On the Efficient Design of Scalable Indoor Positioning Systems Based on Wi-Fi Fingerprinting

Emad Ali Abdalla Ebaid,
School of Computing and Communications
Lancaster University

A thesis submitted for the degree of
Doctor of Philosophy

August 2024

Declaration

I declare that the work presented in this thesis is, to the best of my knowledge and belief, original and my own work. The material has not been submitted, either in whole or in part, for a degree at this, or any other university. This thesis does not exceed the maximum permitted word length of 80,000 words including appendices and footnotes, but excluding the bibliography. A rough estimate of the word count is: 48561

Emad Ebaid
August 2024

On the Efficient Design of Scalable Indoor Positioning Systems Based on Wi-Fi Fingerprinting

Emad Ali Abdalla Ebaid, .
School of Computing and Communications, Lancaster University
A thesis submitted for the degree of *Doctor of Philosophy*. August 2024

Abstract

This thesis investigates the design and implementation of Wi-Fi fingerprinting-based Indoor Positioning Systems (IPS), with a focus on enhancing their efficiency, scalability, and accuracy. Wi-Fi fingerprinting, particularly utilising Received Signal Strength Indicator (RSSI) data, offers a cost-effective and non-intrusive method for indoor positioning. Despite its advantages, existing systems encounter challenges such as high computational complexity, the need for frequent manual updates, and difficulties in managing large datasets.

The research commences by evaluating various position estimation algorithms, including k-Nearest Neighbour (k-NN) and its weighted variant (Wk-NN), identifying the correlation distance function as a highly effective approach when combined with exponential data representation. This combination was found to balance accuracy with computational simplicity, making it a viable option for efficient IPS.

To address scalability and reliability, the thesis introduces a cloud-based Indoor Positioning System (CB-IPS) framework that leverages cloud computing, edge computing, and cache technologies. This framework significantly enhances the management of large fingerprint databases, optimises computational resources, and supports real-time processing, thereby improving the overall performance of the IPS.

Furthermore, the research addresses the complexity of database management by implementing data preprocessing techniques, dimensionality reduction through Principal Component Analysis (PCA), and auto-update mechanisms. These strategies effectively reduce computational load and storage requirements, thereby ensuring that the system remains scalable and efficient.

The findings demonstrate that the proposed optimisations can substantially enhance the performance of Wi-Fi fingerprinting-based IPS, making them more competitive with state-of-the-art systems. The research contributes to the advancement of indoor positioning technologies, offering practical solutions that address current limitations while laying the foundation for future innovations.

This thesis concludes by outlining potential directions for future research, including the integration of advanced machine learning techniques, and further optimisation of real-time implementations. These efforts are essential for fully realising the potential of Wi-Fi fingerprinting-based indoor positioning systems across various real-world applications.

Publications

E. Ebaid and K. Navaie, “Optimum NN Algorithms Parameters on the UJIIndoorLoc for Wi-Fi Fingerprinting Indoor Positioning Systems,” Wellington, New Zealand: IEEE, Dec. 2022, pp. 280–286, ISBN: 978-1-6654-7104-6. DOI: 10.1109/ITNAC55475.2022.9998385

E. Ebaid and K. Navaie, “Efficient Design of Scalable Indoor Positioning System Based on Wi-Fi Fingerprinting,” in *Proceedings of the CPS Summer School PhD Workshop 2022 co-located with 4th Edition of the CPS Summer School (CPS 2022), Pula, Sardinia (Italy), September 19-23, 2022*, R. L. López, D. M. Quintín, F. Palumbo, *et al.*, Eds., ser. CEUR Workshop Proceedings, vol. 3252, CEUR-WS.org, Sep. 2022. [Online]. Available: https://ceur-ws.org/Vol-3252/short1.pdf

Acknowledgements

This doctoral journey has been shaped and enriched by the generosity and support of many people, to whom I owe my deepest gratitude.

My profound thanks go to my PhD supervisor, **Professor Keivan Navaie**, for his exceptional guidance, patience, and generosity. His unwavering support, insightful advice, and belief in my work have been instrumental to my academic growth and personal development.

I am grateful to my internal and external examiners for their thoughtful engagement, constructive feedback, and valuable suggestions that strengthened this thesis.

Thanks to the **School of Computing and Communications**, Lancaster University, for providing a stimulating and supportive environment, and to colleagues and all administrative and technical staff for their collegiality, assistance, and many contributions.

I also extend heartfelt gratitude to the **Libyan Embassy** in London for sponsoring my doctoral studies and for their commitment to advancing higher education and research.

Above all, my deepest thanks go to my family. I cherish the memory of my beloved father, grandmother, and grandfather, whose encouragement and belief in me remain a constant source of strength.
To my mother — your love, sacrifices, and prayers have sustained me through every challenge.
To my brothers — your humour and solidarity have been a source of comfort and motivation.
To my wife — your patience, courage, and faith have been my anchor.
To my children — your laughter and light have lifted me through the steepest climbs.

This thesis stands as much on your love and faith as on my own efforts.

Contents

List of Figures

List of Tables

List of Abbreviations

2D: Two-dimensional
3D: Three-dimensional
AoI: Area of Interest
AP: Access Point
BLE: Bluetooth Low Energy
CB-IPS: Cloud-based Indoor Positioning System
CC: Cloud Computing
CDF: Cumulative Distribution Function
CNN: Convolutional Neural Network
CFR: Channel Frequency Response
CSI: Channel State Information
DCT: Discrete Cosine Transform
DFL: Device-free Localisation
DNN: Deep Neural Network
EC: Edge Computing
EKF: Extended Kalman Filter
FC: Fog Computing
FTM: Fine Time Measurement
GNSS: Global Navigation Satellite System
GPS: Global Positioning System
HPC: High-Performance Computing
HR: Hit Rate
IaaS: Infrastructure as a Service
IEEE: Institution of Electrical and Electronic Engineering
IMU: Inertial Measuring Units
INS: Inertial Navigation System
IoT: Internet of Things
IPS: Indoor Positioning System
IR: Infrared
k-NN: k-Nearest Neighbor
KL: Kullback-Leibler
LBS: Location-based Service
MAC: Medium Access Control
MAE: Mean Absolute Error
MC: Mist Computing
MD: Mobile Device
MdE: Median Error

ML: Machine Learning
MSE: Mean Squared Error
NDFL: Non-device-free Localisation
NIC: Networking Interface Card
NLOS: Non-Line-Of-Sight
NN: Nearest Neighbor
OFDM: Orthogonal Frequency-Division Multiplexing
OSI: Open System Interconnection
PaaS: Platform as a Service
PCA: Principal Component Analysis
PS: Positioning Server
RF: Radio Frequency
RFID: Radio Frequency Identifier
RMF: Radio Map Fingerprinting
RMSE: Root Mean Squared Error
RNN: Recurrent Neural Network
RN: Reference Node
RP: Reference Point
RSS: Received Signal Strength
RSSI: Received Signal Strength Indicator
RTT: Round-Trip Time
SaaS: Software as a Service
SLA: Service Level Agreement
SLAM: Simultaneous Localization and Mapping
SVM: Support Vector Machine
TCP: Transmission Control Protocol
ToA: Time of Arrival
UI: User Interface
UWB: Ultra Wideband
VLC: Visible Light Communication
VM: Virtual Machines
Wi-Fi: Wireless Fidelity, IEEE 802.11 Wireless LAN
Wk-NN: Weighted k-Nearest Neighbor
WLAN: Wireless LAN
WSN: Wireless Sensor Networks

Chapter 1

Introduction

1.1 Overview

Positioning and navigation services have become increasingly important in our daily lives. Currently, the accessibility and affordability of Global Navigation Satellite System (GNSS) technology for outdoor positioning and navigation are at their peak. The GNSS services, such as Global Positioning System (GPS) technology, used in smartphones, provide an accurate position within approximately 4.9 metres with 95% probability in outdoor environments [3]. However, in indoor environments, GPS cannot function properly due to obstacles that block GPS Radio Frequency (RF) signals from penetrating walls and objects inside buildings. Therefore, alternative technology is needed to replicate GPS in indoor environments with greater accuracy than can be achieved outdoors. This has led many academic and industrial researchers over the past decades to work out how to emulate GPS in an indoor setting. These efforts have resulted in an Indoor Positioning System (IPS), as illustrated in Figure 1.1.

