms, five experiment scenarios including 360-degree rotations, straight-line motion, U-turns motion, L-shape motion, and rectangular motion are considered. The experiment scenarios are illustrated in Figure 4.1.

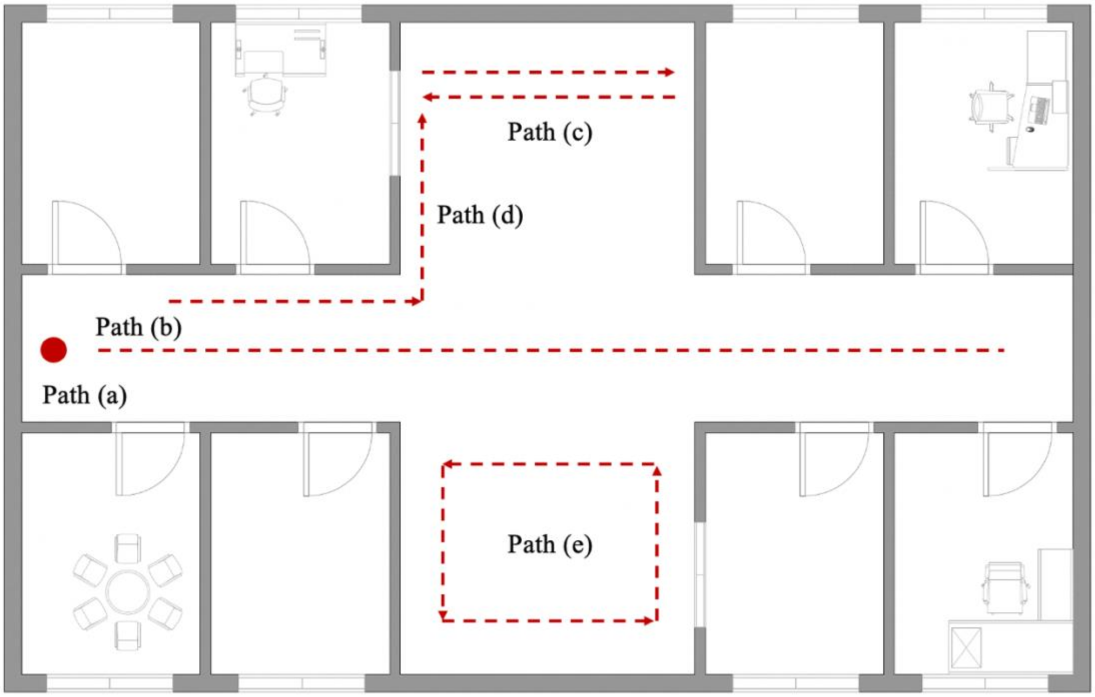

Figure 4.1: Experiment Scenarios

4.3. Experiment Results & Discussions (& Performance Analysis)

Performance of Mode Detection

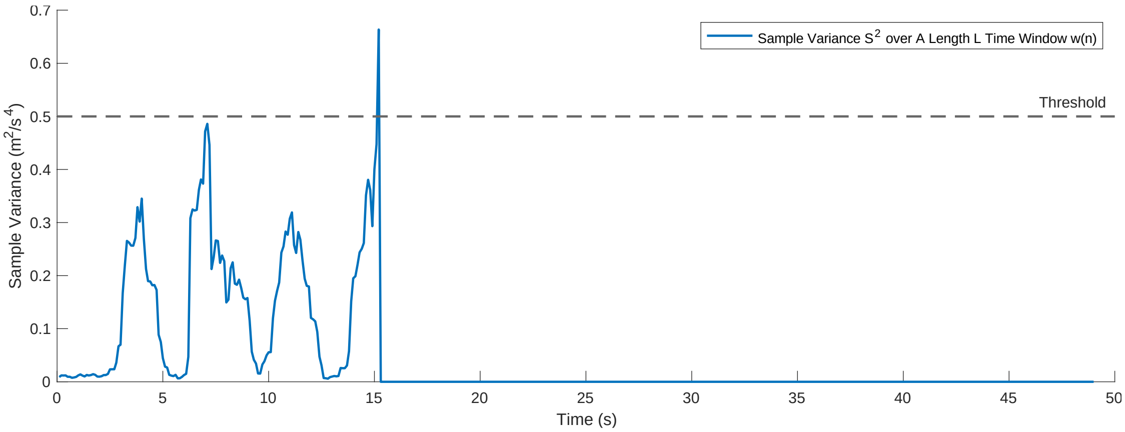

Figure 4.2: Mode Detection Result

Table 4.1: Accuracy of Proposed Mode Detection Module

Static Mode Period (second)			
Test No.	Actual	Detected by Proposed Mode Detection Module	Error
1	5	4.7	6%
2	10	10.2	2%
3	15	15.2	1%
4	20	20.3	2%
5	25	25.3	1%
6	30	30.1	0.3%
Mean Relative Error			2.1%
Mean Squared Error (MSE)			6%

To evaluate the accuracy of the proposed mode detection module, the path of straight-line motion is chosen. Figure 4.2 shows the mode detection result of an actual scenario in which a pedestrian starts to walk after performing 15-second-long rotation motions. From Figure 4.2 and Table 4.1, it can be seen that the proposed model detection module detected the start time of the dynamic mode at 15.2 seconds. The signal shown in Figure 4.2 is the sample variance $S^2$of the magnitude a_i of accelerometer readings A_r over a length L time window $w(n)$. When S^2 exceeds the threshold, the proposed system enters the dynamic mode. Six rounds of experiments with different actual length of the static mode period have been performed to evaluate the accuracy of the proposed model detection module. Table 4.1 compares the actual moment with the detected moment that the system leaves the static mode and enters the dynamic mode. The mean relative error of all six rounds of experiments is 2.1% and the MSE is 6%. The step detection module is triggered after the system enters the dynamic mode.

