Dead Reckoning Methods For Pedestrian Navigation

Deepak Kiran BS

Master thesis presented for the degree of
Masters in Systems Control and Mechatronics

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Signals and Systems
Chalmers University of Technology
Gothenburg, Sweden

Dead Reckoning Methods For Pedestrian Navigation

-Deepak Kiran BS

Supervisor:
David Törnqvist
- Senion AB

Examiner
Lars Hammarstrand
-Chalmers University Of Technology

Master's Thesis EX036/2017
Department of Signals and Systems
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset: LaTeX
Linköping, Sweden

Abstract

The growing size of modern infrastructures and the ubiquitous presence of smartphones has brought into place indoor maps and navigation systems that facilitate indoor navigation for pedestrians. Indoor navigation methods rely on two main research approaches; solutions that depend on radio frequencies and solutions that depend on dead reckoning methods with sensors installed on the pedestrian to be tracked. Dead reckoning methods are useful to extend navigation into areas where other navigation systems are unavailable or to improve the accuracy of the existing navigation methods. With the increased offerings of sensors in smartphones several pedestrian dead reckoning **(PDR)** methods based on Inertial Measurement Units (IMUs) have been proposed, these methods have practical limitations restricting them to be applied in all scenarios. Thus, there is a need to explore this area and develop methods that can be used in all practical scenarios.

PDR based on hand-held IMUs in devices like smartphones, has been dealt in the thesis. It is desired to track the path of a pedestrian by integrating his step lengths along the respective heading directions. The heading direction (or, heading angle) estimation is a well known problem for PDR based on hand-held devices (smartphones). As it is desired to estimate the heading of the pedestrian based on IMUs in a device, the heading angle indicated by the device is dependent on its orientation and independent of the heading of the pedestrian. In such a case, the heading of the pedestrian cannot be directly obtained from the orientation of the device and there exists an angular deviation between the direction indicated by the device and the pedestrian, which is referred to as the **crab angle**; this angle needs to be taken into account to estimate the heading angle of the pedestrian. The problem of estimating this heading angle irrespective of the orientation of the device has been the focus of the thesis. A novel method to estimate the heading angle along with the position of the pedestrian has been proposed by implementing an extended Kalman filter.

The proposed method has been tested under various scenarios for numerous walking paths, such that the orientation of the device changes multiple times during each path taken. The results obtained by implementing the proposed method "passes" all the test scenarios by closely tracking all the reference trajectories. Further the close tracking of the reference trajectories, proves that the pedestrian heading angle independent of the orientation of the device has been obtained by using the IMU data.

Keywords: Pedestrian dead reckoning, pedestrian heading angle, crab angle, smartphone heading angle, smartphone orientation estimation, step detection, inertial measurement units, extended Kalman filter

Acknowledgements

I would like to heartily thank and express my sincere gratitude to everyone without whom it would not have been possible to complete this thesis.

I am grateful to Senion AB for providing me this wonderful opportunity to carry out my Master thesis with them. First and foremost, I would like to thank my supervisor David Törnqvist at Senion AB, for continuously guiding me and supporting me throughout the thesis. David was available all the time for any sort of help and has answered all my queries and provided me with his meticulous thoughts for all the questions that I had. Thank you David!

I would like to thank Per Skoglar at Senion AB for his valuable inputs that helped me narrow down complicated thoughts to simple approaches. David and Per have reviewed my report multiple times and have provided their inputs to make it simple and clear. Thank you both!

I would like to thank Lars Hammarstrand, my examiner at Chalmers. All my discussions with him about the approaches in the thesis have helped me eliminate redundant methods and stick to rigorous methods. I am grateful to him for all the time he has taken to carefully listen to my approaches and steer me to a rigorous path.

Further, I would like to thank my thesis opponents, Lars Sjöberg and Sangeetha Nagaraju for their opposition and also for their valuable feedback on the report. Also, I would like to thank Rasmus Vilhelmsson, my fellow colleague at Senion AB for all the discussions, and tips on latex during the thesis.

Finally, I would like to thank my family and friends for believing in me and their constant support throughout.

Deepak Kiran
Linköping, Sweden

Contents

List of Figures

Chapter 1

Introduction

The growing size of modern infrastructures like airports, shopping malls, hospitals, hotels *etc.* and the ubiquitous presence of smartphones has brought into place indoor maps and navigation systems that facilitate indoor navigation or localization of pedestrians. Accurate indoor localization has the potential to transform the way people navigate indoors in a similar way that GPS transformed the way people navigate outdoors. Consequently indoor localization can benefit location-aware applications such as, intelligent spaces, personal or asset tracking and guidance of persons with mobility problems [8].

Indoor navigation methods rely mostly on 2 main research approaches: 1) Solutions that depend on radio frequencies *i.e* a network of emitters and receivers placed at known locations (beacon based systems); and 2) Solutions that depend on dead reckoning methods with sensors installed on the person to be tracked(beacon free systems) [8]. The beacon based systems perform localization using triangulation from a set of measured ranges or angles, and they depend on different wireless technologies such as cellular, Wi-Fi, ultra-wide band (UWB), radio-frequency identification (RFID), Bluetooth *etc.* Whereas, the beacon free approach or the dead-reckoning method perform localization by calculating one's current position by using a previously determined position and advancing that position based upon known or estimated speeds over elapsed time and course. Dead reckoning methods are useful in cases where there is no external infrastructure available and pedestrian navigation that does not rely on external aids is referred to as pedestrian dead reckoning (PDR) [12]. Over the last 15 years, several pedestrian dead reckoning methods based on Inertial Measurement Units (IMUs) have been proposed and these methods have practical limitations restricting them to be applied in all scenarios. Thus, there is a need to explore this area and develop methods that can be used in all practical scenarios.

Pedestrian dead reckoning using IMUs is used to estimate the movement of a person, by detecting steps, estimating stride lengths and the directions of motion. Estimating the direction of motion is a well known challenging problem in the field of location based applications [7]. One of the way to implement IMU based PDR is through hand-held devices like smart phones, and the challenge in implementing this PDR is that the IMUs are located in a smartphone and obtaining the direction of motion of the pedestrian depends very strongly on the orientation of the smartphone with respect to the walking direction. In this thesis PDR based on hand-held devices shall be dealt with, and the objective of the thesis is to obtain accurate direction of motion irrespective of the orientation of the device to estimate the position of the pedestrian.

1.1 Background

An IMU is an electronic device that measures and reports a body's specific force, angular rate and the magnetic field surrounding the body, using a combination of accelerometers, gyroscopes and magentometers respectively. IMU based PDR is done by integrating the step lengths and heading angles at each detected step to estimate the user's position. It can be carried out in various ways, few of which are stated below:

- *PDR based on foot-mounted IMU:* The IMU mounted on the foot gives an advantage of accurate step detection, which improves the efficiency of estimating the positions [2]. Measurement of the foot's acceleration allows the precise identification of separate stride segments, thus providing improved step length estimation [10]
- *PDR based on head-mounted IMU:* Emerging wearable technologies like Google glass, offer new sensor placement options on the human body, where the accelerometer data is used to predict steps and the direction of motion is provided by a magnetic compass [13].
- *PDR based on hand-held IMU:* With built-in IMUs in smartphones, this has become the most common and popular way for PDR. Accelerometers can be used as a pedometer and built-in magnetometer as a compass heading provider. In a trivial implementation, the pedestrian holds his phone in front of him and each step causes position to move forward a fixed distance in the direction measured by the compass.

PDR based on hand-held devices like smartphones shall be dealt in the thesis. In the most practical scenario, the heading of the device is not the same as the heading of the user, as the IMUs are located inside a phone, its orientation can change based on the hand movements of the pedestrian. Hence the heading direction indicated by the IMU (or the smartphone) may not be the same as the walking direction. In such a case the heading of the pedestrian cannot be directly obtained from the heading of the device and there exists an angular deviation between the direction indicated by the device and the pedestrian, which is referred to as the **crab angle**, that needs to be taken into account to estimate the heading angle of the pedestrian. Thus, this situation surfaces the necessity to estimate 2 quantities, 1) The orientation of the device through an orientation filter; 2) The angular deviation between the heading of the phone and the pedestrian, or the **crab angle**. Estimating these quantities using the IMU data helps in estimating the pedestrian heading angle.

1.2 Thesis Objective and Problem Formulation

The objective of the thesis is to obtain the position and heading angle of a pedestrian, with an IMU based, hand-held device like smartphone. In order to achieve this the following sub problems shall be looked into:

1. An orientation filter to estimate the orientation of the device shall be implemented.
2. A step detector, to detect steps shall be implemented.
3. A step length estimator shall be implemented.
4. The crab angle shall be estimated.
5. The heading angle of the pedestrian shall be estimated.

6. Step lengths shall be integrated along the respective heading directions to estimate positions in a global frame of reference.

The various calculations involved to estimate the position and heading angle are from the IMUs that are noisy, hence the estimation problem can be set-up as a stochastic filtering problem, with a well defined motion model, measurement model and tuned covariance matrices for the uncertain states. Integrating the step lengths and the heading angle to obtain the positions is a non linear motion model, hence a non linear filter such as an extended Kalman filter shall be implemented to form the estimates.

1.3 Scope and Limitations

PDR methods themselves are not highly accurate and in general they can be used to extend navigation into areas where other navigation systems are unavailable, or they are used in conjunction with beacon based approaches to improve the accuracy of navigation.

The accuracy of PDR methods is limited by accurate step detections, accurate step length estimation and accurate heading angle estimation. Another challenge is differentiating walking from running, and recognizing movements like bicycling, climbing stairs, or riding an elevator. Further accuracy is limited by the sensor precision and magnetic disturbances inside structures which affect the readings from the magnetometer.

1.4 Related Works

As already mentioned PDR is based on step detection, step length estimation and pedestrian heading angle estimation. Significant works have been done in all of these areas. Below is a brief review of few works that are relevant in the context of the thesis; as well as works that are actually considered in the thesis.

1.4.1 Step Detection

In [14] the authors have described a robust step detection method that uses the signals from the accelerometer, magnetometer and gyroscope of the smartphone to transform the device reference frame to the earth reference frame, and then employs the vertical acceleration to implement the step detection. In [4] the author employs band pass filtering of accelerometer signals with cut off frequencies that are within the range of normal walking speed and frequencies. This step detection algorithm works in most of the normal walking cases and this method has been considered in the thesis owing to its light-weight, simplicity and accurate results.

1.4.2 Step Length Estimation

In [11], the authors have presented a step length estimation model based on accelerometer signals, the paper reveals the relation of frequency and the variance of accelerations to step length. Most

other approaches as in[3] and [6] have a data set of step lengths, height and weight of various people and these data sets are used to train neural networks and then estimate the step length based on the height and weight of the pedestrian. In the scope of the thesis, since the PDR methods developed are to be used in conjunction with radio beacons, hence it is just sufficient to assume an average step length for all pedestrians with a certain standard deviation.

1.4.3 Heading Angle Estimation

In [7], a Master thesis by *Nirupam Roy*, a method to estimate heading direction of a pedestrian has been presented by analyzing the acceleration and deceleration signals during each step. The direction is obtained by projecting the accelerometer data in the phone onto a horizontal plane and since these forces are due to the legs walking in the horizontal plane, finding the resultant force direction using the projected data gives the direction of the pedestrian. The limitation of this method is that the heading vector determined by the algorithm gives the direction of the walk in the coordinate system of the smartphone and estimating the heading direction due to orientation changes increases the error of estimation. Further in the context of the thesis this paper also serves as a basis to understand the bio mechanics of walking and gives a clear idea regarding how step detection can be done. In [1] a paper by *Zhi-An Deng,* the heading has been obtained by using orientation estimate of the device and accurately projecting the accelerometer data in the phone co ordinates to a fixed coordinate system X_l, Y_l in the ground plane. Known the magnitudes of acceleration along X_l and Y_l, enables one to find the walking direction. This method has shown promising results, but sometimes falls short due to gyroscopic drifts. In [5] a paper by *Valérie Renaudin.et al* a quaternion based approach using the gyroscope accelerometer and magnetometer is presented, in this approach the heading direction in the global co ordinates is obtained using the heading information from accelerometer and magnetometer, but assumes that the phone does not change orientations as frequently which is not a most practical case to deal with.

All the heading angle estimation methods stated above fall short in estimating the heading angle accurately due to orientation changes of the device (smartphone). In this thesis it is desired to obtain accurate heading angle irrespective of the orientation of the device. In order to achieve this a heading angle method patented by Senion AB [12] shall be used and evaluated. Based on the calculations presented in the patent, the heading direction can be obtained in 2 different ways both of which are briefly discussed below:

Method-1

In the first approach the heading angle is obtained by summing up the heading of the phone and the crab angle. Fusion of gyroscope, accelerometer and magnetometer readings gives the heading of the phone and the crab angle is calculated by using the accelerometer data.

Method-2

In the second approach the heading direction shall be obtained by projecting the accelerometer data onto the horizontal plane and known the magnitude of acceleration in a coordintate system the heading angle can be obtained. This method is very similar to the works presented by *Zhi-An Deng, et al.* in [1], except that the method of calculation is different. Other than that the idea and approach of obtaining the heading angle is the same. But for the method presented in this thesis the calculations have been taken from the patent developed by Senion AB [12].

1.5 Major Contributions

The thesis is centered on implementing a PDR for hand-held devices. In doing so the following sub problems have been solved:

- Modelling the PDR based on the orientation of the phone and the crab angle.
- Formulating the problem as a stochastic filtering problem and implementing an EKF thereby modelling a motion model, a measurement model, and tuning the co variance matrices for the uncertain states.
- Collecting data for various kinds of situations that would arise when a pedestrian walks with a smart phone in his hand. When a person walks with the phone in his hand, he can change the direction of his path or just change the orientation of the phone and not his direction or do both *i.e.* change both direction and orientation. Data for all such cases for different paths has been collected.
- Generating and analyzing the results for various kinds of data collected.

1.6 Thesis Outline

The remainder of the thesis is structured as follows. In Chapter 2 the theory required to implement pedestrian navigation algorithms are described; the main parts of this chapter cover stochastic filtering and device orientation estimation. Next, in Chapter 3 the concepts central to PDR, which are step detection, step length and heading angle estimation are discussed, which is followed by Chapter 4 that covers the implementation of a couple of dead reckoning methods based on the concepts described in Chapter 3. After the dead reckoning methods are presented, the results and discussions based on both the methods are presented in Chapter 5, which is followed by a comparison of the methods and results in Chapter 6. Chapter 7 then discusses the limitations and the possible future work of the thesis, followed by concluding remarks in Chapter 8.

Chapter 2

Theory

This chapter introduces the theory that is necessary to solve the PDR problem. The chapter focuses on the concepts of stochastic filtering, device orientation estimate and particularly how to obtain, roll and pitch angles of a device.

2.1 Stochastic Filtering

Filtering is about recursively estimating parameters of interest based on measurements. In the theory of stochastic processes, filtering problem is a mathematical model for a number of state estimation problems in signal processing and related fields. The idea is to obtain a "best estimate" for the true value of particular system from an incomplete, noisy set of measurements on that system. Based on the linear or non linear nature of the system dynamics, the filtering techniques vary. For a system with linear dynamics, Kalman filter is the optimal recursive filter and for non linear dynamics the derivatives of Kalman filter are used based on the application to obtain the "best estimates".

