Research on Enhancing Object Localization in GPS-Challenged Environments

Submitted to
Graduate School of Information Science and Technology
Osaka University

July 2024

Heng ZHOU

List of Publications

1. Journal Paper

1. Heng Zhou, Takuya Maekawa: GPS-assisted Indoor Pedestrian Dead Reckoning, Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 6(4), pp.1-36 (January 2023) (Presented at *UbiComp 2023*).

2. Heng Zhou, Takuya Maekawa: Predicting Signal Reception Information from GNSS Satellites in Indoor Environments without Site Survey: Towards Opportunistic Indoor Positioning based on GNSS Fingerprinting. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 2024 (Major revision)

2. Domestic Conference Paper (with peer-review)

1. Zhou Heng, Maekawa Takuya: Preliminary Investigation of Predicting GPS Satellite's Signal Strength in Indoor Environments, The 37th Annual Conference of the Japanese Society for Artificial Intelligence, 2023.

2. Zhou Heng, Momoe Sukegawa, Ryoma Otsuka, Hirotsugu Azechi, Kaoru Ide, Susumu Takahashi and Maekawa Takuya: Preliminary Investigation of Estimating 3D Trajectory of Mice Using Inertial Sensors and Magnetic Landmark Correction, The 47th Annual Meeting of the Japan Neuroscience Society (NEURO), 2024. (To appear)

3. Domestic Conference Paper

1. Zhou Heng, Maekawa Takuya: Preliminary Investigation of Using GPS Information to Improve Indoor Pedestrian Dead Reckoning, IPSJ SIG-UBI Technical Reports, May 2021.

2. Zhou Heng, Maekawa Takuya: Preliminary Investigation of Indoor GPS Satellite Reception Prediction via Neural Network, IPSJ SIG-UBI Technical Reports, September 2023.

4. Invited Talk

1. Zhou Heng: UbiComp/ISWC2023 Attendance Report, IPSJ SIG-UBI Technical Reports, November 2023.

5. Awards

1. Zhou Heng, Maekawa Takuya: “Students Award,” IPSJ SIG-UBI Technical Reports, June 2022.
2. Zhou Heng, Maekawa Takuya: “International Conference Presentation Award,” IPSJ SIG-UBI Technical Reports, November 2023.
3. Zhou Heng, Maekawa Takuya: “Students Award,” IPSJ SIG-UBI Technical Reports, February 2024.

Abstract

The Global Positioning System (GPS) has revolutionized location tracking and navigation, enabling applications ranging from commercial logistics to personal navigation and wildlife monitoring. However, GPS technology is significantly hampered in environments where satellite signals are obstructed by physical barriers such as buildings, dense foliage, or subterranean layers. Such environments necessitate alternative solutions to ensure reliable and precise location data, especially in indoor and underground settings where traditional GPS is ineffective.

The primary challenge in these GPS-challenged environments is to mitigate the loss of reliable satellite signals, which is crucial for the accuracy of GPS systems. In response to this, localization methods such as dead reckoning (DR) have been adapted. Dead reckoning estimates an entity's current location based on its previously known position, employing internal sensors like accelerometers and gyroscopes to track movement. However, this method is inherently prone to accumulating drift errors due to inaccuracies in sensor data and the absence of external reference points to recalibrate the path calculated by DR.

To overcome the limitations of dead reckoning, several sophisticated approaches integrating with environmental landmarks have been developed. These landmarks, which are annotated with distinct physical or signal-based features in the environment, provide external references that help calibrate and correct the estimated trajectory. However, this integration of landmarks with dead reckoning poses specific challenges that vary significantly across different environments. Firstly, in infrastructured settings—environments that have been artificially constructed with features like floors, walls, and Wi-Fi signals—such as buildings, subways, and underground malls, the selection of appropriate landmarks involves identifying low-cost solutions that are ubiquitously available and do not require extensive site surveys. Currently, there are some kinds of landmarks used in indoor environments that might include Wi-Fi or Bluetooth signal heat-map, and even architectural features like staircases but they are limited due to their high cost and low density, respectively. The challenge here lies in choosing landmarks that are both detectable and reliable over time without necessitating substantial upfront investment or ongoing maintenance. The second challenge is to effectively utilize these landmarks

with dead reckoning systems to correct drift errors, which requires the development of sophisticated algorithms. This algorithm must not only output the user's trajectory based on sensor data from accelerometers and gyroscopes but also correct this trajectory using environmental landmarks based on location and time, with the assistance of another smartphone-embedded sensor. Furthermore, the requirement of prior known starting positions is also a restriction to dead reckoning's applicability. To achieve our research goal of object localization in GPS-challenged environments, we must also consider scenarios where artificial infrastructure is absent. Therefore, the third challenge is to adapt dead reckoning for underground environments, such as animal-dug tunnels, where traditional landmarks are typically non-existent. This is particularly crucial for tracking small animals, where the absence of human-made infrastructure necessitates innovative approaches. Here, the approach shifts towards adapting dead reckoning to operate in three dimensions, accounting for the intricate and dynamic nature of subterranean tunnels. This adaptation is technically demanding and requires innovative thinking to identify new types of landmarks that can function in such environments. Current mainstream methods for animal tracking often rely on computer vision technologies, but installing such systems, such as cameras, in underground conditions presents practical and technical difficulties.

To address the first challenge of finding easily accessible, cost-effective landmarks and generalizability, which refers to the ability of a system to fuse with different systems under various conditions, in indoor environments, we have innovated a new kind of landmarks – "GPS landmarks." We developed a method that integrates indoor floor plans, outdoor buildings maps, and satellite positions to generate a detailed GPS map of signal reception like WiFi signal strength heat-maps. We used collected information of GPS satellites signal strength as output to train a neural network to learn signal propagation patterns, focusing on both regression and binary classification tasks. We collected data from 10 different environments in both sub-urban and urban areas and evaluated our method. In practice, this method has achieved a 74% accuracy of predicting high signal strength position.

To address the second challenge of utilizing new landmarks to correct dead reckoning in indoor environments, we proposed a new method with existing GPS landmarks. Specifically, we identified areas near windows where GPS signals, although not consis-

tently reliable throughout the entire space, can provide useful satellite signal information, including satellite identifiers, positions, and signal strengths at various times. By analyzing this information, we can infer the spatial relationship between the user and the GPS landmarks. This information is then fused with continuous dead reckoning data through a particle filter, which efficiently combines these sporadic GPS readings with the sensor-driven estimates of movement. This solution does not rely on a predefined starting position, enhancing flexibility for users. Our evaluation in diverse indoor settings shows that this GPS landmark-based dead reckoning system effectively minimizes drift errors, achieving a mean absolute error of approximately 1.2 meters relative to the ground truth.

Finally, to solve the third challenge of accurately tracking movements of small animals in underground environments, where traditional GPS and indoor positioning systems falter, we proposed an innovative method tailored to complex, three-dimensional spaces with additional magnetic field information. Our approach utilizes a magnetometer to capture the characteristics of magnetic fields generated by nearby rotating magnets. The captured features of magnetic fields by a magnetometer are then integrated with 3D change of direction and length predictors to accurately calculate the trajectory of a mouse by particle filter. To validate the efficacy of our proposed method, we collected 1 hour and 30 minutes of active movement data from mice. From this dataset, we performed cross-validation for detailed evaluation. Our method could achieve a mean absolute error of only 3.4 centimeters.

Contents

Chapter 1

Introduction

1.1 Background

Positioning technologies are integral to modern life, enabling a plethora of applications that extend from everyday conveniences to advanced scientific research. These technologies facilitate accurate location tracking, which is crucial for navigation, logistics, emergency response, and environmental monitoring [1, 2, 3, 4]. The importance of precise positioning is underscored by its role in optimizing routes for transportation, enhancing the efficiency of delivery services, managing assets in real-time, and studying animal behaviors in their natural habitats [5, 6, 7]. Positioning applications vary significantly between indoor and outdoor scenarios, each presenting unique challenges and requirements [8]. Outdoors, where the environment is open and less obstructed, Global Positioning System (GPS) has emerged as the fundamental solution. GPS operates through a network of satellites that transmit signals to receivers on Earth [9, 10]. By calculating the time it takes for these signals to travel from satellites to the receiver, GPS devices can determine accurate positions, velocities, and time synchronizations. This capability has led to widespread applications in fields such as aviation, maritime navigation, and personal devices, where GPS's strength lies in its global coverage and consistent accuracy in open spaces [11, 12, 13, 14]. However, the transition to indoor environments reveals the limitations of GPS. In these settings, obstacles such as walls, roofs, and other structural elements significantly attenuate or block the satellite signals,

leading to decreased accuracy or complete loss of positioning capability. This weakness poses a considerable challenge in developing reliable indoor positioning systems, as the robust and ubiquitous nature of GPS cannot be directly replicated in such obstructed environments. As a result, alternative approaches and technologies are required to achieve precise location tracking indoors, where GPS's inherent strengths are diminished by physical barriers and signal interferences [15].

In environments where GPS signals are weak or unavailable, such as indoors or underground, Dead Reckoning (DR) emerges as a valuable method for navigation. DR operates by utilizing inertial sensors like accelerometers, gyroscopes, and magnetometers to estimate the user's position based on previous known locations [16, 17, 18]. This method is particularly effective because it can track movements even without external signals, adapting to various activities such as walking, running, or climbing stairs. While DR provides autonomy and robustness against signal jamming, it faces challenges like sensor errors and the accumulation of drift over time. These errors arise from the inherent limitations of inertial sensors like sensor biases [19, 20]. Over time, these inaccuracies accumulate, leading to drift—a gradual deviation from the actual path [21, 22, 23, 24]. To counter these issues, several solutions have been proposed. One common approach is to integrate additional sensors or algorithms that can compensate for or correct the drift, especially incorporating external cues or "landmarks" into the DR system [25, 26]. Landmarks are distinctive features within an environment and they can be physical objects like doors, stairs, or elevators, or they can be signal-based, such as positions with strong Wi-Fi signals or special magnetic fields [25, 27]. The idea is to use these features as known reference points to reset or adjust the DR system whenever a user passes them, effectively reducing the cumulative errors from sensor drift [28].

This integrated system has some characteristics which differs from single systems. For the GPS-challenged environments, there are two kinds of situations. On one hand, environments with artificial infrastructure, like buildings, subways, and underground malls, feature human-made modifications such as physical structures (elevators, stairs, rooms) and technological installations (Wi-Fi networks, electrical systems). These modifications provide a variety of potential landmarks. On the other hand, environments without artificial infrastructure, like wild underground areas, lack these human-made

features, presenting a more challenging scenario for navigation due to the absence of ready landmarks. First, in environments with artificial infrastructure, there are **various options for potential landmarks** since current sensors can capture a wide range of information, both visible and invisible. Given the large spatial areas of these environments, even if there are few or no obvious landmarks, new landmarks can be created by adding additional equipment. Moreover, landmarks based on different sensors or wireless technologies can be layered together to cover areas that might be missed by others, creating a more comprehensive landmark system. Secondly, **utilizing landmarks to correct errors in dead reckoning** enhances the reliability of landmark-based dead reckoning accuracy. DR is a self-localization technique that offers continuous relative position estimation but is prone to drift, making it unsuitable for long-term use. On the other hand, wireless localization provides an absolute position but struggles to maintain continuous location estimation when access points or beacons are insufficient. By combining DR with wireless localization, both the drift issues and the intermittent coverage of wireless localization methods can be mitigated, resulting in continuous and accurate tracking [29]. Thirdly, landmark-based dead reckoning methods can be applied to **various environments and objects**. In environments completely lacking additional contextual information, new types of landmarks can be artificially added to reduce long-term dead reckoning errors. For example, in large animal tracking experiments using dead reckoning, fixed-position visual markers can serve as effective landmarks [30].

However, current landmark-based methods are not perfect and face various issues. By addressing these challenges, it is possible to enhance the localization systems for humans or other objects in different GPS-challenged scenarios.

1.2 Challenges

In this section, we focus on the specific challenges we face in developing DR systems enhanced with environmental landmarks. As previously mentioned, accurate localization in these settings is crucial due to the wide range of applications that rely on robust navigation solutions. To address this, there has been a growing focus on landmark-based methods, which utilize environmental features to improve the precision and reliability of DR systems. However, this method also faces many challenges, making it

currently unsuitable for large-scale commercial deployment. There are various challenges associated with landmark-based DR in GPS-challenged environments, including the need for low-cost and widely applicable solutions, seamless sensor integration, handling environmental changes, and ensuring reliability in diverse settings. Among these various challenges, we have identified three specific ones that are critical to advancing the field of GPS-landmark-based DR systems in GPS-challenged environments. These challenges are selected based on their potential to significantly improve the effectiveness and practicality of DR systems in diverse and demanding environments:

- **Low-cost & Ubiquitous Landmarks**

- **Fusion Methods With New Landmarks**

- **Landmark Used DR in Wild Underground.**

In the following, we will specifically introduce these challenges and their positions within the overall challenges.

1.2.1 Low-cost & Ubiquitous Landmarks

One prevalent approach to enhancing indoor navigation in GPS-challenged environments involves utilizing wireless signals that carry information, such as Wi-Fi or Bluetooth [31, 32, 33, 34]. These signals are particularly valuable as landmarks for localization due to their pervasive presence in many indoor settings, and the fact that in different areas, these signals contain distinct information, enabling them to serve effectively as landmarks. However, leveraging these wireless signals is not without its challenges. For instance, using Wi-Fi or Bluetooth as navigation aids often necessitates additional hardware installations (i.e. Bluetooth beacons or tags), leading to increased costs. Moreover, to harness these signals effectively as landmarks, comprehensive site surveys must be conducted. This process involves collecting data at every potential landmark location within the environment, a task that can be both time-consuming and labor-intensive. The need to meticulously map out signal strength and consistency across different areas requires significant effort and resources, making it less feasible for rapid deployment or adaptation in dynamic or expansive spaces. Additionally, environmental changes

may necessitate the recollection of data, further complicating the use of these signals as reliable landmarks.

Another approach to overcoming the limitations of GPS in indoor navigation is the use of physical activity-based landmarks, such as detecting the act of climbing stairs through accelerometers embedded in mobile devices [25, 35, 36]. This method capitalizes on the natural human interaction with the environment, avoiding some of the drawbacks associated with wireless signal-based approaches. However, this method also has its limitations, primarily the low density of such landmarks. Physical structures like stairs, ramps, or other unique architectural features may not be present in sufficient frequency or uniformity across all indoor environments. Consequently, while these physical activity-based landmarks can provide valuable navigation cues, their sparse distribution may not guarantee continuous and reliable navigation support. Therefore, there is a necessity for low-cost indoor landmarks that offer substantial coverage, are minimally affected by tiny changes from the surrounding environments, and do not require preliminary data collection. Finding such solutions can greatly enhance the feasibility of deploying DR systems in diverse indoor environments, making them more accessible and practical for a wide range of applications.

1.2.2 Fusion Methods With New Landmarks

There are numerous methods aimed at reducing the drift errors in dead reckoning by using landmarks. These approaches detect stable features in the environment and adjust the computed trajectory accordingly. For instance, Gu et al. [37] use multiple types of detectable landmarks, each identified by unique changes or patterns in sensor data. The locations of these landmarks are initially estimated using trajectories from dead reckoning methods. A distance constraint-based K-Means clustering algorithm then refines these locations by grouping similar sensor readings within a defined radius. This method processes a sequence of potential landmarks, using a distance threshold to determine cluster membership and forming clusters whose centers represent the refined locations of sensory landmarks. However, this type of methods has a difficulty that they are not suitable for novel “GPS” based landmarks we will present in the following chapter because of the integration of GPS landmarks with dead reckoning systems.

Although detecting the presence near a window using GPS is possible, this information alone can be ambiguous. Compared to current methods which has absolute and accurate landmarks, the definition of window-side is much wider in space. Therefore, a flexible integration framework is required, incorporating additional information such as which specific window-side the user is near and their walking direction. This flexible integration is necessary to enhance the clarity and accuracy of the positioning information.

Another significant issue with existing dead reckoning systems is their dependence on a known starting point. Some systems attempt to establish this starting point using additional sensors or through manual input by the user. For example, a system might use BLE beacons to determine the user's starting position [38], or require the user to select their starting location on a map [39]. However, these solutions introduce their own set of problems, including extra costs, which often needs to be manually input or determined using additional sensors, thereby increasing system complexity. Addressing this challenge can lead to more robust navigation solutions that are less dependent on fixed starting points and more adaptable to different environments and different kinds of wireless technologies.

1.2.3 Landmark Used DR in Wild Underground

Many studies focus on analyzing behaviors like movement trajectories of animals, primarily those inhabiting land, air or sea, which can be easily captured by GPS or cameras [40, 41, 42]. However, research on underground animals is relatively sparse, largely due to the complexities and challenges associated with tracking these creatures in their subterranean habitats. Underground environments, such as intricate tunnel systems, require the use of 3D dead reckoning to accurately calculate animal movements in all spatial dimensions. This necessity introduces a higher level of complexity due to the need to account for vertical as well as horizontal displacements, which are not typically a factor in 2D plane dead reckoning. The three-dimensional aspect is essential because underground animals navigate complex terrains, including steep inclines, declines, and winding paths, which demand precise measurement in all axes to understand their natural movements and behaviors fully.

Moreover, underground settings are vastly different from indoor environments, pri-

marily because they lack stable, recognizable landmarks. This absence of identifiable features complicates recalibrating or correcting the positioning system, leading to potential inaccuracies in tracking animal movements. Unlike indoor settings with existing infrastructure where landmarks like walls, doors, or specific installations can help refine location estimates, subterranean environments often offer no such aids, making it challenging to maintain consistent tracking accuracy over time.

Additionally, while the current mainstream methods for tracking and navigation often rely on computer vision, this approach faces significant obstacles in underground conditions [43, 44]. The lack of ambient light, combined with the presence of dust and moisture, can significantly impair the operation of cameras. Moreover, the complex and irregular geometry of tunnels and other underground structures can further complicate visual data interpretation. Considering the costs and maintenance involved in installing cameras in animal tunnels, the use of computer vision-based methods for real-world applications is nearly impractical. These challenges necessitate exploring alternative sensing technologies or methodologies that can function effectively in the demanding underground environment.

This challenge is selected because it addresses an extreme case of GPS-challenged environments, demonstrating the versatility and adaptability required for DR systems. While it may seem off-topic, tackling this challenge can provide insights and solutions that are applicable to other complex environments, thereby broadening the scope and impact of the research.

1.3 Approach

In this research, we address the problems related to the challenges of object localization in GPS-challenged environments, as detailed in Section 1.2. Specifically, we focus on developing solutions for both artificial infrastructure-existing environments and scenarios where no such artificial infrastructure exists, such as underground habitats for animals. The overview and the structure of this research are presented in Fig. 1.1.

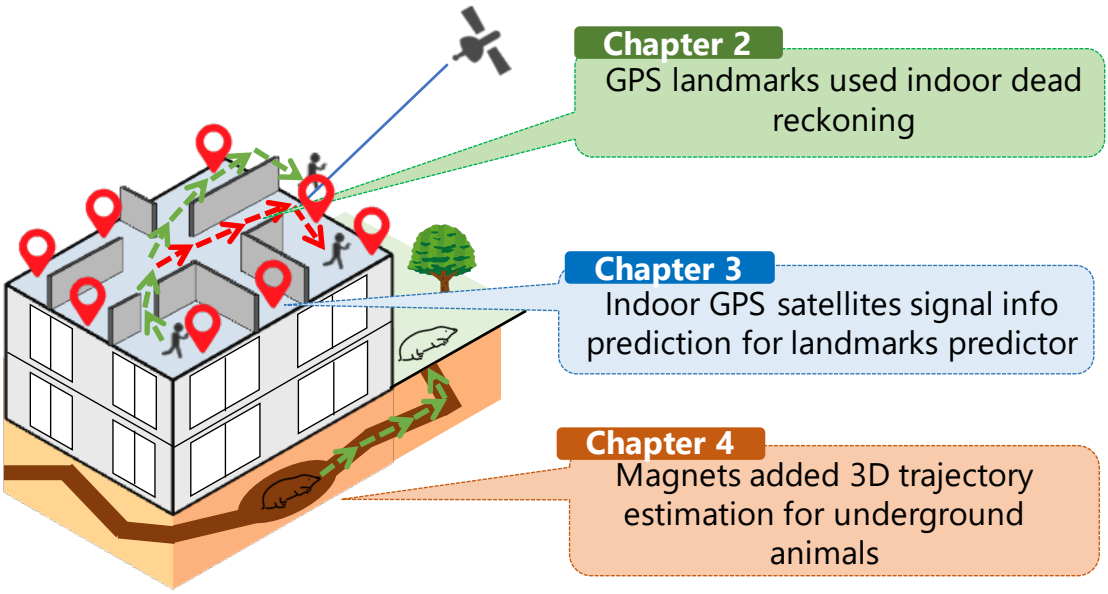


Figure 1.1: Overview of this study

1.3.1 GPS-assisted Indoor Dead Reckoning

In Chapter 2, we delve deeply into the challenges of identifying ubiquitous and low-cost landmarks and the integration of landmark refinement with dead reckoning, as highlighted in Sections 1.2.1 and 1.2.2, with a focus on indoor environments. This landmark should be robust against environmental changes and not require extensive pre-survey or installation efforts, addressing the key problems of cost and practicality. We discovered that GPS satellite signal reception possesses unique features in infrastructure-existing indoor environments [45, 46, 47]. Additionally, since GPS is an existing infrastructure that requires no extra installation, is applied in almost all indoor environments, and is easily accessed, it presents a valuable resource. Moreover, GPS satellite information data resembles time-series data and can be combined and synchronized with inertial sensor data for dead reckoning. Our approach leverages GPS satellites' information to pinpoint landmarks that effectively correct accumulated errors in dead reckoning. Specifically, we identify window-side areas as key GPS landmarks due to their unique reception characteristics. Despite GPS-challenged indoors, we observed that it often delivers strong signals near windows, particularly when a satellite's path aligns with a window's direction. This phenomenon enhances signal reception for individuals near

these windows. Therefore, to solve the sub-challenges from the second challenge, we designed multiple novel components utilizing this insight. Specifically, we designed a system that detects if a user is nearing a window-side area and identifies which window-side area they are approaching within a given floor plan. Moreover, to integrate the results from GPS data with inertial sensor data effectively, we implemented a novel dead reckoning system using machine learning techniques. To further facilitate this integration, we developed a method based on a particle filter with initial positions generator, forward and backward tracking that merges GPS satellite information with dead reckoning data. This method is particularly innovative because it does not require a known starting position, significantly broadening its applicability and utility in various indoor scenarios.

1.3.2 Indoor GNSS Satellites Signal Reception Information Prediction

In Chapter 3, we tackle the remaining issue in Section 1.2.1: generalizability in GPS-challenged indoors. Our approach significantly enhances generalizability compared to current mainstream landmark methods, as it is designed to seamlessly integrate with other indoor positioning systems. We find the propagation of GPS signals in indoor environments follows certain patterns [45] and also aligns with the universally low-cost landmarks we previously mentioned. We propose a method that combines an indoor floor plan, outdoor buildings map, and satellite position as inputs to a machine learning based model to predict GPS satellite signal reception information in position of interest in a target environment. Then, these predictions can be combined to generate a signal reception map, similar to a Wi-Fi signal strength map. These inputs are easily accessed because they are from online sources or building administrators so that it does not need a site-survey in advance. This design allows our system to be easily fused with other systems, enhancing its adaptability and utility across various indoor environments.

1.3.3 3D Trajectory Estimation of Underground Animals

In Chapter 4, we overcome the challenge of implementing a dead reckoning for underground animals in wild underground environments, as mentioned in Section 1.2.3. Although there are no directly usable landmarks, we can create low-cost landmarks using features of magnetic fields generated by cheap magnets. We explore the application of neural network models to estimate the 3D trajectory of underground animals equipped with a tiny inertial sensor. This model is designed to calculate the walking distances and changes in direction within the 3D space, accurately capturing the nuances of animal movements. To enhance the precision of these trajectory estimates, we incorporated magnetic field information into the model. By utilizing a novel correction method based on a particle filter, similar to our approach in the first solution, we were able to refine the trajectory predictions further. This integration of magnetic field data generated by additional low-cost magnets helps correct any drifts or inaccuracies that might occur due to the inherent limitations of inertial sensors alone.

1.4 Research Contribution

In this thesis, we make the following contributions.

In Chapter 2:

- This research represents the inaugural effort to rectify the accumulated inaccuracies in indoor Pedestrian Dead Reckoning (PDR) by utilizing window-side GPS landmarks. This approach eliminates the need for additional infrastructure and avoids the collection of training data within the specific target environment.
- To enable GPS-assisted indoor PDR, we have developed the window-side detector, window orientation classifier, and walking direction predictor. These components are integrated seamlessly into the PDR methodology. The key technical contributions of this study include i) The creation of these modules, which are adaptable for training in various settings outside the target environment by isolating the universal characteristics of GPS signals. ii) The introduction of an innovative framework that effectively merges GPS signal reception information with inertial sensor data, thereby ensuring precise PDR.

- Comprehensive experiments conducted on diverse trajectories in real-world settings have demonstrated the effectiveness and practicality of using GPS landmarks for indoor positioning enhancements.

In Chapter 3:

- This research is the inaugural effort to forecast GPS signal reception information utilizing readily accessible data, eliminating the need for manual data gathering in the target environment.

- To facilitate automatic indoor signal reception estimation, we introduce a predictive methodology that merges data from GPS satellite positions, indoor floorplans, and 3D maps of both the target and its surrounding buildings. The technical contributions of this study include i) The establishment of a pipeline that efficiently incorporates readily accessible information, such as satellite positions, to predict indoor signal reception at specific positions of interest. ii) The creation of a model that adeptly fuses feature maps at different scales derived from the accessible information.

- Extensive experiments across diverse buildings in real-world scenarios have validated the viability and effectiveness of our approach.

In Chapter 4:

- To enhance the precision of animal trajectory estimates, we integrate magnetic field information into the model. By adopting a novel correction technique based on a particle filter, we refine the trajectory predictions, significantly improving the accuracy of our underground tracking system. The introduction of an innovative model that leverages neural networks to estimate 3D trajectories of underground animals, capturing their movement in all spatial dimensions without the need for direct visual confirmation.

- Extensive experiments conducted in simulated underground settings have demonstrated the robustness and applicability of our approach.

1.5 Structure of the Thesis

This thesis is structured as follows.

- *Chapter 1 - Introduction*

  Firstly, we introduce the thesis, providing a background and outlining the challenges of localization and tracking systems using multimodality sensors in GPS-challenged environments.

- *Chapter 2 - GPS-assisted Indoor Pedestrian Dead Reckoning*

  In this chapter, we present a solution for using GPS signal reception information to correct pedestrian dead reckoning (PDR) by smartphones in indoor settings. We proposed a model that detects the proximity of a user to a window and the window's orientation. The research in this chapter is based on work [47] previously published.

- *Chapter 3 - Predicting Signal Reception Information from GNSS Satellites in Indoor Environments without Site Survey*

  Here we introduce a method to predict GPS signal reception for positions of interest indoors. The inputs include real-world 3D building maps, indoor floor plans, and GPS satellite position information, all of which are easily accessed. A neural network baed model that effectively fuses different scales of maps is also proposed. The study in this chapter is based on our work published in [48].

- *Chapter 4 - 3D Trajectory Estimation of Underground Animals*

  In this chapter, we introduce a method for estimating the 3D trajectory of underground animals in simulated underground environments. We propose a model that integrates inertial sensor data inputs and correction mechanisms by additional magnetic field data to enhance accuracy. This chapter is based on our work published in [49].