Performance of Step Detection

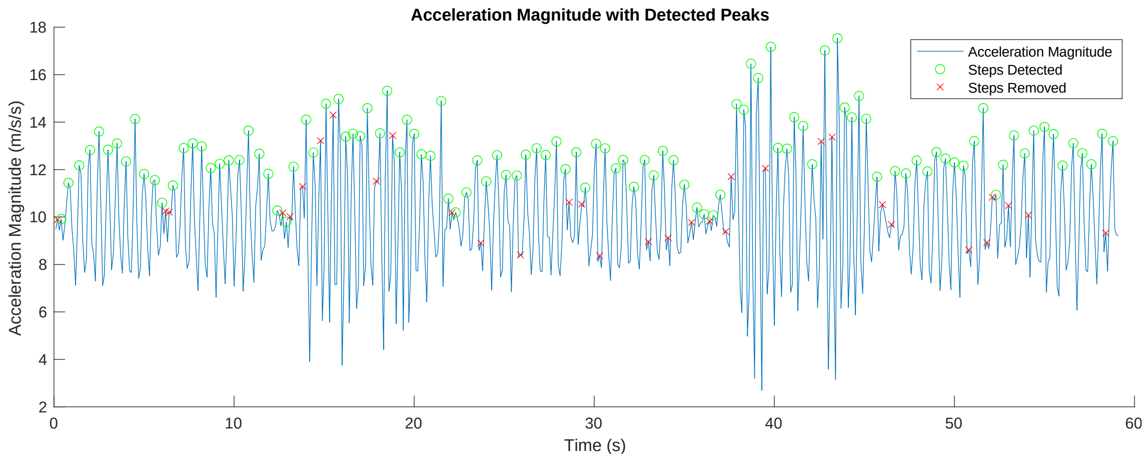

Figure 4.3: Step Detection Result

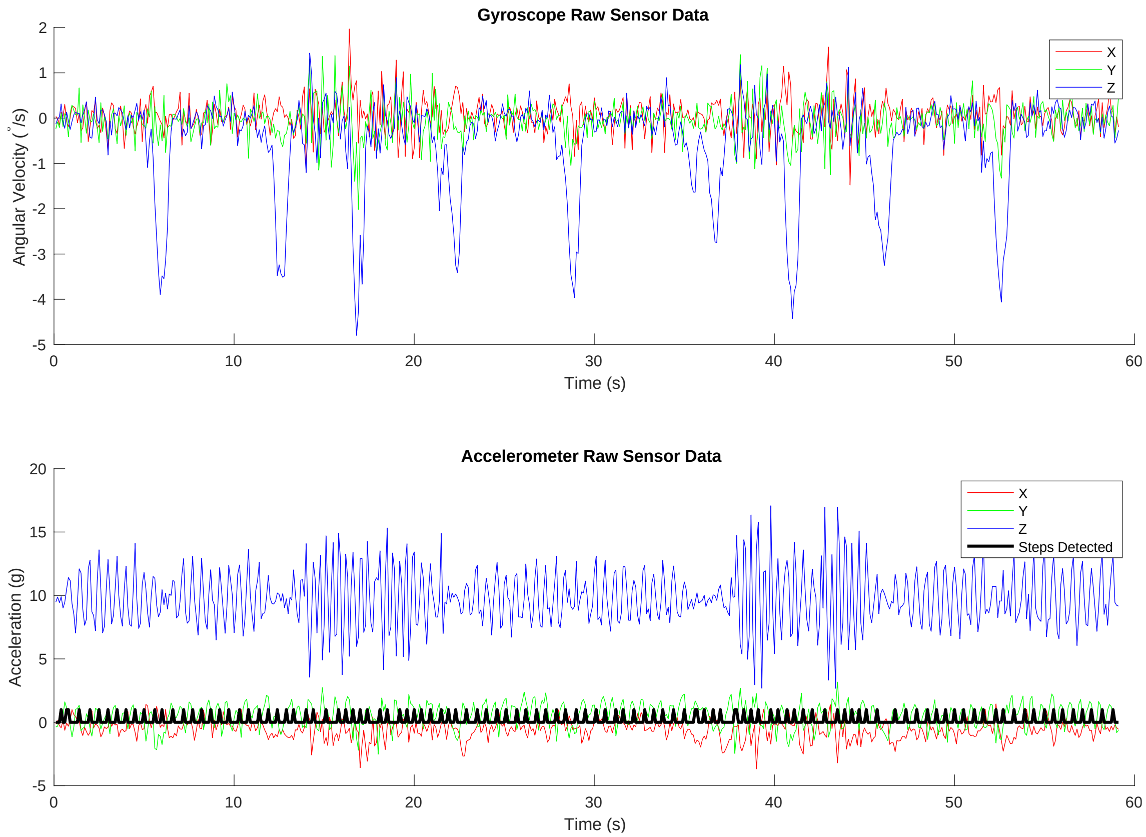

Figure 4.4: Gyroscope and Accelerometer Raw Sensor Data with Steps Detected by the Proposed Step Detection Module

Table 4.2: Performance of Step Detection Module

Test No.	True # of Steps	Threshold-based Detection	Peak/Valley-based Detection
Straight-line Path – Constant Speed			
1-1	63	62	64
Straight-line Path – Varying Speed			
2-1	62	62	61
U-turns Path - Constant Speed			
3-1	116	114	115
U-turns Path - Varying Speed			
4-1	110	112	111

The paths of straight-line motion and repetitive U-turns motion are chosen to evaluate the performance of the proposed multi-layer step detection module. Specifically, four sets of experiments including straight-line path with constant speed, straight-line path with varying speed, U-turns path with constant speed, and U-turns path with varying speed have been performed. The U-turns path is chosen to test the step refinement layer. Table 4.2 shows the experimental results of the proposed step detection module for the four scenarios. From Table 4.2, the results show that both threshold-based step detection and peak/valley-based step detection demonstrated good performance results with high true positive rates, and there is not a significant performance gap between the two step detection methods. However, the performance of the threshold-based detection tends to degrade when the optimal threshold is not selected. Compared with the threshold-based detection, the peak/valley-based detection is more stable.

As elaborated in Section 3.8, three layers of step detection techniques are used to eliminate undesired false positive steps. In Figure 4.3, the y-axis represents acceleration magnitude, and the x-axis represents time. The curve drawn in the blue solid line is the digital signal of acceleration magnitude. The green circle markers indicate the steps detected after applying the three layers of step detection. The red cross markers indicate the removed steps, which were detected as peaks initially but filtered out during the multi-layer step detection process. Figure 4.3 shows all the steps that were detected after the multi-layer step detection process and also the false positive ones that were removed during the adjustment-refinement-validation process. It is observed in the experiments that the constraints imposed in the first layer have a dominant impact on rejecting a large number of false positive steps. Figure 4.4 illustrates the signals of gyroscope and accelerometer raw sensor data with detected steps placed on the same timeline. As shown in Figure 4.4, the signal pattern matching technique used in the second step refinement layer is effective in eliminating the falsely detected peaks especially for paths that contain repetitive motions such as the U-turns path. The third step validation layer cleans up one to two false positive steps when any abnormal velocity is detected.