2.1.1 Kalman Filter

Kalman filtering, is an algorithm that uses a series of measurements observed over time, containing statistical noise and other inaccuracies, and produces estimates of unknown variables that tend to be more accurate than those based on a single measurement alone or based on the dynamics model alone. The estimations are based on a motion model (dynamics of the system), measurement model and observations.

A motion model that is linear with additive Gaussian noise can be represented by the following general equation:

$$x_k = A_{k-1}x_{k-1} + G_{k-1}e_{k-1} \tag{2.1}$$

A measurement model that is linear with additive Gaussian noise can be represented by the following general equation:

$$y_k = H_k x_k + L_k m_k \tag{2.2}$$

where, x_k is a vector of states to be estimated, A_{k-1} is a transition matrix, e_{k-1} is a Gaussian noise with zero mean denoted by $\hat{e}_{k-1}$and covariance E_{k-1}, G_{k-1} is the matrix that scales the noise vector

e_{k-1}, y_k is the measurement prediction, H_k is the measurement model matrix, m_k is a Gaussian noise with zero mean denoted by $\hat{m}_k$ and covariance M_k and L_k is the matrix that scales the noise vector m_k

For the motion and measurement models in (2.1) and (2.2) the Kalman filter estimates are given by the following prediction and update steps:

Prediction Step:

$$\begin{aligned} \hat{x}_{k|k-1} &= A_{k-1}\hat{x}_{k-1|k-1} \\ P_{k|k-1} &= A_{k-1}P_{k-1|k-1}A_{k-1}^T + G_{k-1}E_{k-1}G_{k-1}^T \end{aligned} \tag{2.3}$$

Update Step:

$$\begin{aligned} v_K &= y_k - H_k\hat{x}_{k|k-1} \\ K_k &= P_{k/k-1}H_k^T S_k^{-1} \\ S_k &= H_k P_{k|k-1}H_K^T + L_k M_k L_k^T \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k v_K \\ P_{k|k} &= P_{k|k-1} - K_k S_k K_k^T \end{aligned} \tag{2.4}$$

where $\hat{x}_{k-1|k-1}$ are the estimated states at time step $k-1$, $P_{k-1|k-1}$ is the estimated covariance of the states x_{k-1} at time step $k-1$, $\hat{x}_{k|k-1}$ are the predicted states, $P_{k|k-1}$ is the covariance in the prediction, $\hat{x}_{k|k}$ are the estimated states, $P_{k|k}$ is the covariance in the estimated state, K_k is known as the kalman gain, v_k is known as the innovation, S_k is the predicted covariance of y_k.

The Kalman filter estimates above are for special case of linear and Gaussian Models. Non Linear models are handled differently. There are numerous ways to perform non linear filtering, like Extended kalman Filter(EKF), Unscented Kalman Filter, Particle filters *etc.* The non linear filtering technique used in the thesis is an Extended Kalman filter which shall be discussed in the next subsection.

2.1.2 Extended Kalman Filter (EKF)

An Extended Kalman filter can be implemented for non linear motion and measurement models. The general form of non linear models are shown in equation (2.5)

$$\begin{aligned} x_k &= f(x_{k-1}, e_{k-1}) \\ y_k &= h(x_k, m_k) \end{aligned} \tag{2.5}$$

where, f_{k-1} is a non linear function of the states and motion model noise e_{k-1} and h_k is a non linear function of the states and measurement model noise m_k.

The approach of filtering the states in an EKF is by linearizing the non linear models. The linearization of the motion model can be carried out at $(\hat{x}_{k-1|k-1}, \hat{e}_{k-1})$, after which a prediction of the states $\hat{x}_{k|k-1}$ can be obtained. Using these predicted states the Measurement model can be linearized at $(\hat{x}_{k/k-1}, \hat{m}_k)$.

The linearization of the motion and measurement models is shown below

$$\begin{aligned} x_k &\approx f(\hat{x}_{k-1|k-1}, \hat{e}_{k-1}) + f_x(x_k - \hat{x}_{k-1|k-1}) + f_e(e_{k-1} - \hat{e}_{k-1}) \\ y_k &\approx h(\hat{x}_{k|k-1}, \hat{m}_k) + h_x(x_k - \hat{x}_{k|k-1}) + h_m(m_k - \hat{m}_k) \end{aligned} \tag{2.6}$$

where

$$f_x = \frac{\partial f}{\partial x}\bigg|_{\hat{x}_{k-1|k-1}, \hat{e}_{k-1}}, f_e = \frac{\partial f}{\partial e}\bigg|_{\hat{x}_{k-1|k-1}, \hat{e}_{k-1}}, h_x = \frac{\partial h}{\partial x}\bigg|_{\hat{x}_{k|k-1}, \hat{m}_k}, h_m = \frac{\partial h}{\partial m}\bigg|_{\hat{x}_{k|k-1}, \hat{m}_k}$$

After the linearization the estimates of states can be obtained by using the prediction and update steps of the EKF as shown below.

Prediction Step:

$$\begin{aligned} \hat{x}_{k|k-1} &= f(\hat{x}_{k-1|k-1}) \\ P_{k|k-1} &= f_x P_{k-1|k-1} f_x^T + f_e E_{k-1} f_e^T \end{aligned} \tag{2.7}$$

Update Step:

$$\begin{aligned} v_K &= y_k - h(\hat{x}_{k|k-1}, \hat{m}_k) \\ K_k &= P_{k|k-1} h_x^T S_k^{-1} \\ S_k &= h_x P_{k|k-1} h_x^T + h_m M_k h_m^T \\ \hat{x}_{k|k} &= \hat{x}_{k|k-1} + K_k v_K \\ P_{k|k} &= P_{k|k-1} - K_k S_k K_k^T \end{aligned} \tag{2.8}$$

2.2 Device Orientation Estimate

The data for dead reckoning is obtained from an IMU whose readings vary with the orientation of the device hence it becomes necessary to estimate the orientation of a device. The orientation shall be obtained by modelling the rotations with quaternions. It is sufficient to obtain the orientation estimate with respect to the horizontal plane of the earth using only the gyroscope and accelerometer data. This section gives a brief overview of modelling the orientation with the quaternions.

2.2.1 Quaternions

The quaternions are a number system that extend the complex numbers. They are generally represented by the following form.

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \tag{2.9}$$

The normalized quaternions are referred to as unit quaternions, and when unit quaternions are used to represent rotations they are referred to as rotation quaternions. Any rotation in three dimensions can be represented as a combination of a vector $\bar{u}$ and a scalar θ, where $\bar{u}$, known as the *Euler-axis*, is a vector about which a rigid body or a coordinate system is rotated by an angle θ. Quaternions give a simple way to encode this axis–angle representation in four numbers. A rotation through an angle of θ around the axis defined by a unit vector $\bar{u}$ can be represented by a quaternion by the extension of the *Euler's formula* as shown in (2.10)

$$\begin{aligned} \bar{u} &= u_x\mathbf{i} + u_y\mathbf{j} + u_z\mathbf{k} \\ q &= e^{\frac{\theta}{2}(u_x\mathbf{i}+u_y\mathbf{j}+u_z\mathbf{k})} \\ q &= \cos(\frac{\theta}{2}) + (u_x\mathbf{i} + u_y\mathbf{j} + u_z\mathbf{k})\sin(\frac{\theta}{2}) \end{aligned} \tag{2.10}$$

2.2.2 Modelling the Orientation

In order to obtain the orientation estimate, the gyroscopic readings can be integrated to obtain the rotation represented by quaternions. The rotation model using gyroscopic data is given in (2.11) where q represents the quaternions ω represents the gyroscopic readings and v represents the Gaussian white noise in the gyroscopic readings with zero mean and covariance V_w

$$\begin{aligned} \dot{q}(t) &= S(w+v)q(t) \\ S(w_{k-1}) &= \begin{bmatrix} 0 & -\omega^x & -\omega^y & \omega^z \\ \omega^x & 0 & \omega^z & \omega^y \\ \omega^y & -\omega^z & 0 & \omega^x \\ \omega^z & \omega^y & -\omega^x & 0 \end{bmatrix} \end{aligned} \tag{2.11}$$

The above equation (2.11) is a continuous differential equation in time. Since the gyroscopic measurements are obtained as discrete samples from the IMU, it is necessary to discretize this equation. The discretized equation is shown below in equation 2.12 and the discretization is shown in the Appendix A.1

$$\begin{aligned} q_k &= F(w_{k-1})q_{k-1} + G(q_{k-1})v_{k-1} \\ F(w_{k-1}) &= \begin{bmatrix} 1 & -\frac{T}{2}\omega^x_{k-1} & -\frac{T}{2}\omega^y_{k-1} & -\frac{T}{2}\omega^z_{k-1} \\ \frac{T}{2}\omega^x_{k-1} & 1 & \frac{T}{2}\omega^z_{k-1} & -\frac{T}{2}\omega^y_{k-1} \\ \frac{T}{2}\omega^y_{k-1} & -\frac{T}{2}\omega^z_{k-1} & 1 & \frac{T}{2}\omega^x_{k-1} \\ \frac{T}{2}\omega^z_{k-1} & \frac{T}{2}\omega^y_{k-1} & -\frac{T}{2}\omega^x_{k-1} & 1 \end{bmatrix} \\ G(q_{k-1}) &= \frac{T}{2}\begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & -q_1 & -q_0 \end{bmatrix} \end{aligned} \tag{2.12}$$

Since the gyroscopic measurements have an error and with integration the error grows with time, hence it is desired to correct this drift using the accelerometer measurements by implementing an Extended Kalman Filter (EKF). The measurement model is given below in (2.13) where R_q is the rotation matrix, g^0 is the nominal gravity vector and m_k is the measurement model noise which is Gaussian with zero mean and covariance matrix M_a. By implementing this filter the angular drifts in the roll and pitch shall be corrected but the angular drift in the yaw would still remain uncorrected.

$$
\begin{aligned}
y_k^a &= R_q^T(q_k)g^0 + m_k \\
R_q &= \begin{pmatrix} 2\,{q_0}^2 + 2\,{q_1}^2 - 1 & 2\,q_1\,q_2 - 2\,q_0\,q_3 & 2\,q_0\,q_2 + 2\,q_1\,q_3 \\ 2\,q_0\,q_3 + 2\,q_1\,q_2 & 2\,{q_0}^2 + 2\,{q_2}^2 - 1 & 2\,q_2\,q_3 - 2\,q_0\,q_1 \\ 2\,q_1\,q_3 - 2\,q_0\,q_2 & 2\,q_0\,q_1 + 2\,q_2\,q_3 & 2\,{q_0}^2 + 2\,{q_3}^2 - 1 \end{pmatrix} \\
g^0 &= \begin{pmatrix} 0 \\ 0 \\ 9.81 \end{pmatrix}
\end{aligned} \tag{2.13}
$$

With the orientation models given in (2.12) and (2.13) the orientation estimate can be obtained by implementing an EKF as described in section 2.1.2. The EKF implementation for the orientation estimate is shown in Appendix A.2

2.3 Estimation of Roll and Pitch of the Device

In the above section the orientation estimates provide the information of the roll pitch and yaw angles. In order to find the horizontal plane, it may just be sufficient to know the roll and pitch angles. One of the way to obtain them is by converting the quaternions to Euler angles. The Euler angles that represent the roll pitch and yaw angles are commonly referred to as **Tait Byran** angles. These angles can be obtained from the quaternions estimated in section 2.2 by using the equation (2.14)

$$
\begin{bmatrix} \theta_r \\ \theta_p \\ \theta_y \end{bmatrix} = \begin{bmatrix} \text{atan2}(2(q_0q_1 + q_2q_3), 1 - 2(q_1^2 + q_2^2)) \\ \arcsin(2(q_0q_2 - q_3q_1)) \\ \text{atan2}(2(q_0q_3 + q_1q_2), 1 - 2(q_2^2 + q_3^2))) \end{bmatrix} \tag{2.14}
$$

where, θ_r is the roll, θ_p is pitch and θ_y is yaw. Only roll and pitch angles are of interest from the above equation.

Further the rotation matrix formed using only the roll and pitch angles is shown in (2.15)

$$
R_e(\theta_p, \theta_r) = \begin{bmatrix} \cos(\theta_p) & 0 & \sin(\theta_p) \\ 0 & 1 & 0 \\ -\sin(\theta_p) & 0 & \cos(\theta)p) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_r) & -\sin(\theta_r) \\ 0 & \sin(\theta_r) & \cos(\theta_r) \end{bmatrix} \tag{2.15}
$$

Chapter 3

Step Detection, Step Length and Heading Angle Estimation

This chapter introduces the 3 main concepts central to PDR, step detection, step Length Estimation and Heading Angle Estimation.

3.1 Step Detection

To estimate the position of the pedestrian it is necessary to determine the distance travelled by him, which is approximated by summing up the step lengths at each step, in order to do this it is first necessary to detect the steps. Prior to delving into the math for step detection, it is good to have an overview about the forces acting on a person during his walk. When a person walks the swinging leg experiences an acceleration during the first half of the step and then experiences a deceleration to come to a stop [7]. This means that during a step, there is an increasing acceleration followed by a decreasing acceleration, which indicates that there exists a local maxima for the acceleration data, plotted against time for each step and detecting this local maxima would indicate a step.

When a pedestrian walks with a hand-held IMU, the accelerometer captures all the forces acting on it due to other body motions as well (vibrations due to holding the device in the hand) and also the accelerometer measurements are inherently noisy. Therefore the norm of the $3-dimensional$ accelerometer signal from the accelerometer shall be filtered using a band pass filter such that only the forces due to the pedestrian's walk are considered and all the other forces are filtered out. Further after filtering few disturbances could still be present in filtered acceleration as well, hence it would be a good idea to consider the filtered values only above a certain *threshold*. After analyzing a considerable amount of walking data a suitable *threshold* value of 0.5 has been chosen. Also during a normal walk it is unrealistic to have a consecutive steps in a very short span of time, hence there exists a *minimum time* between consecutive steps and this minimum time condition could be used to eliminate false step detections.

A step detection method proposed in [4] shall be considered in this thesis. In the paper a fourth order band pass Butterworth filter with the lower cut-off frequency of 0.2 Hz and an upper cut-off frequency of 2.75 Hz has been proposed and this filter has been obtained using the *butter* command in matlab. Denoting the norm of the acceleration at any time instant t as $a(t)$ and the filtered

acceleration by $f_a(t)$ then the obtained filter equation in time domain is given by

$$\begin{aligned} f_a(t) =& 0.0058a(t) - 0.0115a(t-2) + 0.0058a(t-4) \\ &+ 3.7701f_a(t-1) - 5.338f_a(t-2) + 3.3661f_a(t-3) - 0.7973f_a(t-4). \end{aligned} \tag{3.1}$$

In support of the above description for step detection, Figure 3.1 below shows the plot of, norm of the accelerometer data, filtered data and steps detected.