- *Chapter 5 - Conclusion and Future Work*

In Chapter 5, we conclude the thesis with a comprehensive summary of the contributions made throughout the research and a discussion on potential future directions in GPS-challenged environment scenarios.

Chapter 2

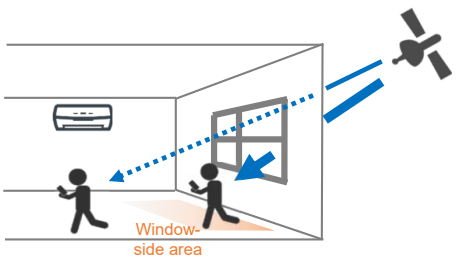
GPS-assisted Indoor Pedestrian Dead Reckoning

2.1 Introduction

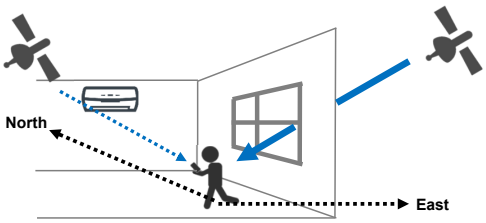
2.1.1 Background and Problems

Since ancient times, positioning technology has been crucial in various fields. With the advancement of technology, the advent of GPS has significantly facilitated human activities. However, as people's needs continue to grow, the limitations of GPS for indoor positioning have become increasingly apparent. To achieve accurate positioning in indoor environments where GPS is not available, researchers have focused on indoor pedestrian dead reckoning (PDR) using embedded inertial sensors in smartphones. PDR has various practical applications, such as indoor navigation, augmented reality, and healthcare monitoring [50]. However, PDR relying only on inertial measurement units (IMUs), including accelerometer, gyroscope or compass, has two main drawbacks: i) accumulated errors in the estimated walking trajectory due to gyroscope drift and stride prediction errors, and ii) the need for initial position and direction information since inertial sensors alone cannot provide absolute coordinates.

To address the first problem, researchers have utilized indoor landmarks detected by various smartphone sensors to correct accumulated errors [25]. For example, Bluetooth modules can detect nearby BLE beacons installed in the environment [33, 34], and



(a) Detecting window-side area



(b) Identifying orientation of window-side area

Figure 2.1: Basic idea of using GPS signal information to detect indoor landmarks. (a) When a smartphone is close to a window, it receives strong GPS signals, enabling us to know whether the smartphone is at a window-side area. (b) When a smartphone is close to a window facing the east, it receives strong GPS signals of a satellite from the east, enabling us to know the orientation/direction that the window faces.

magnetic sensors can detect magnetic fields emitted from infrastructure like elevators [27, 51]. Additionally, location-specific activities and actions, such as climbing stairs, can be tracked by smartphone sensors and used as landmarks [35, 36, 25]. These methods, combined with map matching, enable PDR without initial position information [39, 52].

However, these landmark-based methods opportunistically correct errors only when a user is near a detectable landmark, highlighting the need for increasing landmark density. The cost of installing extra devices, such as BLE beacons, and the labor-intensive data collection for detecting landmarks, such as magnetic fingerprints and human activities, must also be considered.

2.1.2 Approach

In this study, we propose to use 'GPS' landmarks that do not require installation of new devices or pre-acquisition of training data in a target environment. We propose to focus on the fact that, at window-side areas inside buildings, we can receive GPS signals from several GPS satellites, although these signals cannot provide accurate location measurements indoors. Compared to areas far from the windows, the smartphone

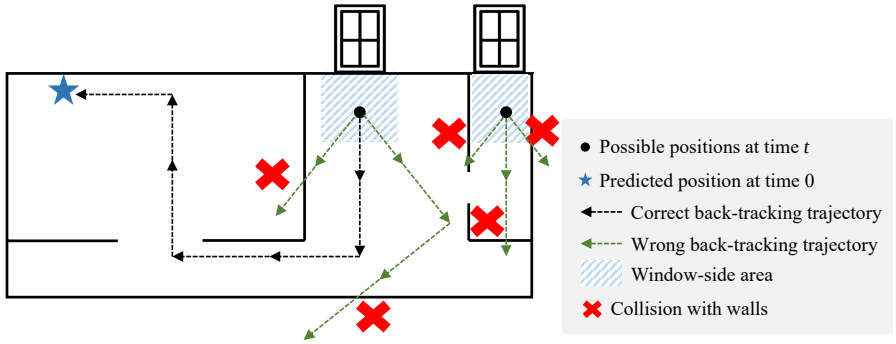


Figure 2.2: Back-tracking smartphone positions starts from time t, when the device is detected near window-side areas facing the top direction in the figure. At t, multiple position candidates are generated within these window-side areas because the exact location is unknown. We then back-track positions using inertial sensor data, creating movement paths in various directions since the exact movement direction is also uncertain. Paths that collide with walls are discarded as incorrect. Forward-tracking follows a similar approach.

receives stronger GPS signals at areas near the windows as shown in Fig. 2.1 (a); therefore, it is possible to estimate the timing when the smartphone is at close vicinity to the window. Moreover, the signal from a GPS satellite contains information about the azimuth of the satellite, enabling us to estimate which direction the window is facing. For example, when the smartphone is at the window-side of an outer wall facing east in a building, the signal strength from GPS satellites in the eastern direction will be stronger, as shown in Fig. 2.1 (b). Therefore, when the signal strength from GPS satellites located at the eastern direction is strong, it can be estimated that the smartphone is located at the window-side of the eastern wall. However, we cannot uniquely identify a single landmark, i.e., window-side area, based on GPS information because the target building usually has multiple window-side areas close to walls facing the same orientation. To deal with this problem, we leverage map matching using the floor plan including information about obstacles, i.e., inner walls, of the target environment to eliminate the ambiguity.

In addition, we explore PDR that does not require initial positions as inputs by leveraging GPS signals. As mentioned above, GPS signals can be used to identify the

time t and the orientation o of a window-side area the smartphone is in. In other words, we can generate some candidates of the absolute coordinates of the smartphone at time t within window-side areas facing said orientation o. Then, from the absolute coordinates at time t, the absolute coordinates at time 0 (i.e., initial position) can be estimated by back-tracking the movement using inertial sensors with map matching, as shown in Fig. 2.2. As shown above, we can implement PDR, which does not require known initial positions. Note that, as mentioned above, when there are multiple window-side areas close to walls facing the same orientation, there exist multiple position candidates at time t. Because the walking direction relative to the window normal at time t is also unknown, it is necessary to generate a candidate moving toward each direction (from 0 to 360 degrees) and then back/forward-track each candidate, which is computationally expensive. Although it is possible to measure the walking orientation at time t through the digital compass in the smartphone, its accuracy is low indoors owing to magnetic interference. Therefore, a digital compass is used together with GPS signal information to roughly estimate the walking direction. The walking direction is estimated based on the idea that the GPS signal strength gradually increases as the user walks toward a window.

The proposed method is composed of a window-side detector and a window orientation classifier for window-side landmark detection, and then, by incorporating them into the PDR module, the method corrects PDR accumulated errors. In the window-side detector and window orientation classifier, we use time-series GPS signals to estimate the proximity to window-side areas and the orientation of the window normal of a window-side area at each time step. In addition, a walking direction predictor in the proposed method estimates a walking direction relative to the window normal by using GPS signals. Note that, to reduce installation costs of indoor landmarks, we assume that training data for detecting window-side areas, window orientation prediction, and walking direction prediction are collected in environments other than the target environment. However, the shapes and orientations of outer walls of different buildings are also different, and positions of GPS satellites change with time (training data collection time vs. test data collection time), making it difficult to detect GPS landmarks independent of environments, which is the technical challenge of this chapter. To deal with these issues, we develop modules for window-side detection, window orientation prediction,

and walking direction prediction that can be used in an environment with any outer wall orientations by extracting GPS-related information independent of the wall orientation and positions of GPS satellites.

2.1.3 Contributions

Our study provides three main research contributions.

- This study is the first work that corrects accumulated errors of indoor PDR by using window-side GPS landmarks that do not require additional infrastructure and training data collected in the target environment.
- To achieve GPS-assisted indoor PDR, we design the window-side detector, window orientation classifier, and walking direction predictor, which are incorporated into the PDR method. The technical contributions of this study are i) the designs of these modules that can be trained in environments other than the target environment by extracting environment-independent features of GPS signals, and ii) a pipeline of efficiently introducing GPS signal information into inertial sensor data to achieve accurate PDR.
- Extensive experiments on different trajectories collected in real environments showed the feasibility of GPS landmarks.

2.2 Related Work

2.2.1 Pedestrian Dead Reckoning (PDR)

Conventional PDR

Pedestrian Dead Reckoning (PDR) systems have been developed as a solution to navigation in indoor environments or in areas with weak and unstable GPS signals. The main advantage of PDR relying on inertial sensors is the robustness against environmental differences, as it uses accelerometer readings to detect steps and estimate walking distance, and uses gyroscope signals to compute the heading direction [17]. From accelerometer

data, PDR systems detect the user's steps. A peak detection algorithm leverages a centered moving average to smooth the acceleration magnitude and applies windowed peak detection to find peaks related to heel impact [53]. For walking distance estimation, the distance traveled by the user is determined by the cumulative strides, with each stride representing the distance traveled within one step. Most step-length estimation methods are based on empirical methods using walk frequency (step frequency) and acceleration variance as main factors [54]. Gyroscope readings are directly used for heading direction, with angular velocity being iteratively integrated to generate the relative direction. However, the main problem of PDR is the accuracy of the heading direction because gyroscope drift causes errors that accumulate over time. To improve the accuracy of PDR, researchers have focused on reducing errors in heading direction estimation [55, 56]. A popular solution is Zero Velocity Updates (ZUPTs), which close the integration loop of angular velocity periodically by applying external constraints to the PDR system to deal with drift [21, 22]. These methods detect the stance phase and apply zero velocity during the stance duration. Finally, with a known starting position, the system can integrate each step's displacement with direction change to get a full trajectory [19].

Machine Learning-based PDR

Unlike conventional PDR systems with step displacement segmentation and stride estimation, state-of-the-art neural network-based PDR systems process raw inertial data in batches to estimate displacement and angular change [57, 58]. Although raw inputs (acceleration and angular velocity) are independent of continuous IMU measurement, they have strong temporal dependencies and represent walking activities. Therefore, deep recurrent neural networks (RNNs), especially long-short term memory (LSTM) networks, can capture these dependencies and leverage them to identify potential connections between data features and walking characteristics. To employ neural network-based PDR in low-end devices like smartphones, Chen et al. [59] designed a PDR method based on convolutional neural networks. In this study, we leverage state-of-the-art neural network-based PDR as the foundation and integrate our GPS landmark module to correct accumulated errors in the PDR method.

2.2.2 Landmark-assisted PDR

As PDR systems relying only on inertial sensors suffer from accumulated errors, mobile and ubiquitous computing researchers have developed PDR systems that correct accumulated errors by leveraging various smartphone sensors to detect indoor landmarks. These systems use smartphone sensors to detect some "unusual" positions and mark them as landmarks. When a smartphone user approaches or passes by a landmark, the PDR systems correct the position estimate to the landmark's position. Currently, there are many methods to detect indoor landmarks. Abdelnasser et al. [25] classified landmarks into two categories by data features: Seed Landmarks (SLMs) and Organic Landmarks (OLMs). SLMs are positions known a priori that exhibit obvious changes in sensors in smartphones, such as elevators and staircases [60, 36], whereas OLMs are positions that may not be physical but have obvious signatures by clustering data from sensors, such as Wi-Fi or GSM signals [61, 32]. For example, positions without any Wi-Fi and GSM signals in a building can be used as OLMs.

Wireless signaling devices have been widely used as indoor landmarks, e.g., BLE beacons for proximity detection and Wi-Fi access points for creating Wi-Fi fingerprints [62, 31]. Abnormal magnetic fields detected in indoor environments have also been used as indoor landmarks [51, 27]. Lee et al. [51] used a robot to collect magnetic fields in all areas of a target building and then picked areas with abnormal magnetic fields. After labeling them and training a convolutional neural network for detecting the magnetic landmarks, these landmarks could be detected by smartphones. Additionally, visible light has been used as indoor landmarks [63, 64]. Sakshi et al. [63] used visible light communication-based indoor positioning, where light emitted by dedicated bulbs is used to send position signals for identification. Therefore, a smartphone can determine its position after receiving signals from LED lamps by obtaining information from a database storing LED lamp identifiers and their positions. Moreover, location-specific activities and actions, such as ascending/descending stairs, have been used as landmarks [35, 65]. A method called ActionSlam aims to build landmarks by activity recognition to improve PDR [35]. In their work, they used different sensors to record some daily activities, such as eating or opening/closing doors, and then created a ground truth map with action landmarks marked at positions where they expected the corresponding ac-

Table 2.1: Comparison of different types of landmarks. Note that FP stands for fingerprinting

Landmark type	Training data collected in advance in target env	Installation of additional infra	Spatial ambiguity of landmarks	Density of landmarks
Magnetic FP	Yes	No	Low	High
Visible light	No	Yes	Low	Depends on # devices
Wi-Fi FP	Yes	No	Middle	High
Bluetooth FP	Yes	Yes	Middle	High
Bluetooth proximity	No	Yes	Low	Depends on # devices
Acceleration-based	No	No	High	Low
Activity recognition	Yes	No	Middle	Middle/High
GPS (this work)	No	No	Middle/High	Middle

tions to happen.

However, many of these methods require substantial costs for equipment installation and/or training data collection. We summarize the features of these methods and our proposed GPS landmarks in Table 2.1. While the proposed GPS landmarks have significant advantages regarding the costs of device installation and training data collection, the spatial ambiguity of the landmarks is high when a target environment has multiple window-side areas facing the same orientation. Therefore, this study leverages map matching and GPS-based walking direction prediction to reduce the ambiguity. While acceleration-based landmarks using staircases also do not require device installation and training data collection, these landmarks are sparse in many cases.

Note that we do not believe that error correction using GPS landmarks will replace other types of indoor landmarks. As smartphones have various sensors, landmark-based error correction should be implemented by using various types of landmarks to increase

the density of landmarks in a target environment. We believe that the proposed GPS landmark is one of the landmarks with the lowest installation costs because it does not require extra infrastructure or training data in the target environment.

2.2.3 Multi-modal Indoor Localization

With the development of smartphones and small sensing devices, various sensors can be equipped in a single device. Fusion of multi-modal sensors has become a trend in the studies on indoor localization. These studies correct deviations in walking direction by using various sensors such as Wi-Fi modules, magnetometers, and cameras [66, 67, 68]. For example, Venkatnarayan et al. [68] proposed a Wi-Fi based indoor localization method without fingerprints. In their study, they used Wi-Fi signals to calculate the distance traveled by a subject as well as an accelerometer to predict the distance, and then used the Kalman filter to correct the wrong walking direction by fusing the distance.

2.3 Proposed Method

2.3.1 Preliminaries

We assume that a pedestrian holds a smartphone equipped with inertial sensors including a accelerometer as well as a gyroscope, a digital compass, and GPS module. From these sensors, we collect time-series data $\mathcal{S} = \{\mathcal{S}^i, \mathcal{S}^c, \mathcal{S}^g\}$, where $\mathcal{S}^i$ consists of the time-series 6-axis inertial measurement data from the accelerometer and gyroscope, $\mathcal{S}^c$ includes the time-series orientation data relative to north, and $\mathcal{S}^g$ comprises the time-series GPS satellite data. Additionally, $\mathbf{s}_t^i \in \mathbb{R}^3$ represents the IMU data at time t, $\mathbf{s}_t^c \in \mathbb{R}^3$ is the orientation data at time t, and $\mathbf{s}_t^g \in \mathbb{R}^3$ includes GPS information such as the elevation angle, azimuth, and signal strength of visible satellites at time t.

Moreover, we assume the availability of floor plan information (as shown in Fig. 2.2) for the target floor. This floor plan includes details about the outer shape and orientation of each wall. In this study, the orientation of a wall is defined as the angle, relative to north, of the line perpendicular to the surface of the wall. The floor plan also specifies

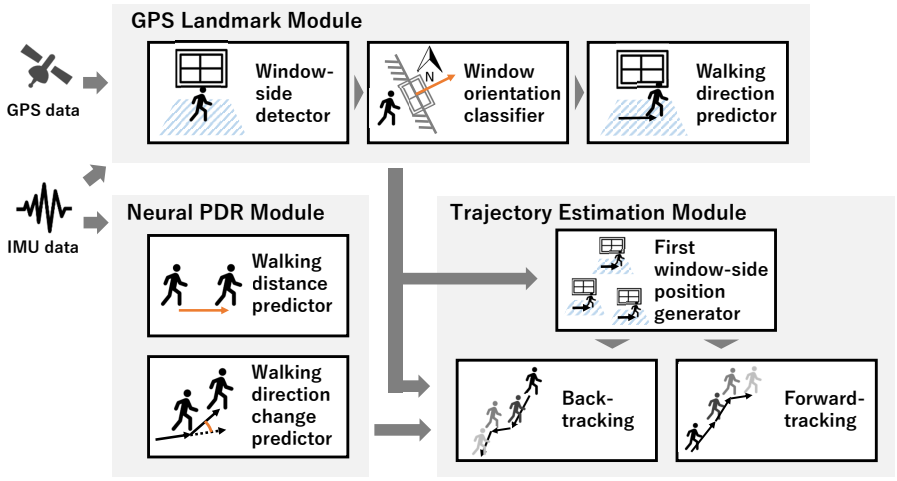


Figure 2.3: Overview of the proposed method

the position of each window, indicating which wall each window is installed on. The window-side area is defined as an area within d_w meters from the window, where d_w represents the distance from the window. Additionally, the floor plan provides positional information about the inner walls, which is used for map matching.

2.3.2 Overview

Modules

Figure 2.3 presents an overview of the proposed method, which comprises three main modules:

- **Neural PDR module**: This module includes the walking distance predictor and the walking direction change predictor, which respectively estimate the walking distance and relative change of walking direction within a time window, using accelerometer and gyroscope data.
- **GPS landmark module**: This module identifies GPS landmarks and includes three sub-modules: the window-side detector, window orientation classifier, and walking direction predictor. The window-side detector utilizes GPS information to determine whether the user is within a window-side area at each time t. When the user is detected to be in a window-side area at time t, the window orientation

classifier estimates the orientation of that window. The walking direction predictor estimates the walking direction relative to the window normal, aiding in reducing computational costs in the trajectory estimation module.

- **Trajectory estimation module**: Based on a particle filter, this module integrates outputs from the previous modules to estimate a walking trajectory. It comprises three sub-modules: first window-side position generator, forward-tracking, and backward-tracking. When the GPS landmark module first detects the user in a window-side area facing orientation o at time t_i, the first window-side position generator creates random candidate positions in window-side areas facing o, using the walking direction predicted by the walking direction predictor. Forward-tracking then employs the Neural PDR module and map matching to reconstruct the trajectory from t_i onward. The outputs of the window-side detector and window orientation classifier correct accumulated errors in the Neural PDR module. Backward-tracking recovers the trajectory from t_i to time 0.

Procedures

The procedures of these modules are outlined as follows:

(i) Initially, the user begins walking from an unknown initial position at time 0, making the trajectory indeterminable at that moment.

(ii) As the user moves and passes through a window-side area, the window-side detector and window orientation classifier identify the user entering a window-side area facing orientation o [degree(s)] at time t_i with high confidence. The first window-side position generator then generates candidate position estimates at time t_i within window-side areas facing orientation o [degree(s)]. The walking directions of these candidates at time t_i are determined by the walking direction predictor.

(iii) The trajectory estimation module then uses map matching and the Neural PDR module's output to backtrack the trajectory from t_i to time 0.

(iv) Simultaneously, the trajectory from t_i onward is forward-tracked using the Neural PDR module's output. Throughout both forward- and back-tracking, the window-side detector and window orientation classifier correct accumulated trajectory errors.

2.3.3 Neural PDR Module

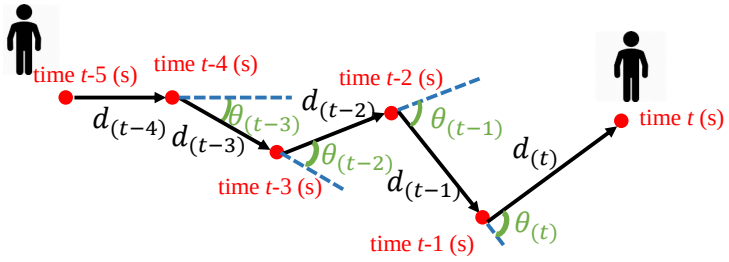


Figure 2.4: Displacement d_t and relative walking direction change θ_t are predicted every second using sensor data between $t-1$ and t. The movement path is then reconstructed using these predicted values.

In this module, we estimate the displacement and change in walking direction within a W-second time window, as shown in Fig. 2.4 ($W = 1$ in this study). Because it utilizes state-of-the-art neural network-based techniques [58, 57, 69], we will provide a brief explanation of this module. The method for acquiring ground truth values (i.e., d_t and θ_t in Fig. 2.4) will be discussed later.

Walking Distance Predictor

The input for the walking distance predictor is a 1-second window of accelerometer and gyroscope data $\mathbf{s}_t^i = \{s_{t-W_i}^i, ..., s_{t-1}^i, s_t^i\}$, where W_i corresponds to the 1-second window size. The input data is fed into a neural network for distance prediction, which comprises an LSTM layer with 32 units using the ReLU activation function, a fully connected layer with 8 units, a dropout layer, and an output layer with a linear activation function. The output of this network is the displacement d_t between time $t-1$ and time t, as shown in Fig. 2.4. The model is trained using the Adam optimizer [70] to minimize the mean squared error (MSE) between the predicted displacement and the ground truth.

Walking Direction Change Predictor

The input for the walking direction change predictor is a 1-second window of gyroscope data. A neural network for predicting directional change comprises a bi-directional LSTM layer with 32 units using the ReLU activation function, a fully connected layer with 8 units using the ReLU activation function, and an output layer with a linear activation function. A Advanced model is employed for the walking direction change predictor because simpler models failed to capture slight changes in gyroscope data. The bi-directional LSTM allows the model to consider both past and future context, improving its ability to detect subtle variations in the data. The output of the walking direction change predictor is the angular change in walking direction $\Delta\theta_t$ between time $t-1$ and time t, as shown in Fig. 2.4. The model is also trained using the Adam optimizer to minimize the MSE between the predicted angular change and the ground truth.

2.3.4 GPS Landmark Module

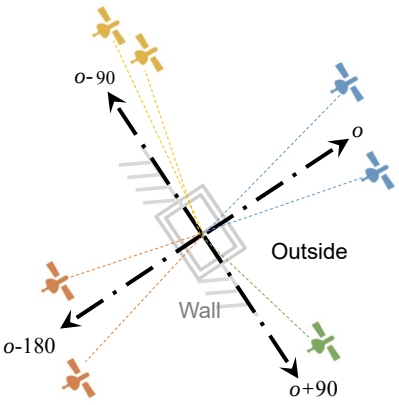
Window-side Detector

The window-side detector estimates whether the user is within a window-side area at each time step. In this study, a window-side area refers to an area within d_w meters from a window. It can be observed that as the user gets closer to the window-side, some GPS signals become stronger. Based on this observation, we use the time series of signal strengths from the top-N_G satellites (in terms of signal strength) as inputs for the window-side detector.

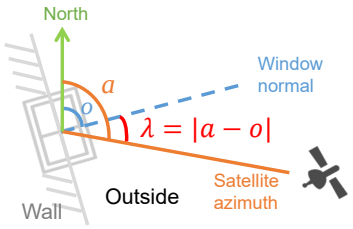
The inputs for the window-side detector are extracted from a time window of GPS data $\mathbf{s}_t^g = \{s_{t-W_g}^g, ..., s_{t-1}^g, s_t^g\}$, where W_g is the window size. The time series of signal strengths from the top-N_G satellites, which have the highest mean signal strength within $\mathbf{s}_t^g$, are chosen as inputs. In other words, the input is an NG-dimensional time series with a length of W_g, where N_G satellites are sorted in descending order of their mean signal strength. This allows us to determine if the smartphone is in a window-side area, independent of the environment and positions of the GPS satellites. The output of the window-side detector is a binary value at time t, indicating whether the user is

in a window-side area or not at time t. The window-side detector is a binary classifier consisting of two bi-directional LSTM layers with 64 and 32 nodes using the ReLU activation function, a densely connected layer with 16 nodes using the ReLU activation function, and an output layer with the softmax activation function. The network is trained to minimize the binary cross-entropy loss using the Adam optimizer.

Window Orientation Classifier



(a) Selecting GPS satellites positioned at four different orientations.



(b) Angle between window normal and satellite azimuth

Figure 2.5: Selecting GPS satellites for the window orientation classifier. (a) Satellites are chosen near the four different orientations: (o (window normal), $o - 90$, $o - 180$, $o + 90$) [degree(s)]. Signals from satellites at orientation o are expected to be strong when the smartphone is near the window. (b) For each orientation, the top-N_G satellites with the smallest λ are selected.

When the window-side detector identifies that the user is in a window-side area, the window orientation classifier estimates the orientation of the window (i.e., window normal), which corresponds to the orientation of the outer wall where the window is installed. The classifier is designed based on the idea that when a smartphone is near a window, the signals from satellites whose azimuth aligns with the window's orientation are stronger than those from other satellites, as illustrated in Fig. 2.1(b).

The window orientation classifier (and other machine learning models) are trained in environments where the wall orientation can differ from that in the target environment. Therefore, the classifier must be designed to handle any wall orientation. To achieve this, the classifier predicts whether the orientation of the window of interest matches a certain candidate orientation or not (binary classification: "True" vs. "False"), mainly using GPS satellite information whose azimuth aligns with that orientation. If the floor plan of the target environment is available, orientation candidates can be derived from it (e.g., north, west, east, south for a building with four walls). The classifier determines if the window of interest faces each candidate orientation, and the candidate with the highest "True" probability is regarded as the window's orientation. This design ensures an environment-independent classifier.

The inputs for the window orientation classifier are also derived from a time window of GPS data $\mathbf{s}_t^g = \{s_{t-W_g}^g, ..., s_{t-1}^g, s_t^g\}$. To determine if the window's orientation aligns with candidate orientation o degree(s), the input is calculated using GPS satellite information from four orientations (o, $o + 90$, $o - 90$, $o - 180$) [degree(s)], as shown in Fig. 2.5(a). When the window faces o, signals from satellites at o degree(s) are expected to be stronger than those from other orientations. For o degree(s), we select the top-N_G satellites whose azimuth is closest to o degree(s), as shown in Fig. 2.5(b). The time series of signal strength, the absolute angle λ between the window normal and satellite azimuth, and the elevation angle of each top-N_G satellite are used as inputs. These satellites are sorted in ascending order of λ. The same procedure is applied to extract time series data for the other orientations ($o + 90$, $o - 90$, $o - 180$) [degree(s)]. As a result, the inputs for the classifier are a $4 \times 3 \times N_G$ dimensional time series with a length of W_g at time t as like this

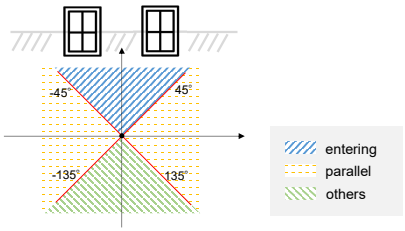


Figure 2.6: Three classes of walking direction when near the window.