Performance of Heading and Orientation Estimation

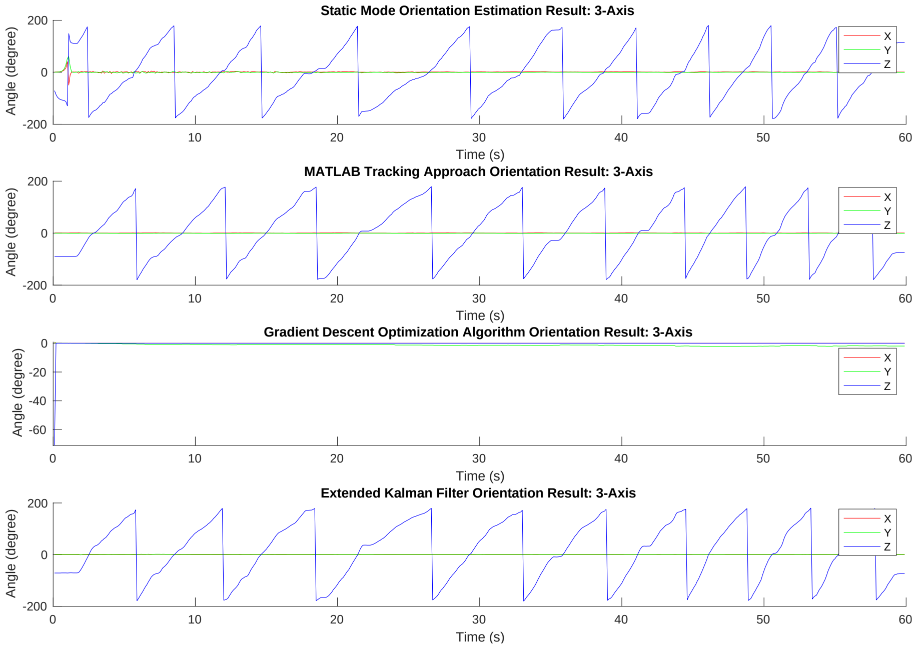

Figure 4.5: Performance Comparison of Heading and Orientation Estimation Algorithms under Static Mode

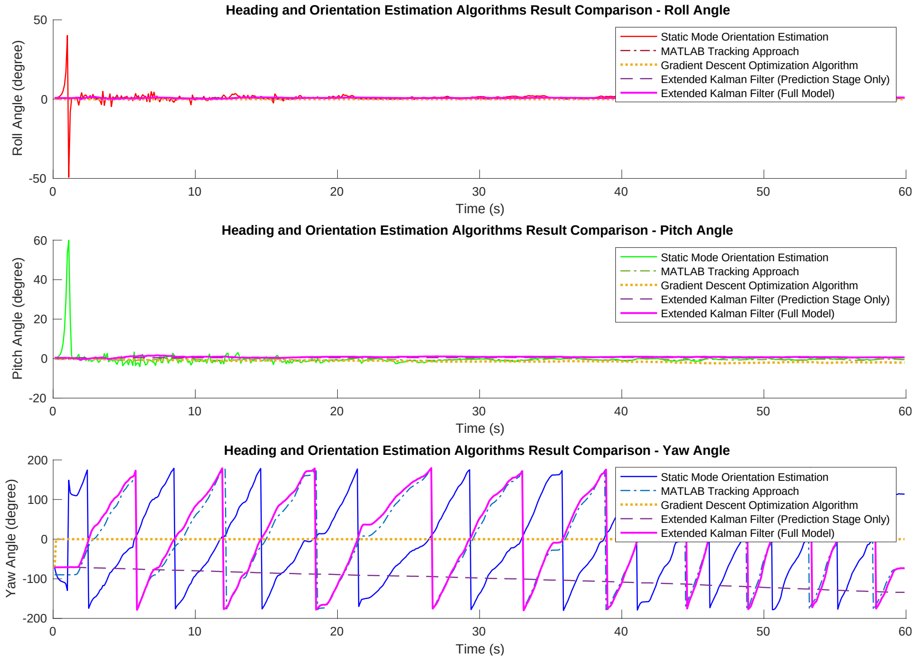

Figure 4.6: Performance Comparison of Heading and Orientation Estimation Algorithms under Static Mode (Individual Axes)

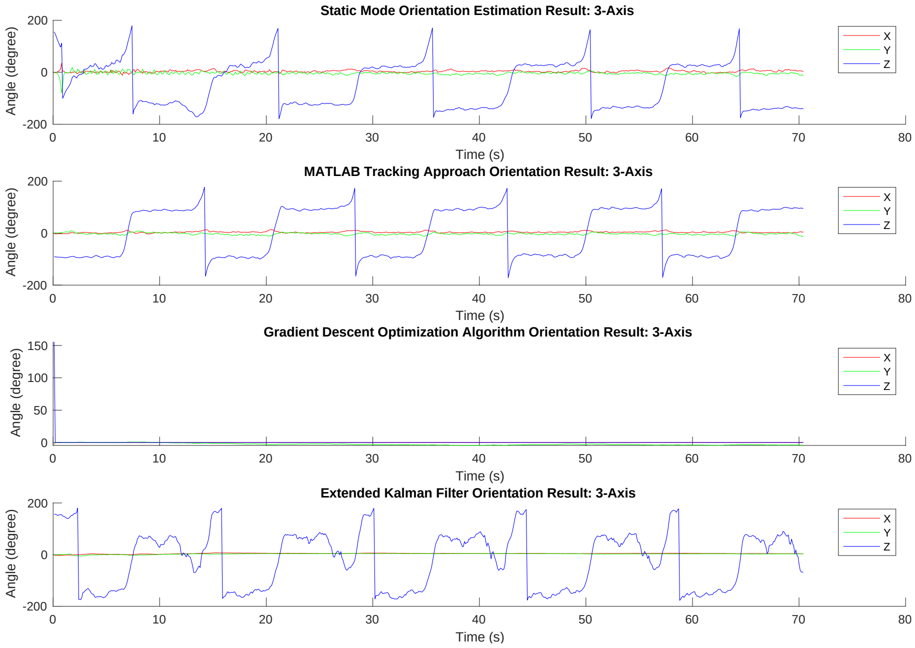

Figure 4.7: Performance Comparison of Heading and Orientation Estimation Algorithms under Dynamic Mode