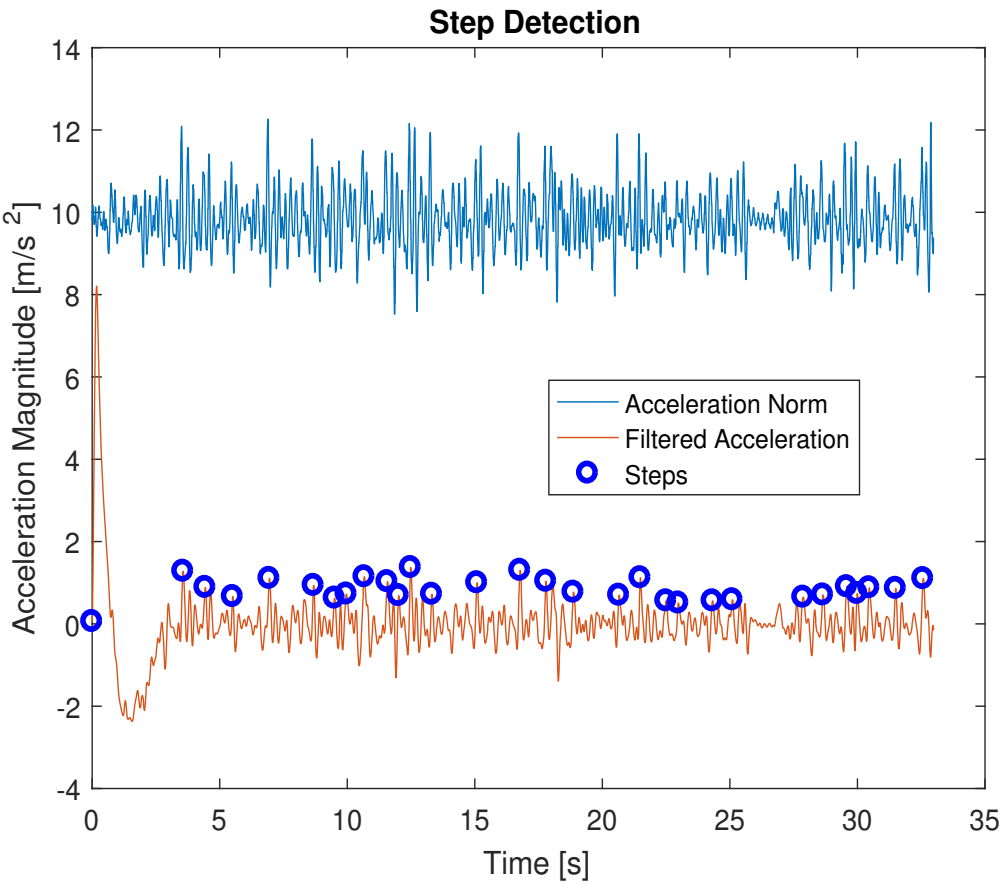


Figure 3.1: Norm of the acceleration, filtered acceleration and steps detected

To further elaborate the description of step detection a step detection algorithm is stated below in Algorithm 1. The algorithm is based on the following **assumptions**.

- The algorithm assumes that it would take greater than 0.5 s for a person to take a step. The assumption is based on the calculation that the speed of a normal human walk is 0.45 m/s at an average step length of 0.7 m.
- A step time of greater than 3 s is considered too low to take a step. Hence if a step time greater than 3 s is detected. The data for that particular step is ignored.
- It is possible to trick the algorithm of step detection by standing still and accelerating the device. The algorithm **does not** handle this case.

The following is the description of the variables that are used in the algorithm:

- *minimumTime* - minimum time for a step is 0.5 s. 0.5 s corresponds to 50 samples from the IMU
- *maximumtime* -maximum time for a step is 3 s. 3 s corresponds to 300 samples from the IMU
- k - corresponds to number of steps-2
- *newSate*, *previousState* - variables used to detect local maxima
- *steptimeIndex*- Array that stores the step time indices

Algorithm 1 Step Detection

Initialize variables

$max{=}-\infty$
$newState$=0;
$steptimeIndex(1)$=1
$minimumTime$=50
$maximumTime$=300
k=2
$threshold$=0.5
$previousStepTimeIndex$=1

for all Accelerometer Measurements i=1,2.... **do**

Obtain total Acceleration
$\mathrm{a}(i) = \sqrt{a_{x_i}^2 + a_{y_i}^2 + a_{z_i}^2}$
Obtain filtered acceleration according to equation 3.1

Detection of local Maxima for a step

$previousState{=}newState$
if $f_a(i) > threshold$ **then**
$newState$=1
if $f_a(i) > max$ **then**
$max = f_a(i)$
$maxIndex = i$
else
$newState$=0

if $previousState! = newState \&\& newState == 0$ **then**
$stepTimeDifference = maxIndex - previousStepTimeIndex$
if $stepTimeDifference ¡ minimumTime$ **then**
$stepDetected$=false
else if $stepTimeDifference ¿ maximumTime$ **then**
$stepTimeIndex(k-1) = maxIndex$
$stepDetected$=false
else
$stepTimeIndex(k) = maxIndex$
$previousStepTimeIndex = stepTimeIndex(k)$
$stepDetected$=true
$k = k + 1$
$max{=}-\infty$

3.2 Step Length Distribution

Estimating the step length varies from person to person and the existing methods for step length estimation depend on the person's height, weight and gender [6]. As an end application of this thesis, the algorithm will be used in cases where the position measurements from radio beacons are available, hence it would be sufficient to approximate the step length with a Gaussian distribution with mean $\bar{L} = 0.72\,\mathrm{m}$ and standard deviation of $0.1\,\mathrm{m}$. According to the University of Oklahoma Health Sciences Center, the average step length of a man is $0.78\,\mathrm{m}$ and the average step length

of woman is 0.66 m [9]. Therefore, an average step length of both the genders of 0.72 m has been chosen.

Mathematically the step length L_k of a pedestrian can be denoted by a Gaussian distribution in the following manner:

$$L_k \sim \mathcal{N}(\bar{L}, 0.01) \tag{3.2}$$

3.3 Heading Angle Estimation

In an ideal case if a pedestrian walks with a smartphone parallel to the direction of the walk, the heading angle indicated by the IMU (or the device) is the heading angle of the pedestrian, but the orientation of the IMU (or the device) changes based on the hand movements of the pedestrian, this might cause the heading of the device to change independent of the heading direction of the pedestrian. In such a case the heading of the pedestrian cannot be directly obtained from the heading of the device and there exists an angular deviation between the direction indicated by the device and the pedestrian, which is referred to as the **crab angle** and this angle needs to be taken into account to estimate the heading angle of the pedestrian. Such a method of estimating the heading angle has been proposed in a patent in [12] and this method shall be considered in the thesis.

Based on the calculations presented in the patent there are a couple of ways to calculate the heading angle. In the first approach the heading angle is obtained either by taking the sum or difference of the heading of the device and the crab angle. In the second approach a heading angle is obtained using the accelerometer data, in a local coordinate system (X_l, Y_l) that lies in the horizontal plane. This angle is referred to as the heading angle offset. Later this heading offset can be converted to the global heading angle if angles between the global and the local coordinate systems are known. The following subsections elaborate both of these approaches.

3.3.1 Heading angle estimation using crab angle

As already mentioned the heading angle in this approach is obtained either by taking the sum or difference of the heading of the device and the crab angle. The heading angle of the device denoted by ψ is obtained from its orientation (this shall be elaborated in the next chapter). The crab angle denoted by δ is obtained using the accelerometer data. In order to define and estimate the crab angle it would seem necessary to define the body coordinate system and the device coordinate system, Figure 3.2 and 3.3 show the body and the device coordinate systems respectively.

Figure 3.2: Body coordinate system

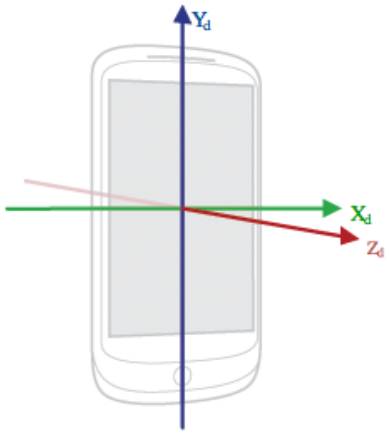


Figure 3.3: Device coordinate system

When a pedestrian walks with the phone in his hand, the velocity of the pedestrian is along Y_b of the body co ordinate system as shown in figure 3.2. The crab Angle δ is then defined as the angle from the walking direction (Y_b) to the Y axis of the device Y_d. This can be visualized on a global coordinate system as shown in Figure 3.4. The Figure indicates crab angle δ as the angle from the walking direction to the device.

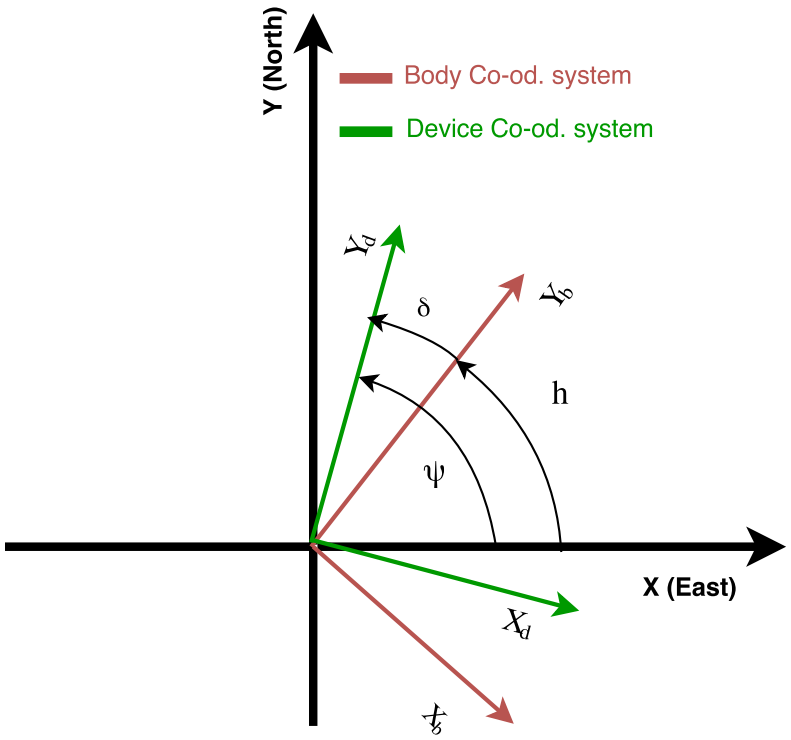


Figure 3.4: The Figure indicates heading angle of the person($h = \psi - \delta$), the heading angle of the device(ψ) and the crab angle(δ)

Having defined the crab angle, we can now proceed to the calculation of the crab angle. The idea behind the crab angle calculation is that when a person walks, the accelerometer captures the force due to the walk and the information of the forces captured by the sensor depends on the orientation of the sensor. For instance, consider a case when a person is walking straight and if the sensor is held relatively flat such that its axes X_d and Y_d are parallel to the ground and Z_d is perpendicular to the ground pointing upwards, then the acceleration due to the lifting of the legs is captured in Z_d and the acceleration due to the forward motion is captured along X_d and Y_d. Further it is worth mentioning that if any of the axes X_d or Y_d is parallel to the walking direction, then the acceleration in that axis will be much larger than its perpendicular axis. By obtaining the correlation of acceleration along X_d with acceleration along Z_d and correlation of acceleration along Y_d with acceleration along Z_d gives more definite information about the forces along X_d and Y_d. Finding the resultant force direction by using the correlation information gives the direction of human legs with respect to Y_d. This angle calculated by the resultant force direction is the **crab angle**.

Unlike the above example, in a more realistic and practical scenario, the orientation of the sensor is not relatively flat and it is held at a different orientation, in such cases it becomes important to find the acceleration in the coordinate system in which the X_d and Y_d are parallel to the ground. This can be done by obtaining the roll an pitch estimates (θ_r, θ_p) of the devices as described in section 2.3 and using the rotation matrix $R_e(\theta_p, \theta_r)$ to transform the acceleration in the device coordinates onto the horizontal frame of reference as

$$a_h = R_e(\theta_p, \theta_r) a_d. \tag{3.3}$$

Where, a_d is the 3 dimensional acceleration in the device coordinates, *i.e.* $a_d = [a_d^x, a_d^y, a_d^z]^T$ and a_h is the 3 dimensional acceleration in a coordinate system whose X and Y axes are on the the horizontal plane, *i.e.* $a_h = [a_h^x, a_h^y, a_h^z]^T$

To make the above description further clear the crab angle calculations is shown as an algorithm below in Algorithm 2. The notations used in the algorithm are: $\bar{a}_h$ is the average acceleration for each step, *i.e.* $\bar{a}_h = [\bar{a}_h^x, \bar{a}_h^y, \bar{a}_h^z]$ and C_{XZ}, C_{YZ} are the correlations of acceleration in the horizontal plane.

Algorithm 2 Crab Angle

1: **for all** accelerometer measurements i during step "k",i.e $i_{k-1} : i_k$ **do**

$$a_h(i) = R_e^T(\theta_{p_i}, \theta_{r_i}) a_d(i) \tag{3.4}$$

where $R_e(\theta_{p_i}, \theta_{r_i})$ is obtained from (2.14) and (2.15) in section 2.3

2: For each step "k" between time indices i_{k-1} and i_k calculate the following

$$\begin{aligned}
\bar{a}_h[k] &= \frac{\Sigma_{j=i_{k-1}}^{j=i_k} a_h(j)}{i_k - i_{k-1}} \\
C_{XZ}[k] &= \frac{\Sigma_{j=i_{k-1}}^{j=i_k} a_h^x(j) a_h^z(j)}{i_k - i_{k-1}} - \bar{a}_h^x[k]\bar{a}_h^z[k] \\
C_{YZ}[k] &= \frac{\Sigma_{j=i_{k-1}}^{j=i_k} a_h^y(j) a_h^z(j)}{i_k - i_{k-1}} - \bar{a}_h^y[k]\bar{a}_h^z[k] \\
\delta[k] &= \text{atan2}(C_{XZ}[k], C_{YZ}[k])^1
\end{aligned} \tag{3.5}$$

[1] If the crab angle were defined as the angle from X_d to Y_b then crab angle $\delta[k] = \text{atan2}(C_{YZ}, C_{XZ})$

Equation (3.5) essentially calculates the direction of force due to human legs in a given co ordinate system. So with slight modifications in the coordinate systems it should be possible to calculate the direction of the human legs in a global coordinate system using the same equations, which has been focused in the next subsection 3.3.2.

3.3.2 Approach 2- Heading Angle offset using accelerometer data

In this method the idea is to obtain the heading angle of a pedestrian in a fixed local coordinate system X_l, Y_l that lies in the horizontal plane whose origin is the starting point of the pedestrian navigation. This heading angle obtained in the local coordinate system is referred to as the heading angle offset denoted by ϕ_k.

