Tampere University

Siva Sankar Kannan

MODELLING OF CROWDSOURCED WI-FI FINGERPRINT DATA

Faculty of Engineering and Natural Sciences
Master's Thesis
June 2024

ABSTRACT

Siva Sankar Kannan: Modelling of crowdsourced Wi-Fi fingerprint data
Master of Science Thesis
Tampere University
Automation Engineering (MSc)
June 2024

The lack of and/or the unreliability of GPS signals indoors poses unique challenges with accurate indoor navigation. This thesis proposes an idea that aims to addresses these challenges by leveraging Wi-Fi fingerprinting to augment Pedestrian Dead Reckoning (PDR) based Inertial Navigation Systems (INS).

Wi-Fi fingerprinting involves the collection of Wi-Fi signal strengths from multiple access points, which are then used to model the relationship between fingerprint dissimilarity and real-world distances. Wi-Fi fingerprint data can be modelled through crowdsourced Wi-Fi fingerprint data. This model is crucial for enhancing indoor navigation accuracy where GPS data is unavailable. The research introduces a sophisticated approach using Weighted Least Squares regression with linear scaling weights to refine the estimation process. The Wi-Fi fingerprint model is used to filter out unreliable PDR data, considerably improving the location estimation accuracy. It employs a dual-model approach that allows utilisation of known reference points such as GPS fixes when available or Bluetooth beacons as indoor landmarks to further enhance the reliability of the navigation system.

A weighted algorithm prioritizing data points based on their estimated reliability, effectively reducing the influence of poor-quality data on the overall system performance is used. This method shows a marked improvement in positioning accuracy, thus demonstrating the feasibility and effectiveness of integrating Wi-Fi fingerprinting with traditional inertial navigation methods.

The findings showcase the potential of using Wi-Fi fingerprint modelling as a powerful augmenting technology for PDR-based INS (Inertial Navigation System), offering improvements over existing methods, particularly in complex indoor environments. The research also lays the groundwork for future advancements in indoor navigation technologies, opening avenues for more reliable and accurate indoor positioning solutions that can operate without GPS.

Keywords: thesis, weighted least squares, Wi-Fi positioning, Wi-Fi fingerprints

PREFACE

This thesis titled ‘Modelling of Crowdsourced Wi-Fi fingerprint data’ is a culmination of my research and work as part of my Master of Science Degree in Automation Engineering at Tampere University. The work aims to address the challenges with indoor navigation systems, focusing on the potential of using Wi-Fi fingerprint data to improve the accuracy and reliance of the indoor navigation system.

The main motivation for the study stems from the practical troubles and tribulations relating to indoor navigation, be it in the form of finding the right classroom at a new building at school, or the right store in a mall. The limitations of the current indoor navigation technologies, often struggling to provide consistent and accurate results, stem from the nature of the complex dynamics in an indoor environment. By taking advantage of crowdsourced data and statistical modelling techniques, the thesis proposes a novel approach in refining the estimation capabilities of the indoor navigation system.

Throughout the research process, I had the privilege of working under the invaluable guidance and support of my supervisors and professors, Robert Piché and Simo Ali-Löytty. Their expertise, patience and insights have been invaluable to the progress of the research. An additional thanks to Simo for his unparalleled patience, objective reasoning and emotional understanding which were quintessential in me completing the thesis. I would also like to acknowledge the financial support from HERE maps, which funded the research project along with the necessary tools and support needed to progress the project. I am also indebted to the University for the continuous support with the software and the computational power that aided in the quick resolution of the research.

Additionally, A special thanks to my managers and co-workers at Trimble who always had my back and helped keep up the motivation when things were rough and helped alleviate the stress to finally rack up and finish the document.

Finally, the undying moral support and endless motivation from my friends and family set the bedrock for my perseverance through the academic pursuit of Masters.

This thesis reflects my academic journey through Tampere University and lays the foundational groundwork for future research in the field of indoor navigational systems. I hope that my work will inspire further research and innovations and contribute towards the development of a more robust indoor navigational system.

Sincerely yours,

Siva Sankar Kannan.

Finland, 2024.

CONTENTS

TABLE OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

AP	Access Point
GPS	Global Positioning System
IMU	Inertial Measurement Unit
LSQ	Least-Squares
PDR	Pedestrian Dead Reckoning
PF	Particle Filter
RSS	Received Signal Strength
UILoc	Unsupervised Indoor Localization
Wi-Fi	Wireless Fidelity
Wi-Fi RTT	Wireless Fidelity Round-Trip-Time
LSR	Least-Squares Regression
R	Residual from Least-Squares Regression
d	distance/dissimilarity between fingerprints
d_e	Estimated distance/dissimilarity between fingerprints
d_o	Observed distance/dissimilarity between fingerprints
x	x-coordinate
y	y-coordinate
F	Wi-Fi fingerprint
$Distance_{1,2}$	Euclidean distance between two points
A_1	Design Matrix for known points
A_2	Design Matrix for Wi-Fi fingerprint distance/dissimilarity
b_1	Observation Vector for known points
b_2	Observation Vector for fingerprint distance/dissimilarity
$\vec{x}$	Parameter vector that contains the coordinates.
v	Final velocity
u	Initial velocity
a	Acceleration
t	Time
m	Slope
c	y-intercept

LIST OF EQUATIONS

1 INTRODUCTION

This section intends to introduce the research and give some context behind the research. It lays the groundwork for the scope of the research while outlining the research question and explains the order of steps that were taken in getting to the final research conclusion.

To preface the introduction, indoor navigation has been getting increasingly essential, and given the prevalence of Wi-Fi in most public spaces and buildings, a Wi-Fi supported indoor navigation becomes a quintessential landmark in the field of indoor navigation (Davidson et al, 2017). In this study, we seek to explore the efficacy of indoor navigation using Wi-Fi fingerprint, and Pedestrian Dead Reckoning data.

The main question that the study attempts to challenge is ‘How effective is a Wi-Fi supported indoor navigation system in accurately representing the user’s location’, especially in complex indoor environments. Through the course of the study, we aim to not only assess the current capabilities of the Wi-Fi assisted navigation, but also present the groundwork for future enhancements, for instance, Wi-Fi Round-Trip-Time (RTT) technology, or Bluetooth beacons for more precise location estimations (Perez-Navarro A et al, 2022). However, it is to be noted that while Wi-Fi RTT does offer promising improvements, it is outside the scope of this study, and is rather an avenue for future research.

The Thesis aims to explore the viability and effectiveness of a Wi-Fi assisted indoor navigation system, leveraging the precision of high-end Inertial Measurement Unit (IMU) for obtaining high-accuracy location data. The accuracy of this data is crucial for evaluating the system’s performance. By combining Pedestrian Dead Reckoning (PDR) data (A localized positioning method that utilizes Inertial sensors like Accelerometers, Gyroscopes, etc.) and Wi-Fi Fingerprint data (A collection of the Wi-Fi Access Points and their corresponding signal strengths) collected from a standard smartphone, the study compares the navigational efficacy against the precise reference data. This analysis is pivotal in determining the potential enhancements that Wi-Fi could bring about in the field of indoor navigation systems. A model is proposed which would serve as the

basis for crowdsourced Wi-Fi fingerprint distance estimation, which is then in turn used to assist in estimating the coordinates in an indoor navigation application.

A critical component to the methodology is the application of Least-Squares Regression, a statistical method, to determine the mathematical correlation between the Wi-Fi signal strength dissimilarities to their corresponding real-world distances (K. He, Y. Zhang et al, 2015). Integrating these advanced techniques together results in a comprehensive assessment of Wi-Fi's capability in augmenting indoor navigation systems.

To brief the idea behind the thesis, Pedestrian Dead Reckoning based location estimation in indoor navigation suffers from its downsides of accumulating error and needing re-calibration (X. Liu et al, 2022). The study aims to estimate the efficacy of using Wi-Fi fingerprint data to get a better estimate of the indoor coordinates.

The process plan of the thesis can be split into 5 key parts.

1. Methodology
2. Data collection
3. Data processing
4. Location estimation
5. Comparison

Here's a small introduction on the process plan before delving into them deep in the further chapters later in the thesis.

1.1 Methodology

The primary technique that is used in the study is 'Least-Squares Regression'; it is a statistical method that works well in overdetermined situations (Chapra et al, 2010). It is used in two different stages, the first is in determining the mathematical relationship between Wi-Fi fingerprint dissimilarities and their real-world geometric distances. The second stage is to use the distance information between timestamps in the recording, to find the cartesian coordinates of the timestamps. The second processing method that is used in this study is using Newton's Laws of motion, along with filtering algorithms to estimate distances

from Pedestrian Dead Reckoning (PDR) data. For the purposes of this study, this processing was done with the help of HERE maps.

For the first stage, data is collected by walking around the campus in Tampere University, whilst recording the Wi-Fi signal strengths of all the Access Points reachable on the phone at the given time and location. Access Points (AP) here are essentially Wi-Fi routers that transmit Wi-Fi signals. This will be explained in more detail in the later parts of the thesis. The recorded data is then used along with the information of the distance travelled between recordings, to estimate the relation between the dissimilarity of Wi-Fi signal strengths at two timestamps and the real-world distance between the two timestamps.

For the second stage, a set of distances recorded between timestamps, is used with Least-Squares Regression to estimate the position in cartesian coordinates. The distances recorded here in this dataset is the displacement between two timestamps. This is then cross referenced against the reference data to evaluate the efficacy of using Wi-Fi in indoor navigation use case.

1.2 Data Collection

Data collection was done with the help of two devices. The ‘MTw XSENS’ Inertial-Measurement-Unit (IMU) sensor which is a high-accuracy sensor to record the Inertial information, such as the Linear Acceleration in 3D space and the Angular acceleration in the 3 axes. The second device that was used was a Sony Xperia X, and android phone, which was used to record the Wi-Fi fingerprint data along with the Inertial information (same as the MTw XSENS sensor). There were two different data collection segments. The initial segment was to identify mathematical correlation between Wi-Fi signal strength dissimilarities and their corresponding real-world distances. And the second segment being the actual test segment which compares the results against each other. Both the segments were recorded simultaneously. The data collection methodology is like that proposed by (Zhuang et al, 2015) which uses a dedicated IMU (Inertial Measurement Unit) and the smartphone for measuring the Wi-Fi fingerprint data and the PDR (Pedestrian Dead Reckoning) data.

For the first segment, only the smartphone was used to record the Wi-Fi fingerprint data. This recording was done with the help of a smartphone app which

recorded the fingerprint data and then allowed it to be exported to a file. As for the location and distance, a straight corridor was chosen with a fixed length. Walking through the corridor at a constant pace and inferring the data allows us to get a hold of the approximate distance/location at any given point in time. Since the recording was done with known distances being covered, it wasn't necessary to record high-accuracy inertial data from the main sensor.

The second segment uses the combination of both the MTw XSENS sensor and the smartphone. Both being strapped to the chest of the user, while walking through corridors in Tampere University. The recording was done in short bursts to avoid the accumulation of error from inertial measurements. And both the devices were synchronized with a GPS clock to ensure easy and automatic alignment of the datasets. The full reasoning and data collection would be detailed later in the study.

1.3 Data processing

Before the collected data can be used to validate the efficacy and possibility of using Wi-Fi, it needs to be processed. This section will also house the information on modelling the mathematical correlation between Wi-Fi signal strength dissimilarities and their real-world distances. This is done in MATLAB by calculating the pair-wise distance between the Access Points and their respective Received Signal Strength (RSS) values. Pair-wise distance in this context is the dissimilarity between each pair of Wi-Fi access points in the dataset. The next step is to map this RSS dissimilarity to the real-world distances. This is done by fitting a curve to find the mathematical correlation between them, giving us a way to estimate the approximate distance from Wi-Fi RSS dissimilarities.

The second set of processing comes from preparing the Inertial Data collected from the Inertial Measurement Units (IMUs) so that it can be used for reference data. This processing employs filtering to filter out unnecessary data, noise, and other unwanted aspects of the inertial data, while accounting for the error that is inherent to Inertial sensors. Then, Newton's laws of motion are employed to estimate the distance between different timestamps. For the purposes of this study, this was done with the help of HERE maps, who were responsible for converting the Pedestrian Dead Reckoning (PDR) data into a distance map. The

same is done for the PDR data from the smartphone. With these done, the pair-wise distance map can be now used to estimate the location at the different timestamps in real-world cartesian coordinates.

1.4 Location Estimation

Since we have processed the data to have pair-wise distance between a lot of points, we have an overdetermined system, which is to say, we have a lot more equations than variables. To estimate the location in cartesian coordinates, we assume the start point as the origin, and then use Least-Squares Regression to minimize the Squares Residual error and find the coordinates that would best suite the data that we have. This is done for both the reference data and the data from the phone.

For the reference data, we have multiple points in our dataset where the location is already known, such as the start of a corridor or an end of a corridor. These points of data are marked as ‘known’ points. This is helpful in making sure our reference is as accurate as possible. And to use this while determining the coordinates for the timestamps, we employ a weighted Least-Squared Regression where the known coordinates are marked with maximum weight, giving us a reliable coordinate set for reference.

For the observed/test data, the same is performed with both Inertial (Pedestrian Dead Reckoning, PDR) data and the distance estimates from the Wi-Fi fingerprints. By careful selection of fingerprint pairs and filtering out fingerprint pairs from their Wi-Fi fingerprint dissimilarities and adding weights to fine-tune the model; a good estimate for the coordinates is determined (Vilaseca, D et al, 2013) (Zhuang. Y et al, 2015).

Another aspect of the study was to also check the possibility of augmenting this accuracy even further. This was done by assigning some of the coordinates as ‘known’ points, much like in the case of the reference data. This is done to mimic having known points of interest in a place such as a mall, or having additional supportive elements such as Bluetooth beacons which would stand out as landmarks. Estimating distance using methods such as Wi-Fi RTT (Round Trip Time) to have a high accuracy distance estimation is not part of this study but is a good place to expand the study on. This thesis serves as an enabler for future

studies pertaining to other technologies that might augment the accuracy even further.

1.5 Comparison and inference

All the mathematical modelling, distance estimation and location estimation are done through the software called MATLAB (Mathworks, 2024). Licensing of which was provided by Tampere University. The way data is stored in the software, and modelled will be explained in more detail later in the study. The way the data is compared, and inference is drawn is done through graphs that are plotted with the help of MATLAB.

To ensure clarity and consistency, the representation of the reference coordinates, obtained from the high-end IMU sensors are marked with a blue circle in the graphs. While the estimated/observed values derived from the smartphone data (combination of Wi-Fi fingerprint data and the PDR data) are marked with a red asterisk. Given different configurations, we compare the proximity of the coordinate results from the phone against the reference values from the high-end IMU sensors. All distances showcased in the graphs are in meters unless explicitly stated otherwise.

While the graphs seek to present a visual understanding of how accurate the estimated values are, it is necessary to note that the quantitative analysis of the standard deviation and mean error is crucial in determining the system's performance. These statistical metrics provide a way of evaluating the real-world implications of the indoor navigation system. Given the practical use-case, such as a mall or public buildings, achieving sub-meter accuracy isn't paramount. The main goal of the study is to shine light on the possibility of a powerful indoor navigation system assisted by Wi-Fi fingerprint data, while enabling further research and studies.