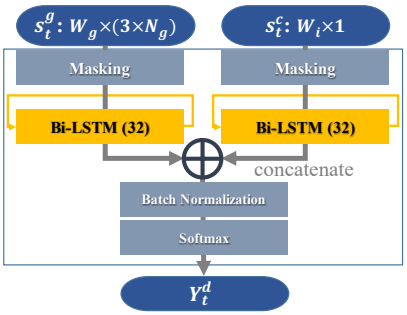


Figure 2.7: Network architecture of the walking direction predictor.

$$\mathbf{In}_t = \begin{bmatrix} \begin{bmatrix} ss_{t,o,1} & \lambda_{t,o,1} & el_{t,o,1} \\ ss_{t,o,2} & \lambda_{t,o,2} & el_{t,o,2} \\ \vdots & \vdots & \vdots \\ ss_{t,o,N_G} & \lambda_{t,o,N_G} & el_{t,o,N_G} \end{bmatrix}, \begin{bmatrix} ss_{t,o+90,1} & \lambda_{t,o+90,1} & el_{t,o+90,1} \\ ss_{t,o+90,2} & \lambda_{t,o+90,2} & el_{t,o+90,2} \\ \vdots & \vdots & \vdots \\ ss_{t,o+90,N_G} & \lambda_{t,o+90,N_G} & el_{t,o+90,N_G} \end{bmatrix} \\ \begin{bmatrix} ss_{t,o-90,1} & \lambda_{t,o-90,1} & el_{t,o-90,1} \\ ss_{t,o-90,2} & \lambda_{t,o-90,2} & el_{t,o-90,2} \\ \vdots & \vdots & \vdots \\ ss_{t,o-90,N_G} & \lambda_{t,o-90,N_G} & el_{t,o-90,N_G} \end{bmatrix}, \begin{bmatrix} ss_{t,o-180,1} & \lambda_{t,o-180,1} & el_{t,o-180,1} \\ ss_{t,o-180,2} & \lambda_{t,o-180,2} & el_{t,o-180,2} \\ \vdots & \vdots & \vdots \\ ss_{t,o-180,N_G} & \lambda_{t,o-180,N_G} & el_{t,o-180,N_G} \end{bmatrix} \end{bmatrix}$$

where $ss_{t,o',n}$, $\lambda_{t,o',n}$ and $el_{t,o',n}$ represent signal strength, absolute angle and elevation angle of satellite n for orientation o' at time t. The output is a binary value indicating whether the window orientation is o degree(s) at time t.

The neural network for the window orientation classifier is a binary classifier comprising one bi-directional LSTM layer with 16 units using the ReLU activation function, a Batch-Normalization layer, and an output layer with the softmax activation function. The network is trained to minimize binary cross-entropy loss using the Adam optimizer.

Walking Direction Predictor

The walking direction predictor uses digital compass and GPS data to estimate the "rough" direction of walking when the user is detected entering a window-side area.

The trajectory estimation module, detailed below, estimates the trajectories before and after the user initially enters the window-side area, based on the predicted walking direction. The output of the walking direction predictor is classified into three categories: entering, parallel, and others, as illustrated in Fig. 2.6.

- "Entering" indicates that the user is walking towards the window, defined as angles between the walking direction and the wall normal line from -45 degrees to 45 degrees.

- "Parallel" means the user walks parallel to the wall surface, defined as angles from 45 degrees to 135 degrees or from -45 degrees to -135 degrees.

- "Others" means the user is moving away from the window area, defined as angles from 135 degrees to 225 degrees (or -135 degrees).

Because usually period of "others" classification covers half second of the whole period of window-side area, almost in few cases they will be used as candidates of initial directions.

The inputs of this walking direction predictor come from a time window of digital compass data $\mathbf{s}_t^c = \{s_{t-W_c}^c, ..., s_{t-1}^c, s_t^c\}$, where W_c is the window size, and a time window of GPS information $\mathbf{s}_t^g = \{s_{t-W_g}^g, ..., s_{t-1}^g, s_t^g\}$. Assuming a window facing orientation o degree(s), the digital compass outputs azimuth relative to the north, which we convert to relative angles toward o degree(s), and use the time-series of these relative angles as input. Additionally, we find the N_G satellites in orientation o degree(s) and use their elevation, relative angle (between satellite azimuth and orientation o), and signal strength time-series data as inputs.

As shown in Fig. 2.7, the walking direction predictor comprises a bi-directional LSTM layer with 32 nodes and the tanh activation function for processing orientation data, and another bi-directional LSTM layer with 32 nodes for processing GPS data. The network is trained to minimize categorical cross-entropy loss using the Adam optimizer.

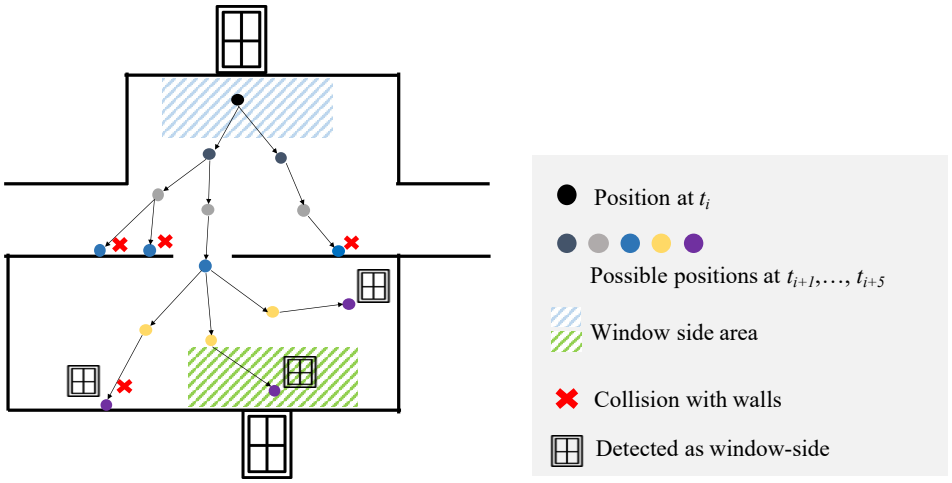


Figure 2.8: Tracking a smartphone user with a particle filter from time t_i, where the user is detected in the blue window-side area. Multiple paths are generated by varying displacement and orientation. Paths colliding with walls are deleted. At time t_{i+5}, the user is detected in the green window-side area, and the position within this area at t_{i+5} is estimated as the true position.

2.3.5 Trajectory Estimation Module

The proposed method uses a particle filter [71], which is typically employed to estimate the internal states of a non-linear system and is also used in map matching-based human tracking [72, 73], for user tracking. This allows us to easily combine outputs from the GPS landmark module and Neural PDR module. In the particle filter, a particle at time t represents one of the user's position candidates at that time.

As shown in Fig. 2.2, the first window-side position generator creates user position candidates (particles) in window-side areas facing direction o_i at time t_i, when the user is first detected entering a window-side area. If there are multiple window-side areas facing o_i degree(s), the user's position at time t_i remains ambiguous. The trajectory before and after time t_i is then estimated using the particle filter with map matching. By using map matching, candidate particles that collide with walls are discarded, reducing the ambiguity of the user's position, as shown in Fig. 2.2.

First Window-side Position Generator

As mentioned above, the window-side detector estimates whether the user is in a window-side area at each time step. When N_c consecutive detections occur after the user starts walking, the user is determined to have entered the window-side area. Using the orientation o predicted by the window orientation classifier at the first detection time t_i, we generate candidates for the user's absolute coordinates at time t_i, as shown in Fig. 2.2. Specifically, N_i particles are randomly generated in window-side areas facing o degree(s). An initial rough walking direction is assigned to each particle based on the estimated probabilities of the three corresponding walking direction classes, making a weighted random selection favoring classes with higher probabilities: i) For "entering", the initial walking direction is randomly selected from $o-45$ to $o+45$ degree(s). ii) For "parallel", the walking direction is randomly chosen from $o+45$ to $o+135$ or $o-135$ to $o-45$ degree(s). iii) For "others", the particle is discarded, as it implies the user is not entering a window-side area.

To reduce computational costs, initial particles with incorrect positions or walking directions are discarded. If t_i is the first window-side detection time and t_{i+N_c} is the last time of the first consecutive window-side detection, we calculate the trajectory between these times for each particle using Neural PDR module outputs. Particles where all predicted positions within the trajectory remain in a window-side area are kept for forward and backward tracking. Otherwise, they are discarded. Additionally, we designate the first window-side detection time as t_i and the final time of the initial consecutive window-side detection period as t_{i+n}. To minimize computational expense, we determine the trajectory between times t_i and t_{i+n} using the outputs from the Neural PDR module, which helps eliminate numerous incorrect initial positions generated at time t_i. For instance, if all predicted positions between t_i and t_{i+n} are within the window-side area, these trajectories will proceed to the next step. Conversely, if a predicted position falls outside the window-side area, the trajectory will be discarded, and its initially generated position will be considered unreliable.

Forward-tracking

The user's trajectory after time t_i is estimated by the particle filter using outputs from the Neural PDR module, as shown in Fig. 2.8. In the particle filter, a particle p_i at time t represents a candidate for the user's position, containing indoor coordinates $\mathbf{x}_{p_i}^t \in \mathbb{R}^2$, walking orientation $o_{p_i}^t$, and weights $w_{p_i}^t$. The GPS landmark module's output corrects accumulated trajectory errors. For example, if the GPS landmark module estimates the user is in window-side areas facing a certain orientation, low weights are assigned to particles outside these areas. Particles colliding with walls have their weights set to 0. In the re-sampling process, particles are re-sampled according to their weights.

The tracking algorithm based on the particle filter operates in three steps: sampling, weight calculation, and re-sampling. In the sampling process, new particles at time t are generated from each particle at the previous time step based on a motion model. The generated particles are prior estimates of the user's location at time t. In the weight calculation process, particle weights are computed based on the GPS landmark module's outputs and the floor plan. In the re-sampling process, particles are re-sampled according to their weights.

- **[Sampling]:** New particles at time t are sampled from candidate areas like a circular sector, determined by coordinates at time $t-1$ and predicted displacement and direction change at time t by the Neural PDR module. The motion model for sampling $\mathbf{x}_{p_i}^t$ is defined as follows:

$$\mathbf{x}_{p_i}^t = \mathbf{x}_{p_j}^{t-1} + \mathcal{N}(\hat{d}_t, \sigma_d)\vec{\mathbf{v}}_{p_i}^t,$$

  where $\mathcal{N}(\mu, \sigma)$ is the normal distribution with mean μ and standard deviation σ, $\hat{d}t$ is the displacement estimate at time t by the walking distance predictor, σ_d is a hyper-parameter, and p_j represents the j-th particle at time $t-1$. Assume N_p particles at time t are generated from p_j at time $t-1$, and p_i is one of them. $\vec{\mathbf{v}}_{p_i}^t$ is a unit vector in $\mathbb{R}^2$ indicating the walking direction of p_i at time t, and the direction $\angle\vec{\mathbf{v}}_{p_i}^t$ is determined by:

$$\angle\vec{\mathbf{v}}_{p_i}^t = o_{p_j}^{t-1} + \mathcal{N}(\hat{\theta}_t, \sigma_o)$$

  where $\hat{\theta}t$ is the predicted change in walking direction at time t by the walking direction change predictor, and σ_o is a hyper-parameter. Then, $o_{p_i}^t$ is updated as

follows:

$$o_{p_i}^t = \angle \vec{\mathbf{v}}_{p_i}^t$$

- **[Weight calculation]:** For each time step t, initial particle weights are calculated so that a particle located at the center of the candidate area at time t has high weight. The weight for p_i is calculated as $w_{p_i}^t = w_d \cdot w_o$, where:

$$w_d = \begin{cases} \text{CDF}(d_{p_i}, \mathcal{N}(\hat{d}_t, \sigma_d)), & (\text{if } d_{p_i} \leqq \hat{d}_t) \\ 1 - \text{CDF}(d_{p_i}, \mathcal{N}(\hat{d}_t, \sigma_d)), & (\text{otherwise}) \end{cases} \quad \text{and}$$

$$w_o = \begin{cases} \text{CDF}(\theta_{p_i}, \mathcal{N}(\hat{\theta}_t, \sigma_o)), & (\text{if } \theta_{p_i} \leqq \hat{\theta}_t) \\ 1 - \text{CDF}(\theta_{p_i}, \mathcal{N}(\hat{\theta}_t, \sigma_o)). & (\text{otherwise}) \end{cases}$$

Here, d_{p_i} is a sampled displacement of p_i, θ_{p_i} is a sampled direction change of p_i, and CDF() represents the cumulative distribution function. As shown, when the value of d_{p_i} or θ_{p_i} is closer to $\hat{d}_t$ or $\hat{\theta}_t$, respectively, the weight of p_i increases. The particle weights are then updated using the floor plan and GPS landmark module outputs. Particles colliding with an inner or outer wall have their weights set to 0. The weights are further updated when the window-side detector output is True, using the following equation:

$$w_{p_i}^t = \begin{cases} \alpha \cdot w_d \cdot w_o + P_t^w + P_t^{o_c}, & \text{(if the particle in a window-side area facing orientation } o_c \text{)} \\ \alpha \cdot w_d \cdot w_o. & (\text{otherwise}) \end{cases}$$

where α is a hyper-parameter, P_t^w is the probability of the "True" class by the window-side detector at time t, and $P_t^{o_c}$ is the probability for orientation candidate o_c by the window orientation classifier. This increases the weight of particles in window-side areas facing the estimated orientation o by the window orientation classifier.

- **[Resampling]:** From the weighted samples, we re-sample N_r particles according to their weights. The probability of re-sampling a particle is proportional to its weight. The posterior estimate of the user's position at time t is the weighted average position of the N_r particles.

Table 2.2: Experimental environments used in this study. Float shows float glass. ALC stands for autoclaved lightweight aerated concrete.

Env.	Floor	Size	Type	Window/wall
A	2F	22.9m x 43.2m	Office	Float/ALC
B	6F	21.7m x 41.9m	Office	Float/ALC
C	3F	22.2m x 26.2m	Multipurpose	Float/ALC
D	3F	35.4m x 49.2m	Library	Float/Concrete

Back-tracking

We also back-track the user's trajectory before time t_i using the particle filter. The procedure is similar to forward-tracking but reverses the movement vectors to back-track the user's trajectory from time t_i to time 0.

2.4 Evaluation

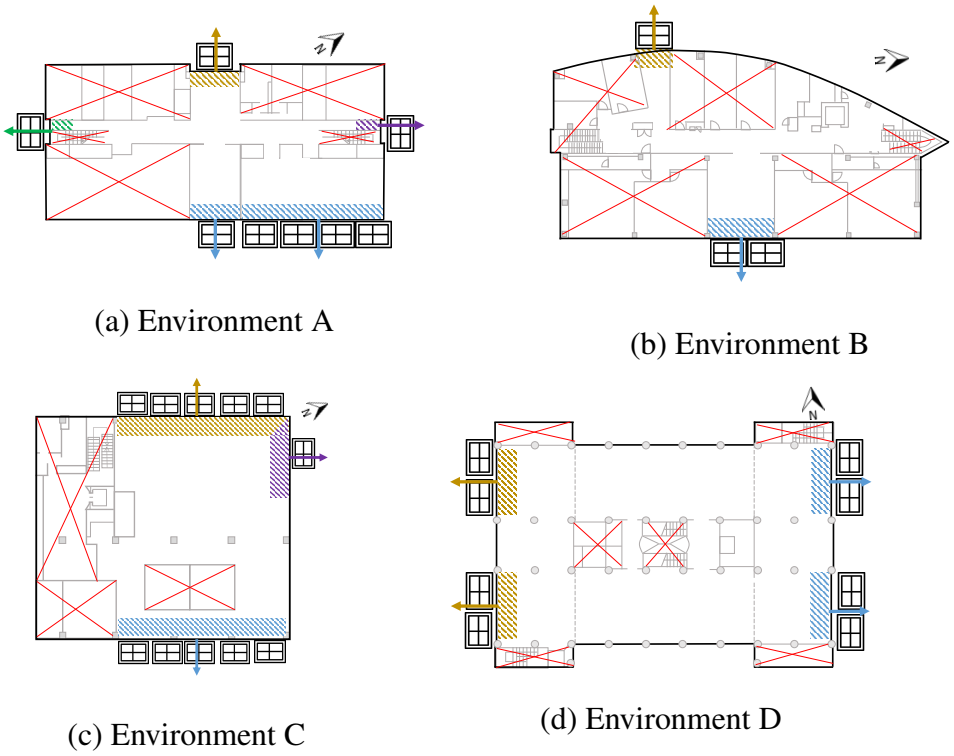
2.4.1 Experimental Settings

As the proposed method is designed to be independent of the PDR models, we evaluated our approach using both user-dependent and user-independent PDR models.

Environments

We collected sensor data from 4 different experimental environments for evaluation. In this dataset, a participant collected data at our four primary environments (buildings) to evaluate the user-dependent PDR model. Details of these four environments are shown in Fig. 2.9 and Table 2.2.

Our dataset comprises GPS, orientation, and IMU data collected from four different environments in distinct buildings, as shown in Fig. 2.9 and Table 2.2. Environment A is an office space on the 2nd floor, featuring four window-side areas located near the north and south staircases, the west open space, and the east conference rooms. There is



(a) Environment A

(b) Environment B

(c) Environment C

(d) Environment D

Figure 2.9: Floor plans of the experimental environments show window locations (window icons) and window-side areas (colored rectangles with diagonal stripes). Different colors indicate different orientations, with arrows showing window normals. Red crosses mark private areas we could not access.

a short neighboring building to the west and a tall building to the south of Environment A. Environment B, also an office floor, is situated on the 6th floor with two window-side areas in the west and east rooms. The curved walls in Environment B are approximated as straight lines in our floor map. Being on the 6th floor, there are no neighboring buildings to the west or east.

Environment C is a multipurpose room on the 3rd floor, with metallic barriers on the exterior walls that interfere with GPS signals. Tall buildings are located on both sides (left and right in the picture). Environment D is a library on the 3rd floor, with very thick walls and windows on all sides. Many desks are situated near the south and north window-side areas. There is a tall building adjacent to the right side of Environment D.

Data Collection

We utilized a Google Pixel 4 running Android 10 to collect IMU, orientation, and GPS data using a custom-developed application, and simultaneously used a Google Pixel 3a running Android 10 to capture ground truth data with ARCore [74]. ARCore was employed to obtain the user's coordinates at 1 Hz, which were then used to compute $d_{(t)}$ and $\theta_{(t)}$, representing the ground truth for the Neural PDR module. The ground truth for the GPS landmark module was derived from ARCore-determined positions and the known locations of window-side areas on the floor map. All sensor data were collected at the smartphone's default sampling rates: 100 Hz for the accelerometer, 50 Hz for the gyroscope and digital compass, and 1 Hz for the GPS receiver. The accelerometer data were subsequently downsampled to 50 Hz. The GPS data included the elevation angle, azimuth angle, signal strength, and PRN (identifier) for each visible satellite.

During data collection, participants holding a smartphone walked randomly within the environment, ensuring they passed a window-side area at least once. Data were gathered in 4 different environments for the first dataset, with multiple trajectories collected in each environment. The default device was the Google Pixel.

Evaluation Methodology

To evaluate the window-side detector, which functions as a binary classifier, we used classification accuracy (the ratio of correctly classified instances). For the window orientation classifier and the walking direction predictor, we also used classification accuracy as the evaluation metric. For the trajectory estimation module, we calculated the mean absolute error (MAE) between the ground truth coordinates and the weighted average of posterior estimates of particles in the surviving trajectories at each time step. Note that if all particles collide with a wall at time t, the method cannot produce results and will ignore walls only at that specific time.

In evaluating the trajectory estimation module, we compared the proposed method against the following methods:

- **Neural PDR**: This is considered a state-of-the-art neural network-based method. It uses only map matching based on the particle filter and the PDR results from the Neural PDR module, with initial position and direction provided.

- **W/o WOC**: This variant of the proposed method does not use the window orientation classifier. Thus, the first window-side position generator creates particles in all window-side areas. In the re-sampling process, only the results from the window-side detector are used to calculate particle weights. The absence of the window orientation classifier limits the amount of information about GPS landmarks, affecting its performance. Despite this, the method can still detect when the user is near a window-side area but cannot identify the window's orientation. As a result, $P_t^{o_c} = 0$ when calculating particle weights. The number of initial particles generated remains the same as in the proposed method.

- **W/o WDP**: This variant does not use the walking direction predictor. Consequently, the method cannot determine the walking direction at the first window-side position, leading to random sampling of walking directions for particles (0-360 degrees). This method helps assess the impact of initialization. The number of first window-side particles generated is the same as in the proposed method.

To further examine the effectiveness of the modules in the proposed method, we also applied the following simpler methods for window-side detection and window orientation classification:

- **Thresholding**: This threshold-based window-side detector uses a decision stump, a one-level decision tree, to determine if a user is near a window-side area based on overall signal strength. The input feature is the average signal strength from all visible satellites within a time window starting at time t. The output is a binary value, indicating whether the user is in a window-side area at time t.

- **Sudden-change**: This method determines the window orientation based on sudden increases in GPS satellite signal strength. When a user is detected in a window-side area at time t, we analyze the signal strength time-series of GPS satellites within the current and previous time windows (starting at t and $t-W$, respectively, each with W_g samples). A sudden increase in a satellite signal strength at time t leads to identifying the wall (window orientation class) whose normal is closest to the satellite's azimuth. Majority voting selects the window orientation class. If no sudden increase is detected in the current window, previous windows

are analyzed (e.g., t vs. $t - W$, t vs. $t - 2W$). A sudden change is assumed when the average signal strength at t is c_s times higher than at $t - W$.

- **Angle-regression**: Unlike our method's binary window orientation classifier, this approach directly outputs the window's orientation. The inputs are signal strength, azimuth, and elevation angle time-series from each of the top-N_G satellites. The network structure is identical except for the output layer, which uses a linear activation function to produce a numerical value representing the orientation. The network is trained to minimize the mean absolute error between the estimated orientation and the ground truth using the Adam optimizer. The wall whose normal is closest to the estimated orientation is then selected.

The user-dependent models were evaluated using a "leave-one-environment-out" cross-validation on the first dataset, where one environment served as the testing environment and the others were used to train the window-side detector, window orientation classifier, and PDR module. Given that the user-dependent PDR model is expected to predict walking distance and direction change precisely, we evaluated the main components of the proposed method, namely the window-side detector and window orientation classifier, in detail when the PDR module achieved optimal accuracy. Note that our method is independent of the PDR module and can integrate any state-of-the-art method that may be proposed in the future. This main experiment used only data from Google Pixel devices.

For evaluating user-independent models with the second dataset, we divided the environments into five training environments (Environments A, B, E, H, and K) and sixteen testing environments. When a participant was involved in a testing environment, their training data was excluded from the training environments. We trained the window-side detector, window orientation classifier, and walking direction predictor for both smartphones held in hand and carried in a pocket. We also explored the impact of various environments and user-independent PDR models, which are significantly affected by differences in walking patterns among participants, on the performance of the proposed method. Table 2.3 presents the experimental parameters used in this study.

Table 2.3: Experimental parameters used in this experiment

Parameter	Value	Description
W_i	50	window size in Neural PDR module
N_G	6	top-N_G satellites are used as inputs
W_g	5	window size in GPS landmark module
W_c	5	window size in walking direction predictor
σ_d	0.05	standard deviation of normal distribution when sampling displacement
σ_o	20°	standard deviation of normal distribution when sampling direction change
N_c	5	N_c consecutive window-side detections are used in the first window-side position generator
N_i	5000	N_i particles generated by the first window-side position generator
N_p	50	N_p particles generated from each particle from each time step in sampling
N_r	20	re-sampled N_r particles after weight calculation

2.4.2 Results

Performance of the Window-side Detector

Table 2.4 presents the classification accuracy of the window-side detector across different environments. Note that we calculated accuracy only for locations where GPS signals were available. When a smartphone is far from window-side areas and does not receive any GPS signals, it is straightforward to assume the smartphone is not in a window-side area. As shown in the table, the window-side detector achieved relatively higher accuracy with $d_w = 3$, which was the setting used in our method.

Moreover, as indicated in Table 2.4, the classification accuracy in Environment B is lower than in the other environments. This could be attributed to the smaller window-side areas in that environment, as shown in Fig. 2.9. The shorter duration of stays within these window-side areas means that detection errors upon entering or leaving the areas had a significant impact on the overall classification accuracy.

Table 2.4: Classification accuracy of window-side detection for the thresholding method and the proposed method of each environment ($d_w = 3$)

Methods	Env. A	Env. B	Env. C	Env. D
Thresholding	59.7%	28.5%	49.9%	61.2%
Proposed method	81.5%	71.4%	74.9%	76.0%

Table 2.5: Sensitivity and specificity of the window-side detector for $d_w = 3$ (window-side corresponds to the positive class)

	Env. A	Env. B	Env. C	Env. D
sensitivity	69.4%	92.8%	70.5%	84.3%
specificity	86.2%	64.4%	80.8%	66.0%

Additionally, Table 2.5 lists the sensitivity and specificity of the window-side detector for $d_w = 3$. As observed, a significant number of window-side instances in Environments A and C were incorrectly classified as non-window-side cases. For example, in the lower right corner in Env A, even when a smartphone is near a window, the window-side classifier often outputs low window-side probabilities. This could be due to the presence of pillars in window-side areas.

Similarly, as depicted in Table 2.5, many non-window-side instances in Environments B and D were incorrectly classified as window-side cases. This misclassification is likely caused by a short delay in the reception of GPS signals by the smartphones, significantly impacting classification accuracy due to the small size of window-side areas in these environments.

Table 2.4 also presents the classification accuracy of the Thresholding method. The results indicate that this simple method, which relies on the average signal strength from all visible satellites within a time window, has comparatively low accuracy. The accuracy in Environment B is notably poorer compared to other environments. This can be attributed to the overall weaker signal strengths in Environment B. Since the Thresholding method determines a threshold for overall signal strength based on training data from other environments, it is significantly affected by environmental differences.

Table 2.6: classification accuracy of Sudden-Change method and the window orientation classifier for each environment

	Env. A	Env. B	Env. C	Env. D
Sudden-change ($c_s = 1.2$)	28.6%	35.9%	38.8%	76.4%
proposed method ($d_w = 3$)	87.3%	100%	69.9%	92.8%

Table 2.7: Classification accuracy of window orientation classifier and macro average of accuracy ($d_w = 3$)

	Env. A	Env. B	Env. C	Env. D
1st wall	36.5%	100%	42.7%	96.1%
2nd wall	98.4%	100%	91.9%	97.7%
3rd wall	16.1%	n/a	100%	n/a
4th wall	99.1%	n/a	n/a	n/a
macro average	62.5%	100%	78.2%	96.9%

Performance of the Window Orientation Classifier

Table 2.6 presents the classification accuracy of the window orientation classifier across different environments. As shown in the table, the window orientation classifier achieved very high accuracy in Environments A, B, and D. The lower accuracy in Environment C, compared to the other environments, can be attributed to difficulties in orientation classification at the top-right corner, as illustrated in Fig. 2.9(c). At a corner, a smartphone receives signals from two directions, leading to high probabilities for classes in both directions. However, this incorrect classification at the corners does not significantly impact the overall PDR performance.

In addition, Table 2.7 provide detailed classification results for the proposed window orientation classifier. The accuracy for the 1st and 3rd walls in Environment A was low, resulting in a lower macro-averaged accuracy in that environment. However, since the window-side areas corresponding to these walls (windows) are small, only a few data instances were collected.