It can be noted that in the crab angle calculation in section 3.3.1 the acceleration a_d in the device coordinates is transformed to acceleration a_h in the horizontal frame with the help of rotation matrix R_e which is formed from the roll and pitch of the device. Instead if the acceleration a_d in the device coordinates is transformed to a_h using a rotation matrix formed from the roll, pitch and yaw of the device and then continue calculations with the same equations in 3.5; the angle then obtained is no longer the crab angle, instead it is a heading angle of the pedestrian in a local coordinate system. Therefore the heading angle offset can be obtained by implementing Algorithm 2 with the following modifications.

1. Modify (3.4) as

$$a_h(t) = R_q(\hat{q}_k)a_d \qquad \text{where } \hat{q}_k \text{ is obtained from section 2.2} \tag{3.6}$$

2. Modify $\delta[k]$ in (3.5) as $\phi[k] = \text{atan2}(C_{YZ}, C_{XZ})$

Chapter 4

Implementation

In the previous chapter the concepts central to PDR have been discussed individually, the following chapter integrates the step detection, step length and heading angle estimation to estimate positions. As already noted, in the previous chapter that 2 methods of heading angle estimation have been described. Consequently 2 methods to estimate the positions shall be discussed in this chapter. The first method uses the the crab angle approach and this problem shall be solved as a stochastic filtering problem by implementing an extended Kalman filter. The second method is a simple integration using the heading angle offset described in 3.3.2 to obtain the positions.

4.1 Method 1- Navigation method using the Crab Angle

This method is based on the heading angle estimate using the crab angle as described in 3.3.1. As shown in Figure 3.4, the heading angle (h) can be obtained as the difference of the device heading (ψ) and the crab angle (δ) *i.e* $h = \psi - \delta$. Prior to getting into the modelling the navigation method, a few pre-calculations need to be done which will act as inputs to the navigation method. These are sequentially stated below:

- Based on IMU measurements obtain the roll θ_r and pitch of the device θ_p as described in section 2.3.
- Implement the step detection as described in section 3.1.
- For each step detected calculate the crab Angle using the roll and the pitch information as described in 3.3.1 and the change in the device heading Δ_k^{ψ} using the gyroscopic measurements (calculation shown in Algorithm 3). These quantities will be used as inputs to the navigation model.

After the crab angle and the change in device heading calculations, implement an extended Kalman Filter to estimate the positions with the motion and measurement models that are described in the succeeding subsections. Subsequently the tuning of the filter for the models is also discussed.

4.1.1 Motion Model

The motion model is shown in (4.1). It calculates the positions X_k, Y_k by integrating the step lengths in the respective heading directions. Here the heading direction is given as $h = \psi_{k-1} + \Delta_k^{\psi} - \delta_k$, which is the difference of the device heading and the crab angle. Since the step length is approximated by a Gaussian distribution as described in 3.2 a noise term v_{k-1} has been introduced to account for the uncertainties in the average step length ($\bar{L}$). The device heading ψ_k is obtained by integrating the gyroscopic readings and to account for the drift due to the integration an additive Gaussian noise q_ψ has been introduced. The motion model in (4.1) is a non linear motion model. In order to implement an EKF it needs to be linearized, the linearization is shown in the Appendix A.3

$$\begin{aligned} X_k &= X_{k-1} + (\bar{L} + v_{k-1})\cos(\psi_{k-1} + \Delta_k^{\psi} - \delta_k) \\ Y_k &= Y_{k-1} + (\bar{L} + v_{k-1})\sin(\psi_{k-1} + \Delta_k^{\psi} - \delta_k) \\ \psi_k &= \psi_{k-1} + \Delta_k^{\psi} + q_\psi \end{aligned} \tag{4.1}$$

The heading of the device is obtained using gyroscopic measurements which drifts with time and this drift shall be corrected by the magnetometer readings using a measurement update. A couple of different measurement models can be used, one model is a linear model and the other is a non linear model. Both the models give almost the same results and both the models shall be stated below, but the results generated in the thesis are based on a non linear measurement model.

4.1.2 Linear Measurement Model:

Equation (4.2) represents the measurement model which is modelled linearly with the state ψ_k with an additive Gaussian noise e_m. The measurement sequences y_k^m are calculated by tracking the change in the device heading ($\Delta\theta_k^m$) for consecutive steps using the magnetometer measurements m_k^y, m_k^x. To be even more precise, the magnetometer measurements m_k^y, m_k^x from the IMU are projected onto a horizontal plane by multiplying with a rotation matrix R_e described in section 2.3 and obtained as $m_k^{y,h}, m_k^{x,h}$. These projected measurements are transformed to indicate an angle with respect to the "geographic East" denoted as θ_k^m. The difference in the angle between consecutive steps is obtained as $\Delta\theta_k^m$. With a known initial global heading direction of the device θ_0, the heading after each step k can be tracked as $\theta_0 + \Delta\theta_k^m$. The measurement sequences modelled mathematically are shown in equation (4.3).

It is worth to mention that the domain of angle indicated by θ_k^m has been transformed to $[0, 2\pi]$ by using the term $sign(\text{atan2}(m_k^y, -m_k^x)) - 1)\pi$ in the expression for θ_k^m. Since the domain of the angles given by $atan2$ is from $[-\pi, \pi]$ and in order to calculate $\Delta\theta_k^m$, it is necessary to have the domain in $[0, 2\pi]$.

Model

$$\begin{aligned} &y_k = \psi_k + e_m \\ &\quad \text{which can be reformulated in a matrix form as follows} \\ &y_k = \underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{H_k} \begin{bmatrix} X_k \\ Y_k \\ \psi_k \end{bmatrix} + e_m \end{aligned} \tag{4.2}$$

Measurement Sequences

$$\begin{aligned} y_k^m &= \theta_0 + \Delta\theta_k^m \\ \Delta\theta_k^m &= \theta_k^m - \theta_{k-1}^m \\ \theta_k^m &= \text{atan2}(m_k^{y,h}, -m_k^{x,h}) - (sign(\text{atan2}(m_k^{y,h}, -m_k^{x,h})) - 1)\pi \\ \theta_{k-1}^m &= \text{atan2}(m_{k-1}^{y,h}, -m_{k-1}^{x,h}) - (sign(\text{atan2}(m_{k-1}^{y,h}, -m_{k-1}^{x,h})) - 1)\pi \end{aligned} \tag{4.3}$$

4.1.3 Non Linear Measurement Model

Equation (4.4) represents the non linear measurement model and is based on the idea that, magnetometer readings projected onto the horizontal plane can be modelled based on the heading of the device ψ_k. The noise in the measurement model appears due to the uncertain device heading angle, hence the noise terms n_x, n_y, that are modelled as white Gaussian noises are added to heading of the device ψ_k. The measurement sequences $[m_h^x[k], m_h^y[k], m_h^z[k]]^T$ in the horizontal plane can be obtained by multiplying the magnetometer measurements in the device coordinates with the rotation matrix R_e as shown in (4.5). It is to be noted that the value m_h^z has no significance in the horizontal plane, only m_h^x, m_h^y are of importance. Further these measurement sequences are to be normalized into a unit vector since the values predicted by the measurement model have a norm of unity. The normalization of these vectors is shown in (4.7).

In order to implement an EKF with this non linear measurement model, it needs to be linearized, the linearization is shown in Appendix A.3. Further the extended Kalman filter calculations are shown in same section.

Model:

$$\begin{bmatrix} m_k^{x,h} \\ m_k^{y,h} \end{bmatrix} = \begin{bmatrix} \cos(\psi_k + n_x) \\ \sin(\psi_k + n_y) \end{bmatrix} \tag{4.4}$$

Measurement Sequences:

$$[m_h^x[k], m_h^y[k], m_h^z[k]]^T = R_e(\theta_p, \theta_r)[m_k^x, m_k^y, m_k^z]^T. \tag{4.5}$$

4.1.4 Filter Tuning

The performance of any filter is highly dependent on proper tuning of its parameters; in particular the noise terms of the motion and measurement model. The noise terms in all the model are assumed to be white Gaussian noises, hence it only remains to tune them in terms of their variances or covariance matrices. The variances of all the noise terms have either been set by intuition or by visual inspection of the position and heading estimates. Following are the tuned noise terms for the motion and measurement models:

Motion Model- The noise appears in the form of uncertain step length denoted by v_{k-1}, since it is assumed that the average step length of 0.72 m would have a standard deviation of 0.1 m, the variance of v_{k-1} is set as 0.01. The variance of the noise term q_ψ has been set as 0.001 based on visual inspection of the estimates.

Linear Measurement Model- The measurement model noise e_m also has been tuned by visual inspection and has been set a variance of 0.08.

Non Linear Measurement Model- The measurement model noise terms n_x, n_y are tuned by visual inspection and have been set a variance of 0.9 each.

4.1.5 Algorithm

To put together the above description of the navigation method, an algorithm for this method is described in Algorithm 3. This algorithm is described considering the Non linear measurement model. Further in the algorithm the variables g_d and g_h are the gyroscopic measurements in the device coordinates and the horizontal plane respectively. The other variables used in the Algorithm are already defined in the previous Sections or Chapters.

Algorithm 3 Navigation Method - 1

1: For all IMU measurements i=1,2....

2: Obtain the roll and pitch estimates r_i, p_i and the rotation matrix for the roll and pitch angles $R_e(\theta_{p_i}, \theta_{r_i})$ from section 2.3

3: Implement the step Detection Algorithm described in Algorithm1
Obtain the the time indices for the step "k"
i.e. obtain $stepTimeIndex(k), stepTimeIndex(k-1)$
which will be denoted as i_k and i_{k-1} respectively in this algorithm

4: **if** *stepDetected==true* **then**
5: **for** $j = i_{k-1} : i_k$ **do**
6: $a_h(j) = [a_h^x(j), a_h^y(j), a_h^z(j)]^T = R_e(p_i, r_i)^T a_d(j)$
7: $g_h(j) = [g_h^x(j), g_h^y(j), g_h^z(j)]^T = R_e(p_i, r_i) g_d(j)$
8: $m_h(j) = [m_h^x(j), m_h^y(j), m_h^z(j)]^T = R_e(p_i, r_i) m_d(j)$

9: For each step k perform the following

10: Obtain crab Angle $\delta[k]$ as described in Algorithm 2

11: Obtain change in heading angle of the device Δ_k^ψ as in (4.6) below

$$\Delta_k^\psi = \Sigma_{n=i_{k-1}}^{n=i_k} g_h^z(n)(t(n) - t(n-1)) \tag{4.6}$$

where $t(n)$ is the time in seconds at IMU measurement index n

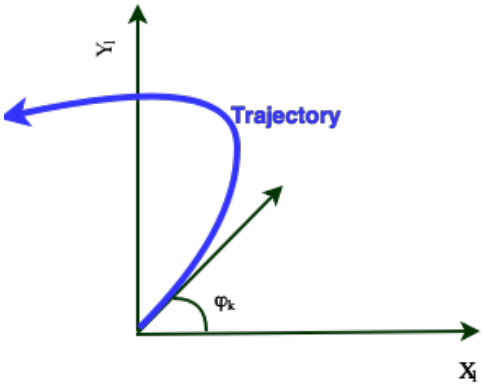
12: Normalize the horizontal plane magnetometer readings at the end of each step

$$\begin{aligned} m_h^x[k] &= \frac{m_h^x(i_k)}{\sqrt{m_h^x(i_k)^2 + m_h^y(i_k)^2}} \\ m_h^y[k] &= \frac{m_h^y(i_k)}{\sqrt{m_h^x(i_k)^2 + m_h^y(i_k)^2}} \end{aligned} \tag{4.7}$$

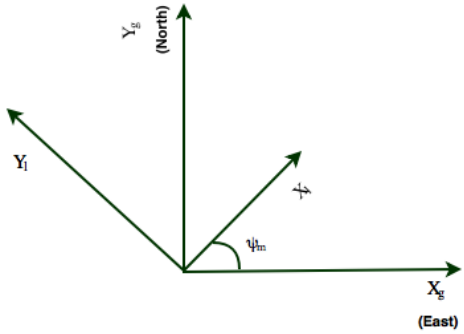
13: Implement an EKF for the motion and measurement models given in (4.1) and (4.4) shown in Appendix A.3

4.2 Method 2-Navigation method using Heading Offset

The idea behind this method of navigation is to use the heading angle offset calculated in 3.3.2. This heading angle offset ϕ_k is converted to the global heading angle if angles between the global and the local coordinate systems are known. Refer Figure 4.1 that further clarifies the idea of the global heading angle calculation.



(a) Figure illustrating Heading offset in a local co-ordinate system



(b) Figure illustrating the angles between the Local and the Global coordinate systems.

Figure 4.1: Local and global Coordinate systems

In the above figure ϕ_k is the heading angle in a local coordinate system. This heading angle can be transformed into the global coordinate system known the angle ψ_m. The transformation is shown later below.

The steps in the navigation method are as follow:

- Use Gyroscope and Accelerometer data to obtain an orientation estimate $\hat{q}_k$ of the device as described in Section 2.2
- Use the accelerometer data from the device to detect steps as described in section 3.1
- For each step, calculate the heading Angle offset using the roll, pitch and yaw information of the devices as described in section 3.3.2
- After obtaining the heading angle, the trajectory of the pedestrian in a local coordinate system can be obtained by a simple motion model described below in (4.8)

$$\begin{aligned} X_k &= X_{k-1} + (\bar{L})\cos(\phi_k) \\ Y_k &= Y_{k-1} + (\bar{L})\sin(\phi_k) \end{aligned} \tag{4.8}$$

- If the angle between the local coordinate system and the global coordinate system is known then the coordinates in the local coordinate system can be transformed to global coordinate system by using a 2d-rotation matrix as shown below:

$$\begin{bmatrix} X_k^g \\ Y_k^g \end{bmatrix} = \begin{bmatrix} \cos(\psi_m) & -\sin(\psi_m) \\ \sin(\psi_m) & \cos(\psi_m) \end{bmatrix} \begin{bmatrix} X_k \\ Y_k \end{bmatrix} \tag{4.9}$$

To put together the above description the navigation method is stated as an algorithm below.

Algorithm 4 Navigation Method - 2

1: For all IMU Measurements i=1,2....

2: Obtain orientation estimate $\hat{q}_i$ as described in section 2.2

3: Implement the step Detection Algorithm described in Algorithm 1
Obtain the the time indices for the step "k"
i.e. $stepTimeIndex(k), stepTimeIndex(k-1)$
which are denoted as i_k and i_{k-1} respectively in this algorithm

4: **if** *stepDetected==true* **then**
5: **for** $j = i_{k-1} : i_k$ **do**
6: Obtain the rotated acceleration a
$a_r(j)$=$R_q(\hat{q}_j)a_d(j)$

7: For each step k obtain the Heading offset $\phi[k]$ as described in section 3.3.2

8: Obtain Position in Local Coordinate system

$$\begin{aligned} X_k &= X_{k-1} + (\bar{L})\cos(\phi_k) \\ Y_k &= Y_{k-1} + (\bar{L})\sin(\phi_k) \end{aligned} \tag{4.10}$$

9: Obtain Position in Global Coordinate system

$$\begin{bmatrix} X_k^g \\ Y_k^g \end{bmatrix} = \begin{bmatrix} \cos(\psi_m) & -\sin(\psi_m) \\ sin(\psi_m) & \cos(\psi_m) \end{bmatrix} \begin{bmatrix} X_k \\ Y_k \end{bmatrix} \tag{4.11}$$

where ψ_m =atan2(m_y^h, m_x^h),$[m_x^h, m_y^h, m_z^h]^T = R_q(\hat{q})[m_x, m_y, m_z]^T$

Chapter 5

Results and Discussions

There are numerous situations that could arise in the context of pedestrian navigation when a user walks with a smartphone in his hand. To validate the algorithm in most practical scenarios, data has been collected for various situations and test cases have been formulated. The test cases along with the results for both the navigation methods shall be presented in this chapter.