In summary, the primary research problem the thesis aims to address is the effectiveness of using crowdsourced Wi-Fi fingerprint data to augment Pedestrian Dead Reckoning (PDR) based Indoor Navigational Systems. Specifically, the research investigates whether this combined approach provides tangible improvements in positioning accuracy within GPS-lacking indoor environments. The findings of this study are expected to highlight the potential enhancements

that Wi-Fi fingerprinting can bring to indoor navigation, offering a foundation for future advancements in this field.

2 LITERATURE REVIEW

In this section, other research and progresses made by other researchers are noted, along with the dictation of the novel approach that is introduced by this paper. Indoor navigation is a problem that is as old as time itself, having been a highly contested market to introduce an effective and efficient navigation system that works indoors, not just for public places such as malls, but also for places like warehouses. Some of the popular technologies that often take the call for research are Inertial measurement units on mobile phones, various wireless technologies such as Wi-Fi, Ultra-Wide band, Bluetooth, and to some level, even NFC (Near Field Communication). This section aims to provide a bigger context by incorporating a comparative analysis with the current methodologies to highlight the novelty showcased in the paper.

Indoor navigation has a lot of innate limitations, being a closed and often highly packed system where wireless technologies that are typically used for navigation fail, as expressed by this study from (Avi Matza et al, 2012) designing an indoor navigation system from a practical perspective. The study explores the issues that are present in an indoor navigation system composed of cellular mobile technology, namely Galileo Satellite Navigation with off the shelf GPS modules. The study concludes with an algorithm that achieves a navigational accuracy of a few meters.

In Literature (Abusara, 2015), the poor performance of GPS indoors is noted and alternative methods for indoor navigation are explored. Some of the methods include ‘Pedestrian Dead Reckoning’, ‘Fingerprint based localization’ and ‘Trilateration’. The proposed hybrid method in the study (Lu et al., 2016) showcases the efficiency improvement towards indoor navigation using a hybrid method which combines multiple technologies to cover up the downsides of the different approaches towards indoor navigation.

A trilateration-based solution works exactly like a conventional GPS system, which uses 3 or more Wi-Fi signals to find the location of the person. The downside to this approach is that Wi-Fi signal strength is prone to a lot of noise and/or error by nature and requires that we know the position of the Wi-Fi nodes

beforehand for it to work. Therefore, it is unusable in places where one does not have the node locations. Such an approach reaches room level accuracy as shown by (Chen, Zhang and Xue, 2018).

A study done by (Moghtadaiee and Dempster, 2014) showcased a fingerprint-based approach that used a K-nearest neighbour approach for indoor localization using FM (Frequency modulation radio waves) as an alternative to Wi-Fi based positioning system. This approach also added a signal strength compensation to account for the fluctuations and clustering to reduce errors. The combination of the K-nearest neighbours and Bayesian probability working simultaneously showcased a significant improvement towards indoor localization. While novel in its approach, the use of FM signals for indoor positioning isn't readily available as a standard compared to Wi-Fi which can be found in all mobile phones and poses as a challenge for integration and adoption.

Striving toward accuracy in indoor positioning using Wi-Fi, the TRIPS (time-reversal indoor positioning system) algorithm was proposed by (Wu, 2014). This approach assumes channel reciprocity and channel stationarity (forward and backward links of the channel are highly correlated and the channel impulse response should be stationary for at least one probing-and-transmission phase). While the above method focused on mitigating residual timing and frequency synchronization error, requires devices to support multiple frequency bands and which support sophisticated Wi-Fi hardware that are also capable of capturing detailed channel information. The dependence on the availability of the bandwidth also adds an additional layer of complexity. While showing promising results, the reliance on sophisticated hardware and setup makes this another approach that's limited by the existing hardware in standard public buildings, and consumer devices.

Pedestrian Dead Reckoning (PDR) takes advantage of the additional inertial sensors that are readily available in smartphones to estimate the relative and absolute position of a target by analysing the acceleration and direction data. Step sizes, lengths and directions are the prime estimates using this data; but magnetometer and gyroscope data can be added to make PDR give reliable estimates of position over short periods, provided the initial position estimate is available. The error accumulation and the mitigation methodologies are studied

in the study by (Ehad A, et al, 2013) whereby a self-resetting algorithm is proposed that aims to minimize the error accumulation in indoor navigation systems involving vehicles. While showcasing promising results, doesn't directly address pedestrian tracking for indoor navigation and is aimed at vehicles in a 2D space without any rotations or turns as a proof of concept.

Other hybrid approaches involving multiple sensors commonly involve the use of Bluetooth based beacons which serve as an anchor to reset the calibration of the PDR sensors and thereby minimize the accumulated error and improve the position estimation accuracy. The study by (Adam Satan, 2018) uses Bluetooth beacons placed at known locations to present an indoor navigation algorithm that works with the radio waves from Bluetooth transmission. Similar to other studies, this study shows shining results, however, is again limited with needing a dense network of Bluetooth beacons for comprehensive coverage, which is both costly and labour intensive to setup and maintain.

The proposed UILoc (Unsupervised Indoor Localization) by (Zhang Yi, et all, 2018) approach is another Hybrid localization approach which uses a PDR module to estimate the concurrent locations after an initial location has been set. The PDR system consists of: A 'particle filter' (PF) module which aims to reduce the error in the estimation of step length and direction; a reliable model which uses landmarks to correct the location for the PDR system; and a database building model which combines all the previous modules to estimate an accurate initial position for the PDR system.

An experiment of walking around a floor which resembles a typical office building was conducted and the proposed system was tested against existing methods. iBeacons (landmarks for the reliable model) were placed in a uniformly distributed pattern around the floor to have a comprehensive coverage of the floor. The test results show that UILoc has a mean error of as low as 1.11 m. These results suggest that the system is a viable low-cost solution to the localization problem.

This thesis aims to study the use of a crowdsourced Wi-Fi model that can be added on to the PDR based indoor navigation system, with beacons for reference position, for position estimation. An experiment is done with 'known' locations uniformly placed throughout the testing area as the premise. And then a

comparative analysis is done to verify the feasibility of using Wi-Fi to improve indoor navigation systems at low cost and maintenance, without requiring extensive pre-mapping of the floor or having to understand the floor plan beforehand.

3 METHODOLOGY

3.1 Least-Squares Regression

Least-Squares Regression (LSR) is a statistical method for modelling and analysing the relationship between variables, particularly useful in overdetermined system (Chapra S C et al, 2010). To break this down, an overdetermined system is one where the number of observations exceeds the number of unknowns, thereby enabling us to derive a model that best represents the system. This is done by minimizing the sum of the squares of the residuals from the equations, as suggested by the name – least squares. A residual is essentially the difference between the observed value and the estimated value from the fitting the model. This model is iteratively improved until we end up with a value that has the least residual squared.

$$arg\ min \sum_{i=1}^{n} \left({R_i}^2\right)$$

Equation 1 - Least-Squares Regression

Where n is the number of observations, and

$$R_i = (d_o - d_e)$$

Equation 2 - Least-Squares Residual

Where d_o is the observed data and d_e is the estimated data.

$$d_e = f(\vec{x})$$

Here, '$f(\vec{x})$' is a function of the coordinates, that represents the distance/dissimilarity between the coordinates. The full model along with how distance/dissimilarity is calculated will be detailed later in the specific sections.

In simple terms, Least-Squares regression attempts to find a line that would best represent a set of data points. For example, imagine trying to plot down the number of crops yielded on a graph against the amount of fertilizer used. Least-Squares regression would help us draw a line that best represents this dataset. The key part here is the 'over-deterministic' nature of this system. We are trying to identify the correlation between the number of crops yielded with respect to the

amount of fertilizer used, so a straight line. A linear line has 2 unknowns, namely, the slope and the intercept. But on the other hand, we have 100s of observations that we have plotted down in the graph, thus making it an ‘over-deterministic’ system. The benefit of having such a system is that it improves the accuracy and reliability of the system.

Residuals as already mentioned are the differences between the observed value (datapoints in the graph) and the estimated value (the value that is predicted by our linear line). Minimizing the overall residuals ensures that our model is as close as possible to the originally observed data. In the case of Least-Squares regression, we aim to minimize the sum of the squares of these differences to find the best fitting line/curve.

Having established a mathematical relationship between the Wi-Fi fingerprint dissimilarity and the real-world distance through Least-Squares Regression, the model can be used to predict future values within the system; to estimate the real-world distance from Wi-Fi fingerprint dissimilarity data, and vice versa (Evennou F et al, 2006).

For the purposes of this thesis, the use of Least-Squares Regression comes in two stages (Zhuang, Y. et al, 2015). The first stage where we try to find a correlation between the dissimilarities of Wi-Fi signal strengths from two timestamps and the real-world distances between the two Access Points. The second stage is to use Least-Squares regression to determine the correlation between the real-world distance between two points and their real-world coordinates/locations. The process of how this is done will be explained in the following sections.

3.1.1 Application within the thesis

In this study, given the number of observations for the dissimilarities in fingerprints at different timestamps far exceeds the number of actual timestamps, we can formulate a model that represents the mathematical relationship between the Wi-Fi fingerprint dissimilarity with their respective real-world Euclidean distances (Li W et al, 2015). To briefly touch upon the number of observations, essentially, for each timestamp that we record the Wi-Fi fingerprint data at, we

have Wi-Fi signal strengths of all the available Wi-Fi access points at that location. Given this information, we can find the dissimilarity of the Wi-Fi fingerprints between any two timestamp locations. For the sake of convenience, the data-recording that is captured at the timestamps will be referred to as 'observations' in this thesis. The second set of use comes in the form of estimating the cartesian coordinates (location) of these observations using the distance between the observations.

3.1.2 Wi-Fi fingerprint dissimilarity to Euclidean distance modelling

The Wi-Fi fingerprint dataset that is recorded has the Wi-Fi RSS (Received Signal Strength) values for all the Access Points (Wireless routers that serve as a Wi-Fi transmitter) that are within range of the location. In a general case, in a public space such as a mall, it is typical to have quite a lot of Wi-Fi APs (Access Points) within range, and not all the APs are generally within range. So, to ensure a proper Wi-Fi fingerprint dissimilarity is established, the dataset has to be pre-processed to allow us to calculate the differences between the different RSS values.

Within the software MATLAB, all the information is stored in the form of a 'Matrix'. A Matrix is a mathematical data-representation system where we have multi-dimension arrays holding the data-points. In case of a 2-dimensional array it would be synonymous to something like a table; where we have the data-points organized under a set of rows and columns. As for the data from the study, a Wi-Fi fingerprint would be the array of all the RSS values from every Access Point within range. And the dataset would be a matrix with the Wi-Fi fingerprint data for each timestamp in the data recording.

Fingerprint Euclidean distance estimation.

Euclidean distance is the distance between two points in the Euclidean space. And since the height of the location is not of concern, the coordinate system in 2D plane is sufficient. The distance between two points in the 2D Euclidean space can be calculated with the Euclidean distance formula.

$$Distance_{1,2} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

Equation 3 - Euclidean Distance in two dimensions

$Distance_{1,2}$ represents the distance between the coordinates at timestamps 1 and 2, and x_1, x_2, y_1 and y_2 are the x-coordinate and y-coordinate values at the timestamps 1 and 2 respectively.

To calculate the pair-wise distance between the different timestamps/observations, the function $pdist$ from MATLAB was used. This calculates the pair-wise Euclidean distance between each pair in the dataset.

Wi-Fi fingerprint Dissimilarity

For the purposes of this study, we define the 'dissimilarity' between Wi-Fi fingerprints as the pair-wise distance/dissimilarity between the two Wi-Fi fingerprints. The pair-wise dissimilarity here is defined as the average dissimilarity of the RSS values between two Access Points at two different timestamps, and can be defined as follows,

$$d_{F_1,F_2} = \sqrt[2]{\sum_{i=1}^{n} \left(RSS_{i,1} - RSS_{i,2}\right)^2}$$

Equation 4 - Wi-Fi fingerprint dissimilarity definition

Where d_{F_1,F_2} is the dissimilarity between the fingerprints F_1 and F_2 which are the Wi-Fi fingerprint values at timestamp 1 and 2 respectively. n represents the total number of access points and i is the $i-th$ access point. And $RSS_{i,1}$ and $RSS_{i,2}$ represent the RSS (Received Signal Strength) values of the $i-th$ access point at timestamps 1 and 2 respectively. For each fingerprint pair, the RSS values from the Access Points are only considered when there is an RSS value from both the fingerprint locations.

The equation is a modified version of the Euclidean distance formula, where the different access points serve as anecdotes to the different axes that are computed in the distance formula. To calculate this in MATLAB, like in the Euclidean distance case the in-built function of $pdist$ is used, which returns an array of the pair-wise distances between the pairs of fingerprints in the dataset.

The output from the function *pdist* is a shortened or condensed version of the pair-wise distances. Since the distance between two points is the same whichever direction, we look at it, the information regarding distance between points 1 and 2, and points 2 and 1 are just stored once in the array. However, for compatibility and better visualization we convert this to a full matrix using the function *squareform* from MATLAB. This converts the pair-wise distance array from *pdist* into a matrix that contains the pair-wise distance between the different data pairs.

Now that all the data has been collected and processed for curve fitting, the next step is to choose the type of curve with which the data is fit. A quadratic curve was selected in this example to better capture the spread of the points in the graph.

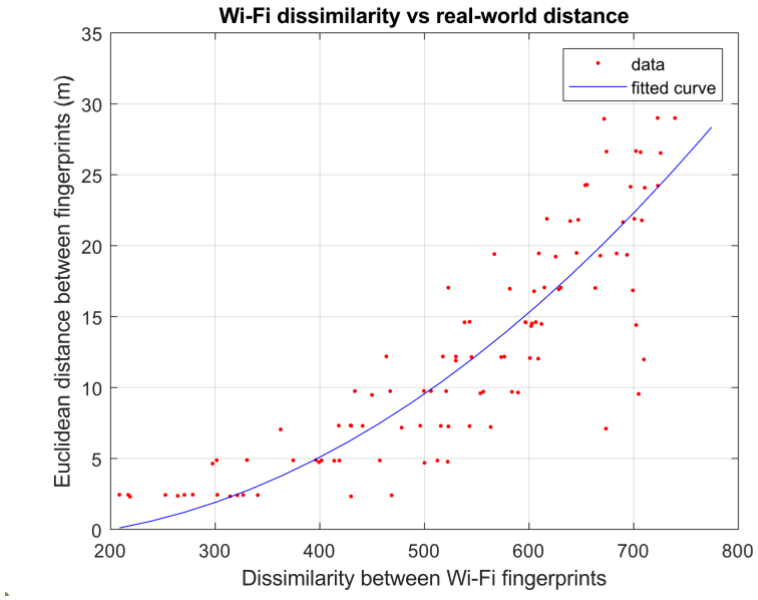

Figure 1 Quadratic fit for Wi-Fi dissimilarity vs distance.