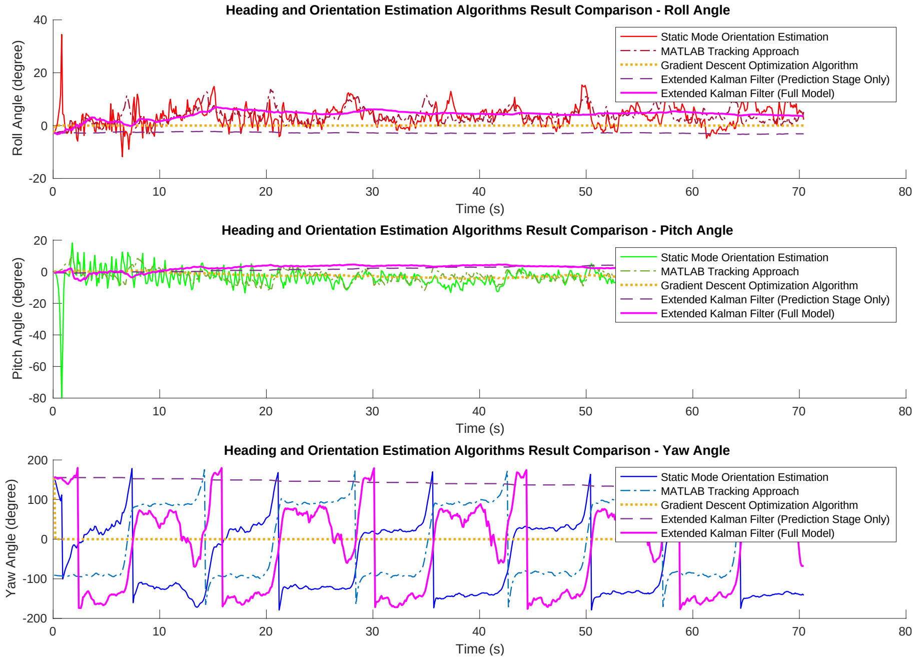

Figure 4.8: Performance Comparison of Heading and Orientation Estimation Algorithms under Dynamic Mode (Individual Axes)

Table 4.3: Volatility/Standard Deviation of Roll Angle & Pitch Angle

Estimation Results under Dynamic Mode

Angle **Algorithm**	Roll Angle ϕ	Pitch Angle θ
Static Mode Orientation Est.	4.09	5.69
Extended Kalman Filter (Prediction Stage Only)	0.25	1.9
Extended Kalman Filter (Full Model)	1.68	1.79
MATLAB Tracking Approach	2.95	3.77
Gradient Descent Optimization Algorithm	0.10	1.44

To evaluate the performance of the heading and orientation estimation module and algorithms, the path of 360-degree rotations is chosen to test the static mode scenario, and repetitive U-turns motion is chosen to analyze the dynamic mode scenario. The former path rotates around the yaw axis at a constant speed, and the latter U-turns motion consists of ten instant direction changes. The proposed heading and orientation estimation module uses sensor fusion techniques to estimate the pedestrian's heading at the current system state. Concretely, the system first predicts a preliminary orientation based on gyro integration, then the static mode utilizes reference vectors to correct the preliminary orientation estimation while the dynamic mode updates the *prior* estimate by fusing accelerometer and magnetometer observations. The static mode orientation estimation algorithm and the extended Kalman filter model are compared with gyro integration, which is the prediction stage model of the EKF, the MATLAB tracking approach, and a gradient descent optimization algorithm (Madgwick *et al.*, 2011). The heading and orientation estimation results of all algorithms under the static mode are shown in Figure 4.5 and Figure 4.6. The heading and orientation estimation results of all approaches under the dynamic mode are provided in Figure 4.7 and Figure 4.8. Figure 4.5 and Figure 4.7

show the 3-axis results of different algorithms, and Figure 4.6 and Figure 4.8 compare the three Euler angles of all five approaches separately.

From Figure 4.5 and Figure 4.6, it is observed that all five algorithms detected neglectable roll angle and pitch angle changes, and the proposed heading and orientation algorithms and the MATLAB tracking approach successfully detected ten actual rotations shown in the yaw angle signal results. Gyro integration and the gradient descent optimization algorithm were unable to detect the rotation motions in the static mode, hence produced false heading estimations. It is noticed that there is a short 1-second convergence phase for the reference vector-based static mode orientation estimation algorithm. Under the dynamic mode, Figure 4.7 and Figure 4.8 show that the proposed extended Kalman filter generated more stable signals in the roll angle and pitch angle results and avoided obvious high frequency signal fluctuations as produced by the static mode orientation estimation algorithm. Table 4.3 summarizes the standard deviation of the roll angle and pitch angle of all algorithms under the dynamic mode. From Table 4.3, it can be seen that the proposed extended Kalman filter achieved better performance in the roll angle and pitch angle than the static mode orientation estimation algorithm and the MATLAB tracking approach. The yaw angle results in Figure 4.8 show that the proposed extended Kalman filter is more sensitive near the actual turning motions than other algorithms and achieves the best performance in orientation estimation under the dynamic mode.