5.1 Test Scenarios

5.1.1 Test Scenario-1

This is a test scenario to validate basic and simple dead reckoning. Following are the conditions in the test scenario:

- The phone is held parallel to the direction of walk, *i.e* Y_d parallel to Y_b.
- The orientation of the phone does not change through out the path.
- There are direction changes on the path *i.e.* the path taken is not a straight line, it can be any path, for instance walking in square or rectangle or circle or even in a zig-zag manner.

5.1.2 Test Scenario-2

This is test scenario to validate the objective of the thesis, i.e to obtain accurate heading angle irrespective of the orientation of the device. Following are the conditions in the test scenario:

- The phone can be held at any orientation with respect to the body.
- The orientation of the phone changes while walking through the path.
- There are no direction changes on the path *i.e.* the path taken is a straight line.

5.1.3 Test Scenario-3

This test case is an extension of "Test Scenario-2", to make it even more realistic and practical. Following are the conditions in the test scenario:

- The phone can be held at any orientation with respect to the body.
- The orientation of the phone changes while walking through the path.
- There are direction changes on the path.

5.2 Results for the Navigation method 1

The results for each test case have been plotted. The plots contain, reference trajectory, obtained trajectory and various angles such as heading angle of the pedestrian, heading angle of the device and crab angle. To generate the results it is assumed that the initial heading of the device in the global coordinates is known (which can obtained from the magnetometer measurements, when the pedestrian just starts his navigation). In order to verify the performance of the algorithm two other trajectories based on individual sensors like gyroscope or magnetometer have been plotted. Below is the list of trajectories and the angles that are plotted:

- Reference trajectory, indicating the actual path taken
- Trajectory obtained after implementing the EKF for the navigation method
- Trajectory obtained if only gyroscopic measurements (contains noise) are used along with the crab angle. (5.1) shows the equations to obtain such a trajectory

$$\begin{aligned} X_k &= X_{k-1} + (\bar{L})\cos(\psi_k^{gyro} - \delta_k) \\ Y_k &= Y_{k-1} + (\bar{L})\sin(\psi_k^{gyro} - \delta_k) \end{aligned} \tag{5.1}$$

- Trajectory obtained if only magnetometer measurements (contains noise) are used along with the crab angle. (5.2) shows the equations to obtain such a trajectory

$$\begin{aligned} X_k &= X_{k-1} + (\bar{L})\cos(\psi_k^{mag} - \delta_k) \\ Y_k &= Y_{k-1} + (\bar{L})\sin(\psi_k^{mag} - \delta_k) \end{aligned} \tag{5.2}$$

- Estimated heading angle of the person obtained using EKF
- Estimated Heading angle of the device
- Crab Angle during the walk

5.2.1 Results for Test Scenario-1

Data Description

The data has been collected based on Test Scenario-1, by walking in a rectangle for 30 steps, with 9 steps along the length and 6 steps along the width. The device was held almost static, flat, and in the direction of walk.

Expected Results and Obtained Results

For the data collected, following are the expected and obtained results:

1. Figure 5.1 illustrates the obtained trajectories along with the reference trajectory, from the Figure it can be observed that filtered trajectory closely tracks the reference trajectory, but however the trajectory obtained by gyroscope alone tracks the reference trajectory even closer, this can be associated with the fact that the filtered trajectory is affected due to the noise in the magnetometer. The noise in the magnetometer is evident from the trajectory tracked by magnetometer alone.
2. Since the device is held static, flat, and in the direction of walk, the expected crab angle must be constant and zero. Figure 5.2 illustrates the obtained crab angle which is also constant and zero, thus showing coherence in expected and obtained results.
3. Since there are no orientation changes during the walk, one can expect, the estimated heading angle of the device and the estimated global heading angle must be the same; the obtained results are as expected which is illustrated in the Figure 5.2. Figure 5.3 illustrates that the heading angle indicated by individual sensors, is roughly the same as the estimated heading angle, which is kind of expected as there are no orientation changes in the device.

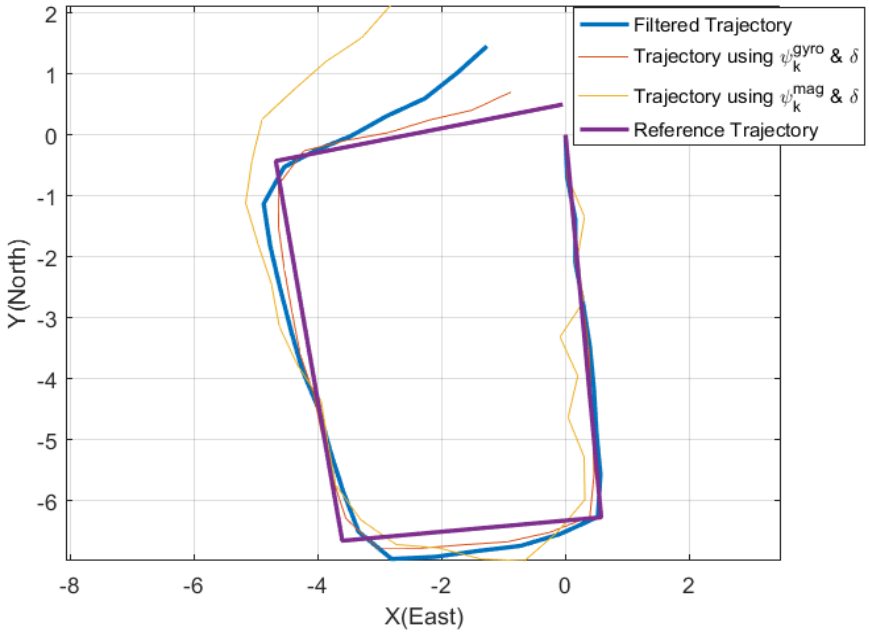


Figure 5.1: Trajectory for walking in a rectangle.

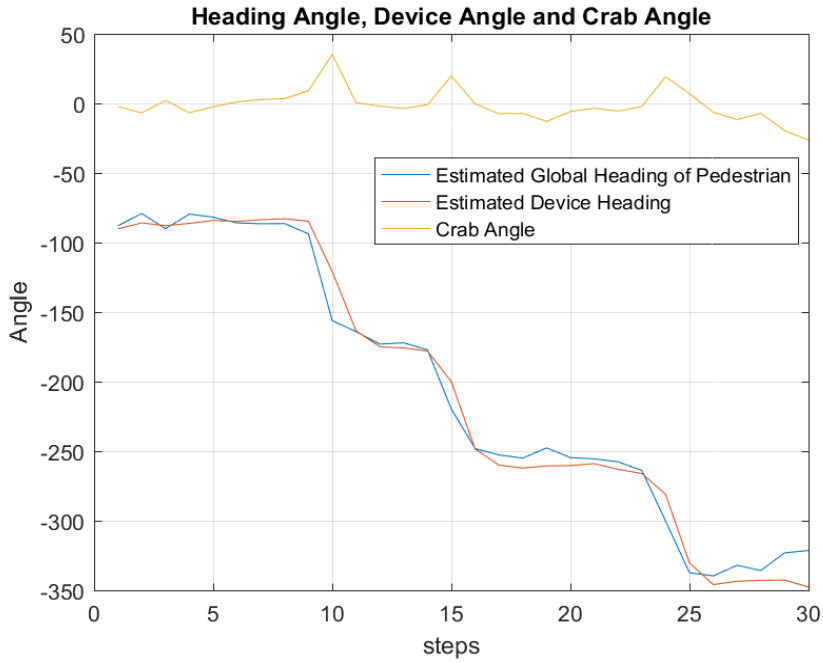


Figure 5.2: Various Angles for the trajectory

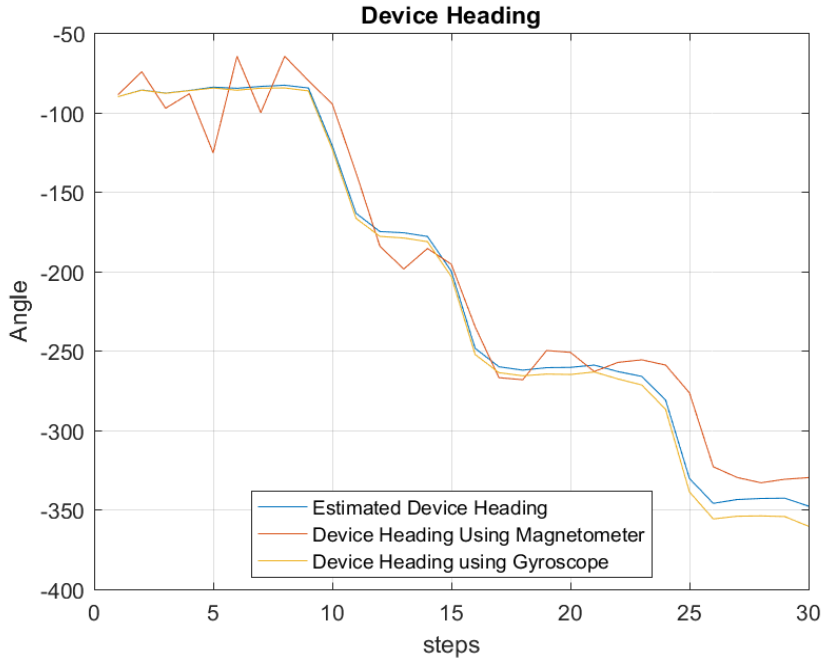


Figure 5.3: Estimated Device Heading angle, along with device heading indicated by individual sensors

5.2.2 Results for Test Scenario-2

Data Description

The data has been collected based on Test Scenario-2, by walking in a straight line and changing the heading of the device in a counter clock wise manner roughly after every 10-12 steps, i.e after every 10-12 steps the angle between the device and the body was roughly changed by 70° to 90°.

Expected Results and Obtained Results

For the data collected, following are the expected and obtained results:

1. Figure 5.4 illustrates the results obtained by walking in a straight line. It can be observed that the filtered trajectory, closely tracks the reference trajectory.
2. Since the walking path is a straight line, irrespective of the orientation changes, the estimated heading angle of the pedestrian must be constant. This is evident from the obtained results as well which is illustrated in Figure 5.5, further it is worth to note that heading angle h and the angle $h + 2n\pi$, $\forall\ n = \{1, 2, 3..\}$ are equivalent. From the Figure it can be seen that the estimated heading angle shows a jump at step number 20 from $-30°$ to 330°, but both these angles are equivalent in a coordinate plane. Therefore irrespective of the heading of the device the heading angle of the pedestrian is constant.
3. Since the orientation of the device is changing after every 10 to 12 steps roughly by 70° to 90°; the plot of the "crab angle" and "estimated device heading" must reflect this and also as there is no direction change in the path, the crab angle must follow the same pattern as the estimated device heading. These results are evident in the obtained results as well, as illustrated in Figure 5.5.

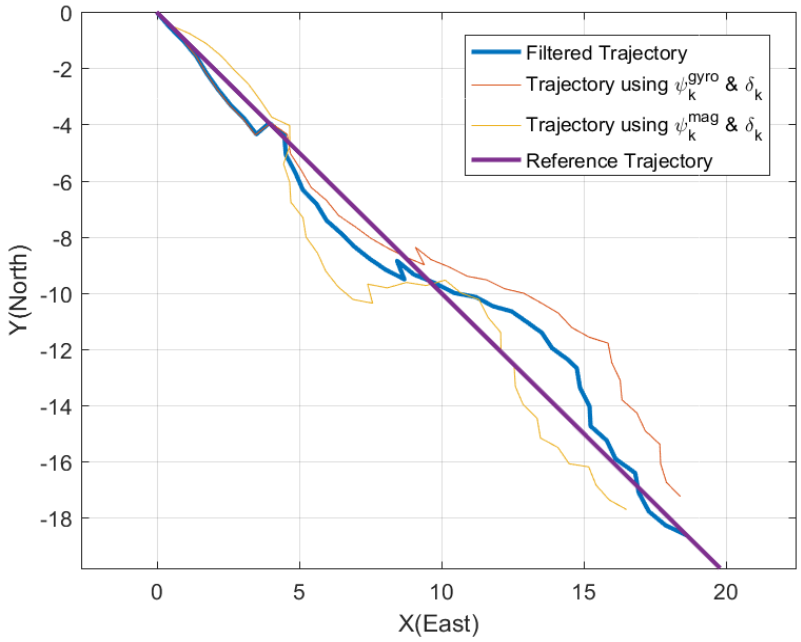


Figure 5.4: Trajectory for walking in a Straight line.

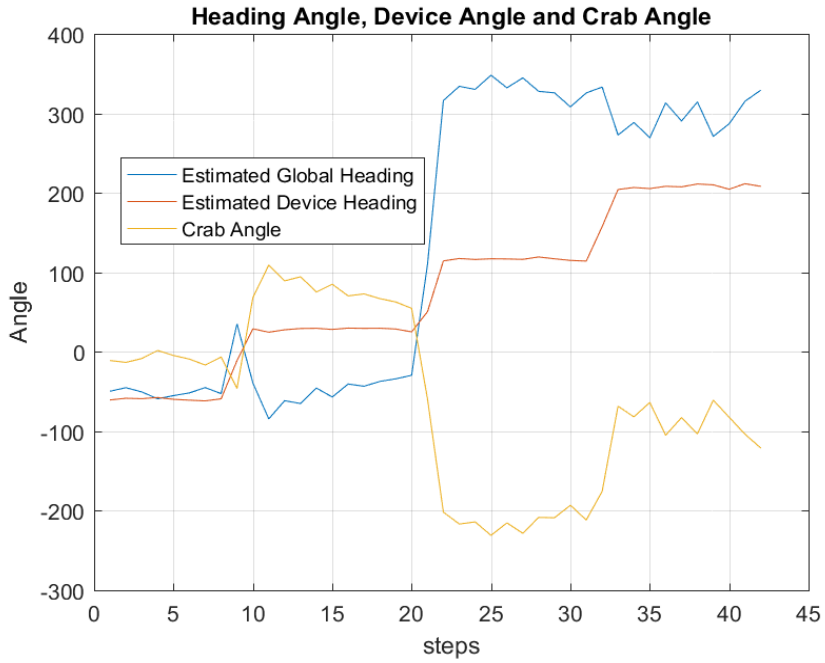


Figure 5.5: Various Angles for the trajectory

5.2.3 Results for Test Scenario-3

Data Description

The data has been collected based on Test Scenario-3, by walking in a L shape for 41 steps. The heading of the device has been changed roughly by 70° to 90° in counter clockwise direction after every 10 to 12 steps and around step number 24, there is a "change in direction" in the path.