Having collected the real-world distance between the different points and having calculated the dissimilarity between the said points, a curve fit can be made between them. For the study, a linear fit with a lower limit of 0 since the scalar distance between two points cannot be negative.

The curve fit used in this study is given by the equation,

$$f(x) = max(0\ , m \cdot x + c)$$

Equation 5 Distance Dissimilarity fit equation

Where $f(x)$ represents the distance estimate between two fingerprint pairs. Where the equation $m \cdot x + c$ denotes the line fit, of slope m and intercept c. The function follows the slope for the distance estimate while maintaining a lower limit of 0. This is done to ensure that the dissimilarity estimate is kept positive.

Fitting a curve over the whole dataset yields a graph like this,

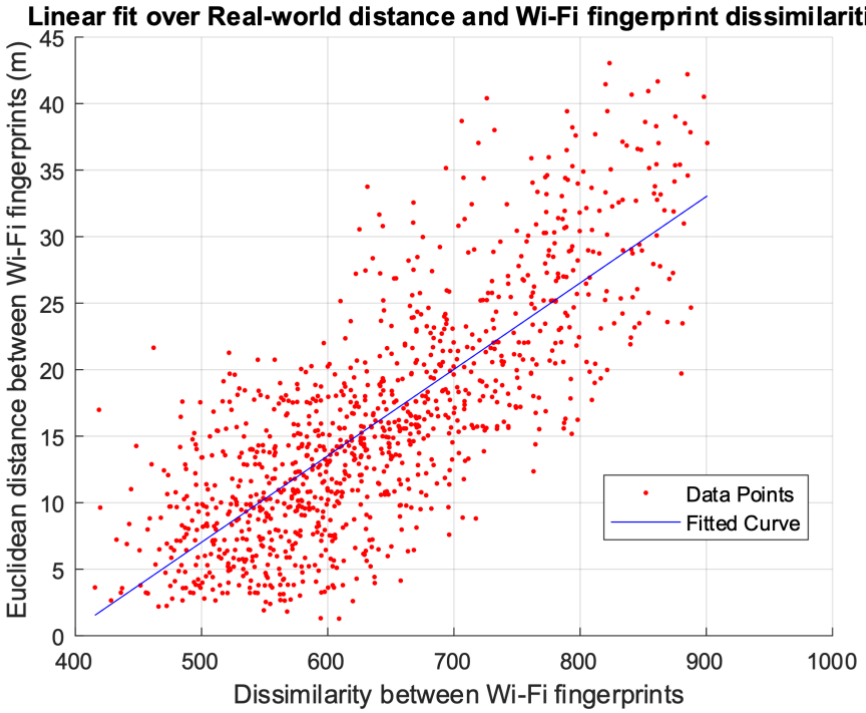

Figure 2 Least-Squares fit over Wi-Fi fingerprint dissimilarity and Real-World Euclidean Distance.

The fit model can then be used to interpret the distance estimate from a pair of Wi-Fi fingerprints. The data from Inertial sensors such as accelerometers and gyroscopes are recorded along with the RSS data at proper timestamps. This distance estimate for the distance between two fingerprint pairs is then augmented with data from the Inertial Sensors on the phone (PDR). When augmenting the distance estimate from the curve fit with the distance estimate from the Inertial Sensors, an individual set of weights is added to each fingerprint data. The different types of weights that were added and their effects will be discussed further in the study.

3.1.3 Cartesian coordinates estimation

With the mathematical model correlating the Wi-Fi fingerprint dissimilarity and Euclidean distance devised, the next step is to use Least-Squares Regression to

map the pairwise distance values to specific coordinates in the cartesian system. Assuming that the information on some coordinates is available, for instance the starting location, or some landmark location. The process is designed in such a way that this is accounted for. To allow for known locations and coordinate points, the Least-Squares based regression is split into two parts. The first set would account for the known values, while the second would be the distance estimation using PDR (Pedestrian Dead Reckoning) and Wi-Fi fingerprint data.

The coordinates in the 2D Euclidean space can be represented in vector form as shown below,

$$\vec{x} = [x_1, y_1 \dots x_n, y_n]^T$$

Equation 6 - Estimated Coordinates vector in MATLAB

Here $\vec{x}$ represents a vector (of size $2n$ by 1) which contains the coordinates (x_1, y_1, $\dots x_n, y_n$) and n represents the number of fingerprints. These coordinates represent the corresponding x and y coordinates of the fingerprint (x_1, y_1 represent the x and y coordinates of the first fingerprint, etc).

The objective right now is to estimate and figure out the coordinate values ($\vec{x}$) given all the information that is available. To allow for leveraging known points/coordinates, the first model can be modelled as

$$A_1\vec{x} = \overrightarrow{b_1}$$

Equation 7 – System Equation for known coordinates model.

Where A_1 is a design matrix of size $2n$ by $2n$ (n is the number of fingerprints) that defines which points that are known, and $\overrightarrow{b_1}$ is the observation vector of length $2n$ which holds the known coordinate values.

The second part of the model uses the information based on distance between fingerprints to find estimate the coordinate values. To recall equation 3, the Euclidean distance between two fingerprints can be defined as such following the Euclidean distance formula.

$$distance_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$$

Where $distance_{i,j}$ represents the magnitude of the distance between a pair of fingerprints (i, j), the x and y coordinates of those fingerprints are represented by

x_i, y_i and x_j, y_j respectively. This distance equation can be further simplified to make it easier for computation by taking vectors into use as shown below,

$$distance_{i,j} = \sqrt{(\vec{x}_i - \vec{x}_j)(\vec{x}_i - \vec{x}_j)'}$$

Equation 8 - Simplified Euclidean distance equation

Here, the distance between the fingerprints i and j can be defined as the dot product of the vector $(\vec{x}_i - \vec{x}_j)$ with itself, where $\vec{x}_i \, and \, \vec{x}_j$ are the cartesian coordinates of the fingerprints at positions i and j respectively. $(\vec{x}_i - \vec{x}_j)'$ is used to denote a transpose of the vector, which is done to enable multiplication between the two vectors. The distance estimates are obtained from a combination of PDR data and the Wi-Fi fingerprint dissimilarity information as shown in the previous section.

$$A_2\vec{x} = f(\vec{x}) = b_2 = distance_{i,j}$$

Equation 9 – System Equation for distance model.

Equation (7) can then be written as shown in Equation (8) where b_2 is the observation vector of the distance between the fingerprints i and j, and A_2 is the design matrix that corresponds to the distance model.

Given the two equations (6) and (8), Least-Squares regression can be employed to minimize the residuals in them to estimate the coordinates of the fingerprints $\vec{x}$ like shown here,

$$R_1 = \overrightarrow{b_1} - A_1\vec{x}$$

$$R_2 = b_2 - f(\vec{x})$$

Equation 10 Residual equations

Here, R_1, R_2 are the residuals from the two equations (6) and (8) that define the model chosen to estimate the indoor coordinates using Wi-Fi fingerprint data and PDR information.

The objective of using Least-Squares regression is to minimize the residuals by iteratively modifying the variables. This is done by starting with an initial guess for the variables ($\vec{x}$), and then observing the difference between the observed values ($\overrightarrow{b_1}$) and (b_2) against the model estimates ($A_1.\vec{x}$) and ($f(\vec{x})$) respectively, and then adjusting the coordinate estimates ($\vec{x}$) iteratively. For the sake of

simplicity, the standard starting value of 0 is chosen for all the coordinates in $\vec{x}$. A new set of guesses for the values of the coordinates are set to find the new residuals. And this is repeated until the residuals are reduced.

Least-Squares Regression is done through MATLAB, which also allows for weights to be assigned, allowing for even more nuanced control over the model. Incorporating weights into the model gives more control by adjusting the importance of certain residuals based on confidence levels, data reliability etc (MATLAB, 2024). This is important in making sure that the known values have strong weight so that the information such as the starting location or known landmarks are used effectively.

3.1.4 Proof of concept – One-dimensional case

To demonstrate the credibility of the proposed algorithm, let's simulate a small dataset and then estimate the coordinates of the simulated points. Let the coordinates of the points be $(1, 2, 3, 4\ and\ 5)$. To recall the least-squares model, the equations (6) and (8),

$$A_1\vec{x} = \overrightarrow{b_1}$$

$$A_2\vec{x} = b_2$$

Combining these two equations, we get

$$\begin{bmatrix} A_1 \\ A_2 \end{bmatrix}\vec{x} = \begin{bmatrix} \overrightarrow{b_1} \\ b_2 \end{bmatrix}$$

Equation 11 Least-Squares Regression Model

In-case of the demo where there's only one dimension, the distance function can be written as,

$$f(x_{1,2}) = |x_1 - x_2|$$

Where $f(x_{1,2})$ represents the distance between the coordinates x_1 and x_2.

$$\Longrightarrow A_2\vec{x} = f(\vec{x}) = \begin{bmatrix} |x_1 - x_2| \\ |x_1 - x_3| \\ . \\ |x_2 - x_3| \\ . \\ |x_{n-1} - x_n| \end{bmatrix}$$

Equation 12 DEMO 1D distance equation

Here, $f(\vec{x})$ is the pair-wise distance between the coordinate pairs of the vector $\vec{x}$ and n is the number of coordinates in the coordinate vector.

Let's define the vector $\vec{x}$ as the vector holding all the coordinate values that we want to estimate with Least-Squares Regression,

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix}$$

Equation 13 DEMO x vector

For the sake of this study, let's assume that the starting coordinate value is known, then following the model that defines the coordinates in equation (6), we get,

$$\overrightarrow{b_1} = [1]$$

Equation 14 DEMO b1 vector

$$A_1 = [1 \quad 0 \quad 0 \quad 0 \quad 0]$$

Equation 15 DEMO A1 Matrix

Given that we know the distances between each point, which is kept at a constant 1, and this being a rudimentary case, the distance vector $b_1$could be simplified to hold only the distance between each consecutive point.

$$b_2 = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Equation 16 DEMO b2 vector

Therefore, the matrix A_2 must be constructed in such a way that we are able to define the function that estimates the distance between consecutive points. This

can be done by assigning the main diagonal to the value of -1 and then the elements above it with the value of 1. As shown by the matrix A_2 here,

$$A_2 = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

Equation 17 DEMO A2 Matrix

Substituting the matrices and vectors A_1, A_2 and b_1, $b_2$and $\vec{x}$ into equation (11), the final model would be,

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & -1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$$

Solving this with Least-Squares Regression in MATLAB, we get,

$$\vec{x} = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{bmatrix}$$

Which is the initial set of coordinates that we started out with. The estimate coordinates can be plotted against the original coordinates as seen in the following graph,

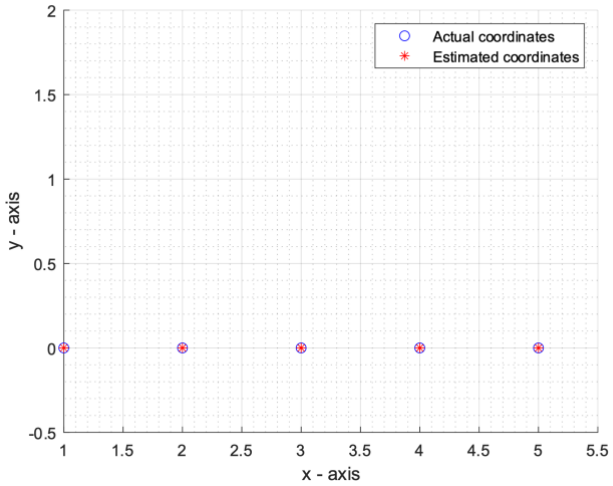

Figure 3 DEMO Actual vs Estimated Coordinates plot.

In case of more complex scenarios, all the information that is available should be used. For example, in case the coordinates of the points are assumed to be, (1, 2, 1, 2, 1) instead of the one in the DEMO, (1, 2, 3, 4, 5), then the full set of

distances from the pair-wise distance equation might need to be calculated as shown in equation (12) for accurate estimation.

3.1.5 Weighted-Least-Squares-Regression

With the Least-Squares Regression method demonstrated for the one-dimensional case, the next step is to ensure that weights are adjusted to make sure that information with different confidence levels are accounted for (Z Zeng et al, 2020). In the model that is chosen for the study, there are two elements. The first element poses the direct coordinate values, and the second element being the model formed through distance estimates between fingerprints. Since the first part of the model is defined in such a way to enable having known coordinates, values from the first part of the model would need to be assigned 100% confidence levels.

In this study however, getting the perfect blend of weights was not prioritized as the main goal of the study is to study the viability and efficacy of using Wi-Fi RSS value supported in-door navigation system. However, a weight system is implemented in such a way that Wi-Fi fingerprint RSS dissimilarities and the PDR distance information between two fingerprints that are of high values get a lower weight, compared to less dissimilar Fingerprints, or closer points.

To account for the two models in the Least-Squares model, two types of weights are added to the model. A uniform weight of 1 is set for the known coordinate values from part one of the model, and a variable weight is chosen for the Wi-Fi fingerprint dissimilarity model. A linear scaled variable weight is chosen for the weights of Wi-Fi fingerprint weights. A comparison between the linear system for weights and the quadratic system of weights was also done to establish a crude estimation of how effective a linear weight system works in case of the Wi-Fi fingerprint dissimilarity weights in the Least-Squares model.

Weights Explanation

To recall the residual equations (9),

$$R_1 = \overrightarrow{b_1} - A_1\vec{x}$$

$$R_2 = b_2 - f(\vec{x})$$

The goal of LSR is to minimize the sum of the squares of these Residuals, a weight can be added to the equation to better control the effectiveness of each of the residuals. As mentioned before, the first part of the model is given a uniform weight of 1, while the second part of the model is given a linear scale variable weight. Adding weights would make the equations as follows,

$$W_1R_1 = W_1\left(\overrightarrow{b_1} - A_1\vec{x}\right)$$

$$W_2R_2 = W_2\left(b_2 - f(\vec{x})\right)$$

Equation 18 Weighted Least-Squares Equations.

Where W_1 and W_2 represent the arbitrary weights that were selected for the model. The values for the weights are selected through empirical data of running simulations in MATLAB.

$$W_1 = ones(numel(b1), 1);$$

The equation above is the MATLAB code that was used to represent the weights for the known coordinates. '*ones*' is an in-built function in MATLAB that allows us to make a matrix of 1s of the size provided. '*numel*' is the in-built function that provides the length of the matrix that was given as the argument to the function. In this case, this returns a vector of 1s of the length of the matrix b_1, which is the list of the known coordinates.

$$W_2 = m \cdot b2 + c;$$

And the above equation represents the weights for the model representing the distance/dissimilarity between Wi-Fi fingerprint pairs. This follows the model of a linear system with a slope (m) and intercept (c), with $b2$ being the observed dissimilarity/distance values.