Table 2.6 also presents the accuracy of the window orientation classifier based on sudden signal strength changes when the c_s threshold is 1.2. As indicated by the results,

Table 2.8: Classification accuracy and mean absolute error (MAE) of orientation classification for the Angle-regression method

	Env. A	Env. B	Env. C	Env. D
classification accuracy	67.30%	97.50%	54.40%	89.10%
MAE in degree	41.56	20.37	70.64	21.98

the orientation classification accuracy using the sudden change method was significantly lower than that achieved by the proposed method. This discrepancy is due to signals received from satellites that are not oriented perpendicular to the window near the user. We observed that while the signals a smartphone receives from such satellites are weak, their strengths can sometimes suddenly increase even when the satellites are not aligned with the window normal. However, classification accuracy was notably higher in Environment D than in other environments. This could be because the walls in this building were made of thick concrete, which often blocked signals from satellites not oriented towards the window normal when the smartphone was close to a window.

Table 2.8 also shows the classification accuracy and mean absolute error (MAE) based on the angle regression method. Similar to our proposed method, in Environments A and C, the classification accuracy is lower and the MAE is higher compared to other environments. However, the classification accuracy using angle regression is slightly lower than that achieved with our proposed approach. This may be due to the limited variation in ground truth labels, specifically the angle relative to the north. In a training environment with four walls, only four distinct angles are possible for the ground truth labels, which is insufficient for a robust regression task.

Performance of the Walking Direction Predictor

Table 2.9 presents the classification accuracy of the walking direction predictor. While the proposed method (GPS+magnetic) does not achieve perfect accuracy, it outperforms methods that use a single sensor. Additionally, because the accuracy of these methods is higher than the random guess ratio (33%), they are useful for generating the walking directions of particles produced by the first window-side position generator. The classification accuracy for the 'only magnetic' method is notably low in Environment

Table 2.9: Classification accuracy of the walking direction predictor by three methods

	Env. A	Env. B	Env. C	Env. D
GPS+magnetic	55.9%	51.4%	48.0%	66.2%
only magnetic	50.2%	55.0%	36.3%	60.4%
only GPS	53.9%	48.6%	48.0%	48.8%

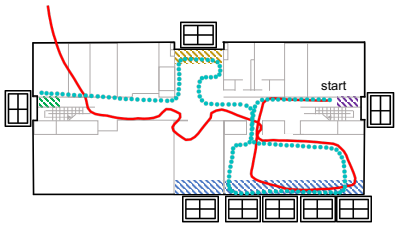
Table 2.10: Positioning errors for the four methods across different environments.

Positioning error (m)	Env. A	Env. B	Env. C	Env. D
Neural PDR	2.595	1.791	2.180	3.978
W/o WOC	6.380	2.762	7.321	12.741
W/o WDP	2.995	2.558	3.247	5.528
Proposed	1.819	1.316	1.267	2.672

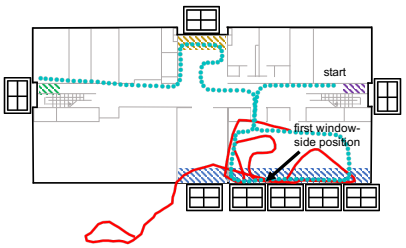
C, likely due to magnetic fluctuations in that environment. Similarly, the 'only GPS' method shows poor accuracy in Environment D, attributed to the thick walls that block GPS signals. The walls in Environment D hinder the collection of long GPS time-series, which is crucial for accurate classification. However, by combining GPS and magnetic readings, the proposed method (GPS+magnetic) achieves more stable performance across the experimental environments.

Performance of Trajectory Prediction

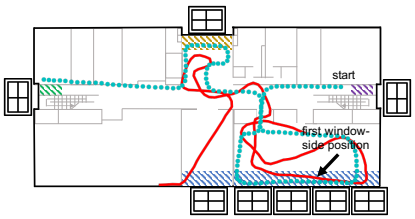
Table 2.10 presents the positioning errors of the PDR methods in each environment. As shown, the proposed method significantly outperformed the state-of-the-art Neural PDR, despite not utilizing information about the initial position and initial walking direction. When the duration of a trajectory is long, accumulated errors in neural PDR can cause all particles to collide with inner and outer walls. In such cases, because the method ignored the wall only at that time (i.e., when all particles were eliminated by map matching), the positioning errors of the Neural PDR method increased. The positioning errors in Environment D were larger than in other environments due to its large size and sparse window-side areas, leading to significant accumulated errors. The



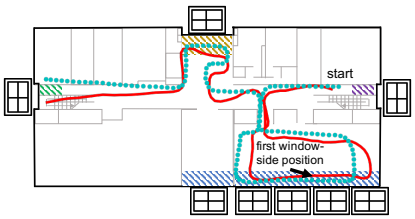
(a) Trajectory estimated by Neural PDR



(b) Trajectory estimated by W/o WOC



(c) Trajectory estimated by W/o WDP



(d) Trajectory estimated by Proposed

Figure 2.10: Examples of estimated trajectories in Environment A: The red line indicates the predicted path using our methods, while the blue dotted line shows the ground truth. The initial window-side position is identified after the first N_c consecutive 'window-side' detections.

W/o WOC (window orientation classifier) variant sometimes failed to generate accurate particles when subjects were first detected in window-side areas. Because W/o WOC generates particles at all window-side areas, the probability of generating particles at actual positions is lower than in the proposed method, resulting in poorer performance for W/o WOC. Similarly, the performance of W/o WDP (walking direction predictor) was inferior to the proposed method. Although the walking direction predictor's accuracy was not very high, the positioning error of the proposed method was much smaller than that of W/o WDP, which randomly determines the walking direction when a user first enters a window-side area. The impact of the walking direction predictor was more significant than expected.

Fig. 2.10 shows examples of trajectories estimated by the methods compared with ground truth trajectories. As mentioned, since all particles are sometimes eliminated by

map matching, the methods occasionally generate trajectories that pass through walls. However, the proposed method, using GPS information, can correct the accumulated errors of the Neural PDR module. The window orientation classifier helps narrow down the candidates for window-side positions. As shown in Fig. 2.10(b), W/o WOC often failed to generate accurate first window-side positions. W/o WDP also sometimes failed to generate accurate walking directions for first window-side positions. In this example, W/o WDP could not find a path that starts with the correct window-side area (purple area). In contrast, the proposed method generated many particles with accurate walking directions at the first window-side position, enabling it to find a path starting from the correct window-side area.

2.4.3 Discussion

Failed Trajectories

We analyzed the reasons for failed trajectory predictions by the proposed method using the first dataset, considering a trajectory with an MAE greater than 3 meters as a failure. The failed trajectories were classified into three types.

1. **False positive predictions by the window-side detector near the window-side areas.** A false positive occurs when the window-side detector incorrectly identifies a user as being in a window-side area when they are not. If a prediction at time t is a false positive but the user was in the window-side area at $t-1$, the trajectory estimation module is likely to assume the user is still in the window-side area at time t, leading to accumulated errors. This issue arises near the boundary between window-side and non-window-side areas. In this case, the ratio is 6%.

2. **All possible particles traversing the nearby wall.** This occurs when the user is close to a wall at time $t-1$ and turns around it at time t, causing all predicted particles to collide with the wall at time t. As a result, all particles are assigned a null weight, and the trajectory estimation module randomly selects particles. This problem is caused by a large error in the predicted change of walking direction and a small σ_o (a particle filter hyperparameter related to walking direction variance). The proportion of this kind of failure is 8%.

3. **Symmetric building structure & lack of information on walls or obstacles.** In Environment D, the floor plan is symmetric but lacks sufficient wall information. This often results in the map matching-based trajectory estimation module predicting trajectories that resemble the ground truth in shape but are translated in 2D geometry. This kind of failed trajectories is only 4% to the total number of trajectories.

We found that the second type of failed trajectory is the most common and presents a significant challenge. One simple solution is to increase the value of σ_o, but this requires generating more particles at each time step, increasing computational cost. Another approach is to generate additional random particles that do not collide with the wall at time t. Future studies will focus on addressing this problem. Additionally, we plan to improve the accuracy of the window-side detector and add more obstacle information to tackle the first and third types of failed trajectories.

Detecting Corners in a Floor

In environments with windows on both sides of a corner area, a smartphone can receive signals from each window direction. Detecting when a user is at a corner can significantly enhance position prediction accuracy. However, such environments are limited. We plan to implement a corner detection module that can be trained with limited data.

Limitations

The proposed GPS landmarks rely on GPS satellites and thus cannot be used in underground environments. Additionally, they cannot distinguish between multiple floors in a multi-floor building. However, as mentioned in the Introduction, smartphones are equipped with various sensors, allowing GPS landmarks to be used alongside other types of landmarks that can distinguish between floors. The primary advantage of GPS landmarks is their ease of installation, as shown in Table 2.1. This study is the first to demonstrate the feasibility of using GPS landmarks.

2.5 Conclusion

This chapter introduced a novel type of indoor landmark based on GPS signals to correct accumulated errors in IMU-based PDR. These GPS landmarks can be utilized without the need for additional signalling infrastructure. Furthermore, we developed a GPS landmark module designed to detect these landmarks using sensor data from environments other than the target environment. We tested the PDR system, integrated with the GPS landmark module, on sensor data collected from real-world environments. The experiments demonstrated that the GPS landmark module provides indoor contextual information that is effective in correcting PDR errors. For future work, we plan to explore additional applications of indoor contextual information obtained from GPS satellites.

Chapter 3

Predicting Signal Reception Information from GNSS Satellites in Indoor Environments without Site Survey

3.1 Introduction

3.1.1 Background

With the widespread adoption of smartphones and the increasing demand for localization, developing effective positioning systems for context-aware applications has become essential in the field of mobile and ubiquitous computing. Positioning techniques are commonly categorized into two primary types: outdoor and indoor positioning systems, each distinguished by their methodologies and specific application areas. Outdoor positioning primarily relies on global navigation satellite systems (GNSS), which offer the significant advantage of high accuracy in open-sky environments by leveraging existing global infrastructure [2]. GNSS includes systems such as GPS, GLONASS, Galileo, and BeiDou. The Global Positioning System (GPS) provides precise location information worldwide through a network of orbiting satellites [9, 10]. However, GPS depends on a clear line-of-sight to satellites, significantly reducing its effectiveness in-

doors where signals are often obstructed [75]. In contrast, indoor positioning employs various methods, with wireless and IMU-based technologies being the most prevalent. Wi-Fi-based fingerprinting, a type of wireless technology, is widely used due to its ease of implementation and compatibility with existing Wi-Fi infrastructures. However, it has drawbacks, such as the need for comprehensive site surveys and the requirement to install additional signaling devices, like ultra-wideband (UWB) or Bluetooth Low Energy (BLE) beacons [76]. Pedestrian dead reckoning (PDR) uses IMUs to estimate step count or displacement and orientation changes, but it faces significant challenges such as the accumulation of errors over time, leading to decreased accuracy [24, 17]. According to survey papers by [77] and [19], conventional Pedestrian Dead Reckoning (PDR) consistently exhibits notable direction errors, resulting in significant cumulative errors.

Recognizing the limitations of existing indoor localization methods highlights the potential of indoor GPS reception. Although GPS systems often fail to provide precise locations indoors, they offer the advantage of low installation costs due to their reliance on existing satellite infrastructure. For example, according to observation, a smartphone close to a window facing a particular direction receives stronger signals from satellites in that direction [46]. This not only helps in deducing whether the smartphone is near a window but also assists in determining which side of the target environment the window is located, offering hints about the smartphone's indoor position. Conversely, even if the device is on the side of a building facing the satellites, the presence of walls and other structural barriers can significantly weaken the GPS signals.

The variation in GPS signal strength, shaped by the indoor settings, presents opportunities to use GPS data in indoor positioning systems, though in a less direct way compared to standard outdoor localization techniques. Signal reception maps from satellites, akin to heatmaps, can be highly beneficial for recognizing indoor positional contexts, much like Wi-Fi fingerprinting maps [78, 79]. For instance, by comparing multiple satellites' signal reception maps with the actual signal reception observed by a smartphone, it becomes feasible to implement opportunistic GNSS indoor positioning that provides a rough indoor estimate when GNSS signals are available. This rough position estimate can complement and mitigate the drawbacks of various existing indoor positioning methods. For example, PDR often requires an initial position or relies on

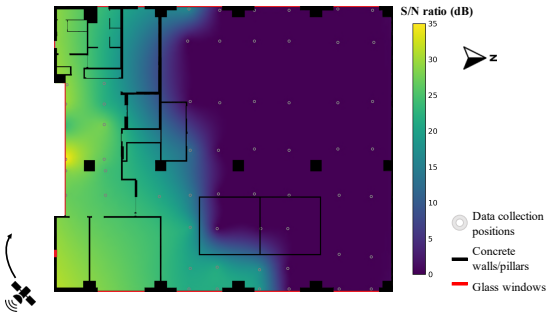


Figure 3.1: Heat map in a floor.

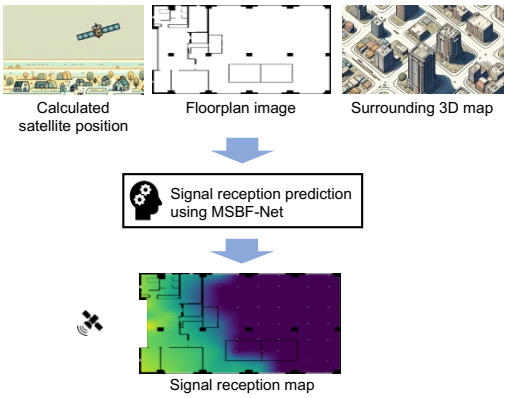


Figure 3.2: Task of this study.

landmarks to manage error accumulation. Integrating GPS signal strength information can provide useful reference points. Furthermore, combining this with Wi-Fi positioning can enhance accuracy, particularly near windows or places with stronger satellite signals because in general, Wi-Fi fingerprinting methods that utilize machine learning or nearest neighbor search algorithms often struggle with positioning accuracy when a target, like a smartphone, is located at the periphery of a given environment, such as near a window. This inaccuracy arises because the fingerprint data is typically collected from within the central areas of the environment, making these techniques poor at extrapolating positions at the edges. Consequently, these methods tend to estimate the position closer to the interior of the environment rather than at its actual edge where the device is located. By merging these technologies and data, we can create more dependable and effective indoor positioning systems that use the strengths of each method to overcome their weaknesses.

3.1.2 Research Objective

Our research objective is to develop a method for constructing signal reception maps for each satellite within specific indoor environments without the need for extensive manual data collection. Traditional manual data collection is costly and time-consuming, and these maps are subject to temporal changes due to the constant movement of satellites [10]. To address these challenges, we propose a strategy to create signal reception maps

at a specific time t **without** the need to collect GPS signals at every position within the target environment. This study utilizes a machine learning-based approach that leverages easily accessible data sources to predict the signal reception map, as illustrated in Fig. 3.2, thereby significantly reducing the associated costs. The key elements of our data sources include:

(i) **Satellite positions**: We utilize publicly available orbit information [80] to calculate the precise locations of satellites.

(ii) **Indoor floorplan information**: This includes images that provide details about the layout of windows and walls, as shown in the upper middle part of Fig. 3.2, typically supplied by the administration offices of the target buildings.

(iii) **3D map of target and surrounding buildings**: These maps, sourced from online mapping services and databases such as Google Earth [81], offer a broader context of the target environment. They are crucial for determining the line-of-sight (LOS) from satellites and understanding how surrounding buildings might reflect or obstruct signal pathways.

We transform these diverse sets of information into 2D feature maps, which are then used as inputs for our proposed CNN-based neural network, named MSBF-Net. This network is designed to estimate signal reception information, as shown in Fig. 3.2, and is trained on data from various training environments, thereby eliminating the need for extensive data collection within the target environment itself.

Moreover, we designed MSBF-Net to estimate signal reception information at specific positions within the target environment rather than generating an entire signal reception map, as shown in Fig. 3.1. This approach is more feasible because generating a complete map for each satellite would limit the number of training datasets to the number of training environments multiplied by the number of visible satellites. Additionally, collecting ground-truth signal strength at multiple positions simultaneously within the same environment is impractical, requiring numerous signal receivers. By focusing on individual positions, we can significantly increase the amount of training data, which expands to the product of the number of environments, the number of observation positions within an environment, and the number of visible satellites. Eliminating the

need for in-situ data collection greatly reduces the time, cost, and logistical complexities typically associated with creating accurate indoor reception maps. We focus on two methodologies: estimating actual signal strength at a given point and binary classification to determine whether the signal is strong or otherwise, with primary emphasis on the latter.

3.1.3 Contributions

Our study offers three significant contributions:

- This research aims to predict GPS signal reception information using easily accessible data, eliminating the need for manual data collection within the target environment.
- To enable automatic estimation of indoor signal reception, we introduce a prediction method that integrates information from GPS satellite positions, indoor floorplans, and 3D maps of the target and surrounding buildings. Our technical contributions include: i) developing a pipeline that efficiently incorporates accessible data, such as satellite positions, for predicting indoor signal reception at specific locations, and ii) designing a model capable of effectively fusing feature maps of varying scales derived from the accessible information.
- Through extensive experiments conducted in real-world environments, we demonstrated the feasibility and effectiveness of our method across different buildings.

3.2 Related Work

3.2.1 Investigation of Indoor GPS Reception

GPS satellites emit specific signals essential for precise location and timing determination [9, 11, 12]. These signals include a navigation message that provides the satellite's location. In addition to the navigation message, GPS receivers measure the signal-to-noise ratio (SNR) of each satellite to determine signal feasibility. SNR, defined as the ratio between the measured signal intensity and noise intensity, is crucial for accurate

GPS measurements [82, 83, 10, 84]. The calculation of SNR in GPS receivers involves analyzing prompt signal outputs from the code-tracking process, aligning SNR values with carrier phase variations caused by multi-path effects. This alignment is significant for identifying and correcting errors in GPS measurements. This study aims to predict signal strength information in indoor environments.

Indoor GPS signal reception differs considerably from outdoor environments due to various factors. Outdoors, GPS signals are received with minimal obstructions under open sky conditions, enabling accurate positioning [85]. Indoors, GPS signals are significantly attenuated by structural elements like walls and roofs, leading to weaker signals and less precise positioning [86]. Studies have highlighted the substantial impact of construction materials on GPS signal quality and positioning accuracy, noting that materials like aluminum and concrete cause considerable multi-path interference, increasing GPS measurement errors [87, 88]. Detailed material information for each building is often inaccessible, known only by construction contractors. Hence, this study predicts signal reception using only accessible floor map information, assuming most modern buildings are constructed from concrete to demonstrate feasibility.

The structure of buildings and their surrounding environments also affect indoor positioning accuracy [45, 89]. GPS positioning is generally more reliable in single-story houses or upper floors of buildings, though signal strength can still be compromised by metal walls [90]. Urban settings further complicate GPS signal reception due to reflections from tall buildings, causing multi-path errors where signals reach GPS receivers through multiple paths, altering phase delay and amplitude. Consequently, GPS signal strength fluctuates [91, 92].

To understand GPS signal reception, researchers use simulation models. Generally, LOS signals in GPS are stronger than multi-path signals, which suffer from reflection or diffraction losses. However, this is not always the case. In certain situations, if an LOS signal is significantly weakened or blocked, multi-path signals may be stronger or the only signals available [93]. Wen et al. [94] found that excluding non-LOS (NLOS) satellites by 3D-radar simulation of GPS signals dramatically reduces the number of available LOS satellites. Another study [95] suggests that while ray tracing simulations are effective outdoors, simulating NLOS scenarios indoors remains challenging due to certain propagation paths or diffraction models that are not adequately captured, leading

to variability in received power. LOS signals appear stable while NLOS signals are very unstable.

Few research projects have attempted to predict signal strengths or signal propagation paths via ray tracing simulations in indoor environments [96, 97]. However, simulation-based approaches for indoor environments are no longer actively researched due to the high cost of preparing detailed 3D models of target indoor environments. Ray tracing simulations require detailed structure/material information about the target and surrounding buildings, which is difficult to obtain. Thus, this study investigates a simple binary classification of received signal strength—whether it is "strong" or "not" for a given satellite at each indoor position—using only accessible information like floor plan images. This approach aims to develop a practical system, as detecting indoor positions with strong signals is crucial for GPS-based indoor applications, such as multi-modal indoor positioning using GPS landmarks.

3.2.2 Indoor Applications based on GPS

Despite the challenges posed by weak and inaccurate GPS signals indoors, researchers have found innovative applications for these signals due to their low cost. The difference in GPS signal strength between indoor and outdoor environments has led to studies focused on detecting transitions between these settings. For example, some studies [98, 99] have explored the use of GPS signals for indoor/outdoor detection. These studies have been enhanced by incorporating additional sensors, resulting in more robust localization systems. Some studies from [100, 101] used a simple fingerprinting method, similar to Wi-Fi fingerprinting, for indoor GPS-based positioning, which involved site surveys. In contrast, our study aims to create a model that can automatically predict similar fingerprint maps independent of the environments by inputting satellite and target environment information.

3.3 Proposed Method

3.3.1 Preliminaries

We assume that multiple source environments and a target environment are available. Our method employs a variety of data sources from these source environments to build a comprehensive predictive model. These data sources are easily accessible through online mapping services and public orbit information of satellites. We denote the signal strength observed at each position at time t in a source environment as $\mathcal{S}_i^j(t)$, where j represents a specific position, and i denotes the satellite number, referring to a unique identity number. This study mainly investigates binary values of $\mathcal{S}_i^j(t)$. This data is essential for establishing the ground truth in our model. Three types of easily accessible information, as shown in the left part of Fig. 3.3, form the foundation for calculating the input features of our predictive model:

- **Satellite Location**: Represented as $L^i(t)$, indicating the 3D position of satellite i at time t. This information is derived from public orbit data.

- **Floor Map**: Denoted as M_n^{in}, visualized as a 2D image to provide structural layout information for the n-th environment. The 2D floorplan image can be easily acquired from an environment's administrator. We used floor maps for both source and target environments.

- **3D Map**: Symbolized as M_n^{out}, a top-view 2D representation that includes the height of each taller building around the n-th environment. The 3D map can be easily retrieved from online 3D map services.

In the target environment, we only have access to these three types of raw information. Our method transforms this data into a set of feature maps, which serve as the input for our prediction model. The output of our method is an estimate of the signal reception for each satellite i at position j in the target environment at time t, defined as $\hat{\mathcal{S}}_i^j(t)$.

3.3.2 Method Overview

Figure 3.3 provides an overview of the proposed method, which consists of two main parts:

1. **Calculation of Input Feature Maps**: In the initial step, we generate input feature maps using $L^i(t)$, M_n^{in}, and M_n^{out}. This process produces three types of feature maps: (i) Small-scale Static Maps: These maps provide detailed static information about the environment's shape and layout. (ii) Small-scale Dynamic Maps: These maps include detailed dynamic information about the satellite position, floor of interest, and surrounding buildings. They help understand signal propagation and occlusion by indoor obstacles and surrounding buildings. (iii) Large-scale Static Maps: These maps cover a significantly larger area and include extensive content, describing large-scale static information about surrounding and target buildings. They are used to learn about signal reflection and occlusion caused by the surrounding buildings.

2. **Multi-Scale Branch Fusion Network for Signal Reception Prediction (MSBF-Net)**: This neural network is designed for signal prediction, integrating multiple scales of input data through its branch fusion design. The network analyzes data at various scales to ensure a comprehensive understanding of the target environment. This architecture is particularly adept at handling the complexities and variations inherent in indoor signal propagation, making it a robust tool for our task.

3.3.3 Calculation of Input Feature Maps

Based on relevant research and our own findings, we categorized the factors affecting signal strength into two types: internal factors, such as building interiors, and external factors, including the height and position of nearby structures. To comprehensively analyze occlusions caused by these elements, we input various detailed maps into our model, ensuring all maps are gray-scale or binary.

Our method generates eight distinct feature maps for a given satellite at a specific position and time t, utilizing three core data types: satellite location $L^i(t)$, indoor floorplan M_n^{in}, and a rough 3D mapping of the surrounding environment M_n^{out}. These maps

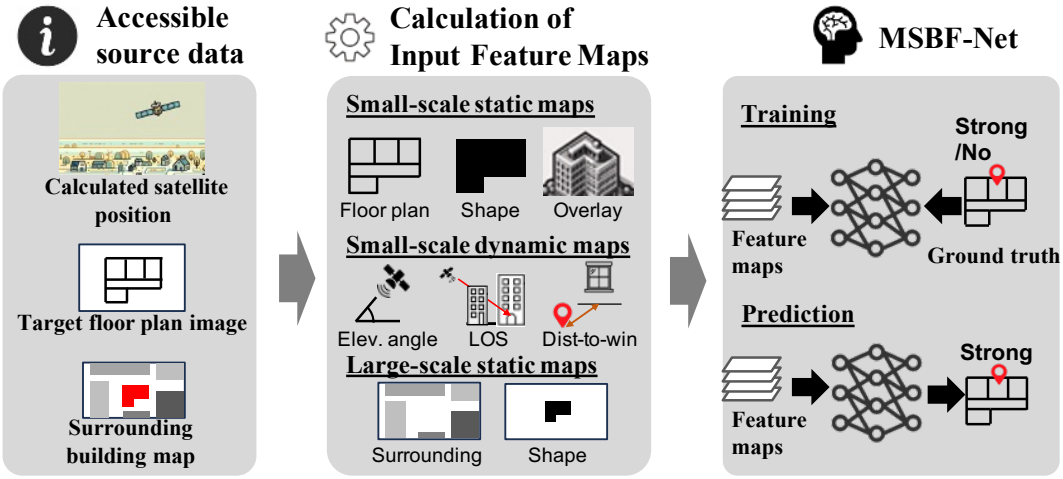


Figure 3.3: Overview of proposed method

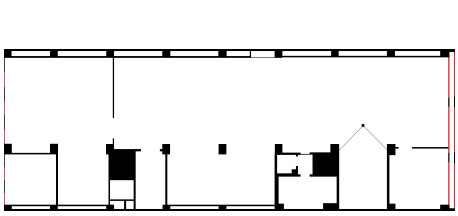
are classified into three categories: (i) small-scale static maps, (ii) small-scale dynamic maps, and (iii) large-scale static maps.

Each feature map is centered at the position of interest j and oriented such that the upward direction aligns with the satellite's direction to facilitate effective training of MSBF-Net. The output $\hat{\mathcal{S}}_i^j(t)$ represents the signal information at position j.

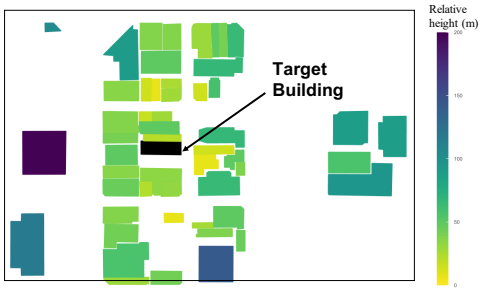
Information about Environment and Satellites

In this subsection, we introduce the crucial information used to calculate input feature maps. We assume access to an image of the target floor's floorplan, along with its positional details: latitude, longitude, altitude, and orientation (angle σ relative to the north). The floorplan image, depicted as a binary image, shows the positions of inner and outer walls (Fig. 3.4(a)). Additionally, we assume the target building is identifiable on online mapping services. Using these services' APIs, we can obtain a top-view 2D map centered around the target building, providing a visual representation of the location and height of each surrounding building (Fig. 3.4(b)).