Expected Results and Obtained Results

For the data collected, following are the expected and obtained results:

1. Figure 5.6 illustrates the results obtained by walking in a L shape. It can be observed that the obtained filtered trajectory is able to track the reference trajectory.

2. The global heading of the pedestrian is supposed to change only around step 24 and nowhere else, which is evident from the obtained results as well, as illustrated in Figure 5.7, but there is also an unexpected result at step 33 where the estimated global heading changes, when in reality there was no direction change, and this can be accounted by the noisy crab angle estimation.

3. Since the heading of the device is changing after every 10 to 12 steps roughly by 70° to 90°; the plot of the "crab angle" and "estimated device heading" must reflect this, further the estimated heading angle of the device must also change in the same pattern as the crab angle. All these are in evident from the obtained results as well, as illustrated in Figure 5.7.

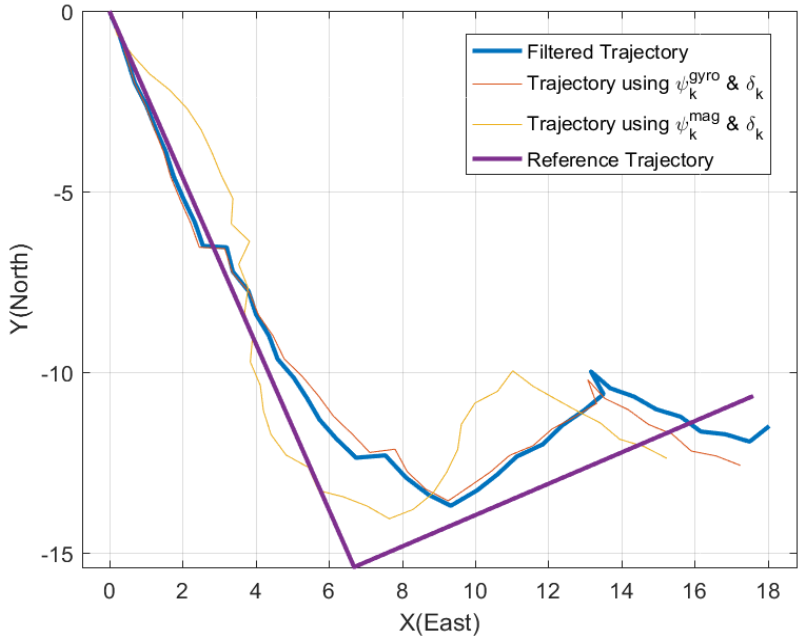


Figure 5.6: Trajectory for walking in L shape.

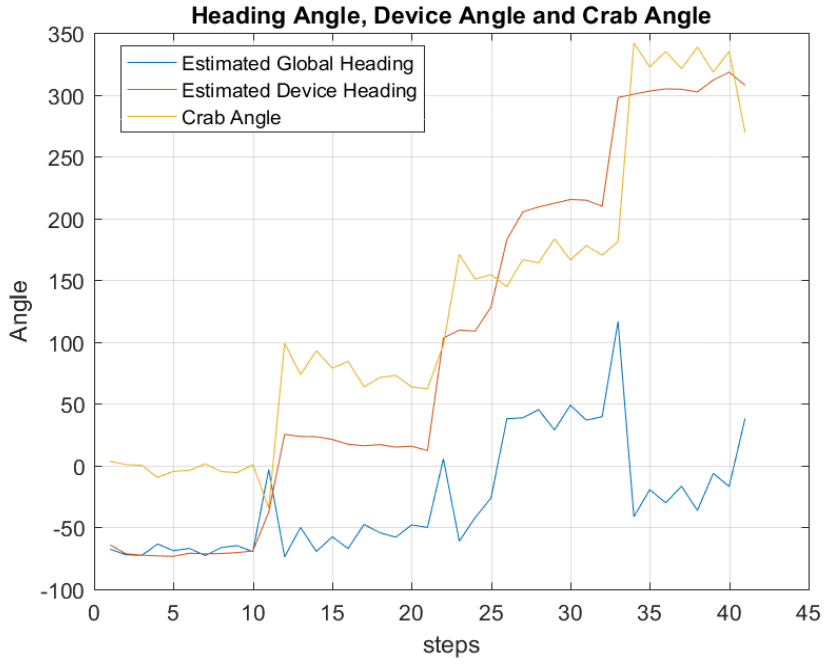


Figure 5.7: Various Angles for the trajectory

5.2.4 Results for Test Scenario-3

Data Description

An additional test result for Test Scenario 3 for a different trajectory has been presented. The data has been collected by walking in a manner shown in the reference path in figure 5.8. The heading of the device has been changed roughly by 70° to 90° in counter clockwise direction after every 10 to 12 steps and around step step 22 and step 38, there is a "change in direction" in the path.

Expected Results and Obtained Results

For the data collected, following are the expected and obtained results:

1. Figures 5.8 illustrates the results obtained by walking along the shown reference path. It can be observed that the obtained filtered trajectory closely tracks the reference path.

2. The estimated heading angle of the pedestrian should change around step 22 and step 38 and must be constant for all other steps; this is evident from the obtained results as well, as illustrated in Figure 5.9.

3. Since the heading of the device is changing after every 10 to 12 steps roughly by 70° to 90°; the plot of the "crab angle" and "estimated device heading" must reflect this, further the estimated heading angle of the device must also change in the same pattern as the crab angle. These results are evident in the obtained results as well, as illustrated in Figure 5.9.

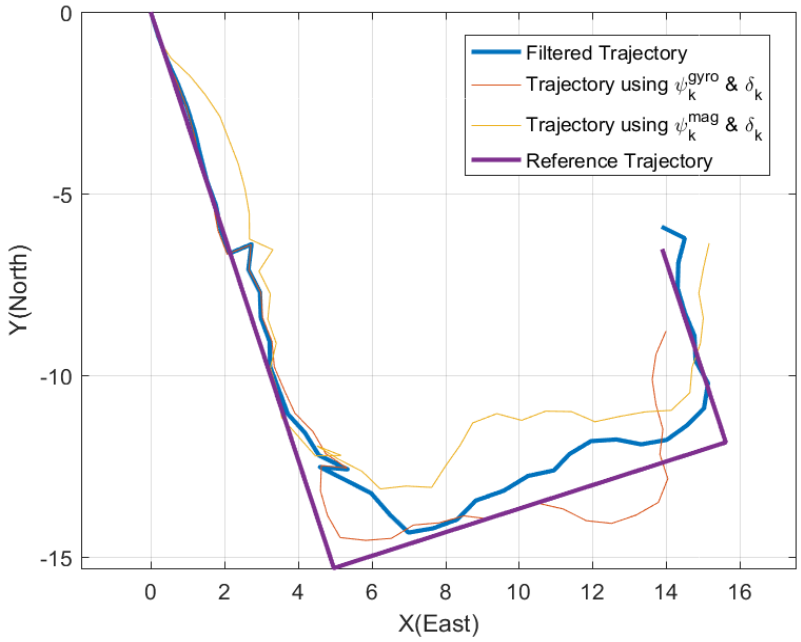


Figure 5.8: Trajectories

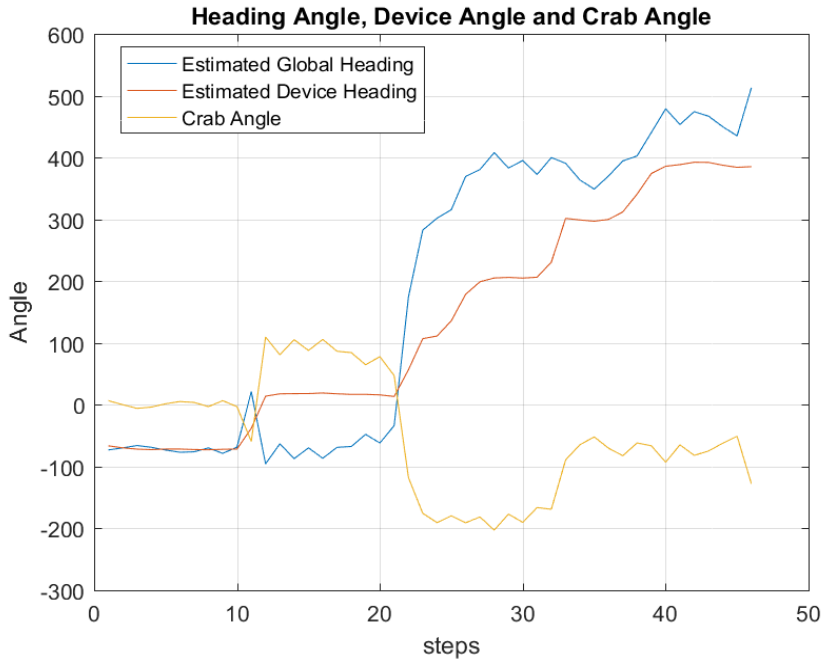


Figure 5.9: Various Angles for the trajectory

5.3 Results for Navigation method 2

The results for each test scenario has been plotted, with a similar data set that has been used to plot the results for "Navigation method 1". The results plotted for this method will contain the following:

- A Reference trajectory of the path.
- Obtained Trajectory using the navigation method.
- The heading angle obtained.
- Heading angle of the device indicated by the gyroscope.

5.3.1 Result for Test Scenario 1

Data Description: This is the same data set obtained by walking in a rectangle as described in section 5.2.1

Expected Results and obtained results

- The obtained trajectory and the reference trajectory are illustrated in Figure 5.10. It can be observed the obtained trajectory slightly deviates from the reference trajectory owing to the noise in the heading angle calculation.

- Since the trajectory is a rectangle, the heading angle should change by 90° roughly. This is evident from the obtained results as well, as illustrated in Figure 5.11
- If the phone is held relatively flat and in the direction of walk, the device heading obtained by gyroscopic data must have the same pattern as the calculated heading angle of the pedestrian. From the Figure 5.12 it can be seen that the obtained heading angle follows the same pattern as gyroscopic angle.

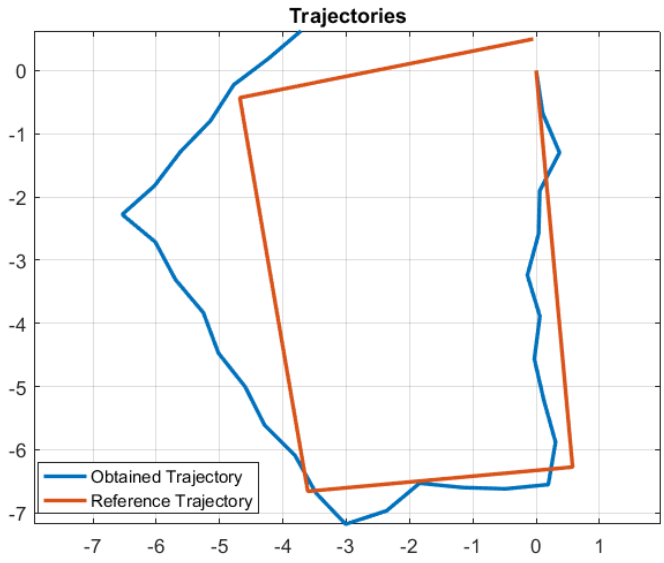


Figure 5.10: Trajectory for walking in a Rectangle

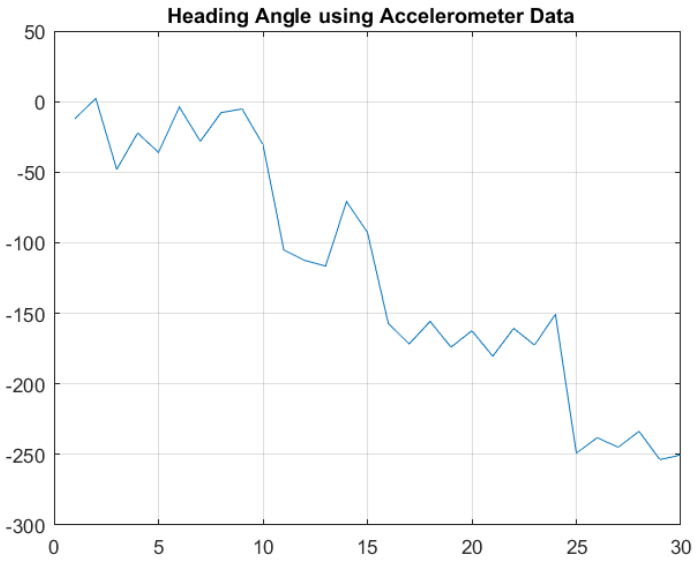


Figure 5.11: Heading Angle of the Pedestrian

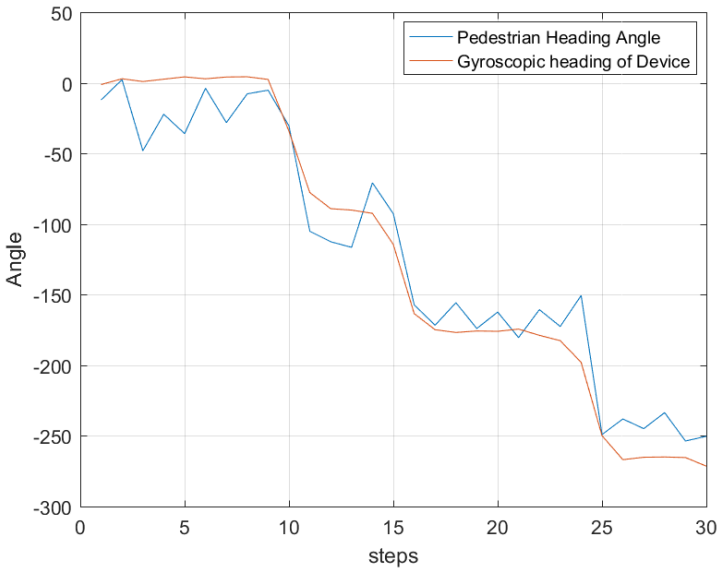


Figure 5.12: Heading Angle of pedestrian and gyroscopic heading of device

5.3.2 Result for Test Scenario-2

Data Description: This is same data set obtained by walking in a straight line as described in section 5.2.2

Expected Results and Obtained Results

- The obtained trajectory and the reference trajectory are illustrated in Figure 5.13. It can be observed the obtained trajectory deviates from the reference trajectory owing to the noise in the heading angle estimation.
- The heading angle must be constant irrespective of the orientation changes of the device. Figure 5.14 shows that the heading angle is constant (since heading angle $h = \theta = \theta + 2n\pi$ $\forall$ $\{n = 1, 2, 3, ..\}$) irrespective of the orientation changes and the orientation changes are indicated by the gyroscopic heading of the device in the same Figure.
- The heading angle calculated using accelerometer is quiet noisy, hence the obtained trajectory deviates from the reference trajectory significantly which is evident from the Figures 5.13 and 5.14.