Weights Demo

To demonstrate the importance of having weights and to serve as a proof of concept, the following graphs were made both with the same data, but one having a weighted Least-Squares Regression while the other doesn't. The data used in this section is the same data that was collected as in the main dataset for this thesis. It was recorded using Xperia X as the mobile device to grab PDR information and Wi-Fi fingerprint data. And an XSENS mTW IMU to collect the more precise PDR information to serve as the reference data. The data collected is from the corridors intersecting the 'Sähkötalo' and 'Rakennustalo' buildings on the second floor in Tampere University. This will be explained in more detail later 33 in the data collection section of the thesis. The information given in both cases are all the actual expected coordinates and Wi-Fi fingerprint dissimilarities for every pair of fingerprints. The expected resulting behaviour is that every fingerprint coordinate estimate is correct. Given that the expected coordinates are given as an input, the output should be the same.

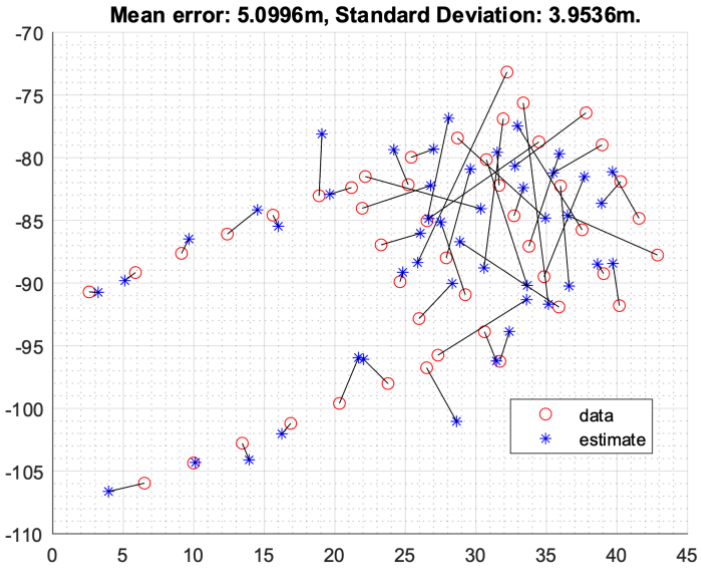

Figure 4 Coordinates estimation without weights.

In the previous figure (5), a Least-Squares estimate was made without weights, as seen above. Even with the known coordinates already provided in the dataset, the results have a mean error of over 5 meters. Now to add weights into the equation. As mentioned before a uniform weight of one is added for the first model, and a linear scale weight is added to the second model.

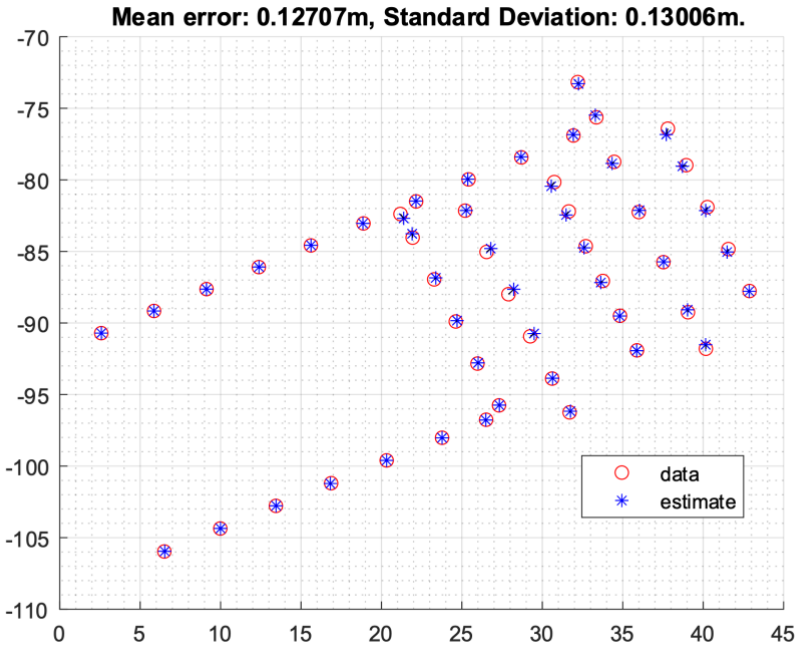

Figure 5 Coordinates estimation with weights.

As seen in the figure above, with weights added, the information from the known coordinate values is given 100% confidence and the result is as expected, with all the estimates being the correct estimates.

Final Least-Squares Equation

Finally, the Least-Squares algorithm is employed to estimate the coordinates and distances by minimizing the weighted sum of the squared residuals.

$$W_1 R_1 = W_1\left(\overrightarrow{b_1} - A_1\vec{x}\right)$$

$$W_2 R_2 = W_2\left(b_2 - f(\vec{x})\right)$$

Re-calling the Weights equations (18), the idea is to find the argument which minimises the Regression from the two equations. This can be formulated as,

$$\vec{x} = \arg min \left\| \begin{matrix} W_1\left(\overrightarrow{b_1} - A_1\vec{x}\right) \\ W_2\left(b_2 - f(\vec{x})\right) \end{matrix} \right\|_2^2$$

Equation 19 Least-Squares Regression Equation

This allows for incorporating weights to adjust the influence of different residuals based on the confidence levels and data reliability.

3.1.6 Initial Estimate selection

A key part of Least-Squares regression is the initial estimate selection. To mitigate the risk of poor convergence, which could lead to suboptimal estimation results, the known points are employed as initial estimates for simplicity and reliability. However, utilizing the general area of the building could also serve as a robust estimate, closely approximating the dataset's spatial median. Using the origin (0,0) as an initial estimate, however, is to be avoided due to its potentially poor convergence outcomes. Although not explicitly demonstrated in this study, an alternative approach could involve averaging the coordinates of the Access Points within the building. This is theoretically sound as it positions the initial estimate at a central point relative to the spatial distribution of the dataset, thereby enhancing the likelihood of achieving more accurate regression outcomes.

3.2 Filtering the dataset

The downside of both PDR based distance estimation and that of Wi-Fi fingerprint dissimilarity is that they both are prone to low accuracy in their own ways. PDR information accumulates error over time and is not very accurate over long distances. Especially in the case of using a mobile PDR data where the user might not have a controlled position of the phone, leading to even more errors being added over time. This will be explained in more detail later in the thesis. Wi-Fi fingerprints on the other hand are susceptible to obstacles and lose their signal strength over heavily. This would lead to cases where the Wi-Fi access points are close in geometry, but due to the interference from other signals and/or the obstacles in between, it loses its signal strength. These pseudo-weak signals pose for more noisy data to be added to the model. To make good use of the data without adding too much noise, it is necessary to not only choose the weights carefully as shown in the previous chapter, but also to only use reliable information. Not all the information from the Wi-Fi fingerprint pair-wise dissimilarity information is equally useful. In this chapter, an algorithm to make the best out of this situation is presented and tested.

Wi-Fi fingerprint dissimilarity is used here to anchor and filter out the fingerprints that were too dissimilar from each other. To do so, a threshold is set, relative to the highest dissimilarity between Wi-Fi fingerprint pairs in the dataset. And all the

fingerprint pairs that are more dissimilar than this threshold is eliminated from the dataset. This method theoretically would mitigate the issue of poor or unreliable information from skewing the estimation results.

3.3 Data collection

- **Wi-Fi -** The common name for the set of protocols used for local area networking and internet access. Wi-Fi typically uses the 2.4 gigahertz and 5 gigahertz frequency bands. These bands can further be divided into multiple channels (IEEE, 2021). Networks can share channels but only one transmitter can locally transmit on a channel at any moment in time. Wi-Fi works best for line-of-sight use, the strength of the signal decays with distance and obstacles.
- **Wi-Fi Fingerprint data –** Wi-Fi Fingerprint data is the data that holds information about the Wi-Fi signal strengths of every access point that is visible or detectable at any given location. Access points are routers or Wi-Fi bridges that emit Wi-Fi signals across the building. The collection of the strength of the Wi-Fi signals from each access point at the place of measurement together is the Wi-Fi fingerprint data.

  The objective of collecting the Wireless signal strength data is to provide input data for the algorithm that is tested and proposed in this thesis. The algorithm proposed requires distance estimates between the point of measurement and the sources (Access points). The Wi-Fi signal strength measured in decibels is used as the distance from the point of measurement and the source in the study.
- **PDR Data –** PDR (Pedestrian Dead Reckoning) data is the method by which previously known location and the inertial information (Acceleration and Orientation) are used to estimate the current location of the person/sensor/device. The speed and direction at which the object is moving is used to estimate the position of the object after a certain amount of time has passed. To calculate the Dead Reckoning, we require the inertial information (Acceleration and Orientation) of the device. This can be measured using Accelerometers and Gyroscopes.

While a person that has the sensors moves and changes direction, the inertial sensors capture the acceleration and orientation of each step. This however is affected by slight deviations in measurement. Be it sensor inaccuracy, human gait irregularity (Walking pattern), or even environmental factors might introduce an error in the dead reckoning. Since each new step relies on the results from the previous step for the new location, over time, the error accumulates and inevitably drifts off from the actual location. Therefore, PDR information is measured in shorter time frames. This ensures that a recalibration of location data is done during each measurement frame, minimizing the impact of the error accumulated throughout.

3.3.1 Location

The Wi-Fi fingerprint data that is being used in this thesis was collected primarily on the campus of Tampere University, Hervanta campus. The readings of Wi-Fi fingerprints were recorded along the corridors intersecting the ‘Sähkötalo’ and ‘Rakennustalo’ buildings on the second floor.

3.3.2 Collection tools and method

To gather fingerprint data (A Wi-Fi fingerprint is the list of all the Wi-Fi access points along with their signal strengths at any given time and location), an Android phone (LG Nexus) along with an in-house Android application provided by HERE Technologies was used. The application tracks the Wi-Fi access points list, with their corresponding signal strengths, the GPS values, and the sensor values, namely Accelerometer and Gyroscope () measurements were recorded. As for reference values, an IoT (Internet of Things) system (Minnowboard) with an MTw (XSENS) Inertial measurement unit was used. The objective of this setup was to collect more accurate reference values for the inertial data (Accelerometer and Gyroscope)

- **Accelerometer –** An accelerometer is a sensor that measures the relative acceleration. Relative acceleration is the relative acceleration while

assuming the sensor is at rest. The acceleration of the system around the sensor, with the sensor being the point of reference; the sensor is at rest. Such a frame of reference is also called a ‘rest frame’. Typical accelerometers that are used in mobile phones are MEMS (Microelectromechanical systems) accelerometers. They are small electromechanical devices that can be embedded in an electrical system. Accelerometers of such type of function by having a mass that is suspended in between two capacitor plates; and the deflection of the mass, which results in a variation in the capacitance is then used to calculate the acceleration of the module. Such accelerometers measure the acceleration in one axis. Therefore, to compensate for the lack of acceleration data on the other axis, a three-axis accelerometer is used. A three-axis accelerometer is nothing but a combination of three accelerometers embedded together, each measuring the acceleration in one of the axes. Hence, giving the acceleration of the sensor in the three axes (x, y and z axes).

- **Gyroscope –** A gyroscope is a sensor that measures the angular momentum. In other words, the gyroscope measures the speed at which the sensor is rotating or spinning about a given axis. The gyroscopes that are embedded within a smartphone are MEMS (Microelectromechanical systems) gyroscopes. These gyroscopes, sometimes referred to as gyro meters, are miniature versions of the prominent types of gyroscopes. One common type of gyroscope that can be found embedded in IMUs and in smartphone sensor modules is a piezo-electric gyroscope. Piezoelectric gyroscopes have a piezoelectric mass that vibrates, and then the deviation caused by the Coriolis effect is measured. Coriolis effect is the deviation of an object from its trajectory that is caused by the presence of an angular rotation in the system. Like the embedded accelerometers mentioned above, this measures the angular momentum in one axis. Therefore 3 different gyroscope modules are coupled together to gather the angular momentum in the three axes (x y and z axes).
- **GPS Receiver –** A GPS (Global Positioning System) receiver is a device that receives the transmission that is sent by GPS satellites around the globe. Though it originated with military applications in mind, it has since

been widely used for civilian purposes. GPS receivers depend on data from a constellation of satellites to determine the current position. Each satellite in the constellation has an atomic clock (synchronised with other satellites and ground station clocks). Given the constant speed of radio waves, the time delay between transmission from satellites.

- **IMU –** An Inertial measurement unit (IMU) is used to detect linear acceleration with a combination of accelerometers and rate of rotation with a (or more) gyroscope. It can also include a magnetometer (like in aircraft navigation applications) for heading reference. Usually, the IMU has accelerometers, magnetometers, and gyroscopes for each of the pitch, roll and yaw axes. A magnetometer essentially provides a sense of direction about the earth's magnetic north. The combination of the three sensors makes IMU a comprehensive suite when it comes to motion detection and navigation.

- **Wi-Fi Receiver –** A Wi-Fi (Wireless Fidelity) receiver is a device that is designed to detect and decode signals that are transmitted through Wi-Fi. It operates within the Radio frequency spectrum, typically at 2.4GHz or 5GHz, providing a wireless network for data exchange between devices and network systems. It consists of an antenna and a receiver, the antenna is responsible for capturing the radio waves transmitted by routers or Access Points, and then the receiver itself is responsible for converting this analogue data into digital information which can then be used for various network/data tasks. This ubiquitous sensor which is present in almost all mobile devices in the modern world, presents a great way to augment the accuracy of position determination. This can be achieved by querying information such as the MAC address and the signal strength of the wi-fi signal in each receiver.

Data collection was done in small portions. The application was switched on to start recording, held in hand at a steady height while taking a short straight walk along the corridors between the buildings Sahkötalo and Rakennustalo. Walking speed was maintained at a steady speed as well, without any sudden increase

or decrease in speed. The reason for having a steady phone and walking straight lines at a steady pace is to have a way to deduce reference data. This will be explained shortly in the reference data section. Measurement frequency was 4 Hertz, or to say each parameter was measured and stored 4 times every second.

The gold standard used as a reference for this research work was the readout from the MTw XSENS IMU. This sensor has a dual GNSS antenna, and MEMS sensors and utilizes Kalman filtering to provide accurate position, velocity, and orientation measurements.

3.3.3 Collected data preparation.

The collected data is grouped into two different parts. The first is the Wi-Fi fingerprint part which encloses the Access Points list with the signal strengths (in dB) for those access points for the location at a given time, and then the PDR (Pedestrian Dead Reckoning) data which consists of the sensor data in from the mobile phone. This includes the data that is parsed from the accelerometer and Gyroscope.

- **Wi-Fi fingerprint data** – Synchronized to the device clock settings, this measurement consists of the Wi-Fi Access points that are visible at the location combined with the signal strength of the same. A matrix is then made with all the different access points that were discovered throughout the course of the measurement as the rows and the number of measurements that were taken as the columns are made. For the access points that did not have a signal strength strong enough to be detected for a given time and location, the signal strength is set to NaN (Not a number) through MATLAB.
- **Pedestrian Dead Reckoning data** – Also synchronized with the clock, the idea behind collecting the accelerometer data and the Gyroscope data was to use the acceleration information in the direction of walking to integrate and find the distance between two different time stamps. Although Newton's laws can be used to estimate the distance between two points of measurement, the accuracy of the estimated distance is not

reliable for longer distances. Which is also the reason for not having longer measurements.