An IPS is a system that can determine the position of the object within a building or in a specific coordinate system [4]. The determination of positioning relies on different methods, which differ based on the technologies employed. These technologies include RF, Optical, Magnetic, and Acoustic methods [5]. However, RF-based technology is commonly used in IPS because radio frequencies can penetrate walls to provide a broader coverage area compared to other technologies. It also has lower-cost hardware, as it often utilises existing infrastructure such as a Wireless Local Area Network (WLAN). Inside indoor spaces, indoor positioning has multi-dimensional challenges such as signal problems, limited infrastructure, and a lack of maps. However, signal problems are a major challenge for IPS RF-based systems. Issues such as signal multipath and attenuation significantly degrade indoor positioning for RF-based systems [6].

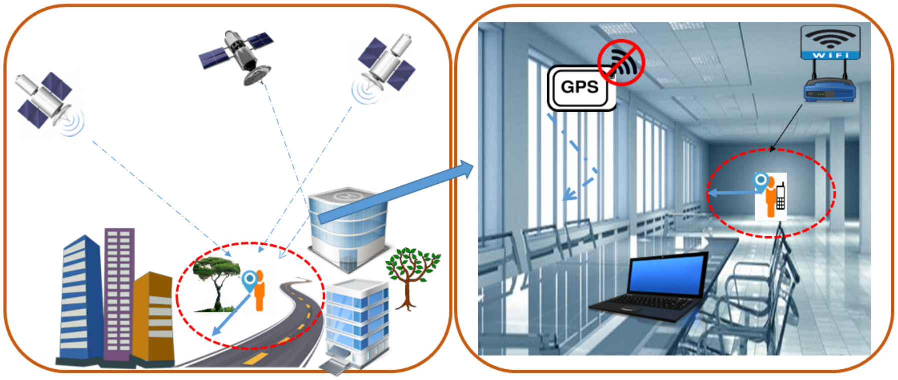


Figure 1.1: Positioning environment GPS vs. IPS

1.2 Motivation

Over the past decade, there has been a substantial increase in the investigation of Indoor Positioning Systems (IPS), with researchers predominantly focusing on various Radio Frequency (RF) technologies [7]. Among these notable technologies are Wi-Fi [8], Bluetooth [9], Ultra-Wide Band (UWB) [10], Visible Light Communication (VLC) [11], Radio Frequency Identification (RFID) [12], and ZigBee [13]. RF-based technologies provide extensive coverage at a relatively low cost, often utilising existing infrastructure such as Wireless Local Area Networks (WLAN). Furthermore, cost-effective methodologies such as Pedestrian Dead Reckoning (PDR) and Inertial Navigation Systems (INS) are frequently integrated [14]. Despite the plethora of

technologies available, only a select few have been identified as optimal solutions for indoor positioning, each presenting distinct advantages and limitations. The pursuit of an enhanced IPS solution has driven researchers to explore hybrid systems that amalgamate multiple technologies through data fusion techniques. Although promising, the implementation of a hybrid approach introduces challenges such as increased costs and complexity [6]. Consequently, numerous proposed IPS systems prefer the simplicity and cost-effectiveness of a single technology. Wi-Fi technology,

widely prevalent and integrated into everyday devices, has emerged as a fundamental enabler for IPS. By leveraging the IEEE 802.11 standard, Wi-Fi chips and access points facilitate the measurement of RF signal intensity, enabling users to ascertain their indoor location using devices such as mobile phones via the Wi-Fi indoor positioning system, thus obviating the need for additional infrastructure costs [7]. For this reason, Wi-Fi fingerprinting is highly esteemed for indoor positioning. This technique offers superior accuracy, scalability, and practicality compared to other indoor positioning solutions such as beacons or RFID tags. The notable advantages of Wi-Fi fingerprinting include:

- **High accuracy:** Wi-Fi fingerprinting can achieve a high degree of accuracy by comparing the current Received Signal Strength Indicator (RSSI) measurements of a device to a pre-collected database of fingerprints, thereby enabling precise location estimation even in complex indoor environments.
- **Cost-effective:** Wi-Fi fingerprinting represents a cost-effective solution for indoor positioning, as it obviates the need for additional hardware such as beacons or RFID tags.
- **Flexibility:** Wi-Fi fingerprinting is compatible with a wide array of devices, including smartphones, tablets, laptops, and other Wi-Fi-enabled devices.
- **Scalability:** Wi-Fi fingerprinting can be effortlessly scaled to encompass larger environments by collecting fingerprints at additional locations and augmenting the number of Access Points (APs).
- **Robustness:** Wi-Fi fingerprinting is resilient to environmental changes, such as furniture reconfigurations or human presence, which can adversely affect other methods like trilateration or triangulation.
- **Privacy:** Wi-Fi fingerprinting can be performed without disclosing personal information, as it solely utilises Wi-Fi signals that can be collected anonymously.

1.3 Research Problem and Questions

Indoor positioning using Wi-Fi technology offers various methods for determining location. Each technology possesses unique indoor positioning capabilities, broadly classified into two models: a 2D model for Bluetooth, ZigBee, and Wi-Fi, and a 3D model for Infrared, UWB, and Ultrasonic technologies. The 2D model, particularly prevalent in Wi-Fi, incurs lower infrastructure costs compared to the 3D model. Despite the 3D model's enhanced accuracy, its implementation cost limits its widespread adoption. Consequently, the 2D model, particularly Wi-Fi using RSSI fingerprinting, remains a widely favoured choice[8]. This research focuses on Wi-Fi and RSSI-based methods that utilise fingerprinting techniques within the 2D model.

Compared to other positioning systems, Wi-Fi fingerprint positioning has the advantages of being low-cost and highly accurate. However, the technology currently requires computationally intensive algorithms for estimating positioning, leading to a relatively long processing time and high algorithmic complexity. Additionally, this method requires data for predefined initial positions and effective database management. Furthermore, maintaining the fingerprinting database each time the environment changes (such as changes in furniture or access points) requires manual calibration to update the fingerprinting. Although previous approaches have attempted to make fingerprinting efficient for meeting IPS requirements at low cost and with less complexity, no system has yet achieved this goal. This is because most existing systems have not considered the quality of their system design or their integration with the infrastructure used, which is crucial for optimal IPS performance.

The advancement of IPS using Wi-Fi technology, particularly fingerprinting techniques based on RSSI within the 2D model, faces significant challenges despite its widespread use and advantages in cost-effectiveness and accuracy. Although RSSI-based methods eliminate the need for additional hardware and are immune to multipath signals, they currently rely on computationally intensive algorithms for position estimation, require manual calibration for database updates, and lack effective strategies for managing large databases generated during the fingerprinting process. Furthermore, existing systems often overlook the integration of design factors within the IPS architecture and positioning framework, which is crucial for optimal performance.

Wi-Fi fingerprint-based approaches are preferred by researchers for several reasons. Firstly, most large buildings come equipped with WLAN services for wireless network coverage, which means that no additional hardware or costs are needed, making them suitable for IPS. Secondly, as mobile and wireless receivers already contain networking interface cards (NICs), they can readily measure Received Signal Strength (RSS) values. Thirdly, path loss modelling might work well under normal circumstances, but proves short-lived inside buildings due to complex signal propagation, which causes RSS signals to fluctuate irrespective of the environment or time.

This research investigates a Wi-Fi-based indoor positioning framework. Specifically, RSSI fingerprinting-based localisation algorithms and techniques are proposed. We assume that examining the system as a whole will lead to improvements in the design of IPS Wi-Fi RSSI-based systems compared to current state-of-the-art systems. This is because the design factors of the IPS architecture and the positioning framework significantly impact the performance of Wi-Fi RSSI-based positioning.