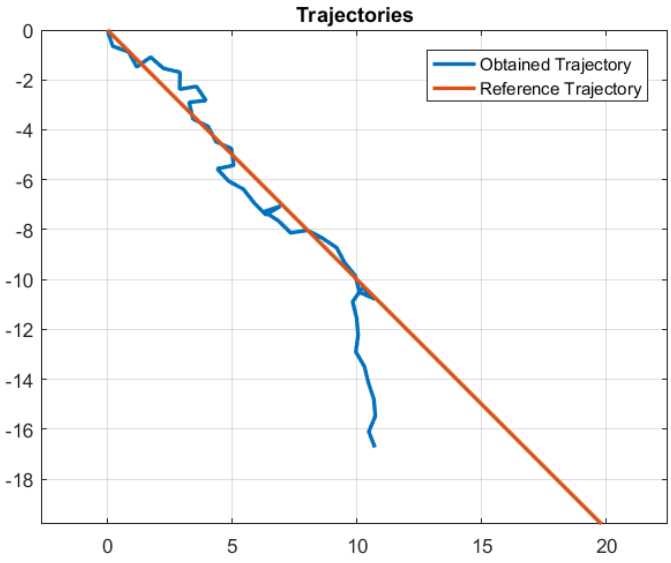


Figure 5.13: Trajectory for walking in a straight line

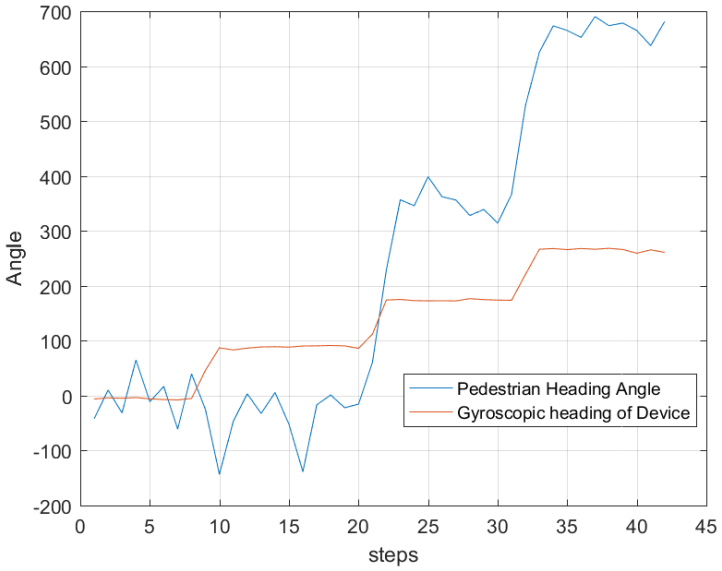


Figure 5.14: Heading Angle of pedestrian and gyroscopic heading of device

5.3.3 Result for Test Scenario-3

Data Description: This is same data set as described in section 5.2.4

Expected results and Obtained Results

- The obtained trajectory and the reference trajectory are illustrated in Figure 5.15. It can be observed the obtained trajectory slightly deviates from the reference trajectory owing to the noise in the heading angle estimation.
- The heading angle must change only around step 22 and step 38 irrespective of the orientation changes. This result is also evident from the obtained results, as illustrated in Figure 5.16.

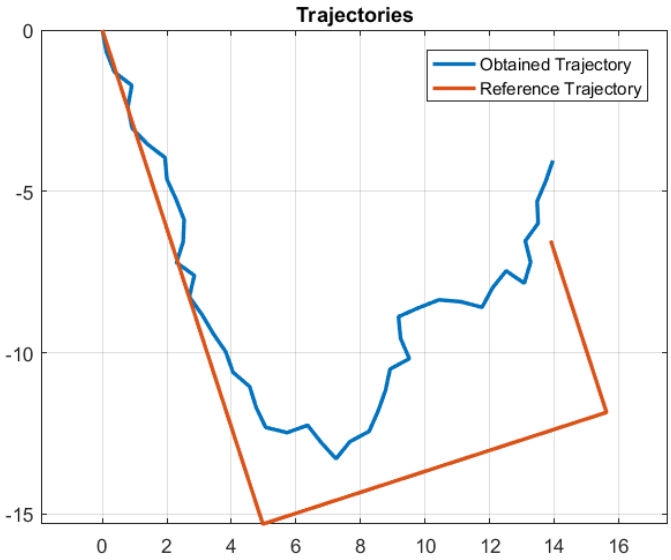


Figure 5.15: Trajectories

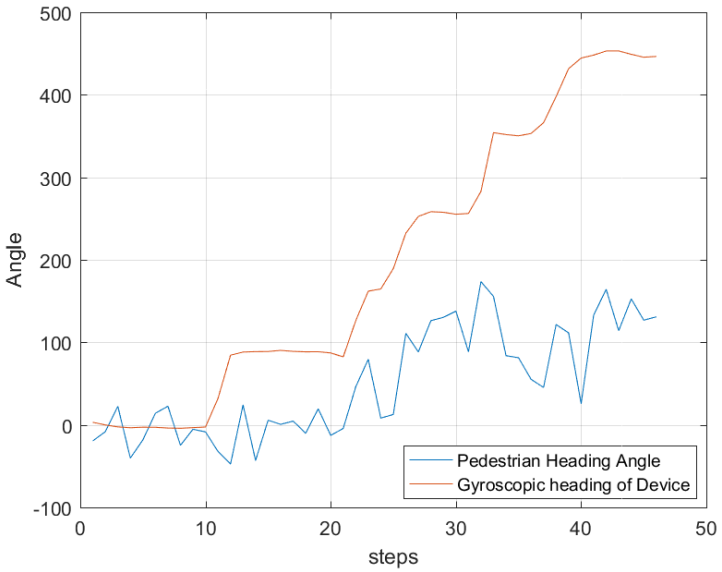


Figure 5.16: Heading Angle of pedestrian and gyroscopic heading of device

Chapter 6

Comparison Of Methods

Having described and presented the results for both the navigation methods a comparison of both the methods can now be done. There is a common shortcoming for both the methods which is regarding the orientation estimate of the device; as the pedestrian walks with the phone in his hand there is an "acceleration component due to the walk" captured by the accelerometer and this acceleration component is not considered in the orientation estimate. Despite this, the orientation estimate still works approximately well, since after every step when both the legs are touching the ground the acceleration due to the walk is close to zero and at this point the estimates of the orientation can be predicted, but any rapid changes of orientation during the walk are not estimated well. This is an important limitation which addresses the fact that the navigation methods will not work as expected for rapid orientation changes during the walk. The individual shortcomings of the methods which affect the results are described below.

Shortcomings of method 1

- Suffers due to noisy magnetometer readings.
- Suffers due to noisy crab angle calculations.

Shortcomings of method 2

- The calculated heading angle is noisy, due to the noise in the accelerometer readings.
- The heading angle calculation depends on orientation estimate due to gyroscope and accelerometer and this orientation estimate drifts in the yaw motion due measurement noise in the gyroscope leading to inaccurate heading angle calculations.

Due to the implementation of an EKF in method 1, it has advantages over method 2 which is a simple kinematic simulation of positions using the heading angle. The purpose of presenting method 2 in the thesis is to prove that the same equations in the patent [12] which are used to calculate crab angle can be used to directly calculate the heading angle as well. This is a potential method, when combined with a good mathematical model using the orientation and the position estimates; further a Kalman filter derivative can be implemented to form the estimates. However this implementation will not be carried out in the thesis and shall be left as a scope of work for the thesis. Following are few plots comparing results for both the methods. It can be observed from the plots that results of "method 1" outperform the results of "method 2".

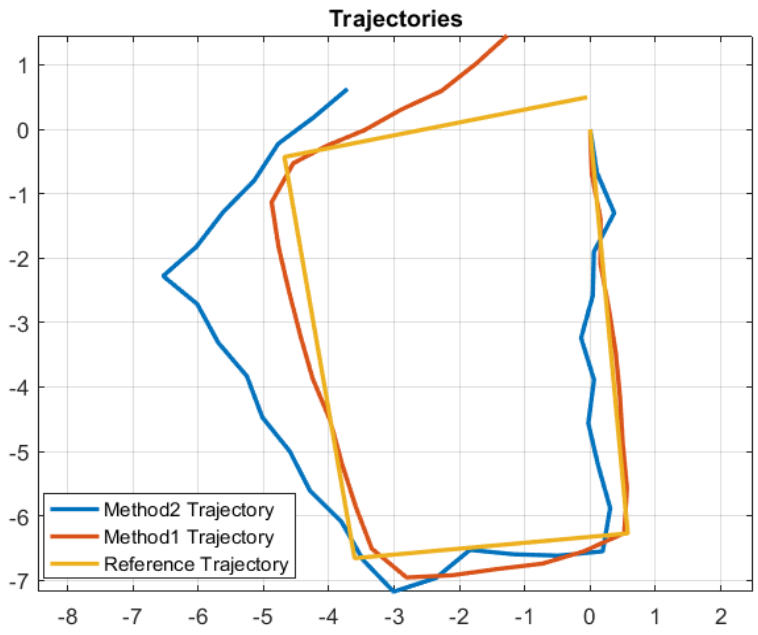


Figure 6.1: Rectangular Walk Comparison

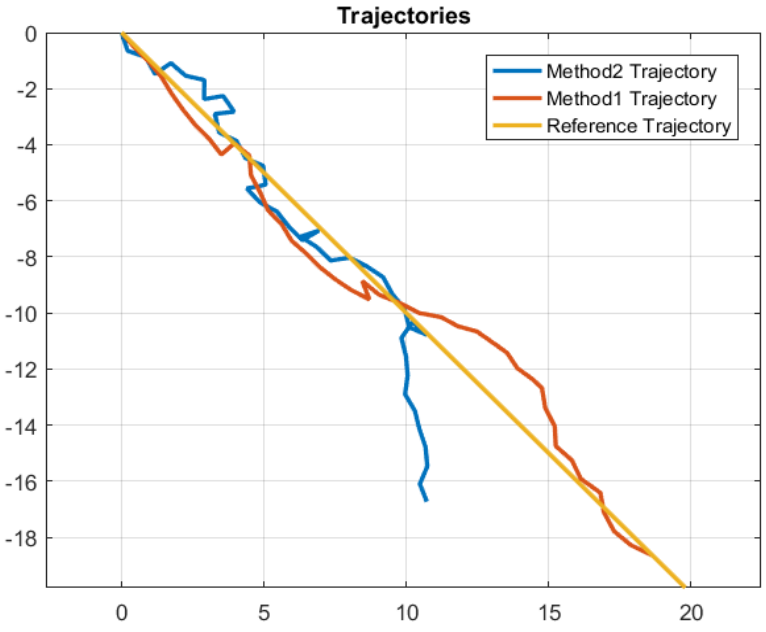


Figure 6.2: Straight Line Walk Comparison

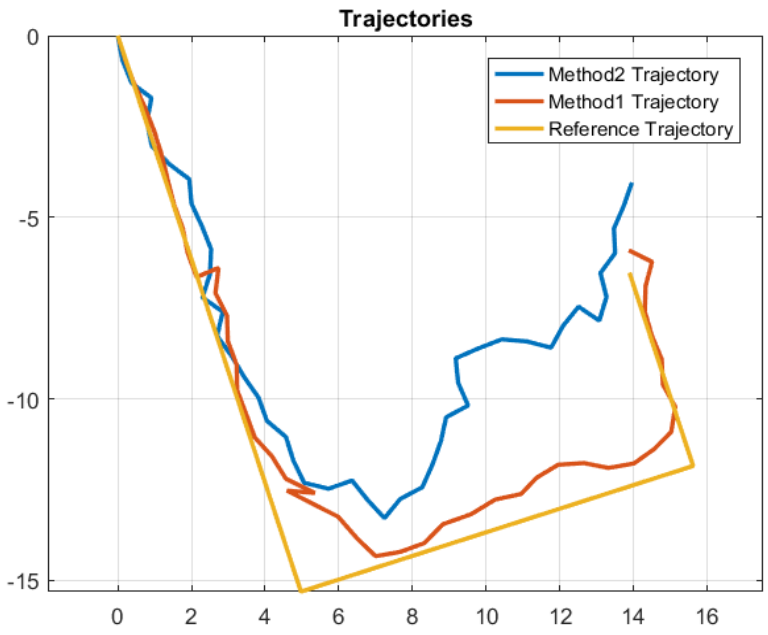


Figure 6.3: Comparing results from both the mwthods against a reference trajectory

Chapter 7

Limitations and Future Works

A couple of navigation methods have been proposed in the thesis and as already mentioned in the previous chapter "method 2" has been presented only to give more insights of the crab angle equation and no more discussions about it shall be done in the current chapter. The thesis mostly focuses on the "method 1" based on the the crab angle and from the results chapter it could be observed that the filtered trajectory based on "method 1" is noisy at certain places and sometimes deviates from the reference trajectory owing to certain limitations in the proposed method. This chapter discusses the limitations in the thesis and the possible future works based on the limitations.

7.1 Orientation Estimate

As already mentioned in the previous chapter the orientation estimate of the device does not consider translation motion of the device in the world frame. This constraint gives rise to approximate orientation estimate, which is very important to determine the horizontal plane. Hence this section can be worked upon by considering the translation of the device in the world frame to improve the orientation estimate

7.2 Step Detection

The step detection algorithm can be tricked of step detection even by standing still and accelerating the device. This algorithm can be improved by detecting standstill mode and neglecting all the measurements in the standstill mode.

7.3 MSE estimates and Evaluation of the Algorithm

The accuracy of the algorithms has not been evaluated numerically in the thesis due to time constraints. This can be pursued in the future by collecting multiple data sets for a given reference path and implementing the algorithm on the data sets which enables one to calculate the error through MSE.

7.4 Adding Radio Measurements

As already stated the algorithms were developed keeping in mind that they shall be used in conjunction with other positioning methods. Since the company Senion AB, where the thesis has been done, deals with indoor positioning systems using radio beacons. The algorithms shall be used in conjunction with the radio measurements for positions. Since only the results in "method 1" are good, a short section about adding radio measurement to navigation method 1 shall be discussed.

In the dead reckoning method, the magnetometer readings have been taken as measurements to estimate the heading of the pedestrian. Since the magnetometer measurements are susceptible to noise due to magnetic material in the surrounding, they can be eliminated and the radio beacon measurements for positions can be implemented using the following motion and measurement models for which an Extended Kalman Filter can be implemented.

Motion Model: same as equation 4.1

$$
\begin{aligned}
X_k &= X_{k-1} + (\bar{L} + v_{k-1}) \cos(\psi_{k-1} + \Delta_k^{\psi} - \delta_k) \\
Y_k &= Y_{k-1} + (\bar{L} + v_{k-1}) \sin(\psi_{k-1} + \Delta_k^{\psi} - \delta_k) \\
\psi_k &= \psi_{k-1} + \Delta_k^{\psi} + q_{\psi}
\end{aligned}
$$

Measurement Model

$$
\begin{aligned}
X_k^m &= X_k + e_x \\
Y_k^m &= Y_k + e_y
\end{aligned}
\tag{7.1}
$$

where e_x and e_y are the noise in the position estimates from the radio beacons.