Since the orbit of a GPS satellite is publicly available, we can calculate its 3D position at a given time t using its two-line element (TLE) set. This involves determining the satellite's exact location in space based on its orbital data. For a detailed explanation of this process, refer to [102]. Given a position of interest j, with λ_j, ϕ_j, and h_j as



(a) floor plan



(b) top-view 2D map with heights information

Figure 3.4: (a) Example of floor plan image. It contains the positions and widths of obstructions represented as black areas in the image. The positions of windows are highlighted in red lines. (b) Example of top-view 2D map with heights information about surrounding environment. The black-colored shape represents the target building. The other colored shapes are surrounding buildings. The colors indicate the relative heights from their rooftops to the target floor level.

longitude, latitude, and altitude, and $\lambda_i(t)$, $\phi_i(t)$, and $h_i(t)$ as the satellite's longitude, latitude, and altitude at time t, we convert these geographical coordinates into Cartesian coordinates:

$$\begin{aligned} x_j &= (R + h_j)\cos(\phi_j)\cos(\lambda_j) \\ y_j &= (R + h_j)\cos(\phi_j)\sin(\lambda_j) \\ z_j &= (R + h_j)\sin(\phi_j) \end{aligned} \quad \text{and} \quad \begin{aligned} x_i(t) &= (R + h_i(t))\cos(\phi_i(t))\cos(\lambda_i(t)) \\ y_i(t) &= (R + h_i(t))\cos(\phi_i(t))\sin(\lambda_i(t)) \\ z_i(t) &= (R + h_i(t))\sin(\phi_i(t)) \end{aligned}$$

where R is the Earth's radius. Using these coordinates, we calculate the azimuth and elevation angles of satellite i relative to position j:

$$\begin{aligned} \theta_{ij}(t) &= \arcsin\left(\frac{z_i(t) - z_j}{\sqrt{(x_i(t) - x_j)^2 + (y_i(t) - y_j)^2 + (z_i(t) - z_j)^2}}\right), \\ \alpha_{ij}(t) &= \arctan 2((y_i(t) - y_j), (x_i(t) - x_j)) \end{aligned} \tag{3.1}$$

Here, $\theta_{ij}(t)$ represents the elevation angle, and $\alpha_{ij}(t)$ represents the azimuth angle of satellite i relative to position j. This orientation is essential for accurately simulating the satellite's position concerning the indoor environment.

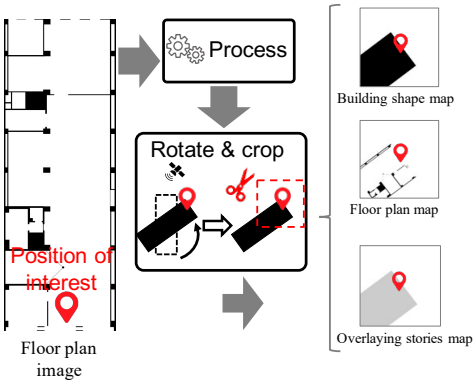


Figure 3.5: Workflow of small-static maps generation

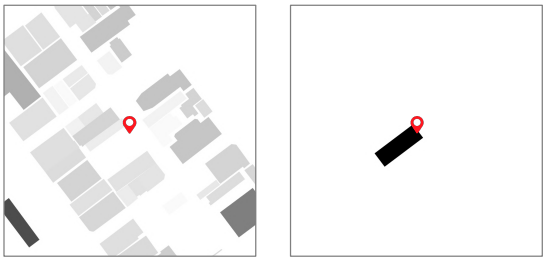


Figure 3.6: Example of large static maps

Small-scale Static Maps

Small-scale static maps in our approach are composed of three distinct input feature maps primarily derived from the floorplan.

- **Floorplan map**: This binary map delineates the positions of internal and external walls within the building, as these structural components can impede GPS signals. Illustrated in Fig. 3.5, the map is created by cropping the original floorplan image, centering it on the point of interest. It provides detailed information by highlighting the locations of walls, pillars, and elevators in black. In contrast, areas representing windows and other non-obstructive spaces are shown in white, based on the premise that window materials have a less adverse impact on GPS signals compared to wall materials, according to related studies [87, 88]. Crucially, the map is rotated by the angle $\alpha_{ij}(t) - \sigma$ at time t around the center of the map, ensuring that the upward direction aligns with the satellite's direction. This orientation is maintained across all subsequent maps to provide a standardized perspective for analysis.

- **Building shape map**: This map is a binary depiction that outlines the shape of the target building, focusing on its exterior walls. In this map, the area within the building's exterior walls is shown in black, while spaces outside the building are

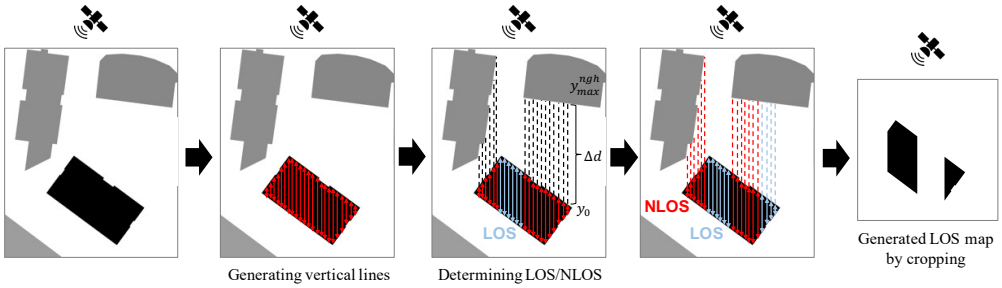


Figure 3.7: Formation of the LOS map used as a feature map. We assume the satellite is infinitely far above the target building, flanked by two adjacent buildings. We generate vertical lines representing satellite signals within the target area. For each line, we determine LOS/NLOS. Unobstructed gaps between buildings are identified as LOS areas. For obstructed areas, we calculate the distance Δd and relative height Δh_m to compare angles with the satellite's elevation to assess LOS. Combining all identified LOS regions forms the final LOS map.

represented in white, as shown in Fig. 3.5. This map offers a relative position of the point of interest within the building's overall layout.

- **Overlaying stories map**: Based on findings from related studies that indicate signal reception is influenced by the structure above the receiver, this map is a gray-scale representation indicating the number of floors above the point of interest. As shown in Fig. 3.5, the building's shape in this map is shaded with values representing the number of overlying floors.

Small-scale Dynamic Maps

Small-scale dynamic maps include three input feature maps, each aimed at capturing the dynamic elements of satellite signal propagation concerning the indoor environment. These maps are essential for comprehending how shifting satellite positions and internal/external obstructions influence signal reception at a specified target location.

- **Satellite elevation angle map**: This map is a gray-scale representation showing the elevation angle of the satellite. The elevation angle is crucial for GPS signal

propagation indoors. For instance, a lower elevation angle helps the signal penetrate deeper into the building. In this map, the non-white area aligns with the floor shape map, but the intensity within the building's section changes based on the satellite's elevation angle. We use min-max normalization on the angular value in degrees to determine the gray-scale value. The normalized value fills the building's shape on the map. The gray-scale value for the building shape is calculated using the formula $1 - \frac{\theta_{ij}(t)-90}{0-90}$, with 90 degrees as the largest elevation angle and 0 as the smallest because we only consider satellites visible above the horizon.

- **Line-of-Sight (LOS) map**: The LOS map is a binary map that indicates the LOS availability from the satellite to each position in the target environment. The process for creating this map is shown in Fig. 3.7. This map identifies direct visibility to the satellite, considering obstructions from surrounding buildings, providing a precise representation of the satellite's LOS and NLOS in urban settings. By using the indoor position j, the satellite's position $L^i(t)$, and the shapes, positions, and heights of nearby buildings M_n^{out}, we determine the LOS areas within the floor shape map. To identify LOS or NLOS areas, we use a calculated approach. We determine the angle to the top of a neighboring building based on the distance and height difference, then compare this angle with the satellite's elevation angle to see if the satellite is blocked by the building. Since a satellite is at a nearly infinite distance, the azimuth angle is constant across the building. This allows us to treat the satellite as an infinite light source, with signals as directional light, a concept used in computer vision [103]. In this context, we assume the GPS satellite projects parallel signals like rays of light from the top of the image after being rotated (the maps are rotated so that the upward direction aligns with the satellite's direction). Like light rays, these satellite signals can't penetrate physical obstacles (nearby buildings). The steps of creating LOS map is shown in Fig. 3.7.

- **Distance-to-window map**: The distance-to-window map is a gray-scale map quantifying signal obstruction by walls in the target floor and the distance from each location within the target floor to the nearest window in the satellite's direction, as shown in Fig. 3.8. This map is important for understanding signal propagation related to windows, which allow better signal penetration than other

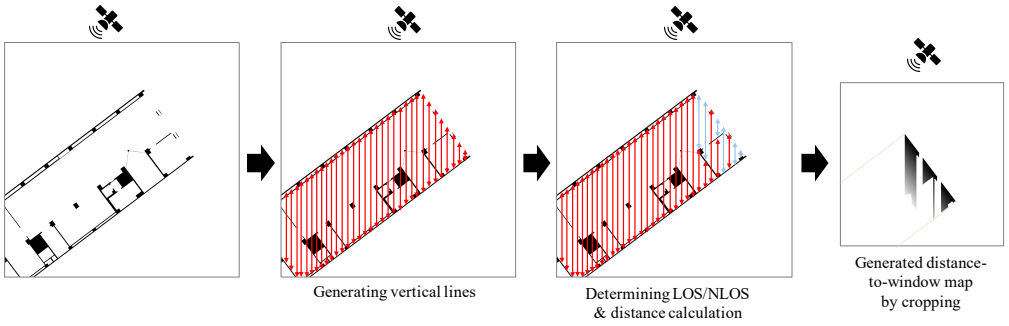


Figure 3.8: Creation of the distance-to-window map, utilized as one of the feature maps. Similar to the creation of the LOS map, we generate vertical lines (red) within the floor area. For each point on the line, we determine LOS/NLOS. If no indoor obstacle like a wall exists above a point, it is marked as LOS (blue line). We then calculate the distance from the point to the upper edge of the floor.

structural elements. The creation of this map considers the satellite's orientation and the wall structure within the target floor. In this map, black portions of the floorplan represent walls, pillars, and large objects, such as elevators, as primary obstacles that block or diffuse the line of sight. Similar to the LOS map, we consider signals from the satellite as coming from an infinite light source, unable to penetrate these dense structural elements. Viewing the floorplan in the Cartesian coordinate system, where each pixel represents a position, we maintain a consistent scale across all environments for uniformity.

Large-scale Static Maps

Large-scale static maps in our approach include two distinct input feature maps. These maps provide a wide view of the target environment and its neighboring areas, essential for comprehending the broader context of signal reception. Figure 3.6 visually represents these two maps, emphasizing their distinct characteristics and contributions to our model.

1. **Neighboring building map**: This grayscale map offers a top-down 2D perspective of the buildings surrounding the target location, as depicted in Figure 3.6. Its unique feature is the focus on the relative heights of these neighboring buildings

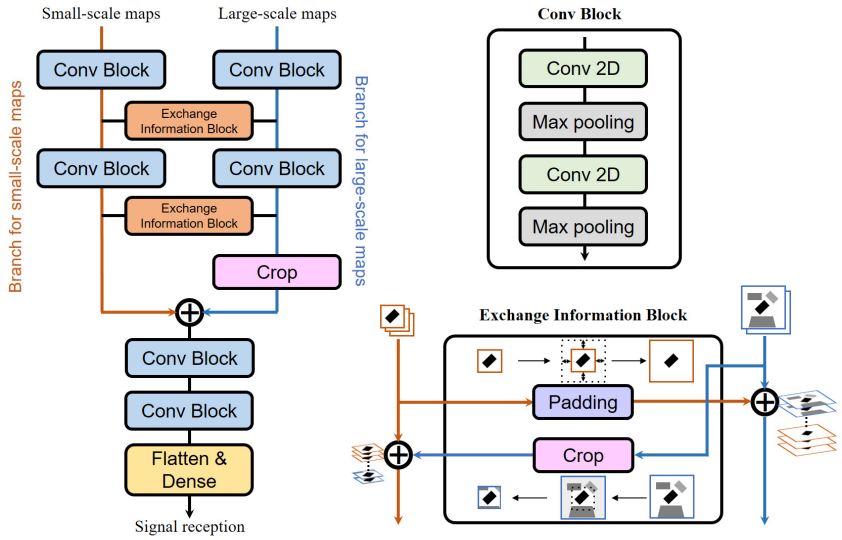


Figure 3.9: Network architecture of MSBF-Net (left). Detailed architectures of convolution and exchange information blocks (right).

in comparison to the target floor. Grayscale values indicate these relative heights, delivering a clear visual depiction of surrounding structures and their potential influence on signal propagation and reception.

2. **Building shape map**: Unlike the neighboring building map, the building shape map is a binary representation that outlines the target building's shape, as shown in Figure 3.6. This map focuses solely on the target building, highlighting its physical boundaries and structural layout. It is similar to the building shape map in the small-scale static maps but presented at a different scale.

3.3.4 MSBF-Net: Multi-scale Branch Fusion Network for Signal Reception Prediction

Network Architecture

An overview of the network architecture is displayed on the left side of Fig. 3.9. In MSBF-Net, the input feature maps are divided into two categories: small-scale and

large-scale feature maps. MSBF-Net is designed to efficiently integrate these feature maps of different scales. Due to differences in sizes, each category is independently processed using two separate input branches within the network. Combining these different-scale maps is essential for effective feature extraction because the information in both categories is strongly interconnected. To achieve this, we introduce the concept of the 'Exchange Information Block' into our network architecture. As shown on the left side of Fig. 3.9, MSBF-Net processes feature maps from each category using two input branches. Each branch utilizes a 2D convolutional neural network (CNN) and a max-pooling layer to handle the provided maps. The kernel size was 3×3, and the stride length was 1 in the 2D CNNs. The exchange information block, which facilitates information exchange between different branches, connects the two branches to form a ladder structure, as illustrated on the left side of Fig. 3.9. Its architecture is shown in the bottom-right part of Fig. 3.9. To transfer the small-scale maps to the large-scale map branch, the small-scale maps were enlarged by padding. Conversely, the large-scale maps were transferred to the small-scale map branch by cropping to a smaller size for concatenation onto the small-scale images. This block's architecture enables the processing of feature maps with appropriate resolutions referencing other feature maps with different scales. In the final stage of the model, the processed feature maps from both the small- and large-scale branches are concatenated after reducing the large-scale maps through cropping. The concatenated data are then fed into flattened and linear layers. The output of these layers is an estimate of the signal strength information for a given indoor position j (the center position of the maps) at time t. Note that this study tests outputs as binary value ("strong" or "not") because the preliminary investigation revealed that signal strength values are almost random in NLOS areas. As shown in Fig. 3.10, the signals within the red-marked areas (expected NLOS areas) appear to have a high level of randomness. For example, in the left red rectangle, a signal strength of 13 dB is observed between the 17 and 18 dB positions, possibly caused by complex multipath signals. Predicting such signal fluctuations using low-cost and easily accessible information is infeasible. Therefore, this study mainly tested a simple binary classification, i.e., stronger than or equal to a threshold or not.

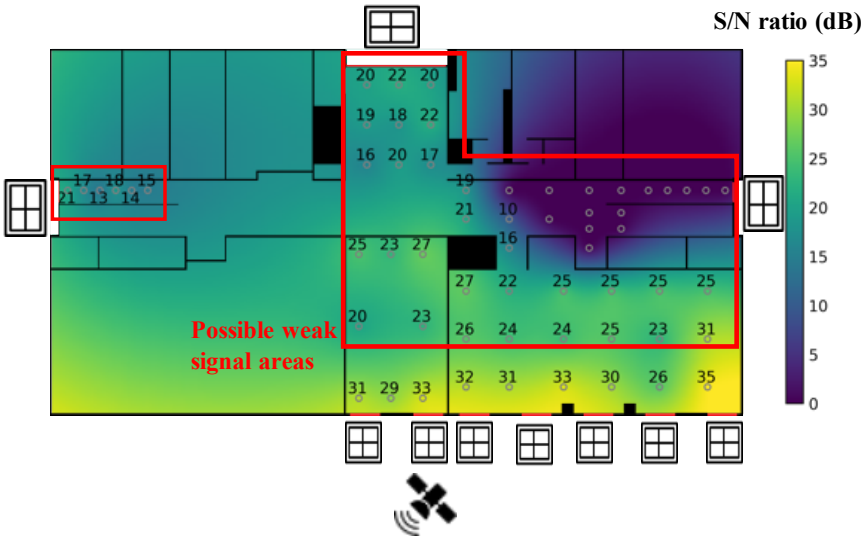


Figure 3.10: The GPS signal heat map for a low-level floor environment shows the satellite positioned at the bottom. Each observation point is numbered to indicate signal strength, averaged from 30 consecutive observations at 1 Hz. The heat map reveals that signal strength is stronger near the windows, as expected. However, in the red areas, signal strength is weaker and exhibits a random pattern rather than decreasing linearly with distance from the window..

Network Training

MSBF-Net was trained using the Adam optimizer [70]. For estimating numerical signal strength values, the loss function used was Mean Squared Error (MSE). In contrast, for predicting binary values (“strong” or “not”), binary cross-entropy was used as the loss function. In the binary classification task, the “strong” class had fewer instances than the “not” class. To address this class imbalance, we balanced the number of instances in both classes by undersampling the “not” class until both had approximately equal numbers of instances in each training batch. This approach helps prevent biased predictions that favor the majority class.

Table 3.1: Experimental environments used in this study.

Env.	Floor	Size	Type	Area	# of visible satellites	# of observation points
A	2F	22.9m x 43.2m	Lab	suburban	16	60
B	6F	21.7m x 41.9m	Lab	suburban	17	81
C	3F	22.2m x 26.2m	Multi	suburban	14	59
D	3F	13.4m x 38.4m	Multi	urban	10	36
E	7F	30.6m x 32.3m	Office	urban	11	50
F	5F	20.9m x 46.5m	Lab	suburban	14	64
G	5F	27.6m x 28.0m	Office	suburban	13	73
H	7F	24.6m x 26.5m	Office	urban	11	45
I	7F	53.0m x 39.6m	Office	urban	12	61
J	3F	19.3m x 17.8m	Multi	urban	11	47

3.4 Evaluation

3.4.1 Dataset

We used a Samsung Galaxy S21 with Android 12 to collect GPS data with a self-developed application, and we simultaneously collected ground-truth coordinates using an AR-based measurement application. For obtaining ground truth coordinates, we utilized Samsung Quick Measure to determine the relative coordinates of observation points, using a corner of a room as the origin. It's important to note that these ground-truth coordinates were not used as a direct ground truth for MSBF-Net but were utilized in the calculation of feature maps. Data collection in these varied environments occurred within a single day for each site. Preliminary experiments indicated that GPS signal strength remains consistent across different days. Additionally, studies such as [104] suggest that weather conditions have a limited impact on GPS satellite signal reception.

GPS signal strength values were collected at a 1 Hz sampling rate across 10 different experimental environments. At each point, GPS data was recorded for 30 seconds, and

the average value was used as the ground-truth value. Only satellites visible above the horizon (elevation angle greater than 0) were considered available, meaning the number of valid instances does not equate to the total count of satellites multiplied by the number of data collection points. Data collection spanned various types of buildings, including research facilities, office buildings, and commercial spaces, as well as multiple floors within these buildings. The surrounding building density also varied significantly. We divided into 2 areas, urban and suburban. The density of tall buildings in urban areas is much higher than that in suburban areas.

3.4.2 Evaluation Methodology

To assess the effectiveness of the proposed method, we compared MSBF-Net against several baseline methods:

- **MSBF-Net**: This is our proposed method.
- **Single (Pad)**: A variant of MSBF-Net that processes all feature maps within a single branch by concatenating the maps through padding the small-scale maps, then processing them with the large-scale branch. The branch for small-scale maps and exchange information blocks are not used.
- **Single (Crop)**: Another variant of MSBF-Net, this method processes all feature maps within a single branch by concatenating the maps through cropping the large-scale maps, then processing them with the small-scale branch. The branch for large-scale maps and exchange information blocks are not used.
- **W/o Small Static**: A variant of MSBF-Net that excludes small-scale static maps.
- **W/o Small Dynamic**: A variant of MSBF-Net that excludes small-scale dynamic maps.
- **W/o Large Static**: A variant of MSBF-Net that excludes large-scale static maps, and hence does not use the branch for large-scale maps or exchange information blocks.

- **SIM**: This is a simulation-based method for the binary classification task. Instead of using a 3D model of the target indoor environment as in ray tracing simulations, it uses the 2D floorplan image, the 3D coordinates of the satellite, and a 3D map of surrounding buildings to simulate signal propagation. It identifies if a position is LOS or NLOS using information about indoor obstacles and surrounding buildings, similar to the calculation of LOS maps and distance-to-window maps. This method assumes that GPS signals are occluded by surrounding buildings, indoor obstacles, and the ceiling, with a window height of 2.5 meters. If a position is identified as LOS, it is classified as "strong;" otherwise, it is classified as "not."

We used leave-one-environment-out cross-validation to evaluate the methods described. Specifically, when testing MSBF-Net on a particular environment, we trained MSBF-Net using data from the remaining environments. For the binary classification task, which is the primary focus of this study, we used the true positive (TP) rate, false positive (FP) rate, and macro-averaged accuracy. In this context, the "strong" class is considered the positive class. In our experiments, the small-scale maps were set to a resolution of 200 pixels, which represents an area of 20 meters in the real world. The large-scale maps had a resolution of 500 pixels, corresponding to an actual size of 200 meters. Based on preliminary experiments, we set the threshold value for binary classification at 27 dB. To acquire 3D maps of surrounding buildings, we utilized Google Earth [81] and the Geospatial Information Authority of Japan [105]. The floorplan images used as inputs for our method were created based on original floorplans provided by the administrators of the respective environments. Using drawing software, we removed lines corresponding to windows and irrelevant objects, such as dimension lines, before binarizing the images.

3.4.3 Results

Prediction Performance

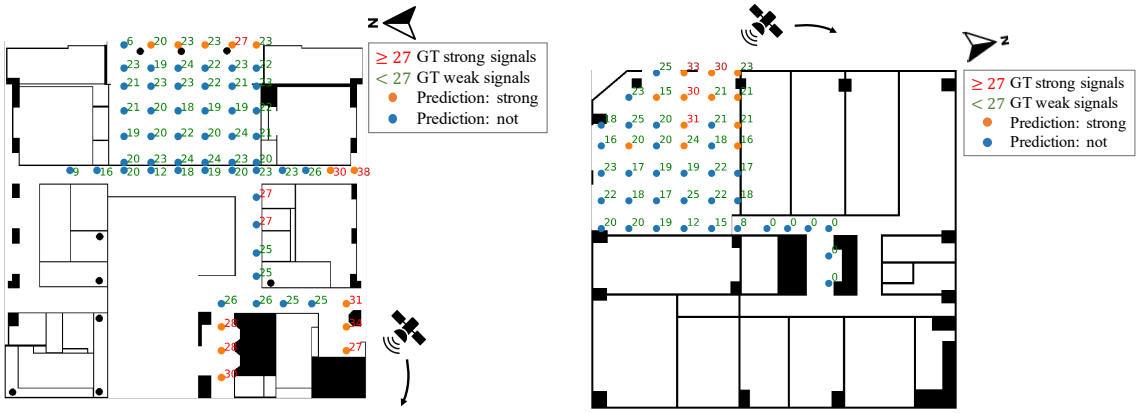
Table 3.2 displays the results of the binary classification task for MSBF-Net. The average true positive rate, false positive rate, and macro accuracy achieved by MSBF-Net were approximately 0.77, 0.28, and 0.74, respectively. The data suggest that MSBF-Net performs better in suburban environments due to the higher number of visible satellites,

Table 3.2: Performance of MSBF-Net in binary classification when threshold is 27 dB

Env.	A	B	C	D	E	F	G	H	I	J	Avg
TP rate	80%	65%	87%	80%	94%	91%	70%	85%	67%	48%	77%
FP rate	21%	18%	41%	24%	45%	33%	16%	36%	31%	15%	28%
Macro Acc	80%	74%	73%	78%	74%	79%	77%	74%	68%	66%	74%

which enhances signal prediction accuracy. Urban environments, with their inherent complexities and fewer visible satellites, present more challenges, leading to higher FP rates and more variability in performance metrics. Fig. 3.11 (a) shows example predictions and ground truth on a floor map of Environment G. In the figure, orange dots signify predictions labeled as "strong," while blue dots represent predictions of "not." Green numbers indicate locations where the ground truth signal strength is less than 27 dB, and red numbers highlight those with a ground truth signal strength of 27 dB or higher. During data collection, the satellite's azimuth was between 210 and 240 degrees, moving from the bottom right to the lower side of the figure, with an elevation angle of around 10 to 30 degrees. As shown, MSBF-Net successfully detected "strong" positions near windows facing the satellite's direction. However, there are two false negative points located in the middle corridor on the right side (strength of 27 dB), likely due to complex multi-path effects in that area. Additionally, high signal strength points on the upper side of the figure may result from signal refraction caused by external building clusters. While MSBF-Net attempted to predict signal refraction based on the neighboring building map, it produced several false positives.

Another prediction example is shown in Fig. 3.11 (b). During data collection, the satellite moved from 290 to 316 degrees in azimuth angle (from top to right in the figure) and from 30 to 40 degrees in elevation angle. MSBF-Net generally classified points close to the upper windows as "strong," achieving reasonable accuracy. However, in the rightmost columns of the left-upper room, the top four positions are false positives where the model incorrectly predicted strong signal strength. These errors may be due to the influence of neighboring buildings, as these positions are close to an NLOS area caused by a building in the above direction. Our investigation found that True Positive



(a) Predictions by MSBF-Net of satellite PRN 16 in Env. G

(b) Predictions by MSBF-Net of satellite PRN 32 in Env. H

Figure 3.11: Example of predictions by MSBF-Net for different satellites in Env. G and H

and False Positive positions are closer to the NLOS area (3.2m and 2.6m, respectively) compared to True Negative and False Negative positions (6.4m and 5.6m). This indicates that proximity to the NLOS area significantly impacts classification accuracy, with closer positions more likely to be misclassified. The model may have overemphasized the surrounding environment's ability to diffract signals, leading to erroneous strong signal predictions. This complex interaction between building architecture and signal propagation likely caused false positives in the model's predictions.