Performance of Position Estimation

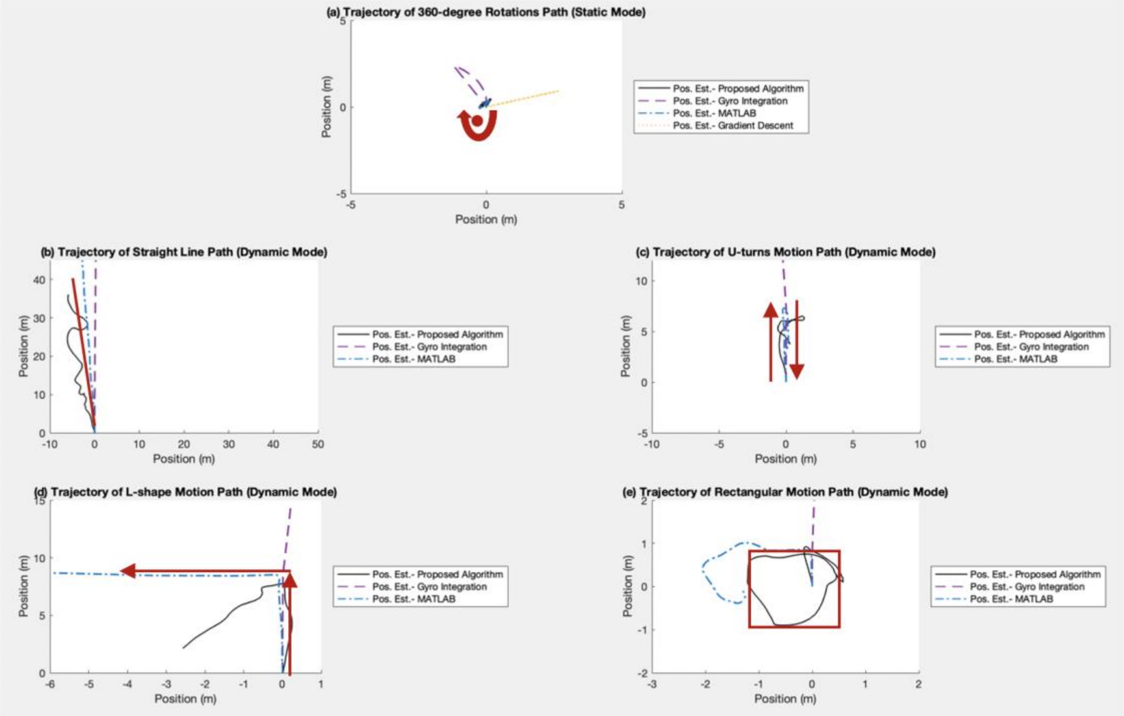

Figure 4.9: Trajectory Tracking Results for Different Path Scenarios

Table 4.4: Displacement Error

Path	**Displacement Error**				
	Static Mode Orientation Est.	Extended Kalman Filter (Full Model)	EKF Prediction Only (Gyro Integration)	MATLAB Tracking Approach	Gradient Descent Optimization Algorithm
(a) 360-degree Rotations	0.1439 m	n/a	3.332 m	0.7378 m	3.541 m
(c) U-turn Motion	n/a	2.3524 m	12.062 m	2.7197 m	12.0658 m
(e) Rectangular Motion	n/a	0.5346 m	4.5946 m	1.304 m	4.5971 m

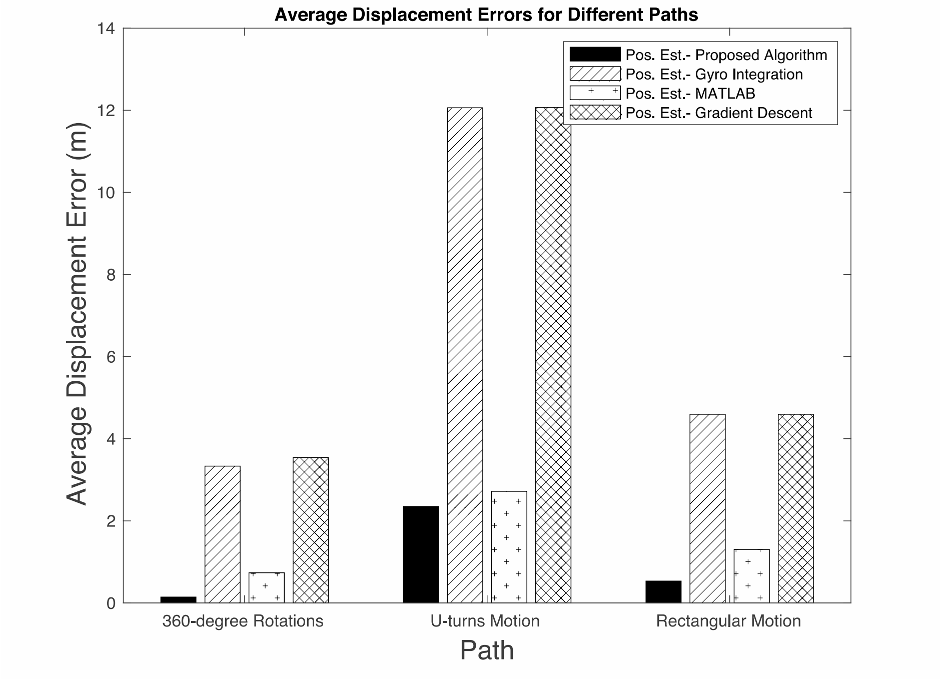

Figure 4.10: Position Estimation Performance Comparison for Different Paths

As shown in Figure 4.1, five motion paths are selected as different experiment scenarios to evaluate the performance and accuracy of the proposed position estimation algorithm and indoor positioning system model. The five motion paths are (a) 360-degree rotations, (b) straight-line motion, (c) U-turns motion, (d) L-shape motion, and (e) rectangular motion. Path (a) is a typical scenario of the static mode, and the other four paths (b), (c), (d), and (e) are considered as scenarios under which the system triggers the dynamic mode algorithms. The pedestrian returns to the starting point for path (a), (c), and (e), so we calculated the average displacement error of these scenarios only. During the evaluation process, four different solutions have been compared. Figure 4.9 depicts the trajectories of the ground truth, the proposed position estimation algorithm, the gyro integration method, the MATLAB tracking approach, and a classical Gradient Descent optimization algorithm for all five experiment scenarios. The reference path in each experimental scenario is shown as the red solid line. The MATLAB tracking approach is considered as a state-of-the-art baseline PDR method. The gyro integration method and the Gradient Descent optimization algorithm serve to evaluate the contribution of sensor fusion techniques of the proposed indoor positioning system. To test the performance of the proposed position estimation algorithm in the static mode, we performed continuous 360-degree rotation motions. Figure 4.9(a) shows that both the proposed algorithm and the MATLAB tracking approach accurately restored the motion path, whereas the other two methods drifted away from the pedestrian's location over time. The reason of the "drifting effect" is that the gyro integration method and Gradient Descent optimization algorithm failed at estimating the correct heading and orientation of the pedestrian. To validate the proposed position estimation algorithm in the dynamic mode, two groups of experiments were carried out with multiple trails. The first group includes the straight-line motion and U-turn motion. In these experimental cases, the pedestrian walked in the corridor in long distance paths. The second group includes the L-shape motion and rectangular motion. These experiments represent the performance of the system for comparatively short distance paths. The experimental results in Figure 4.9 show that the proposed position estimation algorithm and the MATLAB tracking approach demonstrated better performance than the gyro integration method. The only exception is path (d). The proposed algorithm estimated a larger angle than the actual 90-degree turn in the middle of the path, leading to a result that departs from the actual path.

To assess the accuracy of the proposed position estimation algorithm, the displacement errors of paths (a), (c), and (e) are calculated. Concretely, the difference between the actual finishing location and the estimated finishing location is calculated as the displacement error. The displacement errors of different position estimation algorithms under the three experimental scenarios are provided in Table 4.4. For paths (a) and (e), the displacement errors of the proposed algorithm are 0.1439 m and 0.5346 m respectively. The displacement error of the proposed algorithm for path (c) is 2.3524 m. From the table, it can be seen that the proposed algorithm has the lowest displacement error for each of the three paths and achieved a high position estimation accuracy compared to the other methods. Figure 4.10 shows the average displacement errors for different paths. We find that the location errors of the proposed algorithm are significantly less than that of the other conventional methods.