Research Questions

1. *What is the most efficient and yet least complex position estimation algorithm suitable for the IPS based on Wi-Fi Fingerprinting techniques?*

2. *How can a Wi-Fi Fingerprinting-based IPS achieve scalable and reliable performance, making it simple, efficient, and competitive with state-of-the-art systems while ensuring acceptable positioning services?*

3. *How can cloud architectures be utilised to maintain the required accuracy, privacy, and response time while providing scalability to the IPS?*

4. *What approaches can be employed to enhance the positioning accuracy and scalability of Wi-Fi RSSI-based systems, with a focus on simplifying database fingerprinting complexity using an edge-computing architecture?*

1.4 Objective and Contribution

The prevailing approach to solving indoor positioning issues involves using inexpensive and ubiquitous technologies, with Wi-Fi being a prime candidate due to its widespread availability and the use of smartphones. This research aims to enhance indoor positioning systems by implementing a fingerprinting method based on edge computing, leveraging existing WLAN infrastructure without incurring additional costs. Efficient design is crucial for addressing challenges in signal processing and storage, particularly in managing large databases and calculating accurate positions. This research focuses on developing an ideal indoor positioning system, considering factors such as Wi-Fi access point deployment, database management, and cloud administration for optimal performance.

Wi-Fi fingerprinting is central to this research, aiming to design a scalable and reliable indoor positioning system. By examining system design aspects, the research seeks to enhance functionality and accuracy while addressing challenges associated with indoor deployment. Specific objectives include identifying efficient position estimation algorithms, suitable cloud architectures, and improving system scalability and accuracy through Wi-Fi fingerprinting. The research aims to establish system design parameters applicable to indoor Wi-Fi infrastructure, emphasising simplicity, high performance, and satisfactory positioning services. As a result, we will determine the system design parameters applicable to Wi-Fi infrastructure in indoor offices and buildings. Therefore, this research addresses the following:

- Enhancing the positioning accuracy and scalability of Wi-Fi RSSI-based systems.

- Simplifying the complexity of database fingerprinting by implementing an edge computing architecture for efficient resource management and system scalability.

Consequently, this PhD thesis makes significant contributions by overcoming the challenges of accuracy and scalability through innovative approaches and methodologies. The research provides valuable insights and effective design strategies for IPS, particularly in the domain of Wi-Fi fingerprinting techniques.

Most research literature overlooks the system's design elements. In this research, we propose a new approach that considers the system's input, particularly for the online stage of Wi-Fi fingerprinting. We also design a framework that allows multiple solutions from each system element by establishing system metrics. The proposed framework demonstrates a remarkable improvement in IPS accuracy and scalability performance.

The following points highlight the main contributions of this thesis:

- Optimising the Wi-Fi fingerprinting database by studying the characteristics of RSS to design an appropriate algorithm for radio maps and mitigate positioning errors.

- Design an optimised WK-NN algorithm for better positioning estimation. We propose a combination of methods utilising optimal parameters such as k-value, distance weight, and functions that perform well on the selected dataset.

- Propose an indoor positioning system based on edge computing architecture. This system leverages resource optimisation techniques to improve the accuracy and scalability of the IPS, utilising a cache mechanism to reduce computational cost and increase response time.

All these proposed systems are evaluated using publicly available datasets.

1.5 Scope and Limitations

Scope: Building on the motivation discussed in Section 1.2, this research focuses on utilising Wi-Fi as the primary technology for indoor positioning. This choice has implications for the environments and applications that our system can support. Our work targets any area with pre-existing Wi-Fi infrastructure, such as indoor buildings, including offices, markets, and other locations with local hotspots. Specifically, our project emphasises Wi-Fi fingerprinting in indoor environments to enhance the performance and scalability of indoor positioning systems (IPS). We concentrate on the online phase for signal processing and algorithm matching, and we utilise cloud architecture for storage and system scaling. Our proposed system is simulated under indoor conditions using software-based simulations (MATLAB) and the open-source dataset UJIIndoorLoc [15], available from the IndoorLoc repository [16]. The primary aim of this research is to improve the performance and scalability of the proposed IPS system compared to existing state-of-the-art systems.

Simulation Justification. This research adopted a simulation-based approach using MATLAB and the UJIIndoorLoc dataset due to practical limitations in setting up real-world testbeds. The approach enabled rapid evaluation of system design variations and provided a controlled environment for analysing different algorithmic configurations. While simulations offer valuable insights and reproducibility, future work should validate the findings through hardware-based deployments and real-time experiments.

Limitations: This research develops a proof of concept for an indoor positioning system, acknowledging several potential challenges. The focus on simulation, due to environmental and time constraints, may impact the system's validity in real-world scenarios. The accuracy of simulated environments is critical, as it determines the system's performance and generalisability. Furthermore, the system's dependence on existing Wi-Fi infrastructure may restrict its applicability in areas lacking reliable Wi-Fi coverage. Open-source datasets may have limitations in data quality and diversity, potentially compromising the system's robustness.

Although this research offers proof of concept, it may not resolve all deployment challenges, highlighting the necessity for further investigation into real-world applications. Moreover, software-based simulation tools may not fully replicate real-world conditions, necessitating refinement for a more accurate representation.

1.6 Thesis Organization

The structure of the thesis is detailed below:

Chapter 2 provides background information on IPS and Wi-Fi fingerprinting techniques. It includes a review of related work in three areas: Wi-Fi fingerprinting, IPS, IPS design, and cloud-based IPS. The chapter offers a comprehensive analysis of the reviewed work and identifies existing research gaps in Wi-Fi fingerprinting techniques.

Chapter 3 establishes a unified System Model, Notation, and Evaluation Framework for the thesis. It discusses the Wi-Fi fingerprinting method for indoor positioning, focusing on the Received Signal Strength Indicator (RSSI)-based approach and detailing the fingerprinting technique. The chapter also presents baseline deterministic algorithms for RSSI-based fingerprinting.

Chapter 4 addresses position estimation in the online phase, providing comprehensive testing and optimisation of algorithms to enhance performance.

Chapter 5 focuses on optimizing radio map fingerprinting. It reviews various methods and proposes combinations to address issues such as heterogeneity, data size reduction, auto-updating mechanisms, and database management strategies.

Chapter 6 describes the system design and cloud-based indoor positioning architecture. This chapter introduces the proposed system as a platform, incorporating models and algorithms from Chapters 4 and 5. It includes details on system integration, testing, and the results and discussion.

Chapter 7 concludes the research, summarising the main findings and suggesting directions for future work.

Chapter 2

Background

2.1 Overview of IPS

Imagine navigating a large mall, effortlessly finding the perfect store or guiding a loved one through a hospital maze with ease. What was once a dream is now a reality with the revolutionary IPS. Gone are the days of frustrating map searches and aimless wandering. IPS acts as your indoor GPS, pinpointing your exact location and guiding you seamlessly through any building, no matter how complex. This advanced technology uses a network of strategically placed wireless access points or beacons to create a digital map of indoor environments. Whether it is using a smartphone app or a dedicated tag, the IPS system translates complex signals into simple directions, ensuring precision and leading you straight to your destination. Customers, patients, and employees can all benefit from IPS, as it empowers them to navigate indoor spaces with confidence and purpose. Consequently, this gives rise to what we know as Location-based Services (LBS) in many smartphone applications, such as marketing platforms, tracking systems, navigation guides, and emergency response and rescue tools.

Positioning or localisation is an interchangeable term that refers to the use of technology to determine the specific location of a device or person. This mechanism can be classified primarily based on the area of deployment, the underlying technology, and the measurement techniques shown in Figure 2.1. Then, based on user needs and budget, the underlying technology and measurement techniques are chosen accordingly. IPSs have become essential tools for providing accurate location information about people and devices within indoor environments. Various technologies and algorithms have been developed to address the challenges and requirements of indoor positioning. In the following sections, we will explore these technologies, focusing on RF technologies.