As shown in Table 3.2, the macro accuracy in Environments I and J was poor. An expressway next to Environment J was not included in the 3D map services used, so MSBF-Net could not predict signal occlusion by the expressway. At locations near the expressway, the satellite signal strength ground truth is very low, but the model predicts high signal strength for these points. This discrepancy is caused by the lack of expressway data in the neighboring building information. In Environment I, high false positive rates for some satellites with low elevation angles were likely due to super tall buildings in the map's upper right corner, not shown in the neighboring map due to the limited scale, leading to inaccurately detecting LOS for these satellites and causing high FP rates. However, MSBF-Net performed well in other environments as shown in Ta-

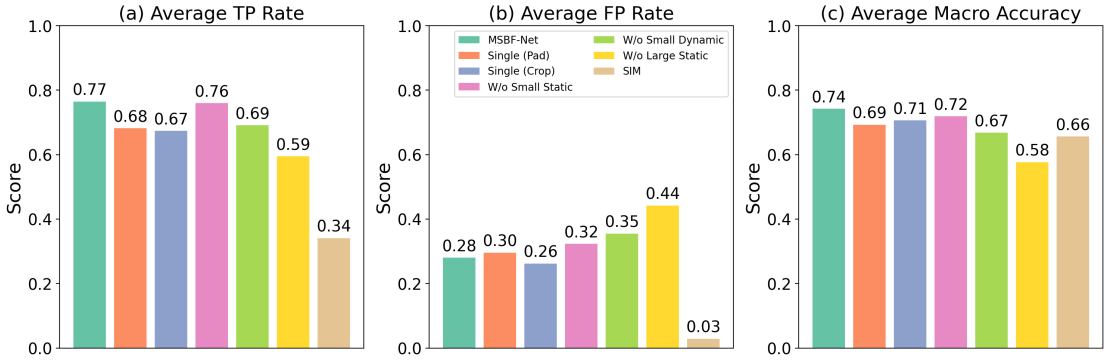


Figure 3.12: Ablation study of performance analysis of MSBF-Net

ble 3.2, despite using only easily accessible information about the target environments. In Section 3.5, we apply MSBF-Net to a localization application (indoor positioning with GPS fingerprinting) to evaluate its performance.

Figure 3.12 illustrates the average true positive rates, false positive rates, and macro accuracy of Single (Pad), Single (Crop), and MSBF-Net. (Supplementary information provides the macro accuracy for each of the ten environments.) The averaged accuracy of Single (Pad) and Single (Crop) was about 0.04 lower than that of MSBF-Net, highlighting the effectiveness of MSBF-Net's architecture, which allows for independent analysis of small- and large-scale maps at their appropriate resolutions while referring to other feature maps with different scales. The average true positive rates for Single (Pad) and Single (Crop) are significantly lower than the average true positive rate for MSBF-Net. This discrepancy is due to the absence of the "Exchange Information Block" in the Single (Pad) and Single (Crop) methods. The padding approach used in Single (Pad) can introduce irrelevant context, diluting the model's ability to capture detailed information in the small-scale maps and resulting in a lower true positive rate. Conversely, Single (Crop) may omit valuable context when cropping parts of the large-scale maps, leading to the loss of critical information necessary for identifying strong signal areas.

Figure 3.12 also presents the results of W/o Small Static, W/o Small Dynamic, and W/o Large Static. Surprisingly, the macro accuracy of W/o Large Static was about 0.2 lower than that of MSBF-Net. This is because surrounding buildings are the most influential factor in signal occlusion. The result also indicates that MSBF-Net effectively

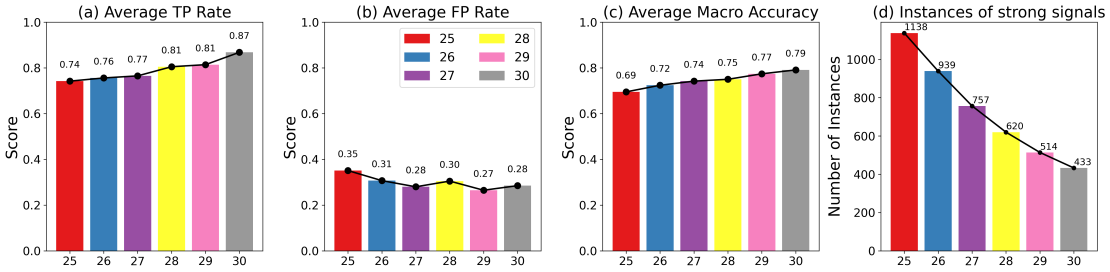
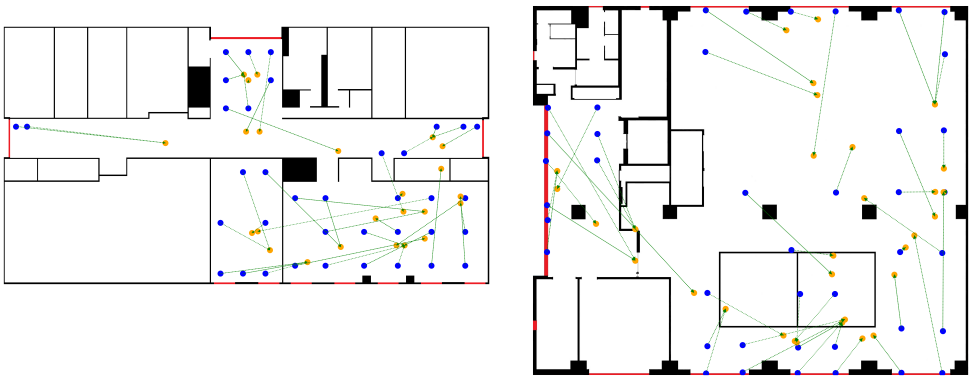


Figure 3.13: Comparative performance analysis of threshold changes in MSBF-Net: (a), (b), and (c) display the average true positive rate, false positive rate, and macro accuracy across all environments for thresholds from 25 to 30 dB. (d) shows the change in the number of "strong" instances with varying threshold values.

fuses the small-scale and large-scale maps to estimate fine-grained signal reception information. Figure 3.12 also includes the results of the simulation-based method, SIM. Although this method also employs information about indoor obstacles and neighboring buildings, its true positive rate was significantly poorer than that of MSBF-Net. As shown in Fig. 3.10, window-side positions occluded by indoor obstacles such as pillars and walls sometimes have strong signals. We believe that capturing complex interactions between GPS signals and indoor/outdoor obstacles is crucial for precise signal prediction. This is likely why MSBF-Net, trained on raw floor images and 3D maps of surrounding environments, outperformed SIM.

Threshold for Binary Classification

In the previous experiments, a threshold of 27 dB was used for binary classification. Here, we analyze how varying the threshold value impacts the performance of MSBF-Net. Figures 3.13 (a), (b), and (c) illustrate the changes in true positive rate, false positive rate, and macro averaged accuracy as the threshold value is adjusted. Additionally, Fig. 3.13 (d) shows the variation in the number of "strong" instances. As the threshold value increases, the performance of MSBF-Net generally improves. However, an increase in the threshold value also reduces the number of "strong" instances, which may limit the applicability of our method in certain contexts, such as multi-modal indoor positioning based on GPS landmarks.



(a) Indoor Positioning Results in Env. A ($k = 5$)

(b) Indoor Positioning Results in Env. C ($k = 5$)

Figure 3.14: (a) and (b) show example indoor positioning results based on predictions by MSBF-Net in Environments A and C, respectively ($k = 5$). The blue dots represents the ground-truth coordinates while the orange dots are denoted as predicted coordinates. The green arrows represent the discrepancies between the the ground-truth coordinates and the prediction coordinates.

3.5 Discussion

We applied MSBF-Net to estimate a smartphone's position in an indoor environment. To do this, we collected signal strength values from multiple satellites at position j at time t in a given environment. These values were binarized using a 27 dB threshold, creating a binary observation vector by concatenating the binary values. MSBF-Net was then used to generate a binary fingerprint vector for each observation point at time t by concatenating its binary estimates for the satellites. These predicted binary fingerprint vectors formed a GPS fingerprint map, with each vector linked to the ground-truth indoor coordinates.

To predict the indoor coordinates of an observation, represented by a binary observation vector, we used the fingerprint map in conjunction with the k nearest neighbor (kNN) algorithm. Observations with no "strong" values were ignored, as they lacked useful information. When k was larger than 1, we averaged the coordinates of the top-k fingerprints. Remarkably, the GPS-based indoor positioning achieved a mean absolute

error (MAE) of 3.38 meters on average when $k = 5$. Although this performance is slightly inferior to general Wi-Fi fingerprinting (which typically has a 2-4 meter error), GPS-based indoor positioning does not require additional infrastructure or extensive site surveys. Figure 3.14 illustrates the relationship between the ground-truth positions and the predicted positions using GPS-based indoor positioning with $k = 5$. As seen in the results, while many predictions exhibit moderate positioning errors, a few show significantly larger errors. Observations with few "strong" values (i.e., values of 1) are particularly susceptible to classification errors by MSBF-Net, which is an inherent limitation of fingerprinting-based methods.

3.6 Conclusion

This study introduced a novel method for predicting GPS signal reception information at specific indoor locations. As far as we know, this is the first approach that uses readily accessible data to predict GPS signal reception without the need for manual data collection in the target environment. Our method generates input feature maps from easily obtainable information, such as the calculated 3D position of a satellite and a 3D map of the surrounding environment. Our proposed MSBF-Net then efficiently integrates these input feature maps at different scales using a ladder-like structure. Experiments conducted in real-world environments validated the effectiveness of our feature maps and the MSBF-Net model.

Chapter 4

3D Trajectory Estimation of Underground Animals

4.1 Introduction

4.1.1 Background

Studying the behavior of underground animals, such as rats and moles, is crucial due to their unique biological traits and significant impact on subterranean ecosystems. These animals exhibit distinct behaviors, including tunneling patterns and foraging strategies, which provide valuable insights into the dynamics of underground environments. Understanding these behaviors can help develop better agricultural practices and more effective ecosystem management strategies. For example, mice and moles play a critical role in soil aeration and nutrient redistribution [106]. Their tunneling activities can enhance soil structure, promote water infiltration, and improve root growth, which are essential for healthy plant development. However, these same behaviors can also cause damage to crops and garden plants, leading to significant economic losses in agriculture. By studying their behavior, researchers can develop strategies to mitigate these negative impacts while harnessing their beneficial effects on soil health.

One of the primary tools used to study wildlife behavior, including that of underground animals, is the bio-logger [107]. Bio-loggers are small, sophisticated electronic devices attached to animals' body to record various types of data, such as location,

movement, temperature and even videos of activities. These devices are indispensable in ecological research because they enable long-term monitoring of animals in their natural habitats, providing comprehensive data on their activities. Bio-loggers have revolutionized the study of animal behavior by providing detailed and continuous data that were previously difficult to obtain. Bio-loggers offer another significant advantage of adaptability to various environments, including terrestrial, aquatic, and aerial habitats [108, 40]. This versatility allows researchers to track animals in diverse settings, providing unprecedented insights into their daily lives, migration patterns, and interactions with their habitat and and environmental conditions. However, tracking underground animals presents significant challenges, primarily due to the limitations of conventional tracking methods in subterranean environments. GPS is effective above ground but becomes ineffective underground [109]. Similarly, the use of cameras in bio-loggers is limited by the absence of light in underground settings, rendering visual tracking impractical. Other wireless technologies also struggle to function effectively in dense underground environments, complicating the tracking process.

In recent years, alternative methods relying on computer vision and machine learning techniques have emerged to study animal movements in controlled environments [110, 111]. While these methods yield detailed insights into animal behavior, they are often confined to controlled environments and may not transfer directly to natural underground habitats. Computer vision techniques require clear visibility and sufficient lighting, unavailable underground. Thus, these techniques offer promising tools for studying animal behavior, but their application to underground environments remains challenging. Given these conditions, a more straightforward and direct approach is necessary.

4.1.2 Research Objective

These challenges require the creation of innovative tracking methods that surpass the limitations of traditional approaches. Therefore, the primary objective of this research is to develop a novel tracking method that effectively studies the movement and behavior of underground animals. Given the complexity of underground environments, particularly the intricate 3D structures of tunnels, we propose implementing a novel trajectory

estimation technique that operates effectively in three dimensions. One promising approach to tracking movement in GPS-denied environments is the use of inertial measurement units (IMUs), which employ accelerometers and gyroscopes to capture valuable movement data, because they are applicable in different scenarios. As discussed in Chapter 2, IMUs can provide detailed information on the movement of animals including movement distances and directions. However, a significant challenge with inertial sensors, especially gyroscopes, is their susceptibility to drift errors over time. These errors can accumulate, leading to substantial distortions in the trajectory data.

To address this challenge, we introduce a corrective mechanism that leverages strategically placed rotating magnets, each emitting a distinct magnetic signature detectable by a magnetometer included in the sensor array [112]. These magnetic landmarks serve as fixed points in the 3D space, providing absolute reference coordinates. In practice, the cost of installing a single magnetic landmark is relatively low. Each unit consists of a small permanent magnet, a servo motor, an Arduino board to connect motor and battery and control frequency of the motor, and disposable lithium batteries. To maintain accuracy, we will install multiple magnetic landmarks based on the size of the environment. If the tunnel is close to the surface, we can place the devices on the ground; if the tunnel is slightly deeper, we place these devices 10cm above the tunnel to avoid corrupting the tunnels. Also, for different magnets, we set their rotation frequency diverse from each other so that each magnet can be identified by corresponding frequency. By incorporating these landmarks, we can correct accumulated drift errors using a particle filter approach.

Building on the data from accelerometers and gyroscopes, our model calculates the distance traveled and directional changes of the target animal (e.g., a rat) at each time interval. Simultaneously, magnetometer data are used to determine proximity to the magnetic landmarks, identifying which specific magnet the animal is near. This dual approach ensures that the model maintains accurate positional data over time. Additionally, our observations indicate that the movement of rats is often intermittent rather than continuous. To account for this behavior and avoid sensor errors that could arise from periods of inactivity, we designed a module to detect whether the animal is moving or stationary. By integrating the results from the accelerometer, gyroscope, and magnetometer, and combining them with the movement detection module, we can accurately

track the animal's trajectory using a particle filter method, as also outlined in Chapter 2. It is important to note that in this experiment, the initial starting position is known. We assume that while GPS signals are unavailable underground, they are accessible above ground. When the animal enters the underground environment, its starting position can be determined using GPS data. This prior condition differs from the one in Chapter 2.

4.1.3 Contributions

- To enhance the precision of animal trajectory estimates, we integrate magnetic field information generated by rotating magnets with different frequencies into the model. By adopting a correction technique based on a particle filter, we refine the trajectory predictions, significantly improving the accuracy of our underground tracking system. The introduction of an model that leverages neural networks to estimate 3D trajectories of underground animals, capturing their movement in all spatial dimensions without the need for direct visual confirmation.

- We conducted comprehensive experiments in simulated underground environments to validate our approach. These experiments demonstrated the robustness and applicability of our method, confirming its effectiveness in accurately tracking the movement of underground animals in complex subterranean settings. The successful application of our system in these controlled settings highlights its potential for broader ecological studies and practical implementations in real-world scenarios.

4.2 Related Work

4.2.1 Animal Tracking Systems

Animal tracking systems have evolved significantly with advancements in technology, enabling researchers to gather detailed data on animal movement and behavior. Various methods and technologies are employed, each with its own set of advantages and challenges. One common method for tracking small animals is RFID technology, which involves attaching RFID tags to the animals [113]. These tags emit signals captured by

strategically placed RFID readers. This method is effective for tracking within confined areas like labs. Some studies [114] demonstrated the effectiveness of an ultra-high-frequency (UHF) RFID system in accurately monitoring the location and movement of tagged animals. However, its main limitation is the short range and the need for multiple readers to cover larger areas.

GPS-based tracking is another widely used method, particularly for studying wildlife in large and open environments. GPS collars provide precise location data over large distances and are invaluable for studying migratory patterns [115, 116, 117]. These systems often incorporate additional sensors, such as accelerometers, to gather data on the animals' activity levels. The major drawback is the weight and battery life of the collars, which can limit their use on smaller animals [118].

Vision-based tracking systems utilize cameras to capture the movement of animals, especially in small or lab settings. These systems can be highly accurate and provide rich data on the animals' behavior and interactions. In some cases, they also have a strong advantage of tracking multiple animals simultaneously by deep learning algorithm [119, 120, 121]. However, they require significant computational power and storage for processing and analyzing the video data. Besides this, there are also some studies using multiple modalities, even with audio recorder to improve the accuracy of tracking systems [122]. In our research, since we decided to use a low-cost method for underground animals so that the computer-vision way is not suitable here.

4.2.2 Animal Activity Recognition

Animal activity recognition systems are designed to classify and interpret the behaviors and activities of animals based on data collected from various sensors. These systems play a crucial role in understanding animal behavior, health monitoring, and ecological studies. Accelerometers are widely used in activity recognition systems due to their ability to capture detailed movement data. By analyzing the accelerometer data, researchers can classify different activities such as walking, running, resting, and foraging. The data from accelerometers can be processed using machine learning algorithms to develop models that accurately classify different behaviors [123, 124].

Vision-based systems for activity recognition utilize video data to monitor and an-

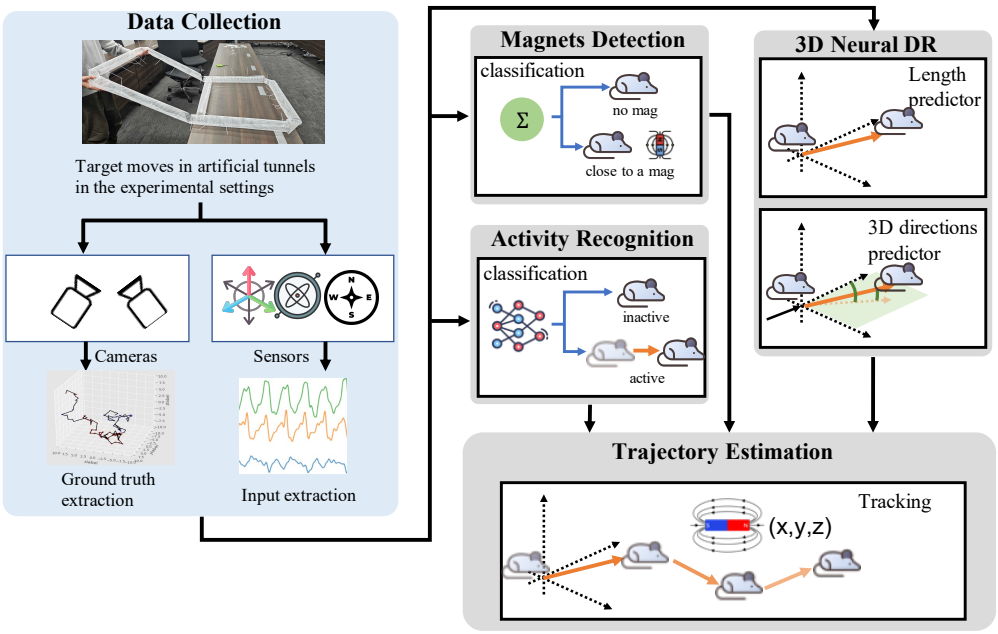


Figure 4.1: Overview of the proposed method

alyze animal behaviors. These systems can provide a more comprehensive understanding of the animals' interactions with their environment and other individuals. The use of advanced Neural Networks for precise object detection and pose estimation has been shown to be highly effective in tracking and recognizing activities in multiple environments [125, 126]. By training the models on annotated video data, researchers can achieve high accuracy in identifying different activities and behaviors. In our study, we also implemented activity recognition by creating a simple binary classifier to determine whether the animal is stationary. This approach helps us minimize the potential drift errors that can accumulate when the animal is not moving.

4.3 Methodology

4.3.1 Preliminaries

We assume that a rat is equipped with inertial sensors, including an accelerometer, gyroscope, and magnetometer. From these sensors, we collect time-series data $\mathcal{S} =$

$\{\mathcal{S}^i, \mathcal{S}^m\}$, where $\mathcal{S}^i$ includes 6-axis inertial measurements from the accelerometer and gyroscope, and $\mathcal{S}^m$ includes measurements from magnetometer. Additionally, $\mathbf{s}_t^i \in \mathbb{R}^3$ represents the six-axis IMU data at time t, and $\mathbf{s}_t^m \in \mathbb{R}^3$ represents the magnetic field strength data at time t. For the environment, we placed artificial tunnels for underground animals to move and ensured the tunnels stationary. We also fixed several magnets in place to serve as static landmarks and the positions of additional installed magnets are denoted as $\mathbf{P}_n^{Mag} \in \mathbb{R}^3$, where n indicates the magnet's number.

4.3.2 Overview of Proposed Method

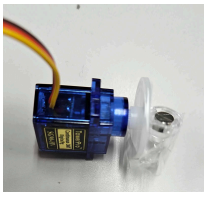
Fig. 4.1 shows the overview of our proposed method. The left part illustrates the experimental data collection, while the right part depicts the inference process using the collected data to predict the path traversed by the animal over a period of time. In the right part, there are 4 parts:

- **Magnets Detection**: This module determines whether the animal is close to a specific magnet and can identify which magnet the animal is approaching at time t by given $\mathbf{s}_t^m$.

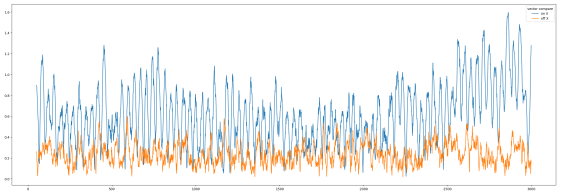
- **Activity Recognition**: This module determines whether the animal is in an active or inactive state at any given time t. If the animal moves from time t_n to t_{n+1}, it is considered active at time t_n; otherwise, it is inactive. Note that the time window between each adjacent time is fixed.

- **3D Neural DR**: This module includes the Length predictor and the 3D directions predictor, which respectively estimate the movement distance and relative changes of movement direction in horizontal and absolute directions in vertical plane within a time window from t_n to t_{n+1}, using accelerometer and gyroscope data $\mathcal{S}^i$. The 3D directions predictor is designed to output two direction values, representing changes in bearing (in the horizontal plane) and absolute elevation (in the vertical plane).

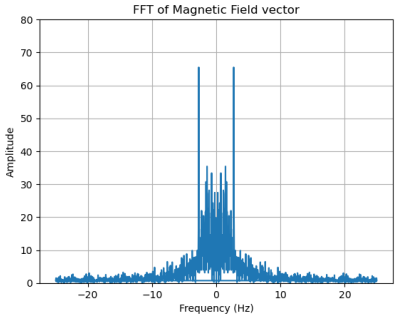
- **Trajectory Estimation**: Utilizing a particle filter, this module combines outputs from the preceding modules to estimate the walking trajectory. In this scenario,



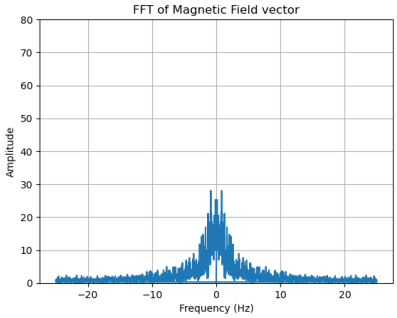
(a) Magnet (right) attached to a servo motor (left blue one)



(b) Raw signal waves when near a magnet (blue curve) VS not (orange curve)



(c) Frequency spectrum when near a magnet



(d) Frequency spectrum when away from a magnet

Figure 4.2: Magnets detection settings

we consider only forward tracking, assuming the initial position (x_0, y_0, z_0) is known. The outputs from the Magnets Detection, Activity Recognition, and 3D Neural DR modules help correct accumulated errors. It provides the positions (x_t, y_t, z_t) up to the end time, forming a complete trajectory.

4.3.3 Magnets Setting and Detection

Firstly, we will set up a corresponding number of "rotating magnets" at several fixed positions. We attach small magnets to servo horns or arms (or called output shaft) of servo motors, and when the servo motors are powered, they maintain a constant rotation speed, causing the small magnets to rotate steadily as shown in Fig. 4.2 (a). Consequently, the small magnets generate a distinctive sine wave that can be captured by sensors, as illustrated in Fig. 4.2 (b). In the case of multiple magnets, we can control

the rotation rates of the servo motors so that each motor has a different speed, resulting in varying magnetic field strengths and unique signatures.

For each time t, we generate the frequency spectrum using a fast Fourier transform (FFT), which transforms the time-domain signal into its constituent frequencies. The FFT output is given by the formula:

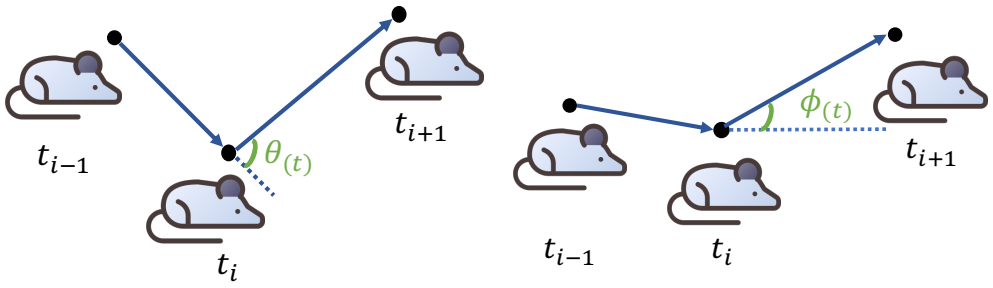
$$X_k = \sum_{n=0}^{N-1} \mathbf{s}_t^m \cdot e^{-i\frac{2\pi}{N}kn} \tag{4.1}$$

where X_k is the FFT output of the magnet No. k, $\mathbf{s}_t^m$ is the input magnetometer signal, f is the frequency, N is the number of samples, and i is the imaginary unit. The FFT allows us to analyze the frequency components of the signal generated by the rotating magnets.

When the animal approaches a rotating magnet, the generated graph, as shown in Fig. 4.2 (c), exhibits a very high amplitude at low frequencies. This distinctive high amplitude indicates the presence of the specific sine wave generated by the rotating magnet. In contrast, in the absence of a magnet, the graph generated, as shown in Fig. 4.2 (d), only shows a slightly higher amplitude near zero frequency. This minor increase is due to the surrounding environmental magnetic fields and the Earth's magnetic field, which are always present. By comparing the high amplitude frequencies at time t, we can directly determine whether the animal is near a rotating magnet at t. Additionally, the differences in frequency values allow us to identify which specific magnet the animal is near. Each rotating magnet, with its unique rotation rate, produces a distinctive magnetic field signature, enabling precise identification.

4.3.4 Activity Recognition

The activity recognition module estimates whether the animal is active or inactive during each time window. In this study, if the animal remains completely still within the time window, it is defined as inactive. Conversely, any movement, even if it involves returning to the same position within a time window, is considered active. Based on this observation, we use the time series of only 3-axis accelerometer data as inputs for the window-side detector. The output of the activity recognition model is a binary value at time t, indicating whether the animal is active or inactive at that specific time.



(a) Animal movement top view

(b) Animal movement side view

Figure 4.3: 3D animal movement by two kinds of view

The neural network is a binary classifier comprising one 1-dimensional convolutional layer with 16 output channels using the ReLU activation function, followed by a Batch Normalization layer, a Max Pooling layer, and a Dropout layer with a probability of 0.5. The output from these layers is flattened and passed through a fully connected layer with 64 units, followed by another ReLU activation function. Finally, the network includes a fully connected output layer with the sigmoid activation function. The network is trained to minimize binary cross-entropy loss using the Adam optimizer, with a learning rate of 0.001 and weight decay of 0.0001.

4.3.5 3D Neural Dead Reckoning for Animal

In this module, we estimate the displacement, horizontal changes in walking directions and absolute vertical walking directions within a W-second time window. This module builds upon the neural PDR approach discussed in Chapter 2, extending its application from 2D to 3D space. Therefore, we will provide a concise summary here. The method for obtaining ground truth values (i.e., displacement d_t, change of bearing angle $\Delta\theta_t$, and elevation angle ϕ_t) will be elaborated in Section 4.4.1.

- **Length Predictor**: The input for the movement length predictor is a time window of accelerometer and gyroscope data $\mathbf{s}_t^i = \{\mathbf{s}_{t-W_i}^i, ..., \mathbf{s}_{t-1}^i, \mathbf{s}_t^i\}$, where W_i is a static window size. The input data is processed by a neural network for length prediction, which includes an LSTM layer with 16 units using the ReLU activation

function, a dropout layer, and an output layer with a linear activation function. The output of this network is the displacement d_t between time $t-1$ and time t. The model is trained with the Adam optimizer to minimize the mean squared error (MSE) between the predicted displacement and the actual ground truth.