Chapter 5

Conclusions

5.1. Summary

We conclude this thesis by briefly summarizing the research and development project in Section 5.1 and discussing a few directions of future work in Section 5.2 for MEMS-MARG-based dead reckoning for indoor positioning and tracking system. In this thesis, we consider the problem of designing and developing a low-cost and practical indoor positioning system. To avoid complicated infrastructure and extra devices, we proposed a complete indoor positioning solution based solely on self-contained MEMS MARG inertial sensors including accelerometer, magnetometer, and gyroscope embedded in smartphones. PDR, signal processing, and sensor fusion techniques were investigated in this work.

The proposed indoor positioning system includes four main modules, mode detection, step detection and moving distance estimation, heading and orientation estimation, and position estimation, with a mode switch feature and an initialization and calibration phase. A detailed system architecture was presented in Section 3.3. It is robust yet flexible as opposed to conventional indoor positioning algorithms which generate cumulative errors over time. Experimental evaluation has shown that it can achieve high position accuracy compared to the other approaches.

The proposed system has two modes: static mode and dynamic mode. Both modes share a preliminary orientation estimate and then use different ways to correct the preliminary estimate. The dual-mode design reinforces the strength of each heading and orientation estimation algorithm in its corresponding mode and increases the overall computational efficiency of the entire system. The mode detection algorithm with the sliding time window technique was described in Section 3.7. Experimental results demonstrated that the system was able to effectively detect the correct mode with a 2.1% mean relative error. To ensure the reliability of magnetic field measurements, magnetometer calibration was performed in the initialization and calibration phase. The hard-and-soft-iron model based on least square fitting of a deviated ellipsoid is used to eliminate potential magnetic field distortions.

For the step detection module, a multi-layer step detection algorithm was presented and illustrated in Section 3.8. Constraints were imposed as preliminary adjustments of the first-round step detection result. The second layer of step detection utilized gyroscope and magnetometer measurements as step refinement to eliminate false positive steps. Eventually, the third step validation layer further guaranteed all true steps. Experimental results have shown that the proposed step detection module and algorithms can achieve a significant reduction in the number of falsely detected steps.

The heading and orientation estimation module and algorithms were presented in Section 3.9. For both the static mode and dynamic mode, the proposed algorithm first generated a preliminary gyro integration-based orientation estimate. Then if the system was in the static mode, it used reference vectors obtained from accelerometer and magnetometer measurements to correct the preliminary orientation estimate. If the system was in the dynamic mode, a double phase extended Kalman filter was adopted. To address the gimbal lock issue, quaternion was used to represent the system state in the mathematical models. Experimental results demonstrated that the proposed orientation estimation algorithms achieved better performance than the rest of the approaches.

The position estimation module was presented in Section 3.10. Experimental evaluation of five scenarios demonstrated the effectiveness and efficiency of the proposed system. Trajectory results of five sets of experiments have shown that the proposed indoor positioning system and algorithms can track the target pedestrian with high accuracy. For the 360-degree rotations path, the displacement error of the proposed system is 0.1439 m. The displacement errors of the proposed system for U-turn motions and rectangular motions are 2.3524 m and 0.5346 m respectively.

5.2. Future Work

We will continue to improve the performance of the proposed indoor positioning algorithms and develop a full-fledged cross-platform indoor navigation system. One major direction we will focus on is to extend the current system so it can track pedestrians over multiple floors. Moreover, as deep learning algorithms have shown tremendous success in various engineering fields, activity recognition based on Long Short-Term Memory (LSTM) can be incorporated into our model to further improve the positioning accuracy.

We will also extend experimental evaluations to scenarios under which the pedestrian is tracked over a very long period of time such as hours. Last but not least, as estimating heading and orientation is critical in the indoor positioning problem, we are interested in enhancing the current system model by applying convex optimization techniques on orientation estimation, which we believe would mitigate magnetic disturbances in a more effective way.

Bibliography

Abid, M., Renaudin, V., Aoustin, Y., Le-Carpentier, E., and Robert, T. (2017). Walking gait step length asymmetry induced by handheld device. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(11), 2075–2083.

Beauregard, S. and Haas, H. (2006). Pedestrian dead reckoning: A basis for personal positioning. *Proceedings of the 3rd Workshop on Positioning, Navigation and Communication (WPNC'06)*.

Borenstein, J., Ojeda, L., and Kwanmuang, S. (2009). Heuristic reduction of gyro drift in imu-based personnel tracking systems. *Journal of Navigation*, 62, 41 – 58.

Brossard, M., Barrau, A., and Bonnabel, S. (2020). AI-imu dead-reckoning. *IEEE Transactions on Intelligent Vehicles*, 5(4), 585–595.

Chen, P., Kuang, Y., and Chen, X. (2017). A uwb/improved pdr integration algorithm applied to dynamic indoor positioning for pedestrians. *Sensors*, 17, 2065.

Cho, S. Y. and Park, C. G. (2019). Threshold-less zero-velocity detection algorithm for pedestrian dead reckoning. In *2019 European Navigation Conference (ENC)*, pages 1–5.

Dai, S., He, L., and Zhang, X. (2020). Autonomous wifi fingerprinting for indoor localization. In *2020 ACM/IEEE 11th International Conference on Cyber-Physical Systems (ICCPS)*, pages 141–150.

Deng, J., Xu, Q., Ren, A., Duan, Y., Zahid, A., and Abbasi, Q. H. (2020). Machine learning driven method for indoor positioning using inertial measurement unit. In *2020 International Conference on UK-China Emerging Technologies (UCET)*, pages 1–4.

Diallo, A., Lu, Z., and Zhao, X. (2019). Wireless indoor localization using passive rfid tags. *Procedia Computer Science*, 155, 210–217.

Diebel, J. (2006). Representing attitude: Euler angles, unit quaternions, and rotation vectors. *Matrix*, 58.

Díez, L., Bahillo, A., Masegosa, A., Perallos, A., Azpilicueta, L., Falcone, F., Astrain,J., and Villadangos, J. (2015). Signal processing requirements for step detection using wrist-worn imu. In *2015 International Conference on Electromagnetics in Advanced Applications (ICEAA)*, pages 1032–1035.

Fan, Q., Zhang, H., Pan, P., Zhuang, X., Jia, J., Zhang, P., Zhao, Z., Zhu, G., and Tang, Y. (2019). Improved pedestrian dead reckoning based on a robust adaptive kalman filter for indoor inertial location system. *Sensors*, 19, 294.