3.4 Reference Data and Observed Data

A reliable set of data that can be used to cross-check the validity of an experiment is vital in understanding the extent to which the experiment is a success. Therefore, it is crucial to have a dataset that is reliable and has the most accurate measurement possible to cross-check the data that is measured through other sources in the experiment. For the experiments and the study, measurements from an MTw XSENS IMU were used as the base standard for reference data. The high precision of this sensor would be monumental towards establishing a robust standard for the reference data to validate the experiments of this study.

The experiment had a meticulous setup to provide accurate measurements for the study. It was conducted by walking at a steady pace over the short, straight corridors. Such a controlled setup helps minimize the variables that could otherwise compromise the integrity of the measurements. Choosing to record in short bursts instead of long stretches allows us to mitigate the issues of error/drift build-up in IMU sensor data. By maintaining a straight path, we ensure a single primary direction, avoiding potential errors and complexities raised by the changes to the movement direction. Furthermore, the start and end points of the recording were carefully chosen to be identifiable real-world positions, such as the start and end of a hallway, hence providing a reliable frame of reference to calibrate the data while also serving as great anchor points for reference. A consistent speed was maintained, to minimize abrupt changes to acceleration. This was done so that the location data at any time could then be interpolated by using Newton's laws of motion with fewer complications.

During the measurement, apart from the Inertial data from the MTw XSENS IMU sensor (serving as reference), the Wi-Fi fingerprint and PDR data from a mobile phone (serving as the object for the experiment), the distance walked in the corridors and the time it took to walk from one end to another were also simultaneously recorded. This comprehensive strategy allows us to collect a

multifaceted dataset, at the same time, eliminating differences from having to record them individually. Hence maintaining a strong integrity.

Additionally, to ensure the timestamps are synchronized and the different devices are in the same temporal framework, a GPS clock was employed to serve as a common frame of reference for time. This also gives us the benefit of then systematically synchronizing all the measurements from the various devices with different recording frequencies without much difficulty. The use of GPS time (being independent of the devices themselves) obviates the need for manually stitching together the different measurement sets.

The resulting dataset comprises of the reference data from the MTw XSENS IMU sensor, the PDR and Wi-Fi fingerprint data from the mobile phone and manually marked anchor points of reference, along with the distance between said anchor points. This robust dataset will serve as the foundation to validate the experiment effectively and accurately. However, before we can compare the reference and experiment data, they need to be processed and interpolated. This will be further explained in the following sections.

3.4.1 PDR Data interpolation

In simple terms, Newton's laws of motion can be used to estimate the distance covered between two timestamps. We calculate the velocity first and then the distance. These can be done with the following equations of motion from Newton.

$$v = u + a \cdot \Delta t$$

$$s = u * \Delta t + \frac{1}{2} \cdot a \cdot \Delta t^2$$

where v represents the final velocity vector, u represents the initial velocity vector, a is the acceleration vector, s is the displacement and Δt is the time.
Given that we have the acceleration information for the different timestamps, we can estimate the velocities for the different timestamps as well, with the initial velocity being 0. And then the displacement to find the distance between the two timestamps in question. As a demonstration, let's look at how this is done for a demo case in a single dimension, for an object starting from rest. So, the initial

velocity is 0, has an acceleration of 2 units per square second, and a timeframe of 0.1 seconds.

$$u = 0, a = 2, \Delta t = 0.1s$$

Applying this to the displacement equation, we can find the displacement in the timeframe, which is the distance covered between the timestamps.

$$s = 0 * 0.1 + \frac{1}{2} \cdot 2 \cdot 0.1^2 = 0.01\, units$$

In a 3D space, we have different accelerations for the different axes. And we are not interested in all the different axes, just the directions that we are walking in.

However, for this thesis, the algorithm, and the interpolation of the IMU inertial data to distance were provided by HERE maps.

3.4.2 Reference Data

The process of compiling the reference data from the MTw XSENS IMU sensor can be categorized into a few key steps. The idea is to convert the raw Inertial data from the IMU sensors into useful real-world coordinates for each timestamp, which would then act as the reference location for said timestamp.

1. *Collecting the Inertial Data* - As mentioned earlier in 3.5, the reference data was collected using the MTw XSENS IMU sensor while walking a straight line, in short bursts. The employment of short bursts for recording helps minimize error accumulation and maintain high accuracy. They are marked with GPS timestamps for seamless and precise synchronization. This provides us with the necessary Inertial data, such as the linear acceleration in the different axes and the angular momentum for the different axes, stamped with GPS timestamps.
2. *Pre-processing, calculating the Geometric distances* - This is the crucial step that determines the accuracy of the estimated coordinates. The first step is pre-processing the data. This entails correcting the data for sensor

drift, filtering out noise, aligning the data, etc to ensure high accuracy. Then it can be processed further to get the distance estimates between timestamps. Again, as mentioned in the section above, IMU (PDR) data is interpolated with Newton's laws of motion to provide us with the distance estimates between time stamps. This was graciously provided by HERE maps, and we are left with an array of distances against time stamps. The distance estimates are for the distance travelled between the two consecutive timestamps.

3. *Transforming into real-world coordinates using Least Squares Regression* - Using an array of distance estimates, Least-Squares regression is applied to accurately interpolate the distances to real-world coordinates in the 2D plane. The detailed explanation can be seen in Section 3.1. In short, it is a statistical approach to determine precise coordinates by minimizing the sum of squares of the differences between the observed and estimated values.
4. *Compilation of the Final Reference Dataset* - The final reference coordinates consist of the estimated coordinates from the Least-Squares Regression coupled with the precisely measured start and end points of the measurements. These points serve as anchor points having predetermined locations which help improve the overall accuracy of the dataset.

4 COORDINATES ESTIMATION AND COMPARISON

Assumptions and Crowdsourcing

Over the course of the thesis and the experiments that have been conducted in this study, the Wi-Fi frequency band that is used is limited to the 2.4 GHz frequency band. This is done to limit the variables that affect measurements. Having multiple different bands of Wi-Fi would add an extra layer of complexity in the modelling of Wi-Fi to real-world standardized distance metrics. Different bands of Wi-Fi propagate differently, leading to a change in the Wi-Fi dissimilarity to real-world distance calculation. To simplify and not have another variable in the mix, a singular band of 2.4 GHz was chosen because of its more common availability and widespread support for a lot of devices. Another reason for selecting 2.4 GHz is its higher range compared to the 5 GHz band at the commercial scale. 2.4 GHz having a higher wavelength also means that it can penetrate obstacles better, ensuring a longer range, while also providing the added benefit of performing better in indoor scenarios which have a lot of obstacles such as objects, furniture, people, etc.

As mentioned earlier in chapter (3.3) about collected data, PDR data is prone to error build-up over time. To recall, IMU sensors are prone to deviations from sensor inaccuracies, human gait movements or even environmental factors. These then affect the distance estimation for that time segment. And since the new location estimate depends on the prior estimate, the error propagates to the next, growing exponentially. To circumvent this issue, the data could be used to estimate the distances in shorter or smaller bursts rather than over longer lengths. By doing so the overall error that is accumulated is reduced. Since the recording is done in parts, the end points of the recording serve as an anchor to recalibrate against. Thereby making the recording as clean as possible. In practice, however this needs to be mitigated by other means. Some of the common ways that this is accomplished are using Bluetooth beacons, which serve as an anchor point to course correct indoor navigation. The study assumes the availability of a sparse setup of Bluetooth beacons across the floor to server as a low-cost solution to serve as anchor points.

The model proposed in the study must be created through crowdsourcing for each building to have the expected accuracy suggested in this study. Since Wi-Fi fingerprint dissimilarities are modelled against real world Euclidean distances, and the signal strength of Wi-Fi access points are heavily affected by obstacles, having crowdsourced data for a building would potentially achieve better results than the model used in the study. Crowdsourcing of data is the process of many people contributing towards the growth of a particular dataset. Typically, the data from the users of a service is compiled to form a big dataset, which is often out of reach for a single individual or a small team to compile on their own. This could be of the form of data being collected consensually in the form of walking around the building and marking different places, while recording the Wi-Fi signal strength in of all the access points and the PDR information in the process. Like how it was described earlier in the study. After the data is collected, it has to be processed and modelled to correlate the Wi-Fi fingerprint dissimilarity against the real-world Euclidean distances. Once the model is prepared, it can be used to estimate the dissimilarity between fingerprints, which can then be used to assist in PDR information as explained in the study. In this thesis, however, the assumption is that such a crowdsourced dataset already exists and that known locations of a few data points (for instance Bluetooth beacons or certain shops in a mall) are identifiable.

Coordinate Estimation and Comparison

In this section, a comparative analysis is done for the positioning algorithm with and without the aid of Wi-Fi to note the improvement, if any, brought in by using Wi-Fi to assist in indoor positioning. All the tests are done with multiple points in the dataset being marked as known points to simulate the availability of Bluetooth beacons or manually adding landmarks. All the graphs will have a red circle marking the original location of the data point, and a blue asterisk to mark the estimated point through the study. A line linking both the estimated point and the original data point is added to help visualize which data point pertains to which actual location. A mean error and a standard deviation are also added to each figure to understand the efficacy of the estimation. The mean distance between consecutive datapoints is around 3 meters.

The first step is identifying the ideal type of weight for the dataset and the parameters for the weights model. This is done through a comparison of two models, particularly, the Quadratic model and the Linear model. For the purposes of this thesis, finding the best theoretically possible weights was not of priority, as much as finding a feasible and reliable way of modelling the crowdsourced fingerprint data.

Weights selection.

The weights for the Wi-Fi fingerprint pairs can be determined from the dissimilarity values of the Wi-Fi fingerprint pairs. Two types of scaled weight systems were tested, namely a linear scaling weights system and a Quadratic scaling weights system. The two weight systems follow the equations,

$$W = a \cdot x^2 + b \cdot x + c$$

Equation 20 Quadratic Weights equation.

$$W = m \cdot x + c$$

Equation 21 Linear Weights equation.

Where W is the weight given to the fingerprint pair, and $a, b, c, m,$ and x are empirical values that were derived from testing to have the best weights for the proposed model. For the purposes of clarity, the values m and c will be referred

to as the slope and intercept in the study; since they are the slope and intercept in the linear equation defined by them.

To check how effective Quadratic scaled weights are compared to linear scaled weights, the estimated coordinates are plotted against the actual data of the coordinates. The mean error and standard deviation can then be used to estimate the efficacy of the selected weights. This can then be done iteratively to find a good weight for the model with empirical evidence to back it up. For the sake of simplicity and reproducibility, the known coordinates are retained at a limited quantity. In the test cases to follow, every 4^{th} coordinate value is set as the original value to provide a good base of reference.

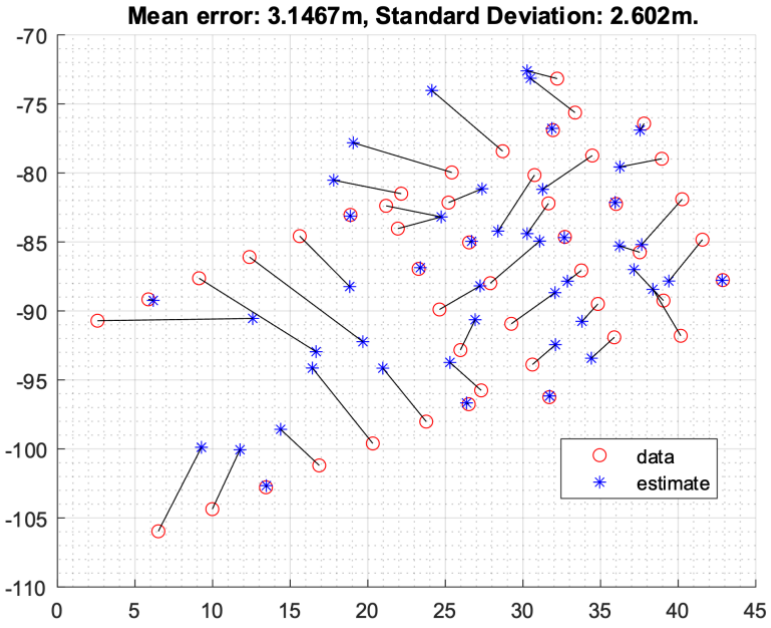

Figure 6 Linear Scaling weights Demo.

The picture above shows that the mean error and standard deviation and mean error and standard deviation of 3.15 meters and 2.6 meters respectively. This is lower than the mean and standard deviation of 5.09 meters and 3.95 meters as seen from the figure (3) where no weights were added. The improvement in the mean error and standard deviation with weights, while using fewer known points given to the model showcases the importance of weights and the effect it has in the estimation of the coordinates. For the plot above, the linear scaling weight has a slope of 1 and an intercept of 2.

Verification against Quadratic scaling weights.

To verify the validation of using Linear Scaling, a small test was done using a Quadratic scaling weight to see its performance against the Linear scaling weights. The Quadratic scaling weights follow the same principle as the basic Quadratic equation with three variables.

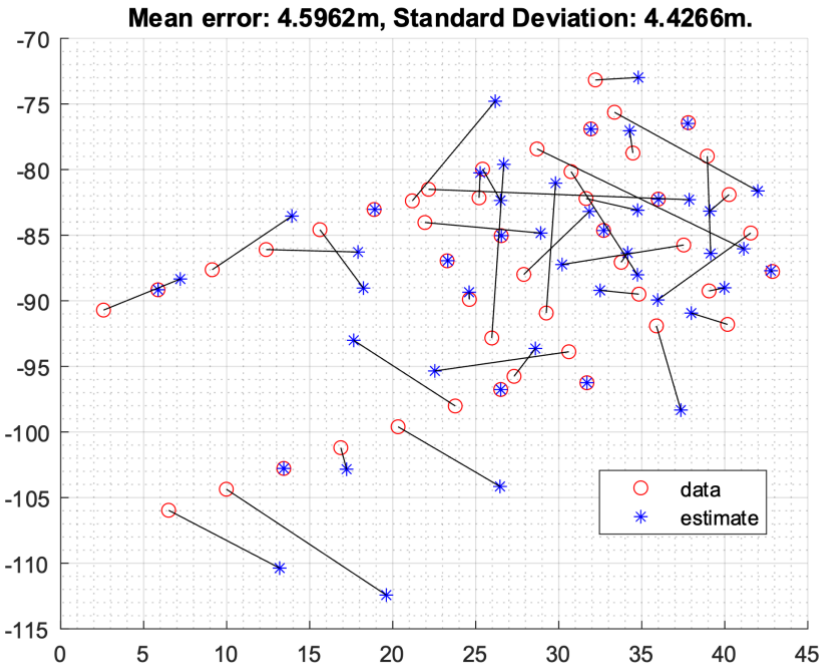

Figure 7 Quadratic Scaling Weights Demo.

As seen from the graph above the mean error and the Standard deviation for the Quadratic scaled weights are 4.59 meters and 4.42 meters respectively. They are less accurate when compared to Linear scaling weights which come up to 3.87 meters and 3.62 meters respectively.