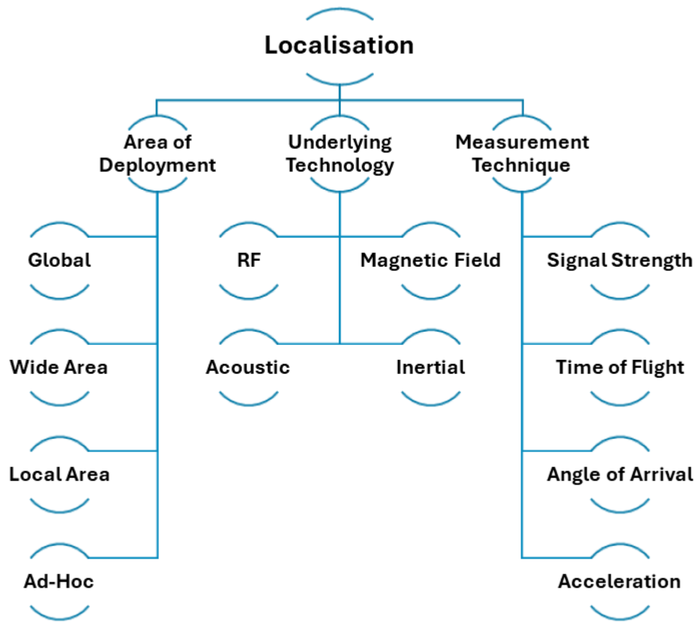


Figure 2.1: General Classification of the Localisation Methods.

2.2 Challenges and Opportunities in IPS

The significance of indoor positioning systems is increasingly recognised due to the growing demand for indoor tracking and navigation. Unlike outdoor environments where satellite-based location services are commonly used, navigating indoors poses significant challenges. Tracking objects and individuals indoors presents a formidable obstacle, constituting a primary barrier to achieving seamless positioning in indoor environments. Enhancements in the performance of indoor positioning systems offer the potential to unlock unprecedented opportunities for businesses.

A fundamental question arises regarding the distinction between indoor and outdoor positioning. Although most positioning systems theoretically function in both environments, their performance varies significantly due to inherent differences.

2.2.1 Challenges

The complexity of indoor environments poses significant challenges for positioning systems, demanding innovative solutions to overcome these obstacles. Identified challenges include:

- **Multipath Effects:** Various obstacles within indoor spaces lead to multipath effects, where signals bounce off surfaces, causing interference and inaccuracies in positioning.

- **Non-Line of Sight (NLoS) Issues:** The lack of direct line of sight to positioning satellites exacerbates inaccuracies in indoor positioning, as signals are obstructed by walls, ceilings, and other structures.
- **Signal Attenuation and Scattering:** The dense concentration of obstacles in indoor environments results in high attenuation and signal scattering, further complicating accurate positioning.
- **Environmental Variability:** Human presence, opening and closing of doors, and changes in occupancy levels introduce dynamic environmental factors that challenge the consistency and reliability of indoor positioning systems.
- **Demand for Precision:** Commercial applications demand high levels of precision and accuracy, placing additional pressure on indoor positioning systems to perform optimally.

2.2.2 Opportunities

Despite the challenges, advances in indoor positioning systems present unprecedented opportunities for various industries. Key opportunities include:

- **Simplified Infrastructure:** Indoor environments typically have smaller coverage areas, simplifying positioning infrastructure setup and reducing deployment complexities.
- **Weather Independence:** Indoor positioning systems are less susceptible to weather influences such as temperature gradients and air circulation, ensuring consistent performance regardless of external conditions.
- **Geometric Constraints:** Fixed geometric constraints, such as planar surfaces and orthogonal walls, provide stable reference points for positioning, enhancing accuracy and reliability.
- **Infrastructure Accessibility:** Availability of essential infrastructure elements, such as electricity and Internet access, facilitates the deployment and operation of indoor positioning systems.
- **Reduced Dynamics:** Slower walking and driving speeds indoors result in lower dynamics, enabling more accurate and reliable positioning measurements.

By addressing these challenges and capitalising on the inherent opportunities, indoor positioning systems can meet the growing demand for LBS and reliable navigation and tracking solutions in indoor environments.

2.3 Positioning Technologies

IPS has become increasingly important in providing location-based services within indoor environments where GPS signals are often unavailable. Several technologies have been developed to address this need, each with its unique advantages and challenges. The common types of these technologies are:

2.3.1 Radio Frequency (RF)

RF-based technologies are commonly used in IPS due to their wide availability and relatively low cost[8]. These technologies include Wi-Fi, Bluetooth, Radio Frequency Identification (RFID), Ultra-Wideband (UWB), and ZigBee [17]. They provide positioning functions with simple modifications and are considered practical for indoor positioning. However, the accuracy of RF-based systems can be affected by signal interference and physical obstructions [18]. Moreover, the accuracy of RF-based IPS can be influenced by various factors, such as multipath propagation, fading, and shadowing. Multipath propagation occurs when signals reflect off surfaces, leading to multiple signal paths reaching the receiver at different times [19], [20]. Fading, on the other hand, results from signal strength fluctuations due to factors such as reflection and interference[8]. Additionally, shadowing occurs when objects obstruct the signal, causing a weaker signal at the receiver [21].

To address these challenges, RF-based IPS systems employ techniques such as signal processing, filtering, and antenna diversity. Signal processing methods such as Time-of-Arrival (ToA), Time-Difference-of-Arrival (TDoA), and Angle-of-Arrival (AoA) can enhance accuracy by mitigating the effects of multipath propagation and fading[22]. Filtering techniques like Kalman filtering and particle filtering are utilised to reduce noise and outliers, thereby improving accuracy[23]. Antenna diversity techniques such as beamforming and spatial diversity help minimise the impact of shadowing and interference, further enhancing accuracy[24]. Overall, RF-based IPS systems provide a practical and cost-effective solution for indoor positioning. However, their accuracy is limited by signal interference and physical obstructions, which can be mitigated by signal processing, filtering, and antenna diversity techniques.

2.3.2 Pedestrian Dead Reckoning (PDR)

PDR is a method that estimates the position of a pedestrian by integrating the step length and heading direction[25]. It uses sensors such as accelerometers, gyroscopes, and magnetometers, commonly found in smartphones. PDR is advantageous because it does not require additional infrastructure beyond the smartphone. However, errors can accumulate over time, leading to a drift in the estimated position [26]. PDR can be combined with other positioning technologies such as Wi-Fi [27], or Bluetooth [28] to mitigate the drift. This hybrid approach can provide a more accurate and robust position estimate [29]. PDR can also be improved by using machine learning algorithms to learn the walking pattern of the pedestrian and correct for errors. PDR has potential applications in indoor navigation, fitness tracking, and location-based services.

2.3.3 Inertial Navigation System (INS)

INS uses motion sensors and rotation sensors to calculate the position, orientation, and velocity of a device[30], [31]. It is often used in combination with other systems like GPS for outdoor navigation, but it can also be used independently for indoor navigation[32]. A method that leverages inertial measuring units (IMU) such as accelerometers and gyroscopes to ascertain the position and movement of objects. INS operates independently of external signals, making it ideal for use in areas with poor signal reception. However, INS may accumulate errors over time and necessitate advanced filtering techniques like the Kalman filter[32]. Additionally, INS demands a network infrastructure for location sensing, which can be expensive and time-consuming to set up[33]. To mitigate this issue, many studies propose an innovative solution that merges existing sensor networks, such as in [31].

2.3.4 Magnetic Field Technology

This technology leverages anomalies in the Earth's magnetic field that are caused by building structures for indoor positioning[34], [35]. Each location inside a building has a unique magnetic signature that can be used for positioning [34]. To identify indoor locations, two methods are commonly used. One relies on the Earth's magnetic field, while the other relies on a magnetic field that is artificially created. The first method identifies location using the unique digital signature of the Earth's magnetic field. The second requires measuring the strength of these fields produced by beacons through coils to determine user location based on the position of the beacons [36]. Magnetic positioning systems are utilised to track locations within enclosed spaces. These systems are resistant to signal interference and do not depend on line of sight to function properly [37]. This approach is beneficial because it does not require any additional infrastructure. However, the accuracy of magnetic field-based positioning can be affected by other magnetic objects in the environment [34].