Fang, J., Sun, H., Cao, J., Zhang, X., and Tao, Y. (2011). A novel calibration method of magnetic compass based on ellipsoid fitting. *IEEE Transactions on Instrumentation and Measurement*, 60(6), 2053–2061.

Forsman, P. M., Toppila, E. M., and Haeggstrom, E. O. (2009). Wavelet analysis to detect gait events. In *2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 424–427.

Fusco, G. and Coughlan, J. (2018). *Indoor Localization Using Computer Vision and Visual-Inertial Odometry*, volume 10897, pages 86–93.

Gartner, G., Frank, A., and Retscher, G. (2003). Pedestrian navigation system for mixed indoor/outdoor environments. *Geowissenschaftliche Mitteilungen*, 66, 161–167.

Giorgi, G., Frigo, G., and Narduzzi, C. (2017). Dead reckoning in structured environments for human indoor navigation. *IEEE Sensors Journal*, 17(23), 7794–7802.

Golestanian, M., Siva, J., and Poellabauer, C. (2017). *Radio Frequency-Based Indoor Localization in Ad-Hoc Networks*.

Graf, B. (2008). Quaternions and dynamics. *arXiv preprint arXiv:0811.2889*.

Hanson, A. (2006). *Visualizing Quaternions (The Morgan Kaufmann Series in Interactive 3D Technology)*.

He, S. and Chan, S.-H. G. (2016). Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys Tutorials*, 18(1), 466–490.

Hellmers, H., Norrdine, A., Blankenbach, J., and Eichhorn, A. (2013). An imu/magnetometer-based indoor positioning system using kalman filtering. pages 1–9.

Herath, S., Yan, H., and Furukawa, Y. (2020). Ronin: Robust neural inertial navigation in the wild: Benchmark, evaluations, amp; new methods. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3146–3152.

Hesch, J. A., Mirzaei, F. M., Mariottini, G. L., and Roumeliotis, S. I. (2010). A laser-aided inertial navigation system (l-ins) for human localization in unknown indoor environments. In *2010 IEEE International Conference on Robotics and Automation*, pages 5376–5382.

Hou, X. and Bergmann, J. (2020). A pedestrian dead reckoning method for head-mounted sensors. *Sensors*, 20, 6349.

Hu, K., Liao, X.-Y., and Yu, M. (2014). Research on indoor localization method based on pdr and wi-fi. In *10th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM 2014)*, pages 653–656.

Huang, S.-Y., Liu, J.-W., Yu, M.-C., and Leu, J.-S. (2019). A pdr-based indoor positioning system in a nursing cart with ibeacon-based calibration. In *2019 IEEE 90th Vehicular Technology Conference (VTC2019-Fall)*, pages 1–4.

Jabbar, M. S., Hussain, G., and Cho, J.-D. (2020). Indoor positioning system: Improved deep learning approach based on lstm and multi-stage activity classification. In *2020 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia*), pages 1–4. IEEE.

Jacobson, N. (2009). Basic Algebra I: Second Edition. Basic Algebra. Dover Publications.

Ji, M., Kim, J., Jeon, J., and Cho, Y. (2015). Analysis of positioning accuracy corresponding to the number of ble beacons in indoor positioning system. In *2015 17th International Conference on Advanced Communication Technology (ICACT)*, pages 92–95.

Jiménez, A., Seco, F., Prieto, J., and Guevara, J. (2010). Indoor pedestrian navigation using an ins/ekf framework for yaw drift reduction and a foot-mounted imu. In *2010 7th Workshop on Positioning, Navigation and Communication*, pages 135–143.

Ju, H. J., Lee, M. S., Park, C. G., Lee, S., and Park, S. (2014). Advanced heuristic drift elimination for indoor pedestrian navigation. In *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 729–732.

Kalman, R. E. (1960). A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1), 35–45.

Kim, P. and Chae, H. (2013). Rigid Body Dynamics for Beginners: Euler Angles &Quaternions. CreateSpace Independent Publishing Platform.

Kok, M., Hol, J. D., and Schön, T. B. (2017). Using inertial sensors for position and orientation estimation. *arXiv preprint arXiv:1704.06053*.

Kone, Y., Zhu, N., Renaudin, V., and Ortiz, M. (2020). Machine learning-based zero-velocity detection for inertial pedestrian navigation. *IEEE Sensors Journal*, 20(20), 12343–12353.

Kunhoth, J., Karkar, A., Al-maadeed, S., and Al-Ali, A. (2020). Indoor positioning and wayfinding systems: a survey. *Human-centric Computing and Information Sciences*, 10.

Kuxdorf-Alkirata, N., Kasolis, F., Bruckmann, D., and Koch, O. (2019). Linear error modeling and noise smoothing for improved low-cost imu-based indoor positioning. In *2019 IEEE 62nd International Midwest Symposium on Circuits and Systems (MWSCAS)*, pages 1069–1072.

Lee, G. T., Seo, S. B., and Jeon, W. S. (2021). Indoor localization by kalman filter based combining of uwb-positioning and pdr. In *2021 IEEE 18th Annual Consumer Communications Networking Conference (CCNC)*, pages 1–6.

Li, X., Wang, J., and Liu, C. (2015). A bluetooth/pdr integration algorithm for an indoor positioning system. *Sensors*, 15, 24862–24885.

Li, X., Wang, Y., and Khoshelham, K. (2018). Uwb/pdr tightly coupled navigation with robust extended kalman filter for nlos environments. *Mobile Information Systems*, 2018, 1–14.

Li, Y., Zhuang, Y., Lan, H., Zhou, Q., Niu, X., and El-Sheimy, N. (2016). A hybrid wifi/magnetic matching/pdr approach for indoor navigation with smartphone sensors. *IEEE Communications Letters*, 20(1), 169–172.

Lu, C., Uchiyama, H., Thomas, D., Shimada, A., and Taniguchi, R. (2019). Indoor positioning system based on chest-mounted imu. *Sensors*, 19, 420.

Ma, R., Guo, Q., Hu, C., and Xue, J. (2015). An improved wifi indoor positioning algorithm by weighted fusion. *Sensors (Basel, Switzerland)*, 15, 21824–43.

Madgwick, S. O. H., Harrison, A. J. L., and Vaidyanathan, R. (2011). Estimation of imu and marg orientation using a gradient descent algorithm. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–7.

Madgwick, S. O. H., Wilson, S., Turk, R., Burridge, J., Kapatos, C., and Vaidyanathan, R. (2020). An extended complementary filter for full-body marg orientation estimation. *IEEE/ASME Transactions on Mechatronics*, 25(4), 2054–2064.