The motion and measurement model above shows how to use inertial navigation systems in conjunction with other indoor positioning systems.

Chapter 8

Conclusion

The aim of the thesis is to track the path of a pedestrian by estimating the positions and the heading angle, with an IMU based hand-held devices like smartphones. In doing so, sub problems of step detection step length estimation and heading angle estimations have been considered. Heading angle estimation is a well known problem for PDR based on hand-held devices, where estimating the heading angle depends strongly on the orientation of the phone; this problem of estimating the heading angle irrespective of the orientation of the phone has been focused in the thesis. A couple of ways to estimate the heading angle based on calculations presented in the patent by Senion AB has been discussed, consequently leading to propose a couple of navigation methods.

Test Scenarios have been formulated to evaluate the performance of the proposed methods. Data has been collected based on the formulated test scenarios for various walking situations with a smart phone in the hand. Results have been generated for both the navigation methods based on the collected data. The approach using the crab angle has passed all the test scenarios by closely tracking the reference paths, while the other approach did not show promising results due to noise in the accelerometer data still being present in the calculations to estimate the heading angle. Finally the limitations of the proposed methods has been presented and possible future investigations on the thesis has been discussed.

The closing statement to conclude the thesis is that crab angle equations presented in the patent [12] can be termed as "leg-trackers", as they track the direction of the legs with respect to the device.

Appendix A

Appendix

A.1 Discretizing and Linearizing the Orientation model

The continuous time gyroscopic orientation model for gyroscopic measurements ω and white Gaussian noise in the measurements v is given by (A.1)

$$\begin{aligned} \dot{q} &= \frac{1}{2} S(\omega(t) + v(t)) q(t) \\ S(w) &= \begin{bmatrix} 0 & -\omega^x & -\omega^y & \omega^z \\ \omega^x & 0 & \omega^z & \omega^y \\ \omega^y & -\omega^z & 0 & \omega^x \\ \omega^z & \omega^y & -\omega^x & 0 \end{bmatrix} \end{aligned} \tag{A.1}$$

Discretizing the above equation in a sampling interval "T" between two discrete time instants $[t_{k-1}, t_k]$ such that the values of $\omega(t)$ and $v(t)$ in the sampling interval remain constant. By this assumption $\omega(t)$ becomes ω_{k-1} where k is any discrete time instant and $v(t)$ becomes v_{k-1}. With these assumptions the time continuous equation can be discretized as follows

$$\begin{aligned} \dot{q} &= \frac{1}{2} S(w_{k-1} + v_{k-1}) q(t) \\ \frac{dq}{dt} &= \frac{1}{2} S(w_{k-1} + v_{k-1}) q(t) \\ \int_{t_{k-1}}^{t_k} \frac{dq}{q} &= \int_{t_{k-1}}^{t_k} \frac{1}{2} S(w_{k-1} + v_{k-1}) dt \\ q_k &= exp^{\frac{1}{2} S(w_{k-1} + v_{k-1})^T} q_{k-1} \end{aligned} \tag{A.2}$$

Expanding the exponential term in the expression for q_k upto 1st order using Taylor Series:

$$q_k = exp^{\frac{1}{2} S(w_{k-1} + v_{k-1})^T} q_{k-1} = (I + \frac{T}{2} S(w_{k-1} + v_{k-1})) q_{k-1} \tag{A.3}$$

where I is the identity matrix

It can be easily verified that $S(\omega_{k-1} + v_{k-1}) = S(\omega_{k-1}) + S(v_{k-1})$. Using this analogy (A.3) can be reformulated as

$$q_k = (I + \frac{T}{2}S(\omega_{k-1}))q_{k-1} + \frac{T}{2}S(v_{k-1})q_{k-1} \tag{A.4}$$

let $I + \frac{T}{2}S(w_{k-1})$ be denoted by $F(\omega_{k-1})$ and let $\frac{T}{2}S(v_{k-1})q_{k-1}$ be reformulated as $G(q_{k-1})v_{k-1}$ which would reformulate the equation (A.4) as (A.5)

$$\begin{aligned} q_k &= F(w_{k-1})q_{k-1} + G(q_{k-1})v_{k-1} \\ F(\omega_{k-1}) &= \begin{bmatrix} 1 & -\frac{T}{2}\omega_x & -\frac{T}{2}\omega_y & -\frac{T}{2}\omega_z \\ \frac{T}{2}\omega_x & 1 & \frac{T}{2}\omega_z & -\frac{T}{2}\omega_y \\ \frac{T}{2}\omega_y & -\frac{T}{2}\omega_z & 1 & \frac{T}{2}\omega_x \\ \frac{T}{2}\omega_z & \frac{T}{2}\omega_y & -\frac{T}{2}\omega_x & 1 \end{bmatrix} \\ G(q_{k-1}) &= \frac{T}{2}\begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & -q_1 & -q_0 \end{bmatrix} \end{aligned} \tag{A.5}$$

A.2 Implementing EKF for device orientation

Motion Model From A.5

$$q_k = F(w_{k-1})q_{k-1} + G(q_{k-1})v_{k-1}$$

Measurement Model From 2.13

$$y_k^a = R_q^T(\hat{q}_k) + e_k$$

Linearized Measurement Model

$$y_k^a = R_q^T(\hat{q}_k) + \frac{\partial R_q^T}{\partial t}(\hat{q}_k)(q_k - \hat{q}_k) + e_k \tag{A.6}$$

The Ekf implementation to estimate orientation for the given motion and measurement models is given by the following prediction and update steps.

Prediction Update

The prediction step update of EKF is given by the following formulas where P_{k-1} is the previous time step posterior covariance matrix and P_k is the covariance matrix at time index k for the prediction step.

$$\hat{q}_k = F(\omega_{k-1})\hat{q}_{k-1} \tag{A.7}$$

$$P_k = F(\omega_{k-1})P_{k-1}F^T(\omega_{k-1}) + G(\hat{q}_{k-1})H_\omega G^T(\hat{q}_{k-1}) \tag{A.8}$$

The prediction update is followed by normalizing the quaternions.

Measurement Update

The measurement model update is obtained by the following equations,

$$S_k = \frac{\partial R_q^T}{\partial t}(\hat{q}_k) P_k R_q^{'}(\hat{q}_k) + E_a \tag{A.9}$$

$$K_k = P_k \frac{\partial R_q}{\partial t}(\hat{q}_k) S_k^{-1} \tag{A.10}$$

$$\hat{q}_{k|k} = \hat{q}_k + K_k(y_k^a - R_q^T(\hat{q}_k)) \tag{A.11}$$

$$P_{k|k} = P_k - K_k S_k K_k^T \tag{A.12}$$

$$\tag{A.13}$$

A.3 Linearization and EKF for Non Linear Motion and Measurement Models

The formula for linearizing a non linear equation with a vector of states x and standard gaussian densities at $(\hat{x}, 0)$

$$y = h(x, r) = h(\hat{x}, 0) + \left.\frac{\partial h}{\partial x}\right|_{\hat{x},0} (x - \hat{x}) + \left.\frac{\partial h}{\partial r}\right|_{\hat{x},0} (r - 0) \tag{A.14}$$

Linearizing the motion model

$$\begin{aligned}
\begin{bmatrix} X_k \\ Y_k \\ \psi_k \end{bmatrix} &= \begin{bmatrix} \hat{X}_{k-1/k-1} + (L_{k-1} + \hat{v}_{k-1}) cos(\hat{\psi}_{k-1} - \delta_{k-1}) \\ \hat{Y}_{k-1/k-1} + (L_{k-1} + \hat{v}_{k-1}) sin(\hat{\psi}_{k-1} - \delta_{k-1}) \\ \hat{\psi}_{k-1/k-1} + T_k w_Z^k + \hat{R}_\psi \end{bmatrix} \\
&+ \begin{bmatrix} 1 & 0 & -(L_{k-1} + \hat{v}_{k-1}) sin(\hat{\psi}_{k-1} - \delta_{k-1}) \\ 0 & 1 & (L_{k-1} + \hat{v}_{k-1}) cos(\hat{\psi}_{k-1} - \delta_{k-1}) \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_{k-1} - \hat{X}_{k-1/k-1} \\ Y_{k-1} - \hat{Y}_{k-1/k-1} \\ \psi_{k-1} - \hat{\psi}_{k-1/k-1} \end{bmatrix} \\
&+ \begin{bmatrix} cos(\hat{\psi}_{k-1} - \delta_{k-1}) & 0 \\ sin(\hat{\psi}_{k-1} - \delta_{k-1}) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_{k-1} \\ R_\psi \end{bmatrix}
\end{aligned} \tag{A.15}$$

Substituting $\hat{v} = 0$, the matrices become

$$\begin{aligned}
\underbrace{\begin{bmatrix} X_k \\ Y_k \\ \psi_k \end{bmatrix}}_{Z} &= \underbrace{\begin{bmatrix} \hat{X}_{k-1/k-1} + (L_{k-1}) cos(\hat{\psi}_{k-1} - \delta_{k-1}) \\ \hat{Y}_{k-1/k-1} + (L_{k-1}) sin(\hat{\psi}_{k-1} - \delta_{k-1}) \\ \hat{\psi}_{k-1/k-1} + T_k w_Z^k \end{bmatrix}}_{B_{k-1}} + \underbrace{\begin{bmatrix} 1 & 0 & -(L_{k-1} sin(\hat{\psi}_{k-1} - \delta_{k-1}) \\ 0 & 1 & (L_{k-1}) cos(\hat{\psi}_{k-1} - \delta_{k-1}) \\ 0 & 0 & 1 \end{bmatrix}}_{A_{k-1}} \begin{bmatrix} X_{k-1} - \hat{X}_{k-1/k-1} \\ Y_{k-1} - \hat{Y}_{k-1/k-1} \\ \psi_{k-1} - \hat{\psi}_{k-1/k-1} \end{bmatrix} \\
&+ \underbrace{\begin{bmatrix} cos(\hat{\psi}_{k-1} - \delta_{k-1}) & 0 \\ sin(\hat{\psi}_{k-1} - \delta_{k-1}) & 0 \\ 0 & 1 \end{bmatrix}}_{G_{k-1}} \begin{bmatrix} v_{k-1} \\ R_\psi \end{bmatrix}
\end{aligned} \tag{A.16}$$

Linearizing the non linear measurement model in equation 4.4 according to the Taylor series expansion stated in A.14 at $(\hat{X}, \hat{Y}, \hat{\psi}, 0)$

$$\begin{bmatrix} m_h^x \\ m_h^y \end{bmatrix} = \begin{bmatrix} \cos(\psi_k + r^x) \\ \sin(\psi_k + r^y) \end{bmatrix}$$

$$\begin{bmatrix} m_h^x \\ m_h^y \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(\hat{\psi}_k) \\ \sin(\hat{\psi}_k) \end{bmatrix}}_{H_k} + \underbrace{\begin{bmatrix} 0 & 0 & -\sin\hat{\psi} \\ 0 & 0 & \cos\hat{\psi} \end{bmatrix}}_{H_{kd}} \begin{bmatrix} X_k - \hat{X}_k \\ Y_k - \hat{Y}_k \\ \psi_k - \hat{\psi}_k \end{bmatrix} + \underbrace{\begin{bmatrix} -\sin(\hat{\psi}_k) & 0 \\ 0 & \cos\hat{\psi}_k) \end{bmatrix}}_{T_k} \begin{bmatrix} r^x \\ r^y \end{bmatrix} \tag{A.17}$$

The motion model is now linearized and for a linear measurement model .Apply the standard EKF formulas written below.

Motion model Prediction step

$$\begin{aligned} \hat{Z}_{k/k-1} &= B_{k-1} \\ P_{k/k-1} &= A_{k-1} P_{k-1/k-1} A_{k-1}^T + G_{k-1} Q_{k-1} G_{k-1}^T \end{aligned} \tag{A.18}$$

Measurement model update step

$$\begin{aligned} \hat{Z}_{k/k} &= \hat{Z}_{k/k-1} + K_k V_k \\ P_{k/k} &= P_{k/k-1} - K_k S_k K_k^T \\ K_k &= P_{k/k-1} H_{kd}^T S_k \\ V_k &= y_k - H_k \hat{X}_{k/k-1} \\ S_k &= H_{kd} P_{k/k-1} H_{kd}^T + T_k R_k T_k^T \end{aligned} \tag{A.19}$$

Bibliography

[1] Zhi-An Deng et al. "Heading estimation for indoor pedestrian navigation using a smartphone in the pocket". In: *Sensors* 15.9 (2015), pp. 21518–21536.

[2] H Guo et al. "Indoor positioning based on foot-mounted IMU". In: *Polska Akademia Nauk. Bulletin of the Polish Academy of Sciences* 63.3 (2015), p. 629.

[3] Julius Hannink et al. "Mobile Stride Length Estimation with Deep Convolutional Neural Networks". In: *IEEE Journal of Biomedical and Health Informatics* (2017).

[4] Parinaz Kasebzadeh et al. "Improved Pedestrian Dead Reckoning positioning with gait parameter learning". In: *Information Fusion (FUSION), 2016 19th International Conference on*. IEEE. 2016, pp. 379–385.

[5] Valérie Renaudin and Christophe Combettes. "Magnetic, Acceleration Fields and Gyroscope Quaternion (MAGYQ)-Based Attitude Estimation with Smartphone Sensors for Indoor Pedestrian Navigation". In: *Sensors* 14.12 (Dec. 2014), pp. 22864–22890. ISSN: 1424-8220. DOI: 10.3390/s141222864. URL: http://dx.doi.org/10.3390/s141222864.

[6] Valérie Renaudin, Melania Susi, and Gérard Lachapelle. "Step length estimation using handheld inertial sensors". In: *Sensors* 12.7 (2012), pp. 8507–8525.

[7] Nirupam Roy. "WalkCompass: Finding Walking Direction Leveraging Smartphone's Inertial Sensors". PhD thesis. University of South Carolina, 2013.

[8] Antonio Ramón Jiménez Ruiz et al. "Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements". In: *IEEE Transactions on Instrumentation and Measurement* 61.1 (2012), pp. 178–189.

[9] "Step Length Approximation". In: (). URL: http://www.livestrong.com/article/438170-the-average-walking-stride-length/.

[10] Ross Grote Stirling. "Development of a pedestrian navigation system using shoe mounted sensors". PhD thesis. University of Alberta, 2004.

[11] Yi Sun, Huaming Wu, and Jochen Schiller. "A step length estimation model for position tracking". In: *Localization and GNSS (ICL-GNSS), 2015 International Conference on*. IEEE. 2015, pp. 1–6.

[12] David Tornqvist, Fredrik Gustafsson, and Per Skoglar. *Determining Sensor Orientation in Indoor Navigation*. US Patent App. 14/736,813. June 2015.

[13] Jens Windau and Laurent Itti. "Walking compass with head-mounted IMU sensor". In: *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE. 2016, pp. 5542–5547.

[14] Xiaokun Yang and Baoqi Huang. "An accurate step detection algorithm using unconstrained smartphones". In: *Control and Decision Conference (CCDC), 2015 27th Chinese*. IEEE. 2015, pp. 5682–5687.