Looking back at the equation used for the Quadratic Scaling,

$$W = a \cdot x^2 + b \cdot x + c$$

Where the values for a, b and c are 0.3, 3 and 1 respectively. The values for the Quadratic equation were selected the same way as the Linear Scaled weights equation, in an iterative manner, minimizing for the lowest mean error and standard deviation.

Linear Weights slope and intercept selection.

To find out the best set of weights for the Linear Scaled weights, the slope and intercept values were iterated and the whole coordinate estimation was done, to find the values for which the mean error and standard deviation is the lowest. This can be visualized with a 3D plot for the slope vs intercept vs mean error/standard deviation.

For the purposes of the test, the slope multipliers were iterated from 1 to 10, and the intercept values were iterated from -10 to 10. The mean error and standard deviation were calculated for each pair of intercept and slope values and then plotted in a three-dimensional surface plot as shown below.

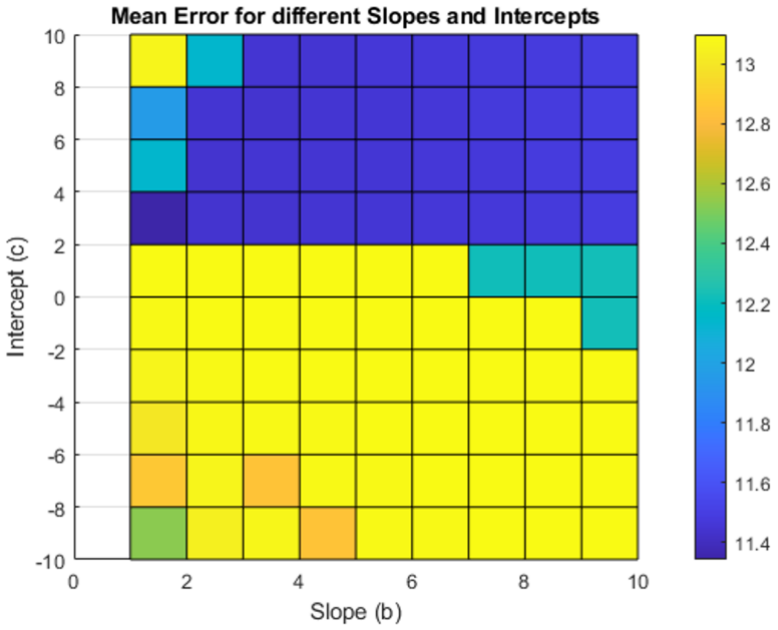

Figure 8 Surface plot for Mean Error.

The surface plot is shown from the z-axis, along with coloured notes to better visualize the Mean error for different values of Slope and Intercepts for the weight equation. From the graph it is evident that the slope and intercept with the lowest mean error is marked by a slope of 1 and an intercept of 2. To help visualize this further, another angle for the surface plot is plotted.

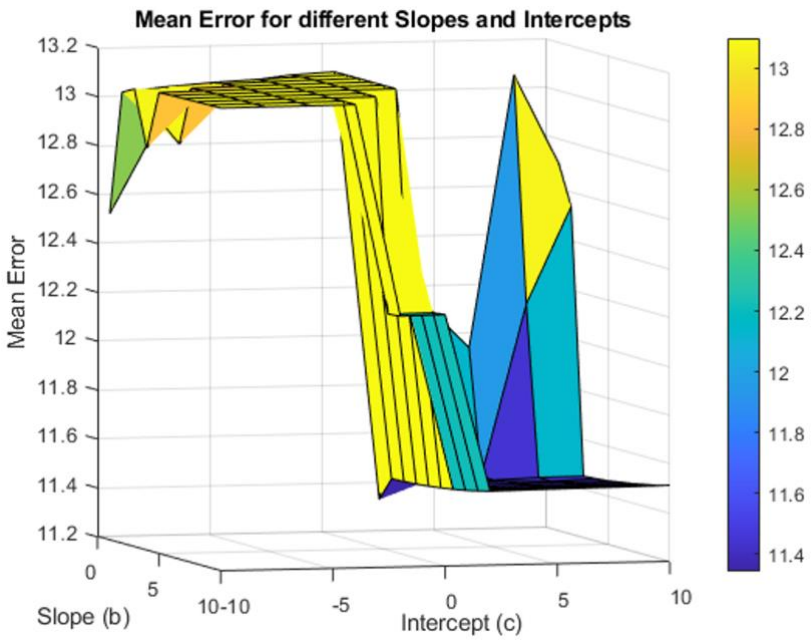

Figure 9 Surface plot for Mean error 2.

Combining the two angles of the surface plot, the optimal weights for the model can be seen more clearly. The ideal slope and intercept for the chosen linear scaled weights equation is 1 and 2 respectively.

The second is to identify the ideal threshold to filter out less reliable distance/dissimilarity data. To identify the ideal threshold for the model, two tests are devised. One test is done with just the PDR information for positioning, and the other is a PDR-based positioning but with some of the points filtered out with the help of Wi-Fi fingerprint dissimilarities.

As with the other demo tests in the thesis, this is done with the help of limiting the information in the dataset and then plotting the estimation results, calculating the mean error and standard deviation. In the figure below, the estimation was done with just the PDR information and no support from Wi-Fi fingerprint information. For the DEMO, every 4^{th} fingerprint's coordinate is set as a known coordinate value for reference, and to help recalibrate the PDR information.

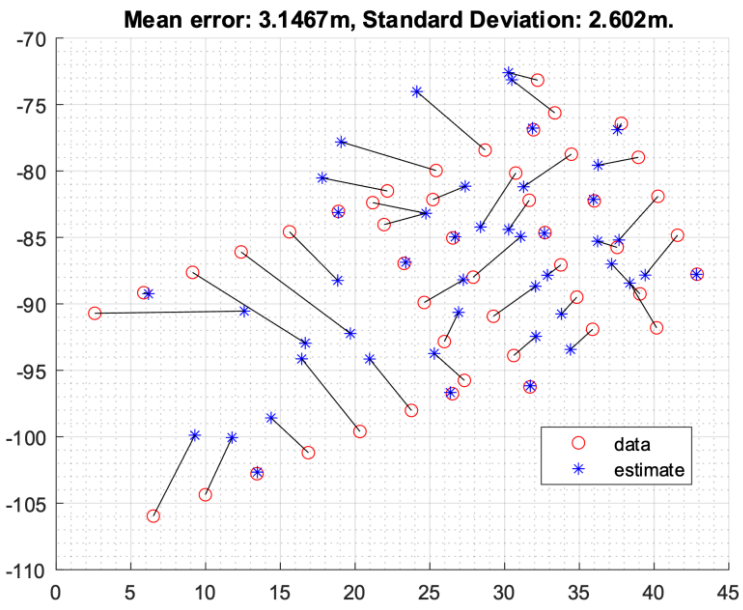

Figure 10 PDR information based indoor positioning.

The mean error while using just the PDR information in this environment turned out to be 3.1467 meters. Filtering out some of the fingerprint pair distance information in the dataset by setting a threshold based off the Wi-Fi fingerprint dissimilarity, the results show a considerable improvement in estimation accuracy.

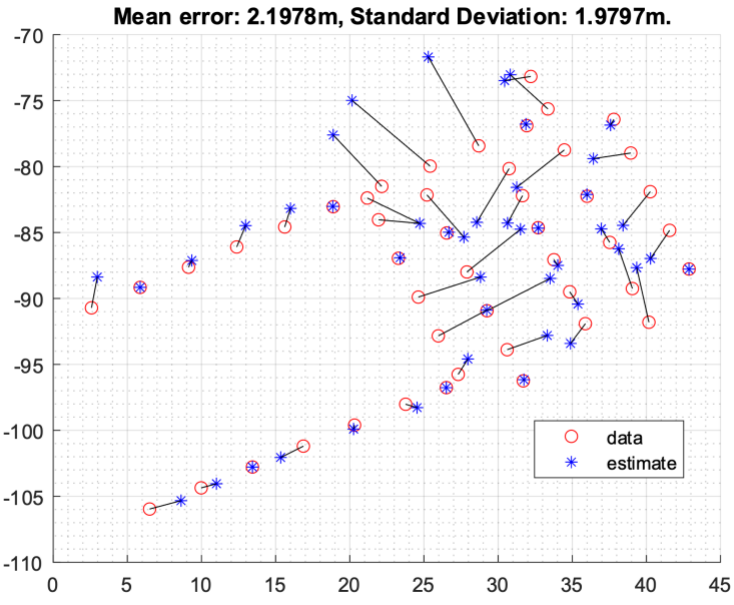

Figure 11 Wi-Fi assisted PDR based indoor navigation.

Using Wi-Fi fingerprint dissimilarity information to filter out potentially unreliable information results in a significant boost to the accuracy of the estimation as can be seen from the figure above. This image uses the best threshold that was

determined for this dataset. And selecting the best threshold is done iteratively like the other sections. Calculating the mean error and standard deviations for different thresholds and finding out the best threshold for minimum mean error and standard deviation.

Like in the previous section, this test setup is also done with every 4th point as a known coordinate value. Then, the coordinate values are estimated for different thresholds iteratively. With the coordinate estimates, the mean error and standard deviation for the estimates can then be calculated and plotted in a graph to find out the best/ideal threshold for this dataset.

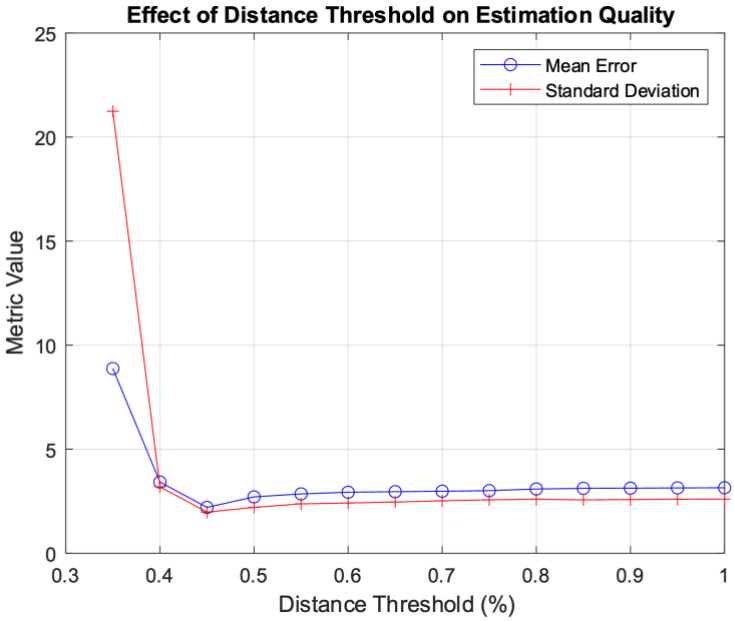

Figure 12 Mean error and Standard deviation at different distance thresholds.

The graph above shows the relation between setting a threshold to the mean error and standard deviation. To calculate the threshold, an arbitrary number is iteratively incremented from 0.35 to 1, and then multiplied with the maximum dissimilarity from the pair-wise distance matrix. Then each dissimilarity in the pair-wise distance matrix is compared against the threshold and any dissimilarity that is higher than the threshold is filtered out.

As noticed from the graph, setting a threshold to withhold and filter out the data from weaker Wi-Fi signals that result in bigger Wi-Fi fingerprint dissimilarities improves the accuracy and reliability of the estimates significantly. The best threshold for this model was found to be 45%. The mean error at the given threshold was 2.1978 meters with a standard deviation of 1.9797 meters.

4.1 Coordinates estimation with PDR information only

To set a baseline, the coordinates estimation is done through using only PDR information, assisted by Bluetooth beacons in a complicated environment filled with multiple turns and crossings. The first part is done with just PDR information, and no added weights or filtering of data. All the data that is available is used directly for the estimation.

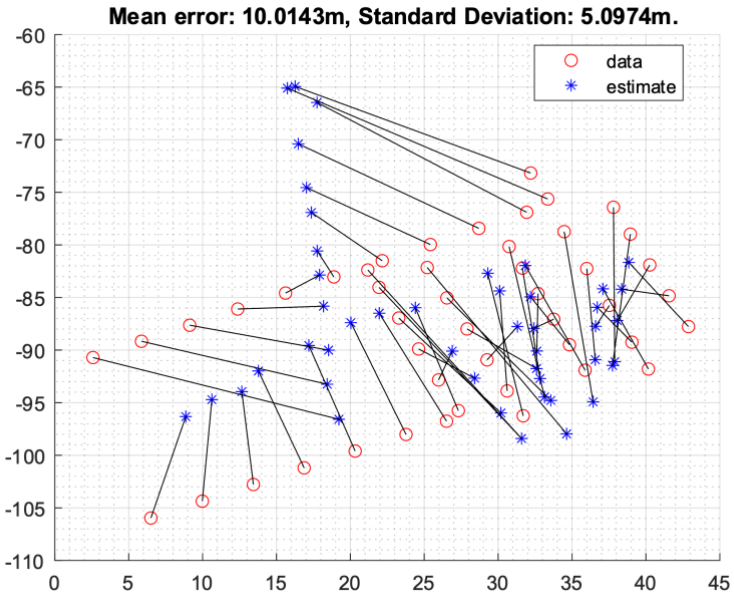

Figure 13 PDR only estimation - no weights/filtering.

As noticed from the image, the points although forming a cohesive set of lines, the lack of weights to anchor the points to relevant points in the floor has thrown the points out of joint. Now adding the weights back into the model to make sure that known points and landmarks are recognized, the mean error should reduce considerably, and the estimation would be more in line with the original data.

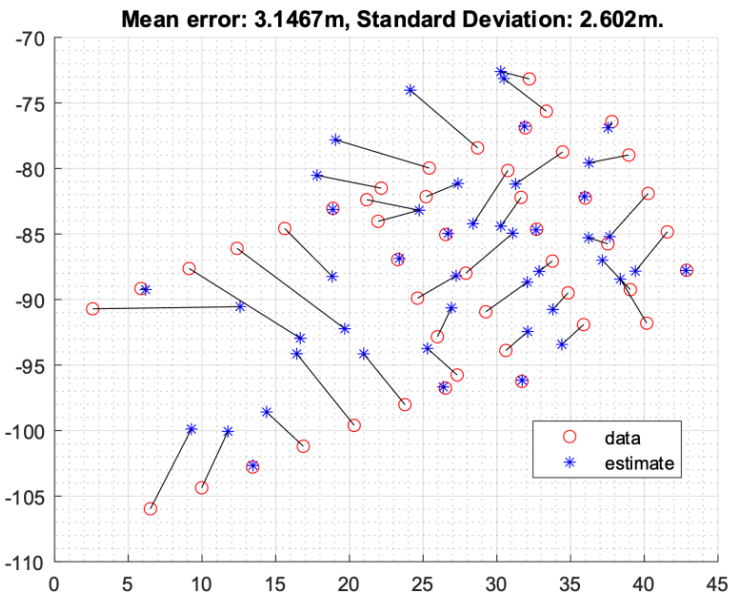

Figure 14 PDR only estimation - With weights.

Having added weights to give more weight to the known coordinates allows the data to be anchored to the actual data points. The Mean error has reduced considerable from over 10 meters to just over 3 meters. However, with this approach, there's still information from datapoints that are too far away interfering with the final estimation. Adding a threshold to filter out information so that only the datapoints that are closer to each other are considered for the estimation, we get the following.