Mahony, R., Hamel, T., and Pflimlin, J.-M. (2008). Nonlinear complementary filters on the special orthogonal group. *IEEE Transactions on Automatic Control*, 53(5), 1203–1218.

Mansur, V., Reddy, S., R, S., and Sujatha, R. (2020). Deploying complementary filter to avert gimbal lock in drones using quaternion angles. In *2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON)*, pages 751–756.

Merenda, M., Catarinucci, L., Colella, R., Della Corte, F. G., and Carotenuto, R. (2021). Exploiting rfid technology for indoor positioning. In *2021 6th International Conference on Smart and Sustainable Technologies (SpliTech)*, pages 1–5.

Mikov, A., Moschevikin, A., and Voronov, R. (2020). Vehicle dead-reckoning autonomous algorithm based on turn velocity updates in kalman filter. In *2020 27th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*, pages 1–5.

Murray, R., Li, Z., and Sastry, S. (1994). A mathematical introduction to robot manipulation. 29.

NAIF (2020). An overview of reference frames and coordinate systems in the SPICE context.

Nazemzadeh, P., Fontanelli, D., Macii, D., and Palopoli, L. (2017). Indoor localization of mobile robots through qr code detection and dead reckoning data fusion. *IEEE/ASME Transactions on Mechatronics*, 22(6), 2588–2599.

Nessa, A., Adhikari, B., Hussain, F., and Fernando, X. N. (2020). A survey of machine learning for indoor positioning. *IEEE Access*, 8, 214945–214965.

Othman, R., Gaafar, A., Muaaz, L., and Elsayed, M. (2021). A hybrid rss+aoa indoor positioning algorithm based on visible light communication. pages 1–5.

Ott, B., Wasle, E., Weimann, F., Branco, P., and Nicole, R. (2005). Pedestrian navigation in difficult environments - results of the esa project shade.

Ozyagcilar, T. (2013). Application note: calibrating an ecompass in the presence of hard- and soft-iron interference. Freescale Semiconductor Inc.

Ozyagcilar, T. (2015). Application note: implementing a tilt-compensated ecompass using accelerometer and magnetometer sensors. Freescale Semiconductor Inc.

Ozyagcilar, T. (2015). Application note: calibrating an ecompass in the presence of hard-and soft-iron interference. Freescale Semiconductor Inc.

Poulose, A. and Han, D. S. (2019). Indoor localization using pdr with wi-fi weighted path loss algorithm. *In 2019 International Conference on Information and Communication Technology Convergence (ICTC)*, pages 689–693.

Poulose, A., Eyobu, O. S., and Han, D. S. (2019). An indoor position-estimation algorithm using smartphone imu sensor data. *IEEE Access*, 7, 11165–11177.

Premerlani, W. and Bizard, P. (2009). Direction cosine matrix imu: Theory. *DIY DRONE: USA*.

Qi, J. and Liu, G.-P. (2017). A robust high-accuracy ultrasound indoor positioning system based on a wireless sensor network. *Sensors*, 17, 2554.

Qiu, S., Wang, Z., Zhao, H.-Y., Qin, K., Li, Z., and Hu, H. (2018). Inertial/magnetic sensors based pedestrian dead reckoning by means of multi-sensor fusion. *Information Fusion*, 39, 108–119.

Sabatelli, S., Galgani, M., Fanucci, L., and Rocchi, A. (2013). A double-stage Kalman filter for orientation tracking with an integrated processor in 9-d imu. *IEEE Transactions on Instrumentation and Measurement*, 62(3), 590–598.

Sant'Anna, A. and Wickström, N. (2010). A symbol-based approach to gait analysis from acceleration signals: Identification and detection of gait events and a new measure of gait symmetry. *IEEE Transactions on Information Technology in Biomedicine*, 14(5), 1180–1187.

Sayed, W., Ismail, T., and Elsayed, K. (2019). A neural network-based vlc indoor positioning system for moving users. In *2019 International Conference on Smart Applications, Communications and Networking (SmartNets)*, pages 1–6.

Schroeer, G. (2018). A real-time uwb multi-channel indoor positioning system for industrial scenarios. In *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–5.

Silhavy, R., Senkerik, R., Oplatkova, Z. K., Prokopova, Z., and Silhavy, P. (2017). *Cybernetics and Mathematics Applications in Intelligent Systems: Proceedings of the 6th Computer Science On-line Conference 2017 (CSOC2017), Vol 2*, volume 574. Springer.

Soken, H. and Sakai, S.-I. (2015). Magnetometer calibration for advanced small satellite missions.

Titterton, D. and Weston, J. (2005). Strapdown inertial navigation technology - 2nd edition. *Aerospace and Electronic Systems Magazine, IEEE,* 20, 33– 34.

Wang, L., Tiku, S., and Pasricha, S. (2021). Chisel: Compression-aware high-accuracy embedded indoor localization with deep learning. *IEEE Embedded Systems Letters*,pages 1–1.

Wang, R., Wan, W., Di, K., Chen, R., and Feng, X. (2019). A high-accuracy indoor positioning method with automated rgb-d image database construction. *Remote Sensing*, 11, 2572.

Wann, C.-D., Yeh, Y.-J., and Hsueh, C.-S. (2006). Hybrid tdoa/aoa indoor positioning and tracking using extended kalman filters. In *2006 IEEE 63rd Vehicular Technology Conference*, volume 3, pages 1058–1062.

Wu, F., Liang, Y., Fu, Y., and Ji, X. (2016). A robust indoor positioning system based on encoded magnetic field and low-cost imu. In *2016 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, pages 204–212.

Yan, H., Shan, Q., and Furukawa, Y. (2017). Ridi: Robust imu double integration.

Yang, B., Dai, C., Ye, H., and Long, Z. (2020). Research on high precision indoor positioning method based on low power bluetooth technology. In *2020 6th International Conference on Big Data and Information Analytics (BigDIA)*, pages 133–137.

Yang, C., Huang, Y., and Zhu, X. (2008). Hybrid tdoa/aoa method for indoor positioning systems. pages 1 – 5.

Yun, X., Bachmann, E. R., Moore, H., and Calusdian, J. (2007). Self-contained position tracking of human movement using small inertial/magnetic sensor modules. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 2526–2533.

Zhang, R., Gao, S., and Cai, X. (2019). Modeling of mems gyro drift based on wavelet threshold denoising and improved elman neural network. In *2019 14th IEEE International Conference on Electronic Measurement Instruments (ICEMI)*, pages 1754–1761.

Zhang, W., Wei, D., and Yuan, H. (2020). The improved constraint methods for foot-mounted pdr system. *IEEE Access*, 8, 31764–31779.