2.3.5 Acoustic

Sound technologies operate within the audio frequency range of 20 Hz to 20 kHz, including both ultrasonic and audible sound. The human ear can perceive sound waves in various environments. Transducers, such as microphones or speakers, are used to convert sound signals into electrical signals[38]. Ultrasonic sensors are commercially available for measuring sound intensity and have beneficial applications, such as distance sensing by estimating the time it takes for sound signals to travel between a source and the sensor. In indoor tracking scenarios, ultrasonic sensors are utilised as distance-based devices within ultrasonic location detection systems, where the echo and trigger pins of the sensor function as both transmitter and receiver [39], [40]. Moreover, ultrasonic sensors have been integrated into soft robotic perception systems for auto-positioning and multimodal sensory intelligence, highlighting their potential in advanced robotics[41].

2.3.6 Optical

Indoor positioning systems utilise optical signals in the form of infrared (IR) and visible light communication (VLC) technologies. These two technologies require a direct line of sight to function, unlike RF technologies. The development of newer technology like LiDAR [42] and computer vision enables precise and real-time navigation. However, this raises concerns regarding the computational power required and privacy issues. To improve positioning systems, sensor fusion enables the integration of data from various sources, including cameras and magnetic fields, with the help of neural networks. These systems have shown promising results of 91% accuracy at 1.34 m, but the cell phone must be held upright [43].

2.3.7 Hybrid techniques

Hybrid techniques in IPS have gained significant attention due to their ability to enhance accuracy and reduce drift by combining different types of data. One common approach is data fusion, where data from various sources, such as Wi-Fi or Bluetooth signals and PDR data, are merged to achieve more precise and reliable results[44]. Wi-Fi and Bluetooth signals can provide information about access points or beacon locations, while PDR data offers information about the user's movement and orientation [44]. By integrating these data through techniques such as sensor fusion, feature fusion, or decision-level fusion, the accuracy and dependability of IPS applications, particularly for indoor positioning and navigation, can be significantly improved [45]. Research has shown that the fusion of multiple wireless signals, such as Wi-Fi and Bluetooth, through machine learning-based IPS, can notably enhance location accuracy [44]. Furthermore, the use of hybrid positioning measurements, such as the Extended Kalman Filter (EKF), has been successful in achieving accurate indoor positioning [45].

Additionally, the integration of synthetic, simulated datasets with data fusion techniques is proposed to eliminate the cost associated with fingerprint collection in IPS [46]. Moreover, the evolution of IPS technologies has resulted in the widespread use of hybrid models in indoor mapping research to address the limitations of individual models [47]. These advances are crucial because accurate indoor positioning systems are now a research priority, considering that a significant amount of time is spent indoors[48]. By leveraging fusion-based techniques and hybrid models, IPS can overcome challenges related to accuracy, reliability, and market penetration[49]. Therefore, data fusion can improve the accuracy and reliability of various applications that rely on sensor data, such as indoor positioning and navigation. This hybrid technique is often used in various studies, including [50]–[56].

In conclusion, the choice of technology for an IPS depends on the specific requirements of the application, including the desired accuracy, availability of infrastructure, cost, and characteristics of the indoor environment. It is also common to use a combination of these technologies to improve system accuracy and reliability. While these technologies offer unique advantages, they also present challenges, including signal interference, hardware compatibility issues, and the need for extensive infrastructure deployment. Each technology offers unique advantages and drawbacks, catering to specific application needs and navigating the complex trade-off between accuracy, cost, scalability, and privacy. As research and development in indoor positioning continue to progress, a combination of these technologies or innovative solutions may lead to even more robust and reliable indoor positioning systems. In the following sections, we will place greater emphasis on RF-based technologies.

2.4 RF Positioning Techniques

Radio frequency technologies used in IPS are versatile, with applications spanning various fields. These technologies leverage signal strength, particularly in wireless communication devices, and employ spread-spectrum signals with narrow bandwidth. These radio waves are produced by sources or devices that generate an electromagnetic field. The IPS incorporates various localisation technologies based on radio frequency, which are essential to wireless communication technologies. These technologies define the physical layer and medium access control (MAC) layer of the open system interconnection (OSI) model. Examples of such radio frequency technologies include Bluetooth, RFID, ZigBee, UWB, WLAN, and mobile networks [17]. Figure 2.2 illustrates how various RF technologies vary in accuracy and coverage, making it

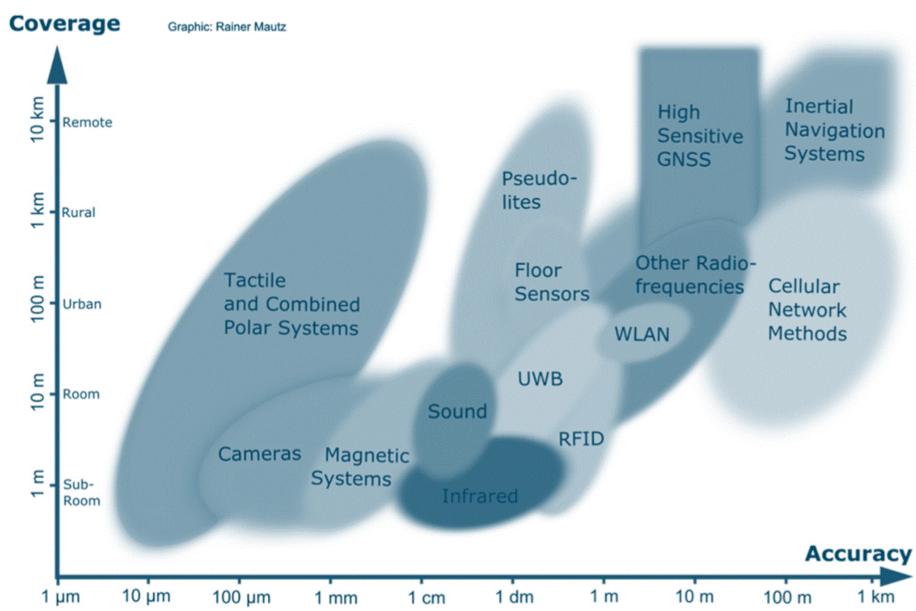


Figure 2.2: RF-based technologies accuracy vs. coverage.(Source:[57]).

important to identify their appropriate use.

Indoor positioning technologies encompass a diverse array of methods, each with its unique principles, advantages, and limitations. One prevalent approach involves Wi-Fi fingerprinting, which utilises the strength of Wi-Fi signals to determine a device's location within a confined space. This method offers high accuracy, scalability, and cost-effectiveness, making it suitable for various applications. Another prominent technology is Bluetooth beacons, which transmit signals to compatible devices, enabling precise localisation. Beacons are particularly useful in retail environments for proximity marketing and navigation [6]. Additionally, UWB technology is gaining traction because of its exceptional accuracy, which makes it capable of pinpointing locations within centimetres. UWB is ideal for applications requiring extremely precise positioning, such as asset tracking and indoor navigation for the visually impaired [58].

The diverse array of RF-based positioning technologies offers unique features that cater to different indoor positioning needs. Understanding the principles, advantages, and limitations of each technology is crucial for selecting the most suitable solution for a given application. As illustrated in Figure 2.3, RF positioning systems have two primary classifications: according to the infrastructure they employ, and based on the devices utilised. Depending on the devices used, the localisation methods can be classified into two categories: **Device-free localisation (DFL)**: This method uses established signals, such as Wi-Fi fingerprinting, to estimate location without needing a person's tag. **Non-device-free localisation (NDFL)**: This approach requires a tag or device attached to the person for precise tracking. Examples include RFID, UWB, Bluetooth, ZigBee, etc. Regarding infrastructure deployment, there are two categories.

Infrastructure-based localisation (i.e., existing RF network): This approach utilises existing networks such as Wi-Fi to estimate location. It is cost-effective but less accurate. In the absence of infrastructure (i.e., infrastructure-free or cooperative localisation): This uses dedicated tags or devices without relying on pre-existing infrastructure. The position of a node is estimated concerning the positions of other nodes in a wireless network, offering better accuracy but requiring setup, which usually incurs additional costs. This is a typical scenario for wireless sensor networks (WSN). In this research, our focus falls under NDFL.

Furthermore, there is another classification presented by [59], which categorises RF localisation according to active localisation and passive localisation. Active IPS encompasses technologies such as RFID, UWB, Bluetooth, ZigBee, IR, ultrasonic, hybrid systems, and WLAN. These systems involve attaching a tag or device to a person to track their position in a dynamic indoor localisation setup. On the other hand, passive indoor localisation is characterised by the absence of any tag or device carried by the person within the location area. Another form of passive localisation is device-free, which operates similarly to the DFL method, such as computer vision. Passive indoor localisation has been studied in [60]–[62].