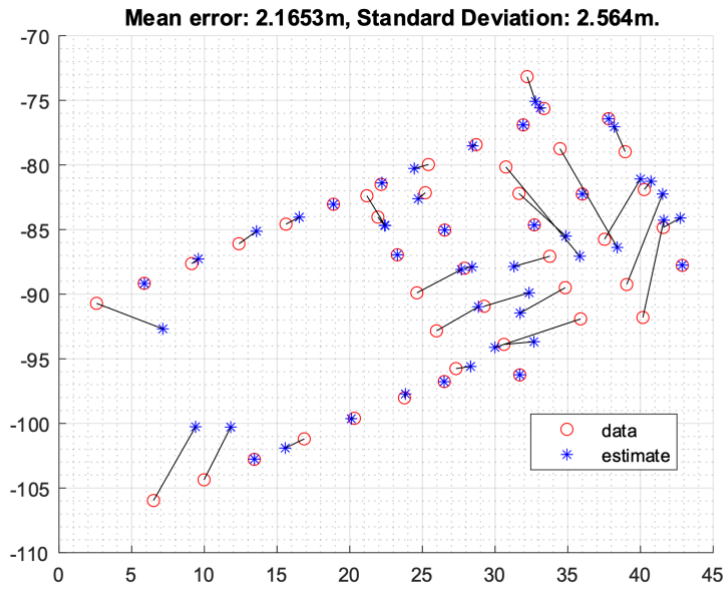

Figure 15 PDR only estimation - With weights and data filtering.

In the graph above, apart from the weights, a filtering was done to only use the data from points that are close to each other. In this case, the information that was used is the pairwise distance information between points that are no more than 2 timestamps away. Thereby limiting the distance between the datapoints and reducing the effects of drift from PDR data. As noted above, the mean error shows a significant improvement over the prior estimation where only weights were used to estimate the coordinates.

4.2 Coordinates estimation using Wi-Fi fingerprint data only.

This section aims to showcase the estimation quality if only Wi-Fi fingerprint data was used to estimate the coordinates. The Wi-Fi fingerprint data is first modelled to represent the mathematical relationship between the Wi-Fi fingerprint dissimilarity and the real-world Euclidean distances. And this information is then used to estimate the distance between the fingerprints, which in-turn is used to estimate the coordinates of the fingerprints.

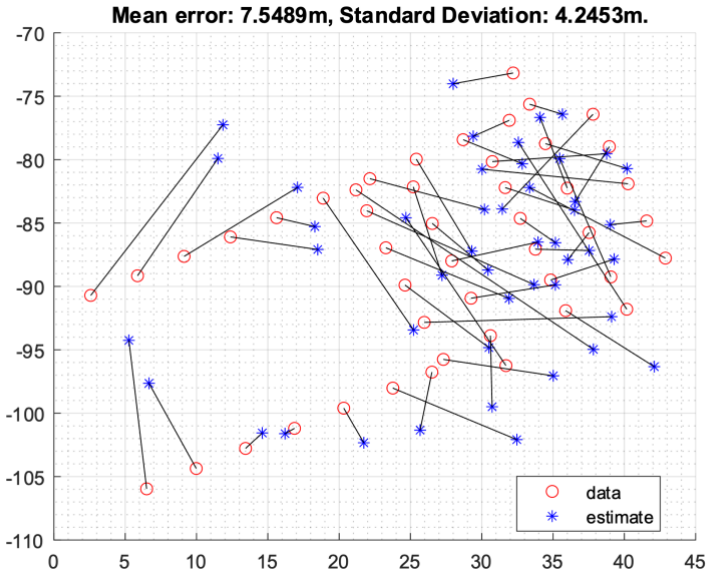

Figure 16 Wi-Fi Only estimation with no weights/filtering.

Using just the Wi-Fi information the mean error from the estimation is considerably lower than just using PDR information. However, the error magnitude it is still far away from the accuracy that is provided by adding weights and filtering to PDR information.

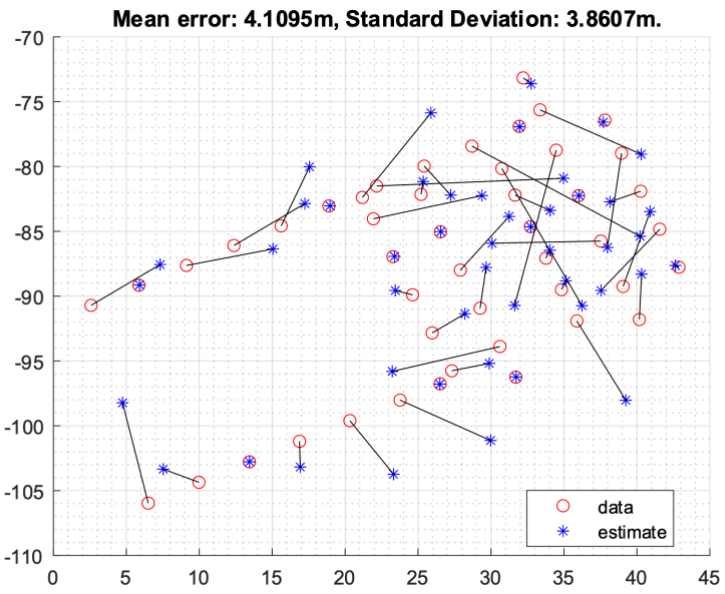

Figure 17 Wi-Fi Only estimation with weights.

Adding weights to the model to anchor the known points in the original position, the mean error reduces substantially, but is not as accurate as PDR based indoor positioning with weights. The Wi-Fi fingerprint dissimilarity information used in this also includes weak Wi-Fi signals which are unreliable. Filtering them out like mentioned before in the methodologies section should give better estimation result.

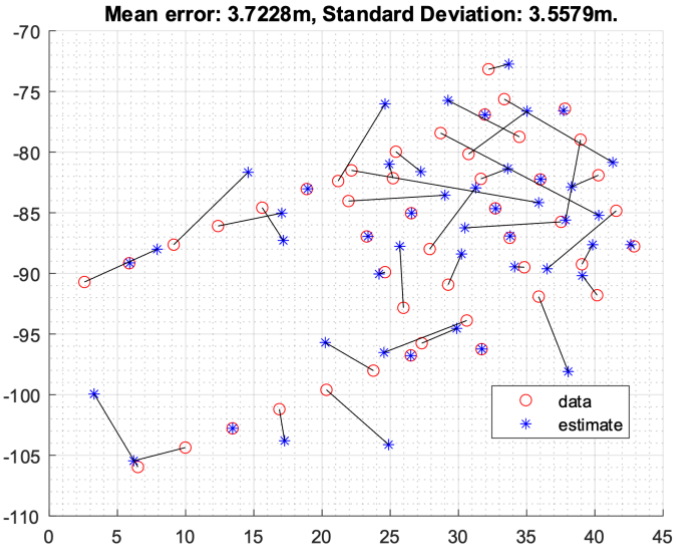

Figure 18 Wi-Fi Only estimation with weights and filtering.

Filtering out the unreliable data from the Wi-Fi fingerprint dissimilarity pairs, the mean error in estimation reduces again. From these graphs in the case of using Wi-Fi only for estimation, the estimation isn't as accurate as using only PDR for

information. But the main objective of the study is to use Wi-Fi to help with the accuracy of PDR based estimation. As noticed from the figure 17, Wi-Fi based estimation, even without anchoring with known points provides better results compared to using just PDR information without any anchors.

4.3 Coordinates estimation with PDR and Wi-Fi fingerprint data

For the final section, all the information that is available is used. Both the information about the Wi-Fi fingerprint dissimilarities and the Euclidean distances from PDR are used to estimate the coordinates. The distance information from PDR is used as the base for estimation, while the Wi-Fi dissimilarities are used to filter out the PDR data to support the estimation.

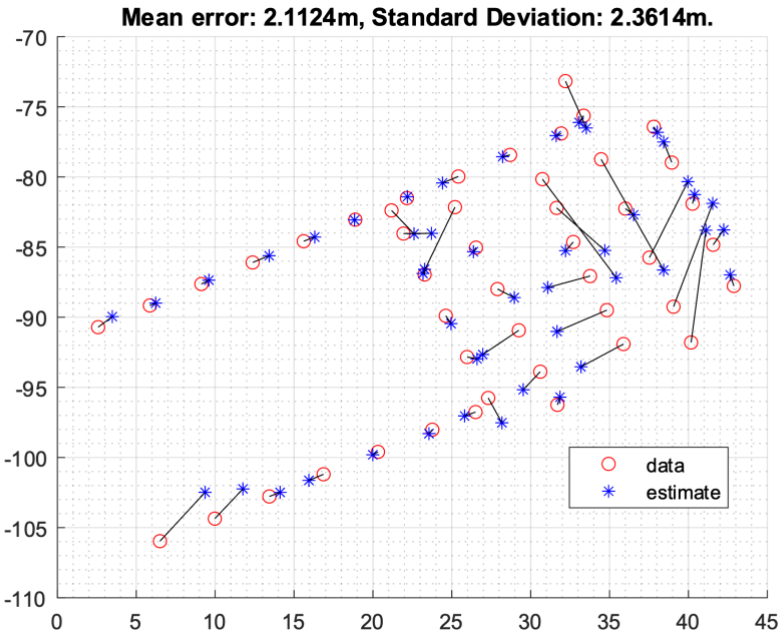

Figure 19 Wi-Fi + PDR with no weights.

Using Wi-Fi fingerprint dissimilarity information to filter out some of the data from the PDR information, the estimation marks a substantial improvement over the estimation from just using the PDR information without weights. Compared the figure (16), where PDR information with weights and filtering, using Wi-Fi to add on to the PDR information provides a slight improvement, even without adding weights to known points and landmarks.

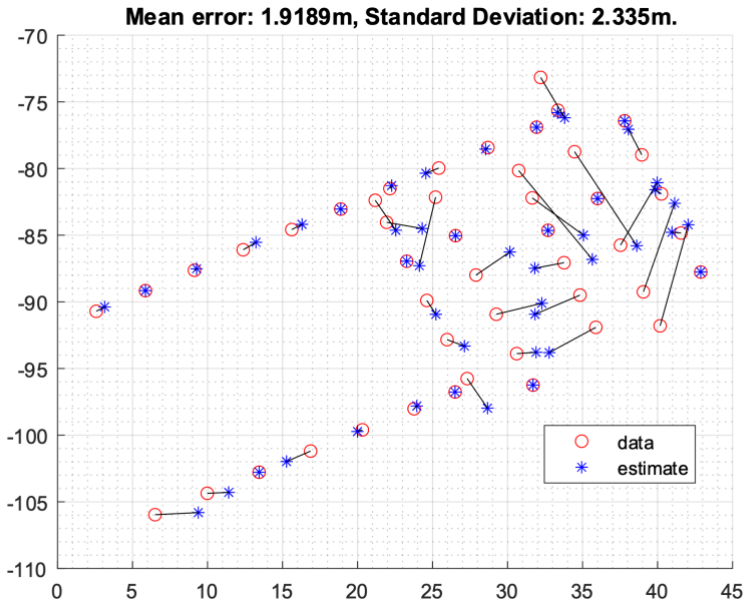

Figure 20 Wi-Fi + PDR with weights.

Adding weights to the regression for known points, the mean error is minimized even more, with an average mean error less than 2 meters.

4.4 Analysis of different scenarios

The results from the different scenarios can be summarized in a table like in the table shown below,

Method of estimation	Mean error	Standard deviation
PDR only without weights/filtering	10.0143 m	5.097 m
PDR only with weights	3.1467 m	2.602 m
PDR only with weights and filtering	2.1653 m	2.564 m
Wi-Fi only without weights/filtering	7.5489 m	4.2453 m
Wi-Fi only with weights	4.1095 m	3.8607 m
Wi-Fi only with weights and filtering	3.7228 m	3.5579 m
PDR + Wi-Fi without weights	2.1124 m	2.3614 m
PDR + Wi-Fi with weights	1.9189 m	2.335 m

PDR only estimation

The high error and variability indicate the significant impact of drift error in PDR-based estimation. Without any anchoring, the cumulative drift leads to substantial inaccuracies, making this approach highly unreliable in complex environments for the purposes of accurate navigation. Introducing weights significantly reduces the mean error and standard deviation. Anchoring to known coordinates mitigates the drift effect by providing reference points that correct the cumulative errors, enhancing both accuracy and consistency. Finally, further improvement is achieved by filtering out unreliable data. This approach ensures that only high quality, closely related data points are used. The combination of weights and filtering provides a robust enhancement for using PDR data alone.

The performance of PDR based estimation will be used as a baseline to evaluate the efficacy and improvements brought in by the modelled crowdsource Wi-Fi fingerprint data.

Wi-Fi only estimation

Wi-Fi only estimation without anchoring is more accurate than PDR due to the absence of drift. However, the variability in signal strength due to environmental factors still results in considerable errors and inconsistencies. Adding weights improves the accuracy and consistency by leveraging known coordinates as anchors. This helps in reducing the impact of variable Wi-Fi signal strengths, though it still doesn't outperform PDR with weights due to the inherent inconsistencies in Wi-Fi signal strengths. And finally, filtering out unreliable data points further refines the accuracy, but again the inherent inconsistencies with the Wi-Fi signal strengths and unreliability due to external and environmental factors limit it to be less accurate than PDR based estimation with proper anchoring and filtering. The results show improvement but highlight the potential for a hybrid approach to showcase the better navigation system.

PDR + Wi-Fi estimation

Adding to the strengths of both forms of estimation, even without weights to anchor the known coordinates, the hybrid approach outperforms the individual performances of both PDR and Wi-Fi based estimation. The nature of the two datatypes used which complement each other help form the robust hybrid system with improve accuracy. The addition of weights to the combined approach yields better results yet. The PDR data benefits from Wi-Fi’s lack of drift, while Wi-Fi

data is stabilized by the more consistent distance estimations from PDR. Using Wi-Fi to filter out and use only the most reliable data from PDR shows promising results with indoor navigation, while anchoring with known points refines the accuracy to create a hybrid system that showcases the benefits of having Wi-Fi to augment the PDR based estimation in a practical sense.

Method of estimation	**Mean error**	**Percentage change against PDR**
PDR only estimation	2.1653 m	0 % (baseline)
Wi-Fi only estimation	3.7228 m	+ 71.93 %
PDR + Wi-Fi estimation	1.9189 m	-11.3795 %

From the experiment, and the results, a hybrid approach where modelled crowdsource Wi-Fi fingerprint data is used to augment the PDR based approach shows promise of viability with an improvement of about 11.38% less mean error.

5 FUTURE STUDIES AND EXTENSION

5.1 Status of the project and future studies

The status of the project is that a model modelling the mathematical relationship between Wi-Fi fingerprint dissimilarity and Euclidean distance to aid in the overall accuracy and reliability of a PDR based indoor navigation system. It works on the premise that there exists the infrastructure to provide known locations, such as Bluetooth beacons placed sparsely across the floor. It employs the distance estimation from Wi-Fi fingerprints as a way to identify the best dataset for location estimation.