- **3D Directions Predictor**: In this predictor, we designed two sub-predictors with similar models but different inputs and outputs. For predicting changes in the bearing angle, the input is 3-axis gyroscope data, and the model we propose includes: a bi-directional LSTM layer with 16 units using the ReLU activation function, a fully connected layer with 4 units using the ReLU activation function, and an output layer with a linear activation function. The output of the bearing change predictor is the angular change in the horizontal plane (parallel to the ground) $\Delta\theta_t$ between time $t-1$ and time t from top view, as shown in Fig. 4.3 (a). The model is also trained using the Adam optimizer to minimize the MSE between the predicted angular bearing change and the actual values. For the other predictor, which estimates the absolute elevation angles, we use only accelerometer data. This is because changes in elevation, such as moving uphill or downhill, are primarily captured by the accelerometer due to the influence of gravity. We employed the same model architecture as the bearing directional change predictor, although they are distinct models. The output of elevation predictor is ϕ_t between $t-1$ and time t from side view where is parallel to the horizon as shown in Fig. 4.3 (b).

4.3.6 Trajectory Estimation

We assume that the position of animal at t_0 is known. The animal's trajectory after time t_0 is estimated using a particle filter with outputs from the 3D Neural DR module, as shown in Fig. 4.4. In this particle filter, a possible position p_n at time t represents a candidate for the animal's position, containing absolute 3D coordinates $\mathbf{v}_{p_n}^t \in \mathbb{R}^3$, accumulated bearing angle $\sum_{t=0}^{t_i} \Delta\theta_t$ of p_n^t, and weights $w_{p_n}^t$. Two other modules correct accumulated trajectory errors. For instance, if the animal is predicted to be inactive, displacement and directions are set to zero. During re-sampling, particles are re-sampled based on their weights.

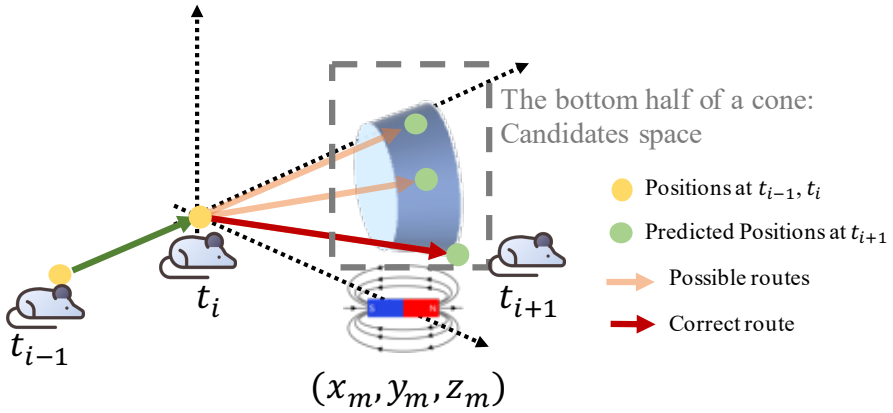


Figure 4.4: An example for weighting the candidates at time t_{i+1}. Given known positions at t_{i-1} and t_i, by the next time step, the sensor detects that the animal is near a magnet. Consequently, the particle filter updates its predictions, selecting the position closest to the magnet as the most accurate prediction for t_{i+1}

The tracking algorithm based on the particle filter operates in three steps: sampling, weight calculation, and re-sampling, similar to the process introduced in Chapter 2. The sampling and re-sampling processes are identical to the previous method, except for the 3D position prediction, which is calculated by the formula:

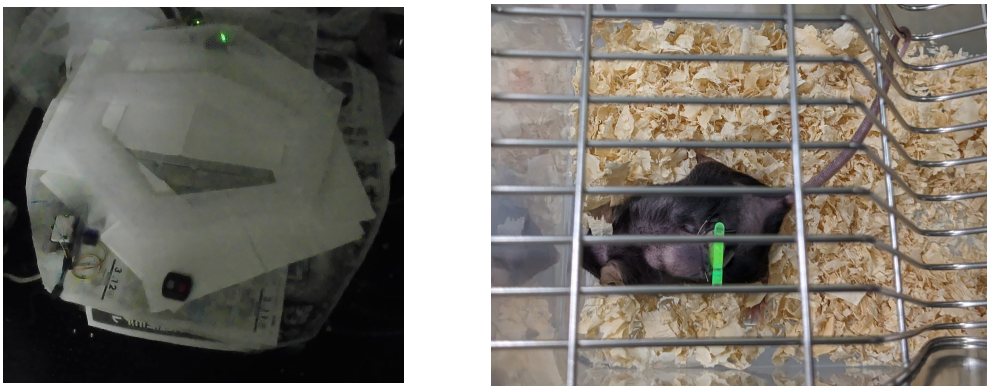
$$x_{t_{i+1}} = x_t + d_{t_{i+1}} \sin(\phi_{t_{i+1}}) \sin(\sum_{t=0}^{t_{i+1}} \Delta\theta_t) \tag{4.2}$$

$$y_{t_{i+1}} = y_t + d_{t_{i+1}} \sin(\phi_{t_{i+1}}) \cos(\sum_{t=0}^{t_{i+1}} \Delta\theta_t) \tag{4.3}$$

$$z_{t_{i+1}} = z_t + d_{t_{i+1}} \cos(\phi_{t_{i+1}}) \tag{4.4}$$

In the weight calculation step, during magnet detection, points closer to the magnet's coordinates (x_m, y_m, z_m) have higher weights, calculated as following:

$$w_{p_i}^t = \frac{1}{||(x_m, y_m, z_m) - (x_{t_{i+1}}, y_{t_{i+1}}, z_{t_{i+1}})||} \tag{4.5}$$



(a) Artificial tunnels for rat to move (b) Data collection device equipped to a rat

Figure 4.5: Settings of data collection

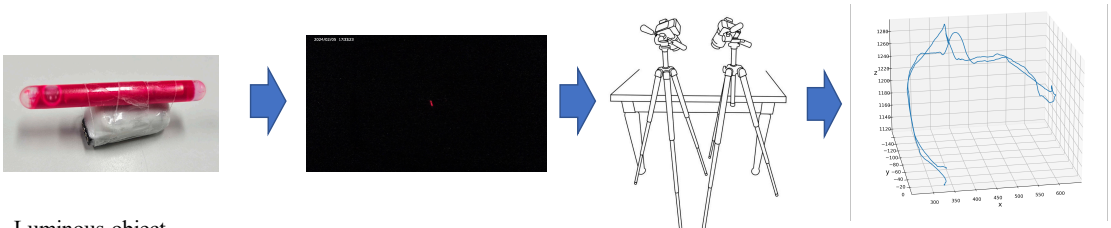


Figure 4.6: The settings of capturing ground truth data by a stereo camera setup.

4.4 Evaluation

4.4.1 Data Collection

We collected data in the environment shown in Fig.4.5 (a). We manually created several cylindrical, translucent plastic hollow tubes, each about 30 cm long, and connected them in a hexagonal pattern. Openings were made on both sides of the hexagonal tube, and two ramps with approximately 15-degree inclines were inserted and connected. For the rat, we glued a necessary tiny sensor onto its back, as shown in Fig.4.5 (b). These sensors collected accelerometer, gyroscope, and magnetometer data at 50 Hz. During data collection, the rat was placed in the tube and allowed to move freely.

To collect ground truth data, we set up a stereo camera system as shown in Fig.4.6

[127]. To allow the rat to move more freely, we simulated a dark environment. Given our use of low-cost cameras, we attached a small fluorescent object, commonly used in night fishing, to the sensor on the rat's back. This way, even in the dark, we could locate the rat using the fluorescent object captured in the video. The video had a frame rate of 30 fps and a resolution of 1080p. The transparent plastic tubes allowed the fluorescent object to be well-captured in the video. We created a simple stereo camera setup using two video cameras placed side by side. The stereo camera system operates by capturing two images from slightly different perspectives, allowing for depth estimation through triangulation. To convert the x and y coordinates from the left and right camera images to 3D coordinates in the real world using a stereo camera setup, we use the following formulas:

$$\begin{aligned} d &= x_L - x_R \\ Z &= \frac{f \cdot B}{d} \\ X &= \frac{(x_L - c_x) \cdot Z}{f} \\ Y &= \frac{(y_L - c_y) \cdot Z}{f} \end{aligned} \tag{4.6}$$

Here, d is the disparity, which is the difference between the x-coordinates in the left and right images. The depth Z is calculated using the focal length of the camera f and the baseline B, which is the distance between the two camera centers. The real-world coordinates X and Y are computed using the x and y coordinates from the left image, the principal point coordinates c_x and c_y, and the calculated depth Z. These equations allow us to convert the 2D coordinates from the left and right images of a stereo camera setup into 3D coordinates in the real world. To convert the coordinates from the stereo camera system to a global reference frame, we use a rigid transformation. Assuming we have two reference points in the environment with known coordinates, we can define the global reference frame. This transformation involves a rotation matrix $\mathbf{R}$ and a translation vector $\mathbf{T}$:

$$\begin{pmatrix} X_g \\ Y_g \\ Z_g \end{pmatrix} = \mathbf{R} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \mathbf{T} \tag{4.7}$$

where (X_g, Y_g, Z_g) are the 3D coordinates in the global reference frame, $\mathbf{R}$ is the 3×3

rotation matrix and $\mathbf{T}$ is the translation vector, accounting for the camera's orientation and position relative to the global frame. Using these principles and formulas, we can accurately calculate the 3D coordinates of the rat's position within the tube for each video frame, providing precise ground truth data for our experiments.

4.4.2 Evaluation Methodology

To evaluate the activity recognition module, which functions as a binary classifier, we used classification accuracy. For the trajectory estimation module, we calculated the mean absolute error (MAE) between the ground truth coordinates and the weighted average of posterior estimates of particles at each time step.

In evaluating the trajectory estimation module, we compared the proposed method against the following methods:

- **3D Neural DR only**: This state-of-the-art neural network-based method uses results from the 3D Neural PDR module, with initial position and direction provided.
- **W/o Activity Recognition**: This variant of the proposed method does not use the results from Activity Recognition module but utilizes Magnet Detection results.
- **W/o Magnet Detection**: This variant of the proposed method does not use results from the Magnet Detection module but utilizes Activity Recognition results.

We used cross-validation to evaluate the methods described. For example, we selected only the last 10 minutes of data collection as the test set, while the remaining data were used as training data for the 3D Neural DR module and the Magnet Detection module as our dataset lasts for around 100 minutes useful data.

4.4.3 Results

The accuracy for the activity recognition module was shown in Table 4.1. As we can see that although the accuracy of proposed model is not very high enough but the recall of inactive classes or true negative rate is really high. The high performance for the inactive class suggests that the dataset might be imbalanced, with more inactive instances than

Table 4.1: classification accuracy of Activity Recognition

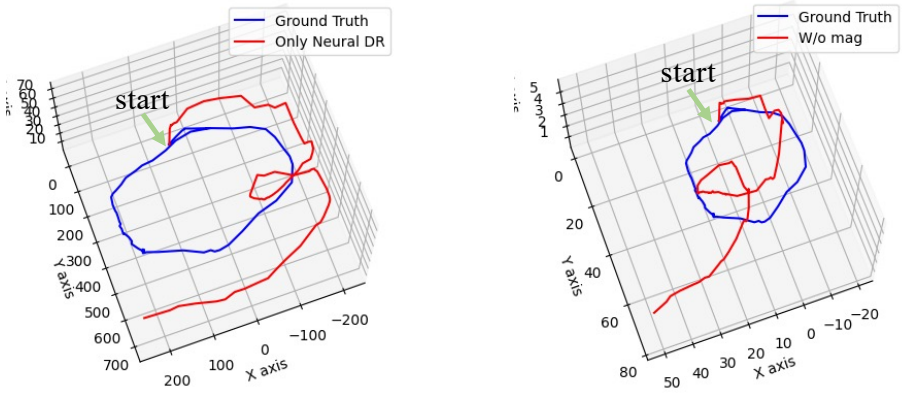
	Accuracy	Recall of Active	Recall of Inactive
proposed method	69.2%	29.4%	84.9%

Table 4.2: Position Errors of Trajectory Estimation for the ablation study

Methods	Averaged MAEs VS. ground truth
3D Neural DR only	14.72cm
W/o Activity Recognition	4.93cm
W/o Magnet Detection	9.14cm
Proposed method	3.42cm

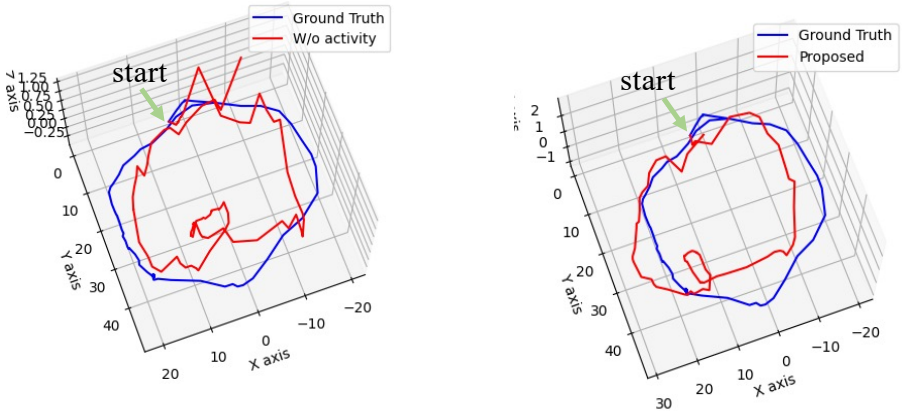
active ones. This imbalance can lead to the model being biased towards predicting the majority class (inactive). Another major reason for this is that when the rat moves, the fluorescent substance might occasionally not be captured simultaneously by both cameras, leading to significant deviations in the generated trajectory and possibly incorrect labeling. Additionally, when the rat is breathing while stationary, the body's movements might be pronounced at certain angles, causing misjudgments. However, for our current research, we prioritize a higher recall for the inactive class. This is because, in the 3D neural DR module, we generate trajectories and angles for both moving and stationary states. Therefore, a low false positive rate and a high recall for the inactive class are more suitable for our study.

Table 4.2 shows the Mean Absolute Errors (MAEs) between the prediction by each method and ground truth for trajectory estimation. As shown, the proposed method significantly outperformed the state-of-the-art 3D neural dead reckoning, as the positioning errors of the 3D neural DR method tend to increase easily. Additionally, the errors without magnet detection are quite large. This is because, even with Activity Recognition module, it is challenging to avoid the cumulative errors that increase during turns. The error magnitude in the method without Activity Recognition is relatively close to the proposed method because we consider 0 distance and 0 directions only when the rat is nearly stationary, which inherently involves minimal movement. However, the results without activity recognition highlight the necessity of the magnet detection module and



(a) Trajectory estimated by 3D Neural DR only

(b) Trajectory estimated by W/o Magnet Detection



(c) Trajectory estimated by W/o Activity Recognition

(d) Trajectory estimated by Proposed

Figure 4.7: Examples of estimated trajectories for one loop example. The red line indicates the predicted path using corresponding methods, while the blue line shows the ground truth..

demonstrate the superiority of our approach in reducing errors by simultaneously utilizing activity recognition and magnet detection.

We also use some examples shown in Fig. 4.7, with all initial positions marked in the 3D plot. The red line in each figure represents a result of the corresponding method, while the blue line is the ground truth. In this case, the rat moved around

a complete loop in the tunnel, passing by two landmarks. We can see that both the proposed method and the method without activity recognition successfully predicted the rat's full loop. In contrast, the other two methods deviated from the ground truth midway due to accumulated drift errors, with the deviations increasing over time. This example demonstrates the necessity of the magnet detection module. Moreover, in this example, the performance of the proposed method and the W/o Activity Recognition method is mainly reflected in the vertical axis, and the proposed method appears smoother shown in Fig.4.7 (a) and (b). The rat is not moving every second. When the rat is stationary, its posture changes can also affect the sensor measurements. This situation can lead to variations in the accelerometer readings, causing misjudgments in the elevation angle. Therefore, our proposed method using the activity recognition module to reduce errors during stationary periods is extremely beneficial.

4.5 Conclusion and Future Work

This study introduced a novel method for estimating trajectories of rats using only inertial sensors. To the best of our knowledge, this is the first approach that combines these technologies to achieve precise estimation with tiny inertial sensors. We developed the 3D Neural DR module and integrated it with particle filtering to estimate the rat's trajectory. Additionally, the activity recognition module accurately classified active and inactive states. By combining magnet detection and activity recognition, our system significantly enhanced trajectory estimation accuracy. Experiments conducted in a controlled environment demonstrated the effectiveness of our approach.

In future work, we aim to address the limitations and complexities associated with real-world environments, specifically focusing on tunnels constructed by rats. Real tunnels exhibit greater variability in size, shape, and width, which can significantly impact the movement and behavior of the rats. To ensure the realism and applicability of our findings, we plan to conduct experiments in natural tunnel environments. This will involve designing and constructing tunnel systems that closely mimic those built by rats in the wild. We will also address the challenges of ensuring smooth navigation for the rats within these tunnels. Furthermore, we will investigate the use of our magnetic landmarks and sensor systems in these real-world settings. By doing so, we can evaluate

the effectiveness and accuracy of our methods in more complex and realistic scenarios. This will help us refine our approach and improve the robustness of our system for practical applications.

Chapter 5

Conclusion and Future Work

5.1 Thesis Summary

In this thesis, we have discussed the methods to enhance object localization systems in GPS-challenged environments.

In Chapter 1, we discussed the significance of localization in GPS-challenged environments and the necessity for landmark-based dead reckoning methods across various settings. We identified three primary challenges that limit the effectiveness of current methods: identifying low-cost, ubiquitous landmarks without the need for extensive site surveys, integrating dead reckoning with new landmarks in artificially structured environments, and implementing landmark-based dead reckoning for underground animals in natural environments. We then developed solutions to address these challenges.

In Chapter 2, we addressed the difficulties of finding ubiquitous, low-cost landmarks and integrating landmark refinement with dead reckoning in indoor environments. Mainstream methods that rely on wireless technologies such as Wi-Fi or Bluetooth require additional hardware installations, which increase costs and necessitate extensive site surveys. Other activity-based methods have low density and cannot ensure continuous and reliable navigation support. Furthermore, current integration methods perform well with landmarks having absolute coordinates but are not suitable for landmarks with broader, less precise areas and usually rely on a known initial position.

To overcome these challenges, we proposed a novel type of indoor landmark based

on GPS signals to correct accumulated errors in IMU-based PDR. These GPS landmarks can be utilized without additional signaling infrastructure. We designed a GPS landmark module that detects when a user is near a window and identifies which window. We tested the PDR system, integrated with the GPS landmark module, using sensor data collected from real-world environments without known initial positions. The experiments demonstrated that the GPS landmark module provides indoor contextual information that effectively corrects PDR errors.

In Chapter 3, we addressed the challenge of creating low-cost and ubiquitous landmarks with high generalizability. Generalizability is defined as the ability to easily integrate with other indoor positioning systems. For example, a signal strength heatmap by wireless technologies are generated for a target floorplan and then fused with other signal strength heatmap like Wi-Fi to provide some additional landmark points where Wi-Fi cannot cover. Also, it is better to generate such signal heatmap without site survey for environments.

To tackle these challenges, we decided to generate signal reception heatmap by predicting GPS satellites reception information. We proposed a method that combines indoor floor plans, outdoor building maps, and satellite positions as inputs to a machine learning-based model to predict GPS satellite signal information at specific indoor locations. We converted these maps into feature maps, which our proposed MSBF-Net then integrates efficiently at different scales using a ladder-like structure. Experiments in real-world environments validated the effectiveness of our feature maps and the MSBF-Net model.

In Chapter 4, we tackled the challenge of implementing dead reckoning for underground animals in natural tunnel systems. Underground environments, with their intricate tunnel systems, require the use of 3D dead reckoning to accurately calculate animal movements in all spatial dimensions, adding complexity compared to traditional 2D dead reckoning. Moreover, these settings lack stable, recognizable landmarks, complicating the recalibration or correction of the positioning system and potentially leading to inaccuracies.

To address these challenges, we proposed a magnet-based dead reckoning system for small underground animals. We explored the application of neural network models to estimate the 3D trajectory of underground animals equipped with tiny inertial

sensors. This model calculates walking distances and directional changes in 3D space, accurately capturing the nuances of animal movements. To enhance the precision of these trajectory estimates, we incorporated magnetic field information into the model. By utilizing a novel correction method based on a particle filter, similar to our approach in Chapter 2, we refined the trajectory predictions further. Experiments conducted in a controlled environment demonstrated the effectiveness of our approach.

Considering the benefits and the challenges addressed by our proposed methods, we believe that our approaches will enhance localization techniques even in GPS-challenged environments such as indoors, underground malls, or even wild underground tunnels. Furthermore, we anticipate that our methods will play a significant role in advancing indoor localization systems and wildlife tracking technologies. For indoor localization, our approach can be combined with current mainstream methods to enhance navigation accuracy in real-time in complex environments such as hospitals and shopping malls, leading to improved operational efficiency and safety. In wildlife tracking, our methods can provide precise but low-cost and long-term monitoring of small animal movements in natural habitats and multiple scenarios, contributing to better understanding of animal behaviors, migration patterns, and habitat utilization.

5.2 Future Work

Through the work in this thesis, we identified the following unresolved issues.

5.2.1 Multiple landmarks used indoor dead reckoning applications

First, we need to integrate the methods from Chapters 2 and 3 to create a fully automated indoor positioning system. This system should allow users to select the relevant building floor, generate a default GPS landmark map, and then combine it with real-time data from the user's smartphone for high-precision localization. To reduce computational costs for real-time positioning on mobile devices, we can incorporate existing landmark technologies such as Bluetooth or Wi-Fi signal heatmaps. By combining multiple landmark sources, we can achieve higher accuracy and better meet commercial demands.

5.2.2 Field test of dead reckoning method in real world underground tunnels

In Chapter 4, we discussed methods for estimating the movements of underground animals using collected data within a controlled environment. However, biologists may require data from real-world scenarios, such as a mole's self-dug tunnel, which is more challenging to access. Therefore, we propose conducting an outdoor field experiment in future research. This experiment will be challenging as we need to find suitable locations for placing signature magnet landmarks and activate the GPS module to correct the trajectory data when the underground animals surface.

Acknowledgment

This thesis encapsulates my academic endeavors at Osaka University and a significant chapter of my life spent studying and residing in Japan over these years. The successful completion of this thesis is attributed to the selfless support and hard work of a number of individuals.

First and foremost, I wish to express my deepest gratitude to my advisor, Associate Professor Takuya Maekawa, who has offered me numerous opportunities since I began my journey in this laboratory. His unwavering support and encouragement were pivotal in my decision to pursue a Ph.D. and in my growth as a researcher. It has been a privilege to be his Ph.D. student. His guidance and mentorship have been indispensable throughout my master's and doctoral studies, and his insightful feedback and encouragement have significantly influenced my research.

I am greatly indebted to my associate advisor, Assistant Professor Daichi Amagata, who has meticulously guided every detail in each step of my research and all the drafts of my papers. Without his kind guidance and valuable advice, it would have been impossible for me to finish my research and this thesis.

I am deeply grateful to my associate advisor, Professor Takahiro Hara, for his generous sharing of time and expertise, and for his invaluable guidance on my research. He also provided me with numerous opportunities, including the JST Fellowship Program, which offered me many enriching experiences. I believe that these experiences and the connections I have made will be valuable assets in the future.

I would like to express my heartfelt appreciation to the members of my thesis committee, Dr. Makoto Onizuka and Dr. Fumio Okura, for their insightful and constructive feedback, which significantly enhanced the quality of my thesis.

I also extend my sincere thanks to Dr. Yihong Zhang, Dr. Daichi Amakata, Dr.

Atsuo Inomata, Dr. Naoto Yanai, Dr. Yuki Arase, Dr. Chuan Xiao, Dr. Yasuyuki Matsushita, Dr. Toru Fujiwara, Dr. Shinji Shimojo, Dr. Susumu Date, Dr. Norihiro Hagita, Dr. Tamami Nakano, Dr. Takahiro Miyashita, and Dr. Satoru Satake for their valuable advice and support during my mid-term defenses and pre-defense.

Special thanks go to Teerawat Kumrai, Ryoma Otsuka, Momoe Sukegawa, Yiming Tian, Xia Qingxin, Naoya Yoshimura, Dissanayake Thilina Madushan, Jaime Morales, Zhi Li, Yuchen Ji, Kei Tanigaki, Guanyu Cao, Yuqiao Wang, and Yang Wang for their stimulating discussions, ideas, and for sharing their research experiences with me.

I am also immensely grateful to the Hara Laboratory's secretary, Makiko Higashinaka, and technical staff, Kana Yasuda, for their assistance in data collection and their support in handling various affairs during my five years in Japan. Without their contributions, my research would not have been possible. Thank you very much.

Additionally, I thank Mr. Masayuki Fujiwara (Daikin Inc.) for his suggestions that highlighted several deficiencies and clarified many aspects during the early stages of my research career.

I would also like to thank all the undergraduate and graduate students in the Hara Lab sensing team for their cooperative and active support throughout my studies.

Last but not least, I would like to express my deepest gratitude to my family. Without their unwavering support, this thesis would not have been possible. Especially to my wife, Jiayu Shen, thank you for your continuous support over the past five years; your companionship has been my driving force.

REFERENCE

[1] Mohinder S Grewal, Lawrence R Weill, and Angus P Andrews. *Global Positioning Systems, Inertial Navigation, and Integration*. John Wiley & Sons, 2007.

[2] Jeffrey Hightower and Gaetano Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57–66, 2001.

[3] Bert Esselink. *A Practical Guide to Localization*, volume 4. John Benjamins Publishing, 2000.

[4] Bernhard Kramer and Angus MacKinnon. Localization: Theory and experiment. *Reports on Progress in Physics*, 56(12):1469, 1993.

[5] Marta Acácio, Philip W Atkinson, João Paulo Silva, and Aldina MA Franco. Performance of GPS/GPRS tracking devices improves with increased fix interval and is not affected by animal deployment. *PLoS One*, 17(3):e0265541, 2022.

[6] Stanley M Tomkiewicz, Mark R Fuller, John G Kie, and Kirk K Bates. Global positioning system and associated technologies in animal behaviour and ecological research. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 365(1550):2163–2176, 2010.

[7] Eli S Bridge, Kasper Thorup, Melissa S Bowlin, Phillip B Chilson, Robert H Diehl, René W Fléron, Phillip Hartl, Roland Kays, Jeffrey F Kelly, W Douglas Robinson, et al. Technology on the move: Recent and forthcoming innovations for tracking migratory birds. *BioScience*, 61(9):689–698, 2011.

[8] Lingxuan Hu and David Evans. Localization for mobile sensor networks. In *Proceedings of the 10th Annual International Conference on Mobile Computing and Networking*, pages 45–57, 2004.

[9] National Coordination Office for Space-Based Positioning, Navigation, and Timing. GPS: The global positioning system. `https://www.gps.gov/systems/gps/`.

[10] Elliott D Kaplan and Christopher Hegarty. *Understanding GPS/GNSS: Principles and Applications*. Artech house, 2017.