The upcoming subsections will briefly cover the most commonly used RF technologies in indoor positioning.

2.4.1 Wireless Fidelity (Wi-Fi)

Wi-Fi, short for Wireless Fidelity, is a standard of the Institute of Electrical and Electronics Engineers (IEEE) 802.11 for WLAN. It facilitates the connection of devices within a limited space using radio frequency signals. Operating on the IEEE 802.11a,b,g and n standards, Wi-Fi utilises electromagnetic waves in the radio frequency range as a means of transmitting data. Although Wi-Fi provides a convenient means of interconnecting devices and accessing a broader network, its coverage is limited due to its high frequency (2.4 GHz and 5 GHz) and a restricted transmission range of approximately 100 metres. As a result, Wi-Fi is typically confined to small areas such as apartments, offices, and markets and cannot span large regions [8]. It is important to note that in both this research and other works, Wi-Fi-based methods are sometimes called WLAN-based methods. While these terms are often used interchangeably, they represent different wireless technologies. WLANs utilise radio waves to link nodes. On the other hand, Wi-Fi standards are simply a subset of WLAN standards.

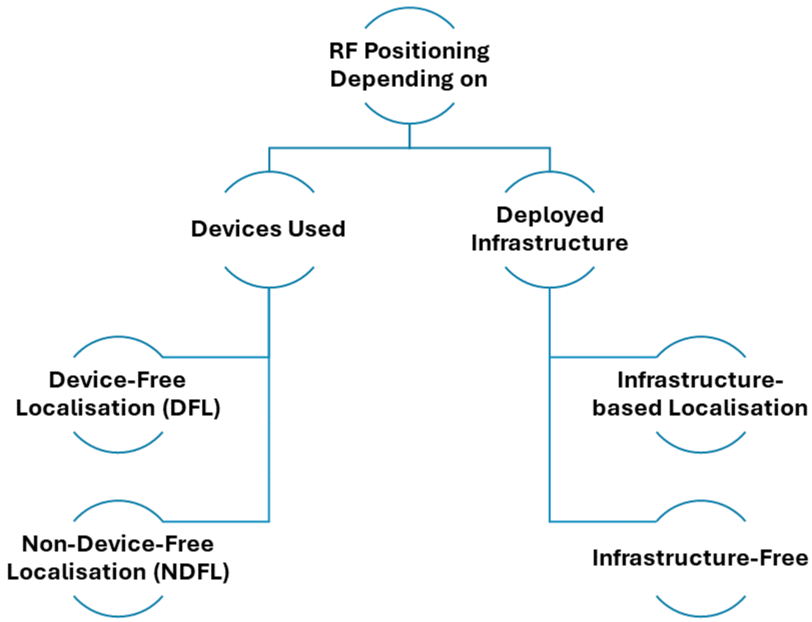


Figure 2.3: Classification of the RF-based Localisation Methods.

Indoor Wi-Fi-based positioning systems utilise the signal strength from multiple wireless access points to estimate the user's position [63]. By analysing RSSI from Wi-Fi access points, these systems can accurately locate users within indoor spaces with a high degree of precision [64]. This technology is attracting considerable attention in research due to its ability to provide high precision, low power consumption, and affordability. As per the IEEE 802.11 standards, Wi-Fi network cards and access points (APs) possess the capability to measure the strength of radio frequency signals. Leveraging this functionality, users can use mobile devices such as smartphones, laptops, and tablet PCs, together with specific algorithms, to achieve indoor positioning using Wi-Fi [8], [65], [66]

Wi-Fi technology has become a prominent solution for indoor positioning systems due to its widespread availability, cost-effectiveness, and accuracy. Leveraging Wi-Fi signals for indoor localisation has been extensively researched, with studies demonstrating the effectiveness of Wi-Fi-based positioning systems. Researchers have explored various approaches to utilising Wi-Fi signals for indoor positioning, including virtual access points, RSSI measurements, and fingerprint-based localisation techniques [24], [67]–[70]. These methods have shown low complexity, high accuracy, and robustness in determining the user's location within indoor environments.

Furthermore, researchers have proposed integrating Wi-Fi signals with other technologies such as sensors, Bluetooth, and IMUs to enhance the accuracy and reliability of indoor positioning systems [30], [52], [66], [71]–[75]. By combining Wi-Fi with sensor data and advanced algorithms, researchers have achieved improved indoor localisation performance. Despite the advantages of Wi-Fi-based indoor positioning, challenges such as signal interference and environmental factors affecting signal propagation have been identified [76]. However, ongoing research focuses on mitigating these challenges through innovative techniques such as machine learning algorithms and hybrid localisation methods[50], [51], [56], [69], [77]–[80].

2.4.2 Bluetooth Low Energy (BLE)

Bluetooth is a part of the IEEE 802.15.1 standard, which is designed to enable short-range wireless communication between devices. Bluetooth technology allows electronic devices to communicate wirelessly using radio waves within the 2.4 GHz license-free Industrial, Scientific, and Medical (ISM) band. Classic Bluetooth uses 79 channels on the 2.4 GHz ISM band for short-range wireless communication between mobile devices, while BLE is designed for low power consumption and small amounts of data transmission for the Internet of Things. The Bluetooth SIG manages the Bluetooth specification and introduced BLE to give the technology a fresh direction. The maximum current of Bluetooth class 4 does not exceed 15 mA, which means that it uses less power than other Bluetooth standards and can transmit small amounts of data to host devices without a physical master-slave connection [81].

Bluetooth, specifically Bluetooth Low Energy (BLE), is a commonly used technology in IPS. BLE beacons can be placed around the indoor environment, and the user's device can estimate its location based on the signal strength of these beacons [20]. Beacon-based Bluetooth systems offer cost-effective solutions with moderate accuracy. Beacons transmit signals, and user devices estimate their position based on received signal strength. However, similar to Wi-Fi, Bluetooth signals can also be affected by physical obstructions. Additionally, scalability can be an issue in large spaces [58]. Researchers have explored different algorithms and methods to improve Bluetooth-based indoor positioning systems. For example, the fusion of Bluetooth beacons with PDR techniques has been proposed to provide metre-level positioning without additional infrastructure [28].

Furthermore, the integration of Bluetooth with Wi-Fi positioning technology has been shown to improve accuracy and robustness in indoor positioning performance [72]. Additionally, studies have investigated the use of BLE for RSSI-based distance estimation algorithms [82], [83]. Despite the success of Bluetooth technology in indoor positioning, challenges such as body-shadowing errors and limitations in response rates of Bluetooth inquiries have been identified [84], [85]. However, ongoing research aims to address these issues through innovative approaches such as IMU-aided error compensation methods [86].

2.4.3 Ultra-Wideband (UWB)

UWB technology has gained significant attention for indoor positioning systems due to its ability to provide exceptionally accurate location information. UWB technology enables precise time-of-arrival estimation, making it more suitable for accurate indoor positioning compared to other technologies like Wi-Fi, Bluetooth, and RFID [87]. UWB-based indoor positioning systems leverage the technology's capability for precise ranging, allowing for centimetre-level precision in determining the user's location [88]. UWB provides very high accuracy and can also measure the distance between the transmitter and receiver, making it suitable for real-time tracking and applications. With its high bandwidth and precise time-of-flight measurements, UWB delivers centimetre-level accuracy.

However, UWB requires specialised hardware, can be susceptible to interference, and is more expensive compared to other RF technologies [10], [89]–[91]. Researchers have explored various applications of UWB in indoor positioning, including dynamic ad hoc systems [92], monitoring and positioning of indoor mobile robots [91], and real-time location systems [93]. UWB technology has been recognised for its superior performance in indoor environments, offering sub-millimetre accuracy and 3D positioning capabilities [88]. Furthermore, UWB has been successfully integrated with IMUs for accurate 3D localisation in various applications [33]. Challenges in UWB-based indoor positioning systems include noise from moving obstacles and non-line-of-sight occurrences, which can lead to unreliable signals [94]. However, research efforts have focused on addressing these challenges through techniques such as ensemble learning and particle swarm optimisation [95].