However, additional new technologies such as Wi-Fi RTT (Wi-Fi Round Trip Time) pave the way for a better estimation of Wi-Fi fingerprint dissimilarity and could potentially improve upon the benefits of adding Wi-Fi fingerprint data to the indoor navigation system. Wi-Fi RTT promises great indoor accuracy with regards to localization and distance estimation between the receiver and the access point. It works by employing the concept of Time-of-Flight (ToF) and the Round-Trip-Time to provide an accurate measure of the distance between the receiver (Often a smartphone) and the transmitter (Often an access point). This could potentially allow for a much better model for Wi-Fi fingerprint dissimilarity and to Euclidean distance and could potentially increase the accuracy of the estimates even further.

5.2 Limitations

- Interference from bigger crowds and the spatial constraints by the mall construction and design.
- Dependent on Bluetooth beacons or other forms of reference anchor points to be viable.
- Wi-Fi signals could be weak and in which case, become unreliable.
- Requires pre-made model specific to the building/floor to be accurate.

6 CONCLUSIONS

The thesis presents a novel approach towards mitigating some of the issues pertaining to indoor navigation with PDR data. It harnesses the power of a crowdsourced Wi-Fi fingerprint data to refine the location estimates in complex indoor environments. The model takes advantage of the Wi-Fi fingerprint data to filter out the PDR information which results in better overall location estimation. The findings from the study affirm that using modelled crowdsourced Wi-Fi fingerprint data does improve the accuracy of the navigation system. The hybrid approach showed a remarkable 11.38% reduction in mean error with regards to the coordinates estimation in the experiments from the thesis, compared against using PDR information alone. This research lays the foundation for further advancements in indoor navigation technologies, emphasizing the potential of Wi-Fi fingerprint modelling as a powerful augmentation technique for PDR-based indoor navigation systems.

REFERENCES

1. Abusara, A. M. (2015) 'Indoor Positioning Techniques and Approaches for WI-FI Based Systems'. Available at: https://dspace.aus.edu:8443/xmlui/handle/11073/7855 (Accessed: 27 February 2019).
2. Chen, J., Zhang, Y. and Xue, W. (2018) 'Unsupervised Indoor Localization Based on Smartphone Sensors, iBeacon and Wi-Fi', *Sensors*, 18(5), p. 1378. doi: 10.3390/s18051378.
3. Lu, Q. *et al.* (2016) 'A hybrid indoor positioning algorithm based on Wi-Fi fingerprinting and pedestrian dead reckoning', in *2016 IEEE 27th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*. IEEE, pp. 1–6. doi: 10.1109/PIMRC.2016.7794982.
4. Moghtadaiee, V. and Dempster, A. G. (2014) 'Indoor Location Fingerprinting Using FM Radio Signals', *IEEE Transactions on Broadcasting*, 60(2), pp. 336–346. doi: 10.1109/TBC.2014.2322771.
5. Blewitt, G. (1997). Basics of the GPS technique: observation equations. Geodetic applications of GPS, 10-54.
6. Youssef, M. A., Agrawala, A., & Shankar, A. U. (2003, March). WLAN location determination via clustering and probability distributions. In Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003.(PerCom 2003). (pp. 143-150). IEEE.
7. Chen, C., Chen, Y., Lai, H. Q., Han, Y., & Liu, K. R. (2016, March). High accuracy indoor localization: A Wi-Fi-based approach. In 2016 IEEE international conference on acoustics, speech and signal processing (ICASSP) (pp. 6245-6249). IEEE.
8. Evennou, F., & Marx, F. (2006). Advanced Integration of WiFi and Inertial Navigation Systems for Indoor Mobile Positioning. *EURASIP Journal on Advances in Signal Processing*, 2006, 1-11. https://doi.org/10.1155/ASP/2006/86706.
9. Xu, Shuang & Wen, Zhigang. (2015). Multi-mode Convergence-based Indoor Wireless Positioning System Design. 10.2991/icmeis-15.2015.75. https://doi.org/10.1016/j.phycom.2020.101232.
10. Zhuang, Y., Lan, H., Li, Y., & El-Sheimy, N. (2015). PDR/INS/WiFi Integration Based on Handheld Devices for Indoor Pedestrian Navigation. *Micromachines, 6*, 793-812.
11. L. Zwirello, Xuyang Li, T. Zwick, C. Ascher, S. Werling and G. F. Trommer, "Sensor data fusion in UWB-supported inertial navigation systems for indoor navigation," 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 2013, pp. 3154-3159, doi: 10.1109/ICRA.2013.6631016.
12. L. -H. Chen, E. H. -K. Wu, M. -H. Jin and G. -H. Chen, "Intelligent Fusion of Wi-Fi and Inertial Sensor-Based Positioning Systems for Indoor Pedestrian Navigation," in IEEE Sensors Journal, vol. 14, no. 11, pp. 4034-4042, Nov. 2014, doi: 10.1109/JSEN.2014.2330573.
13. Y. Zhuang and N. El-Sheimy, "Tightly-Coupled Integration of WiFi and MEMS Sensors on Handheld Devices for Indoor Pedestrian Navigation," in IEEE Sensors Journal, vol. 16, no. 1, pp. 224-234, Jan.1, 2016, doi: 10.1109/JSEN.2015.2477444.

14. Cheng, J., Yang, L., Li, Y., & Zhang, W. (2014). Seamless outdoor/indoor navigation with WIFI/GPS aided low cost Inertial Navigation System. *Phys. Commun., 13*, 31-43.
15. P. Davidson and R. Piché, "A Survey of Selected Indoor Positioning Methods for Smartphones," in IEEE Communications Surveys & Tutorials, vol. 19, no. 2, pp. 1347-1370, Secondquarter 2017, doi: 10.1109/COMST.2016.2637663.
16. Vilaseca, D., & Giribet, J. (2013). Indoor navigation using WiFi signals. *2013 Fourth Argentine Symposium and Conference on Embedded Systems (SASE/CASE)*, 1-6. https://doi.org/10.1109/SASE-CASE.2013.6636772.
17. Barberis, C., Bottino, A., Malnati, G., & Montuschi, P. (2014). Experiencing Indoor Navigation on Mobile Devices. *IT Professional*, 16, 50-57. https://doi.org/10.1109/MITP.2013.54.
18. Tan, K., & Law, C. (2007). GPS and UWB Integration for indoor positioning. *2007 6th International Conference on Information, Communications & Signal Processing*, 1-5. https://doi.org/10.1109/ICICS.2007.4449630.
19. Akeila, E., Salcic, Z., & Swain, A. (2014). Reducing Low-Cost INS Error Accumulation in Distance Estimation Using Self-Resetting. *IEEE Transactions on Instrumentation and Measurement*, 63, 177-184. https://doi.org/10.1109/TIM.2013.2273595.
20. X. Liu, Q. Zhou, X. Chen, L. Fan and C. -T. Cheng, "Bias-Error Accumulation Analysis for Inertial Navigation Methods," in IEEE Signal Processing Letters, vol. 29, pp. 299-303, 2022, doi: 10.1109/LSP.2021.3129151.
21. Matza, A., Zivhon, R., & Ariel, E. (2012). Indoor navigation from a practical perspective. *2012 IEEE 27th Convention of Electrical and Electronics Engineers in Israel*, 1-5. https://doi.org/10.1109/EEEI.2012.6376905.
22. Li, W., Iltis, R., & Win, M. (2013). A smartphone localization algorithm using RSSI and inertial sensor measurement fusion. *2013 IEEE Global Communications Conference (GLOBECOM)*, 3335-3340. https://doi.org/10.1109/GLOCOM.2013.6831587.
23. Atia, M., Korenberg, M., & Noureldin, A. (2012). A WiFi-aided reduced inertial sensors-based navigation system with fast embedded implementation of particle filtering. *2012 8th International Symposium on Mechatronics and its Applications*, 1-5. https://doi.org/10.1109/ISMA.2012.6215167.
24. L. Zwirello, Xuyang Li, T. Zwick, C. Ascher, S. Werling and G. F. Trommer, "Sensor data fusion in UWB-supported inertial navigation systems for indoor navigation," 2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, 2013, pp. 3154-3159, doi: 10.1109/ICRA.2013.6631016.
25. Perez-Navarro A, Montoliu R, Torres-Sospedra J. “Advances in Indoor Positioning and Indoor Navigation”. Sensors. 2022; 22(19):7375. https://doi.org/10.3390/s22197375
26. K. He, Y. Zhang, Y. Zhu, W. Xia, Z. Jia and L. Shen, "A hybrid indoor positioning system based on UWB and inertial navigation," 2015 International Conference on Wireless Communications & Signal Processing (WCSP), Nanjing, China, 2015, pp. 1-5, doi: 10.1109/WCSP.2015.7341240.
27. Chapra, S. C., & Canale, R. P. (2010). Numerical Methods for Engineers (6th ed.). Boston, MA: McGraw-Hill Higher Education.

28. MathWorks. (2024). MATLAB and Simulink for Signal Processing
29. Movella, "MTw IMU Documentation," Online. https://mtidocs.movella.com/home
30. Montgomery, D. C., & Runger, G. C. (2010). Applied Statistics and Probability for Engineers. John Wiley & Sons
31. "IEEE Standard for Information Technology--Telecommunications and Information Exchange between Systems - Local and Metropolitan Area Networks--Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," in IEEE Std 802.11-2020 (Revision of IEEE Std 802.11-2016) , vol., no., pp.1-4379, 26 Feb. 2021, doi: 10.1109/IEEESTD.2021.9363693.
32. Doe, J., & Smith, A. (2021). MEMS Sensors in Mobile Devices. In IEEE (Ed.), Heterogeneous Integration Roadmap . https://eps.ieee.org/images/files/HIR_2021/ch11_MEMS.pdf
33. Z. Zeng, X. Zhang and L. Jia, "Research on a Weighted Least Squares Algorithm in Satellite Positioning," 2020 International Conference on Virtual Reality and Intelligent Systems (ICVRIS), Zhangjiajie, China, 2020, pp. 958-961, doi: 10.1109/ICVRIS51417.2020.00233.

7 APPENDIX

7.1 MATLAB Code

7.1.1 Least-Squares fit for Wi-Fi fingerprint to Euclidean Distance

```
% Calculate the pairwise Euclidean distances between all fingerprint points
euclideanDist = pdist(FPCluster);

% Calculate the maximum Euclidean distance to use as a boundary
dlim = ceil(max(euclideanDist));

% Define the fitting function,
% This ensures that the function never goes below X(1) and
% scales linearly with X(2) and X(3) as slope and intercept. (line fit)
Func = @(X, euclideanDist) max(X(1), X(2) + X(3) * euclideanDist);

% Initial guesses for the parameters in X; setting these as zeros as a
% starting point for optimization.
X0 = [0, 0, 0];

% Set lower bounds for the parameters: no lower bound for X(1),
% no upper bound for X(2), no lower bound for X(3)
lb = [0, -inf, 0];

% Set upper bounds for the parameters: the minimum real distance
% as upper bound for X(1), no bounds for X(2), inf for X(3)
ub = [min(realDist), 0, inf];

% Perform least squares curve fit
X = lsqcurvefit(Func, X0, euclideanDist, realDist, lb, ub);

% Apply the fitting function to the Euclidean distances with the
% optimized parameters to get Wi-Fi dissimilarities
Wdist = squareform(Func(X, euclideanDist));
```

7.1.2 Data filtering and populating model matrices and vectors

```
% Identify indices of distance pairs below the threshold
% Filtering too dissimilar fingerprint pairs,
temp = 0.45;
distthreshold = temp * max(Wdist(:));
neighb = find(Wdist < distthreshold);

% Convert linear indices to subscripts for accessing elements in the PDR
distance matrix
[I, J] = ind2sub(size(pdrdist), neighb);

% Filter to ensure unique pairs (I > J)
selectind = I > J;
```

```
I = I(selectind);
J = J(selectind);

% Initialize model matrices A1 and A2 for least squares regression
A1 = zeros(2 * nkFP, 2 * window);
b1 = zeros(2 * nkFP, 1);
b2 = pdrdist(sub2ind(size(pdrdist), I, J));

% Populate A1 and b1 with known point coordinates
for ii = knownPoints
    A1((2 * ii) - 1, 2 * (ii) - 1) = 1;
    b1((2 * ii) - 1, 1) = latlon_enu_test((ii), 1);
    A1((2 * ii), 2 * (ii)) = 1;
    b1((2 * ii), 1) = latlon_enu_test((ii), 2);
end

% Setup matrix A2 for x and y coordinates
A2 = zeros(2 * numel(I), 2 * nkFP);
A2(sub2ind(size(A2), (1:numel(I))', (2 * I(:)) - 1)) = 1;
A2(sub2ind(size(A2), (1:numel(I))', (2 * J(:)) - 1)) = -1;
A2(sub2ind(size(A2), (numel(I) + (1:numel(I)))', 2 * I(:))) = 1;
A2(sub2ind(size(A2), (numel(I) + (1:numel(I)))', 2 * J(:))) = -1;
```

7.1.3 Weighted Least-Squares

```
% Define the weighted least squares function
% initialize slope and intercept for the weights equation
slope = 3;
intercept = 2;
% Define the weights matrix
W = inv(diag([ones(numel(b1), 1); slope * b2 + intercept]));
% Setup the least squares function
lsqFun_w = @(x, A1, b1, A2, b2, W) W * [b1 - A1 * x; b2 - sqrt([eye(size(A2,
1) / 2), eye(size(A2, 1) / 2)] * ((A2 * x).^2))];

% Prepare initial estimate for least squares regression
x0 = b1;
x0_ = A2\sqrt([eye(size(A2, 1) / 2), eye(size(A2, 1) / 2)]\(b2.^2));
x0(b1 == 0) = x0_(b1 == 0);

% Configure options for nonlinear least squares solver
options = optimoptions('lsqnonlin', 'MaxFunctionEvaluations', 8000);

% Perform the weighted least squares regression
[xest, ~] = lsqnonlin(@(x) lsqFun_w(x, A1, b1, A2, b2, W), x0, [], [],
options);
```

7.1.4 Error calculation and plotting

```
% Calculate the error metrics
error = sqrt((latlon_enu_test(1:window, 1) - xest(1:2:end)).^2 + ...
    (latlon_enu_test(1:window, 2) - xest(2:2:end)).^2);
meanError = mean(error);
deviation = std(error);
variance = var(error);
```

```
% Plot the original data points and the estimates
figure;
hold on;
plot(latlon_enu_test(1:window, 1), latlon_enu_test(1:window, 2), 'ro');
plot(xest(1:2:end), xest(2:2:end), 'b*');
plot([latlon_enu_test(1:window, 1), xest(1:2:end)]', ...
    [latlon_enu_test(1:window, 2), xest(2:2:end)]', 'k-');
% Make the plot prettier and add error information
grid on;
grid minor;
legend('Data', 'Estimate', 'Location', 'Best');
title(['Mean error: ', num2str(meanError), 'm, Standard Deviation: ', ...
    num2str(deviation) 'm.']);
hold off;
```