[11] Mohamed Tamazin, Malek Karaim, and Aboelmagd Noureldin. GNSSs, signals, and receivers. In Rustam B. Rustamov and Arif M. Hashimov, editors, *Multifunctional Operation and Application of GPS*, chapter 6. IntechOpen, Rijeka, 2018.

[12] Bernhard Hofmann-Wellenhof, Herbert Lichtenegger, and James Collins. *Global Positioning System: Theory and Practice*. Springer Science & Business Media, 2012.

[13] Per K Enge. The global positioning system: Signals, measurements, and performance. *International Journal of Wireless Information Networks*, 1:83–105, 1994.

[14] James Bao-Yen Tsui. *Fundamentals of Global Positioning System Receivers: A Software Approach*. John Wiley & Sons, 2005.

[15] Srdjan Čapkun, Maher Hamdi, and Jean-Pierre Hubaux. GPS-free positioning in mobile ad hoc networks. *Cluster Computing*, 5:157–167, 2002.

[16] Cliff Randell, Chris Djiallis, and Henk L Muller. Personal position measurement using dead reckoning. In *Proceedings of the 7th IEEE International Symposium on Wearable Computers (ISWC)*, volume 3, pages 166–173, 2003.

[17] Oleg Mezentsev, Gerard Lachapelle, and Jussi Collin. Pedstrian dead reckoning—A solution to navigation in GPS signal degraded areas? *GEOMATICA*, 59:175–182, 2005.

[18] Ulrich Steinhoff and Bernt Schiele. Dead reckoning from the pocket-An experimental study. In *Proceedings of the 2010 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 162–170, 2010.

[19] Wonho Kang and Youngnam Han. SmartPDR: Smartphone-based pedestrian dead reckoning for indoor localization. *IEEE Sensors Journal*, 15(5):2906–2916, 2015.

[20] Martin Brossard, Axel Barrau, and Silvère Bonnabel. AI-IMU dead-reckoning. *IEEE Transactions on Intelligent Vehicles*, 5(4):585–595, 2020.

[21] Eric Foxlin. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Computer Graphics and Applications*, (6):38–46, 2005.

[22] Raul Feliz, Eduardo Zalama, and Jaime Gómez-García-Bermejo. Pedestrian tracking using inertial sensors. *Journal of Physical Agents*, 01 2009.

[23] Lauro Ojeda and Johann Borenstein. Personal dead-reckoning system for GPS-denied environments. In *Proceedings of the 2007 IEEE International Workshop on Safety, Security and Rescue Robotics*, pages 1–6, 2007.

[24] Stephane Beauregard and Harald Haas. Pedestrian dead reckoning: A basis for personal positioning. In *Proceedings of the 3rd Workshop on Positioning, Navigation and Communication*, pages 27–35, 2006.

[25] Heba Abdelnasser, Reham Mohamed, Ahmed Elgohary, Moustafa Farid Alzantot, He Wang, Souvik Sen, Romit Roy Choudhury, and Moustafa Youssef. SemanticSLAM: Using environment landmarks for unsupervised indoor localization. *IEEE Transactions on Mobile Computing*, 15(7):1770–1782, 2015.

[26] Beakcheol Jang, Hyunjung Kim, and Jong wook Kim. Survey of landmark-based indoor positioning technologies. *Information Fusion*, 89:166–188, 2023.

[27] Yong Hun Kim, Min Jun Choi, Eung Ju Kim, and Jin Woo Song. Magnetic-map-matching-aided pedestrian navigation using outlier mitigation based on multiple sensors and roughness weighting. *Sensors*, 19(21), 2019.

[28] Xi Wang, Mingxing Jiang, Zhongwen Guo, Naijun Hu, Zhongwei Sun, and Jing Liu. An indoor positioning method for smartphones using landmarks and PDR. *Sensors*, 16(12):2135, 2016.

[29] Fuqiang Gu, Xuke Hu, Milad Ramezani, Debaditya Acharya, Kourosh Khoshelham, Shahrokh Valaee, and Jianga Shang. Indoor localization improved by spatial context—a survey. *ACM Computing Surveys (CSUR)*, 52(3):1–35, 2019.

[30] Paul J Wensveen, Len Thomas, and Patrick JO Miller. A path reconstruction method integrating dead-reckoning and position fixes applied to humpback whales. *Movement Ecology*, 3:1–16, 2015.

[31] Zhenghua Chen, Han Zou, Hao Jiang, Qingchang Zhu, Yeng Chai Soh, and Lihua Xie. Fusion of WiFi, smartphone sensors and landmarks using the kalman filter for indoor localization. *Sensors*, 15(1):715–732, 2015.

[32] Guoliang Chen, Xiaolin Meng, Yunjia Wang, Yanzhe Zhang, Peng Tian, and Huachao Yang. Integrated WiFi/PDR/Smartphone using an unscented kalman filter algorithm for 3D indoor localization. *Sensors*, 15(9):24595–24614, 2015.

[33] Ning Yu, Xiaohong Zhan, Shengnan Zhao, Yinfeng Wu, and Renjian Feng. A precise dead reckoning algorithm based on Bluetooth and multiple sensors. *IEEE Internet of Things Journal*, 5:336–351, 2018.

[34] Daisuke Taniuchi and Takuya Maekawa. Automatic update of indoor location fingerprints with pedestrian dead reckoning. *ACM Transactions on Embedded Computing Systems (TECS)*, 14(2):1–23, 2015.

[35] Michael Hardegger, Daniel Roggen, Sinziana Mazilu, and Gerhard Tröster. ActionSLAM: Using location-related actions as landmarks in pedestrian SLAM. In *Proceedings of the 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–10, 2012.

[36] Fuqiang Gu, Shahrokh Valaee, Kourosh Khoshelham, Jianga Shang, and Rui Zhang. Landmark graph-based indoor localization. *IEEE Internet of Things Journal*, 7(9):8343–8355, 2020.

[37] Fuqiang Gu, Kourosh Khoshelham, Jianga Shang, and Fangwen Yu. Sensory landmarks for indoor localization. In *Proceedings of the Fourth International Conference on Ubiquitous Positioning, Indoor Navigation and Location Based Services (UPINLBS)*, pages 201–206, 2016.

[38] Vivek Chandel, Dibyanshu Jaiswal, and Avik Ghose. InLocW: A reliable indoor tracking and guiding system for smartwatches with path re-routing. In *Proceedings of the 2017 IEEE International Conference on Pervasive Computing and Communications Workshops*, pages 697–702, 2017.

[39] Shun Yoshimi, Kohei Kanagu, Masahiro Mochizuki, Kazuya Murao, and Nobuhiko Nishio. PDR trajectory estimation using pedestrian-space constraints: Real world evaluations. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*, pages 1499–1508, 2015.

[40] Joseph Korpela, Hirokazu Suzuki, Sakiko Matsumoto, Yuichi Mizutani, Masaki Samejima, Takuya Maekawa, Junichi Nakai, and Ken Yoda. Machine learning enables improved runtime and precision for bio-loggers on seabirds. *Communications Biology*, 3(1):633, 2020.

[41] Henri Weimerskirch, Francesco Bonadonna, Frédéric Bailleul, Géraldine Mabille, Giacomo Dell'Omo, and Hans-Peter Lipp. GPS tracking of foraging albatrosses. *Science*, 295(5558):1259–1259, 2002.

[42] Roland Kays, Margaret C Crofoot, Walter Jetz, and Martin Wikelski. Terrestrial animal tracking as an eye on life and planet. *Science*, 348(6240):aaa2478, 2015.

[43] Weizhe Hong, Ann Kennedy, Xavier P Burgos-Artizzu, Moriel Zelikowsky, Santiago G Navonne, Pietro Perona, and David J Anderson. Automated measurement of mouse social behaviors using depth sensing, video tracking, and machine learning. *Proceedings of the National Academy of Sciences*, 112(38):E5351–E5360, 2015.

[44] Mackenzie Weygandt Mathis and Alexander Mathis. Deep learning tools for the measurement of animal behavior in neuroscience. *Current Opinion in Neurobiology*, 60:1–11, 2020.

[45] Mikkel Baun Kjærgaard, Henrik Blunck, Torben Godsk, Thomas Toftkjær, Dan Lund Christensen, and Kaj Grønbæk. Indoor positioning using GPS revisited. In *Proceedings of the 8th International Conference Pervasive Computing*, pages 38–56, 2010.

[46] Masayuki Fujiwara, Tomoya Nakatani, Yiming Tian, Joseph Korpela, Takuya Maekawa, and Takahiro Hara. Smartphone-assisted automatic indoor localization of BLE-enabled appliances using BLE and GNSS signals. In *Proceedings of the 7th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation*, pages 80–89, 2020.

[47] Heng Zhou and Takuya Maekawa. GPS-assisted indoor pedestrian dead reckoning. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)*, 6(4):1–36, 2023.

[48] Heng Zhou and Takuya Maekawa. Preliminary investigation of predicting GPS satellite's signal strength in indoor environments. In *Proceedings of the 37th Annual Conference of the Japanese Society for Artificial Intelligence*, 2023.

[49] Heng Zhou, Momoe Sukegawa, Ryoma Otsuka, Hirotsugu Azechi, Kaoru Ide, Susumu Takahashi, and Takuya Maekawa. Preliminary investigation of estimating 3D trajectory of mice using inertial sensors and magnetic landmark correction. In *Proceedings of the 47th Annual Meeting of the Japan Neuroscience Society (NEURO)*, 2024.

[50] Daisuke Kamisaka, Shigeki Muramatsu, Takeshi Iwamoto, and Hiroyuki Yokoyama. Design and implementation of pedestrian dead reckoning system on a mobile phone. *IEICE Transactions on Information and Systems*, 94-D:1137–1146, 2011.

[51] Namkyoung Lee and Dongsoo Han. Magnetic indoor positioning system using deep neural network. In *Proceedings of the 2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–8, 2017.

[52] Christian Ascher, Christoph Kessler, Matthias Wankerl, and Gert F. Trommer. Dual IMU indoor navigation with particle filter based map-matching on a smartphone. In *Proceedings of the 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–5, 2010.

[53] Yan Huang, Huiru Zheng, Chris Nugent, Paul McCullagh, Suzanne M McDonough, Mark A Tully, and Sean O Connor. Activity monitoring using an intelligent mobile phone: A validation study. In *Proceedings of the 3rd International Conference on Pervasive Technologies Related to Assistive Environments*, pages 1–6, 2010.

[54] Jim Scarlett. Enhancing the performance of pedometers using a single accelerometer. *Application Note, Analog Devices*, 41, 2007.

[55] Quentin Ladetto and Bertrand Merminod. An alternative approach to vision techniques-pedestrian navigation system based on digital magnetic compass and gyroscope integration. In *Proceedings of the 6th World Multiconference on Systemics, Cybernetics and Information*, 2002.

[56] Fatemeh Abyarjoo, Armando Barreto, Jonathan Cofino, and Francisco R. Ortega. Implementing a sensor fusion algorithm for 3D orientation detection with inertial/magnetic sensors. In *Proceedings of the Innovations and Advances in Computing, Informatics, Systems Sciences, Networking and Engineering*, pages 305–310, 2015.

[57] Changhao Chen, Xiaoxuan Lu, Andrew Markham, and Niki Trigoni. Ionet: Learning to cure the curse of drift in inertial odometry. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[58] Takuto Yoshida, Junto Nozaki, Kenta Urano, Kei Hiroi, Katsuhiko Kaji, Takuro Yonezawa, and Nobuo Kawaguchi. Sampling rate dependency in pedestrian

walking speed estimation using DualCNN-LSTM. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*, pages 862–868, 2019.

[59] Changhao Chen, Peijun Zhao, Chris Xiaoxuan Lu, Wei Wang, Andrew Markham, and Niki Trigoni. Deep-learning-based pedestrian inertial navigation: Methods, data set, and on-device inference. *IEEE Internet of Things Journal*, 7(5):4431–4441, 2020.

[60] Sheng Guo, Xiong Hanjiang, Xianwei Zheng, and Yan Zhou. Activity recognition and semantic description for indoor mobile localization. *Sensors*, 17:649, 03 2017.

[61] He Wang, Souvik Sen, Ahmed Elgohary, Moustafa Farid, Moustafa Youssef, and Romit Roy Choudhury. No need to war-drive: Unsupervised indoor localization. In *Proceedings of the 10th International Conference on Mobile systems, Applications, and Services*, pages 197–210, 2012.

[62] Yuan Zhuang, Jun Yang, You Li, Longning Qi, and Naser El-Sheimy. Smartphone-based indoor localization with bluetooth low energy beacons. *Sensors*, 16(5):596, 2016.

[63] Sakshi Juneja and Sharda Vashisth. Indoor positioning system using visible light communication. In *Proceedings of the First International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN)*, pages 79–83, 2017.

[64] Ye-Sheng Kuo, Pat Pannuto, Ko-Jen Hsiao, and Prabal Dutta. Luxapose: Indoor positioning with mobile phones and visible light. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking*, pages 447–458, 2014.

[65] Baoding Zhou, Qingquan Li, Qingzhou Mao, Wei Tu, Xing Zhang, and Long Chen. ALIMC: Activity landmark-based indoor mapping via crowdsourcing.

IEEE Transactions on Intelligent Transportation Systems, 16(5):2774–2785, 2015.

[66] Myeongcheol Kwak, Youngmong Park, Junyoung Kim, Jinyoung Han, and Taekyoung Kwon. An energy-efficient and lightweight indoor localization system for Internet-of-Things (IoT) environments. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)*, 2018.

[67] Qun Niu, Ning Liu, Jianjun Huang, Yangze Luo, Suining He, Tao He, S.-H. Gary Chan, and Xiaonan Luo. DeepNavi: A deep signal-fusion framework for accurate and applicable indoor navigation. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)*, 2019.

[68] Raghav H. Venkatnarayan and Muhammad Shahzad. Enhancing indoor inertial odometry with WiFi. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT)*, 3(2), 2019.

[69] Marcus Edel and Enrico Köppe. An advanced method for pedestrian dead reckoning using BLSTM-RNNs. In *Proceedings of the 2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–6, 2015.

[70] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*, 2015.

[71] Fredrik Gustafsson, Fredrik Gunnarsson, Niclas Bergman, Urban Forssell, Jonas Jansson, Rickard Karlsson, and Per-Johan Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on Signal Processing*, 50:425–437, 2 2002.

[72] Hiroaki Santo, Takuya Maekawa, and Yasuyuki Matsushita. Device-free and privacy preserving indoor positioning using infrared retro-reflection imaging. In *Proceedings of the 2017 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 141–152, 2017.

[73] Kazuya Ohara, Takuya Maekawa, Yasue Kishino, Yoshinari Shirai, and Futoshi Naya. Transferring positioning model for device-free passive indoor localization. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp)*, pages 885–896, 2015.

[74] Google. ARCore. https://developers.google.com/ar/, 2018.

[75] Isaac Amundson and Xenofon D Koutsoukos. A survey on localization for mobile wireless sensor networks. In *Proceedings of the Second International Workshop on Mobile Entity Localization and Tracking in GPS-less Environments*, pages 235–254, 2009.

[76] Mai A Al-Ammar, Suheer Alhadhrami, Abdulmalik Al-Salman, Abdulrahman Alarifi, Hend S Al-Khalifa, Ahmad Alnafessah, and Mansour Alsaleh. Comparative survey of indoor positioning technologies, techniques, and algorithms. In *Proceedings of the 2014 International Conference on Cyberworlds*, pages 245–252, 2014.

[77] Zheng Yang, Chenshu Wu, Zimu Zhou, Xinglin Zhang, Xu Wang, and Yunhao Liu. Mobility increases localizability: A survey on wireless indoor localization using inertial sensors. *ACM Computering Surveys*, 47(3), 2015.

[78] Suining He and S-H Gary Chan. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Communications Surveys & Tutorials*, 18(1):466–490, 2015.

[79] Ali Khalajmehrabadi, Nikolaos Gatsis, and David Akopian. Modern WLAN fingerprinting indoor positioning methods and deployment challenges. *IEEE Communications Surveys & Tutorials*, 19(3):1974–2002, 2017.

[80] CelesTrak. CelesTrak: Satellite Tracking and Space Situational Awareness. `https://celestrak.org/`, 2023.

[81] Google Earth. Google Earth: Explore the World from Above. `https://earth.google.com/web`, 2023.

[82] Hongyu Zhu, Linyuan Xia, Dongjin Wu, Jingchao Xia, and Qianxia Li. Study on multi-GNSS precise point positioning performance with adverse effects of satellite signals on android smartphone. *Sensors*, 20(22), 2020.

[83] Andria Bilich, Penina Axelrad, and Kristine M Larson. Scientific utility of the signal-to-noise ratio (SNR) reported by geodetic GPS receivers. In *Proceedings of the 20th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS)*, pages 26–28, 2007.

[84] Penina Axelrad, Kristine Larson, and Brandon Jones. Use of the correct satellite repeat period to characterize and reduce site-specific multipath errors. In *Proceedings of the 18th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GNSS)*, pages 2638–2648, 2005.

[85] Wolfgang Lechner and Stefan Baumann. Global navigation satellite systems. *Computers and Electronics in Agriculture*, 25(1-2):67–85, 2000.

[86] Xingxing Li, Xiaohong Zhang, Xiaodong Ren, Mathias Fritsche, Jens Wickert, and Harald Schuh. Precise positioning with current multi-constellation global navigation satellite systems: GPS, GLONASS, Galileo and Beidou. *Scientific Reports*, 5(1), 2015.

[87] Ting-Hua Yi, Hong-Nan Li, and Ming Gu. Effect of different construction materials on propagation of GPS monitoring signals. *Measurement*, 45(5):1126–1139, 2012.

[88] Dinesh Sathyamoorthy, S Shalini, Mohamad Zainal, A Siti, A Robiah, I Idris, M Mohd, and Hasrol Hisam. Evaluation of the effect of commonly used materials on multipath propagation of global positioning system (GPS) signals via GPS simulation. *Advances in Military Technology*, 9:81–95, 06 2014.

[89] Henrik Blunck, Mikkel Baun Kjærgaard, Torben Godsk, Thomas Toftkjær, Dan Lund Christensen, and Kaj Grønbæk. Empirical analysis and characterization of indoor gps signal fading and multipath conditions. In *Proceedings of the 22nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS)*, pages 2362–2371, 2009.

[90] Frank van Diggelen. Indoor GPS theory & implementation. In *Proceedings of the 2002 IEEE Position Location and Navigation Symposium*, pages 240–247, 2002.

[91] Olcay Yiğit, Havva Bilişik Bakbak, Eren Demir, Radosveta Sokullu, and Korkut Yeğin. GPS signal channel modeling and verification. *Procedia Computer Science*, 113:621–626, 09 2017.

[92] Philip RR Strode and Paul D Groves. GNSS multipath detection using three-frequency signal-to-noise measurements. *GPS Solutions*, 20:399–412, 2016.

[93] Tao Hu, Gérard Lachapelle, and Richard Klukas. Indoor GPS signal replication using a hardware simulator. In *Proceedings of the 18th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS)*, pages 2728–2742, 2005.

[94] Weisong Wen, Guohao Zhang, and Li-Ta Hsu. Correcting GNSS NLOS by 3D LiDAR and building height. In *Proceedings of the 31st International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS)*, pages 3156–3168, 2018.

[95] B. Friebel, M. Schweins, N. Dreyer, and T. Kürner. Simulation of GPS localisation based on ray tracing. *Advances in Radio Science*, 19:85–92, 2021.

[96] Martin Jacob, S Schon, Ulrich Weinbach, and T Kurner. Ray tracing supported precision evaluation for GPS indoor positioning. In *Proceedings of the 6th Workshop on Positioning, Navigation and Communication*, pages 15–22, 2009.

[97] Erwin Löhnert, Wolfgang Bär, Eckart Göhler, and Jochen Möllmer. Galileo/GPS indoor navigation & positioning for SAR and tracking applications. In *Proceedings of the 2010 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 1–6, 2010.

[98] Pengfei Zhou, Yuanqing Zheng, Zhenjiang Li, Mo Li, and Guobin Shen. Iodetector: A generic service for indoor outdoor detection. In *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems*, pages 113–126, 2012.

[99] Van Bui, Nam Tuan Le, Thanh Luan Vu, Van Hoa Nguyen, and Yeong Min Jang. GPS-based indoor/outdoor detection scheme using machine learning techniques. *Applied Sciences*, 10(2):500, 2020.

[100] Masayuki Ochiai, Masahiro Fujii, Atsushi Ito, Yu Watanabe, and Hiroyuki Hatano. A study on indoor position estimation based on fingerprinting using GPS signals. In *Proceedings of the 2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, pages 727–728, 2014.

[101] Masahiro Fujii and Yoshiaki Mori. A study on indoor positioning systems based on SkyPlot mask. In *Proceedings of the IEEE 5th Global Conference on Consumer Electronics*, pages 1–2, 2016.

[102] Oliver Montenbruck and Eberhard Gill. Real-time estimation of sgp4 orbital elements from gps navigation data. In *Proceedings of the 15th International Symposium Space Flight Dynamics*, pages 26–30, 2000.

[103] Bui Tuong Phong. Illumination for computer generated pictures. In *Seminal Graphics: Pioneering Efforts that Shaped the Field*, pages 95–101. 1998.

[104] B. W. Parkinson and J. J. Spilker. *Global Positioning System: Theory and Applications*. Progress in Astronautics and Aeronautics, 1996.

[105] Geospatial Information Authority of Japan. Geospatial Information Authority of Japan. `https://www.gsi.go.jp/`, 2023.

[106] Hans R Widmer, Hans Hoppeler, Eviatar Nevo, C Richard Taylor, and Ewald R Weibel. Working underground: Respiratory adaptations in the blind mole rat. *Proceedings of the National Academy of Sciences*, 94(5):2062–2067, 1997.

[107] Hannah J Williams, Lucy A Taylor, Simon Benhamou, Allert I Bijleveld, Thomas A Clay, Sophie de Grissac, Urška Demšar, Holly M English, Novella Franconi, Agustina Gómez-Laich, et al. Optimizing the use of biologgers for movement ecology research. *Journal of Animal Ecology*, 89(1):186–206, 2020.

[108] Kei Tanigaki, Ryoma Otsuka, Aiyi Li, Yota Hatano, Yuanzhou Wei, Shiho Koyama, Ken Yoda, and Takuya Maekawa. Automatic recording of rare behaviors of wild animals using video bio-loggers with on-board light-weight outlier detector. *PNAS Nexus*, 3(1):pgad447, 2024.

[109] Valery U Zavorotny, Kristine M Larson, John J Braun, Eric E Small, Ethan D Gutmann, and Andria L Bilich. A physical model for GPS multipath caused by land reflections: Toward bare soil moisture retrievals. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 3(1):100–110, 2009.

[110] Anthony I Dell, John A Bender, Kristin Branson, Iain D Couzin, Gonzalo G de Polavieja, Lucas PJJ Noldus, Alfonso Pérez-Escudero, Pietro Perona, Andrew D Straw, Martin Wikelski, et al. Automated image-based tracking and its application in ecology. *Trends in Ecology & Evolution*, 29(7):417–428, 2014.

[111] Alfonso Pérez-Escudero, Julián Vicente-Page, Robert C Hinz, Sara Arganda, and Gonzalo G De Polavieja. idTracker: Tracking individuals in a group by automatic identification of unmarked animals. *Nature Methods*, 11(7):743–748, 2014.

[112] Samuel Jackson Barnett. Magnetization by rotation. *Physical Review*, 6(4):239, 1915.

[113] Luca Catarinucci, Riccardo Colella, Luca Mainetti, Vincenzo Mighali, Luigi Patrono, Ilaria Sergi, and Luciano Tarricone. An innovative animals tracking system based on passive UHF RFID technology. In *Proceedings of the 20th International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, pages 1–7, 2012.

[114] Luca Catarinucci, Riccardo Colella, Luca Mainetti, Vincenzo Mighali, Luigi Patrono, Ilaria Sergi, and Luciano Tarricone. Performance evaluation of a novel animals tracking system based on UHF RFID technology. *Journal of Communications Software and Systems*, 9(1):4–13, 2013.

[115] Alison Matthews, Laura Ruykys, Bill Ellis, Sean FitzGibbon, Daniel Lunney, Mathew S Crowther, Alistair S Glen, Brad Purcell, Katherine Moseby, Jenny

Stott, et al. The success of GPS collar deployments on mammals in australia. *Australian Mammalogy*, 35(1):65–83, 2013.

[116] Magnus Andersen, Andrew E Derocher, Øystein Wiig, and Jon Aars. Movements of two Svalbard polar bears recorded using geographical positioning system satellite transmitters. *Polar Biology*, 31:905–911, 2008.

[117] LF Bandeira de Melo, MA Lima Sabato, EM Vaz Magni, RJ Young, and CM Coelho. Secret lives of maned wolves (Chrysocyon brachyurus Illiger 1815): as revealed by GPS tracking collars. *Journal of Zoology*, 271(1):27–36, 2007.

[118] Troels Gregersen, Timm A Wild, Linnea Worsøe Havmøller, Peter Rask Møller, Torben Anker Lenau, Martin Wikelski, and Rasmus Worsøe Havmøller. A novel kinetic energy harvesting system for lifetime deployments of wildlife trackers. *PloS One*, 18(5):e0285930, 2023.

[119] Fritz A Francisco, Paul Nührenberg, and Alex Jordan. High-resolution, non-invasive animal tracking and reconstruction of local environment in aquatic ecosystems. *Movement Ecology*, 8:1–12, 2020.

[120] Filip Naiser, Matěj Šmíd, and Jiří Matas. Tracking and re-identification system for multiple laboratory animals. In *Proceedings of the Visual Observation and Analysis of Vertebrate and Insect Behavior Workshop at the 24th International Conference on Pattern Recognition (ICPR)*, 2018.

[121] Tristan Walter and Iain D Couzin. Trex, a fast multi-animal tracking system with markerless identification, and 2D estimation of posture and visual fields. *Elife*, 10:e64000, 2021.

[122] Emily K Studd, Rachael E Derbyshire, Allyson K Menzies, John F Simms, Murray M Humphries, Dennis L Murray, and Stan Boutin. The Purr-fect Catch: Using accelerometers and audio recorders to document kill rates and hunting behaviour of a small prey specialist. *Methods in Ecology and Evolution*, 12(7):1277–1287, 2021.

[123] Gaelle Fehlmann, M Justin O'Riain, Phil W Hopkins, Jack O'Sullivan, Mark D Holton, Emily LC Shepard, and Andrew J King. Identification of behaviours from accelerometer data in a wild social primate. *Animal Biotelemetry*, 5:1–11, 2017.

[124] Natasa Kleanthous, Abir Hussain, Wasiq Khan, Jennifer Sneddon, and Panos Liatsis. Deep transfer learning in sheep activity recognition using accelerometer data. *Expert Systems with Applications*, 207:117925, 2022.

[125] Liang An, Jilong Ren, Tao Yu, Tang Hai, Yichang Jia, and Yebin Liu. Three-dimensional surface motion capture of multiple freely moving pigs using MAMMAL. *Nature Communications*, 14(1):7727, 2023.

[126] Vassilis M Papadakis, Ioannis E Papadakis, Fani Lamprianidou, Alexios Glaropoulos, and Maroudio Kentouri. A computer-vision system and methodology for the analysis of fish behavior. *Aquacultural Engineering*, 46:53–59, 2012.

[127] Stephen T Barnard and Martin A Fischler. Computational stereo. *ACM Computing Surveys (CSUR)*, 14(4):553–572, 1982.