2.4.4 Radio Frequency Identification (RFID)

RFID technology has emerged as a valuable tool for indoor positioning systems, offering advantages such as low cost, non-contact communication, and resistance to harsh environments [96]. RFID systems have been widely used for indoor location tracking, particularly in applications involving human tracking [97]. The scalability and efficiency of RFID technology make it a suitable choice for various indoor positioning tasks, including patient identification systems and project management [96]. RFID uses tags and readers for positioning. The tags are attached to the objects to be located, and the readers pick up the signals from these tags. Although RFID can provide high accuracy, the need to install many readers for full coverage can be a limitation [12], [97]. Research has explored the integration of RFID technology with other systems to improve the accuracy and functionality of indoor positioning. For example, a hybrid indoor positioning system has been proposed, combining WLAN and RFID technologies, using passive RFID tags for localisation [98].

Additionally, RFID has been integrated with IMUs for indoor tracking, where RFID provides the primary trajectory estimation complemented by IMU data [99]. RFID-based indoor positioning systems encounter a range of challenges and limitations, including interference from other networks, multi-path and shadow effects, communication distance restrictions, and errors that may accumulate from the use of mobile phone sensors and pedestrian dead reckoning calculations [97]. Although these systems are cost-effective and easy to set up, addressing these issues is crucial for improving their accuracy and performance. Thankfully, advanced algorithms such as Bayesian probability and K-Nearest Neighbour(k-NN) have been developed to boost the performance of RFID-based indoor positioning systems. With the help of these algorithms, the technology's low cost, long life, and ease of deployment can be further optimised, making it a highly promising solution for indoor positioning [100].

The integration of RFID with advanced algorithms and complementary technologies continues to drive innovation in indoor positioning research, enabling enhanced accuracy and scalability for a wide range of applications. However, challenges related to interference, environmental effects, communication distance limitations, and accumulated errors need to be addressed to enhance the performance and accuracy of RFID-based indoor positioning systems [12], [97].

2.4.5 ZigBee

ZigBee is a specification based on the IEEE 802.15.4 standard. It uses the 868 MHz band in Europe, the 915 MHz band in the USA and Australia, and 2.4 GHz in other regions. ZigBee technology has been a subject of interest in indoor positioning systems due to its low power consumption, scalability, and reliability. ZigBee is a low-cost, low-power consumption technology that operates in the ISM radio bands. It is often used in applications that require long battery life and secure networking [13], [17]. Research has explored the application of ZigBee in various indoor positioning scenarios, highlighting its potential for accurate and cost-effective location tracking. Studies have investigated the use of ZigBee in combination with other technologies to enhance indoor positioning accuracy [13], [101].

Additionally, ZigBee has been integrated with Wi-Fi technology for improved indoor localisation [102]. Advanced algorithms have been developed to optimise indoor ZigBee-based positioning systems. For example, research by [103] focused on an indoor positioning algorithm based on ZigBee received signal strength index, emphasising ZigBee's low cost, low hardware power consumption, and easy implementation.

Furthermore, research has explored the use of ZigBee in trilateration algorithms for estimating locations in RSSI-based indoor positioning systems [104]. The advantages of ZigBee technology, such as low cost, high scalability, and support for various topologies, have made it a preferred choice for indoor positioning applications [105].

The ability of ZigBee to establish a fingerprint database and its compatibility with inertial navigation systems have contributed to improved accuracy in location estimation [106]. However, the coexistence of ZigBee with other networks, such as Wi-Fi, can cause significant performance losses due to interference, which can affect the overall reliability of indoor positioning systems based on ZigBee [107].

Despite the fact that ZigBee technology faces challenges related to instability, low accuracy, and interference, addressing these challenges is crucial for improving the performance and reliability of ZigBee-based indoor positioning systems.

2.5 Measurement Techniques

In general, RF-based indoor positioning systems—and more specifically (from now on)—Wi-Fi-based technology in indoor positioning fall into two categories, according to [8]: Time and Space Attributes of Received Signal-Based positioning technology (TSARS) and RSSI-based positioning technology, as shown in Figure 2.4. Each of these categories has its measurements based on the technology used:

TSARS-based techniques encompass ToA, Time Difference of Arrival (TDoA), and AoA. ToA calculates the distance between the AP and the Reference Node (RN) based on signal travel time, while TDoA measures signal arrival delay, and AoA determines signal angle. Achieving high accuracy with TSARS usually requires at least three APs and may necessitate additional hardware in the RN, such as multiple antennas, resulting in a costly deployment. The complexity of indoor environments introduces challenges, as RF signal propagation is susceptible to interference from human movement or environmental factors, such as doors opening and closing. Consequently, this method may yield significant measurement and positioning errors [8], [108].

RSSI-based techniques include trilateration, approximation perception, and scene analysis, commonly known as fingerprint matching. Relying on received signal strength, RSSI-based methods offer advantages over TSARS by eliminating the need for additional hardware and being unaffected by multipath signals[109]. The simplicity and cost-effectiveness of RSSI-based methods, exemplified by technologies like Wi-Fi, have contributed to their widespread use for indoor positioning[5], [8].

In addition, indoor positioning measurement techniques can be broadly classified into range-based and range-free categories.

Range-based techniques use distance measurements between the receiver and transmitters to determine the position. These include:

- **Time of Arrival (ToA):** Measures the time taken for a signal to travel from a transmitter to a receiver.

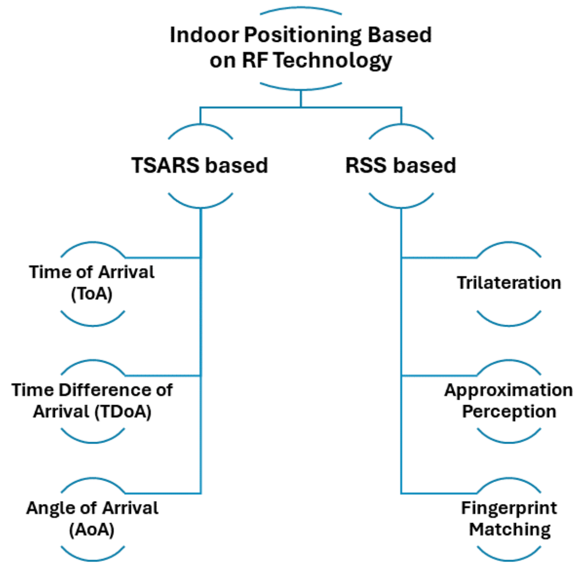


Figure 2.4: Measurements Classification of RF-based IPS (Source [8]).

- **Time Difference of Arrival (TDoA):** Measures the time difference between the arrival of a signal at two or more receivers.
- **Angle of Arrival (AoA):** Measures the angle at which a signal arrives at a receiver, with position determined based on the intersection of angles from multiple transmitters.

Range-free techniques, on the other hand, do not rely on distance measurements but use other parameters such as Received Signal Strength Indicator (RSSI) and AoA. These include:

- **Fingerprinting:** Involves creating a database of signal characteristics at various locations and matching the current signal characteristics to determine the position.
- **Triangulation:** Uses the intersection of circles or spheres to determine the position.
- **Proximity-based methods:** Use the signal strength to estimate the distance between the receiver and transmitter.

In the following subsections, we will provide a brief overview of these positioning measurement techniques for Wi-Fi-based indoor positioning.

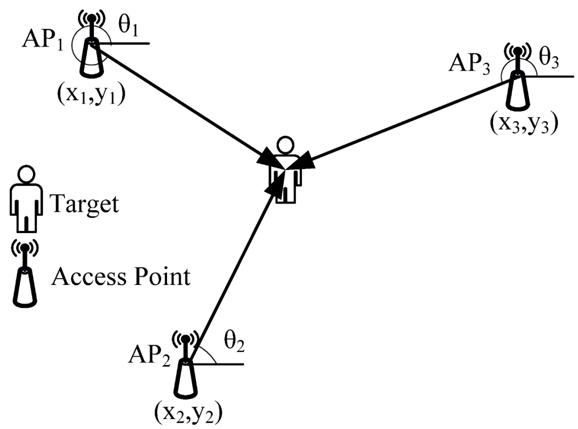


Figure 2.5: AoA measurements (Source[17]).

2.5.1 Angle of Arrival (AoA)

The measurement of the AoA in wireless communication systems is crucial for determining the angle between a mobile device without a fixed location and a fixed anchor, such as an AP with a known position, as depicted in Figure 2.5. Various techniques have been developed to accomplish this, mainly involving the assessment of the amplitude or phase response of the receiving antenna [110]. For accurate localisation, it is imperative to use at least two APs to infer the location of the mobile device [111]. However, as shown in Figure 2.5, it should be noted that the location of the mobile device requires the participation of at least three nodes for a successful implementation [112]. It is noteworthy that employing AoA measurements requires integrating antenna arrays into APs, rendering this technique comparatively more expensive and power-intensive than ToA and RSSI methods [112].

It is crucial to acknowledge that the accuracy of AoA measurements is influenced by various factors, including antenna directivity and environmental conditions such as Non-Line-of-Sight (NLOS) and multipath effects [110]. In particular, angle-based localisation requires a clear line of sight (LOS) due to the potential for significant measurement errors caused by reflected signals [112]. Therefore, careful consideration of these factors is essential in designing and implementing AoA-based localisation systems.

2.5.2 Time of Arrival(ToA)

Determining the distance between nodes is often based on the ToA method, which evaluates the arrival times of signals to calculate the propagation time in one direction, thereby establishing the distance [113]. However, ToA is subject to inaccuracies due to discrepancies in clock synchronisation between transmitters and receivers. Variations as small as a nanosecond can lead to location inaccuracies of up to 0.3 metres, making error control impractical without periodic calibration. Nevertheless, ToA remains a valuable measurement tool in air-based environments due to the consistent propagation rates of radio waves [113].

An alternative approach to addressing this issue involves measuring the round-trip time of signals between nodes, rather than just one-way travel. This method involves sending a signal from one node to another and back again, allowing distance calculation based on the total travel time without requiring precise clock synchronisation. However, delays may occur when the secondary node processes the signal, necessitating calibration procedures to adjust the measurements.

In ToA measurements, the velocity of waves is used to estimate the distance between APs and mobile devices [113]. Various types of waves, such as RF and acoustic signals, can be used for localisation, with RF Receiver resolution, determined by bandwidth, also impacting measurement accuracy, with higher bandwidth resulting in smaller errors. ToA localisation employs the concept of lateration, requiring multiple measurements from different APs to determine the mobile device's coordinates, with a minimum of three APs needed for 2D localisation and four for 3D localisation.

The circular equations derived from ToA measurements are typically solved using methods such as Nonlinear Least Squares (NLS) and Linear Least Squares (LLS). While NLS offers greater accuracy, LLS is more susceptible to noise and non-line-of-sight (NLOS) conditions [114]. Thus, careful consideration of measurement techniques and calibration procedures is essential to ensure accurate distance estimation in localisation systems using ToA.

2.5.3 Time Difference of Arrival(TDoA)

Another related time measurement is the time difference of arrival (TDoA), where the time difference between two ToA measurements is used to formulate a single equation. With three ToA measurements, two TDoA measurements can be formulated [115]. All sensors, including the mobile device, must be synchronised in ToA, as the mobile clock is not as accurate as the base station clock [116]. As a result, there will be errors in estimating the flight time and thus localisation; however, in TDoA, only APs need to be synchronised [116].

On the other hand, ToA makes better use of existing information. One ToA measurement confines the possible locations of the mobile to a circle. With two ToA measurements, the mobile can be located in two possible positions. In contrast, with TDoA, the location lies on a hyperbola. With three measurements, ToA can estimate a unique solution, whereas TDoA may yield one or possibly two solutions

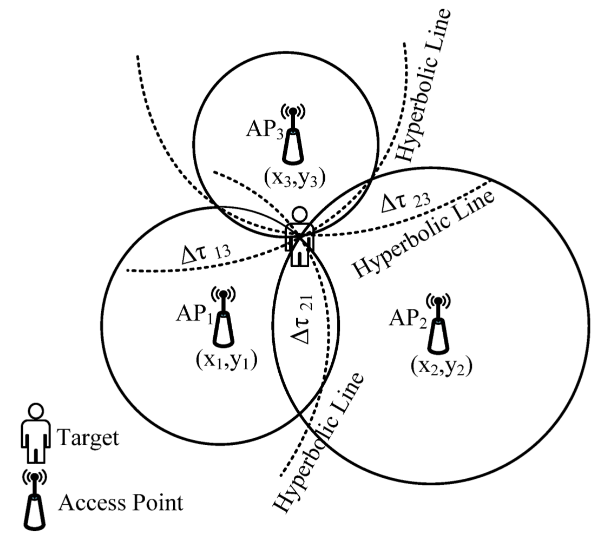


Figure 2.6: TDoA measuremets (Source[17]).

[115]. Another drawback of TDoA is its sensitivity to the existence of LOS; due to its hyperbolic nature, a small error can lead to significant changes in the curve, resulting in reduced accuracy [115]. Several indoor positioning systems using TDoA include [117], [118]

2.5.4 Trilateration

Trilateration is a popular positioning technology that has been extensively researched and applied in indoor positioning systems. Researchers have combined triangular positioning methods with TSARS technology, achieving a certain level of success. However, due to the limitations of these techniques and the advantages of RSSI, RSSI-based triangular positioning has been widely studied. The trilateration method uses three or more APs to send signals that are received by mobile devices and converted into spatial distances. These distances are then used as the radii of circles (with the centres being the APs), and the intersection of the circles determines the user's location, as shown in Figure 2.7.

The complexity of indoor environments can affect the strength of RF signals, leading to inaccuracies in distance determination. To mitigate these errors, scholars suggest techniques to aid trilateration, with the path loss model being the most common method used to estimate the position of APs [8].

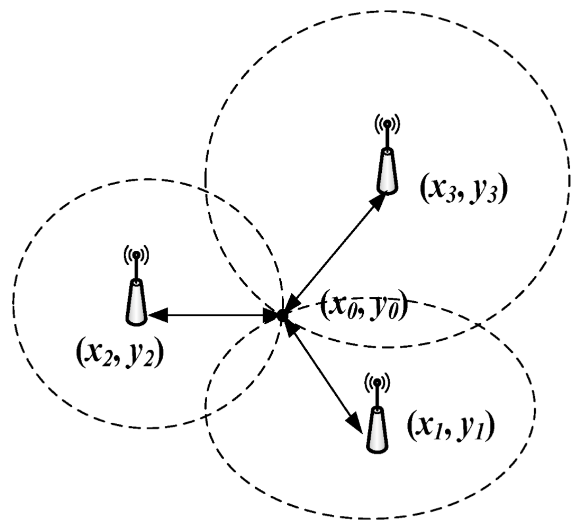


Figure 2.7: Trilateration measurements (Source [8]).

2.5.5 Proximition

Proximity perception, also known as relative positioning, is proposed as an economical and straightforward method for estimating the distance between a mobile device and an AP. The proximity approach is unaffected by whether the mobile device and the AP experience the same fading channel, provided they remain within the communication range [18]. This approach is particularly useful in scenarios where precise synchronisation or alignment of fading channels is challenging, as it emphasises the relative distance between devices rather than specific channel characteristics. Utilising relative positioning methods like proximity perception can be highly advantageous in indoor localisation systems, especially given factors such as multi-path environments and signal attenuation that can complicate traditional distance estimation techniques [17].

By focusing on the proximity of the mobile device to the AP rather than intricate channel characteristics, this approach simplifies the range estimation process and provides a practical solution for indoor positioning applications. In indoor positioning technologies, focusing on relative positioning methods highlights the importance of practical and efficient solutions that do not rely heavily on complex channel modelling or synchronisation requirements. Using proximity between devices as a key metric to estimate range, proximity perception presents a promising approach for achieving accurate localisation results in indoor environments without the need for extensive calibration or synchronisation efforts [119].

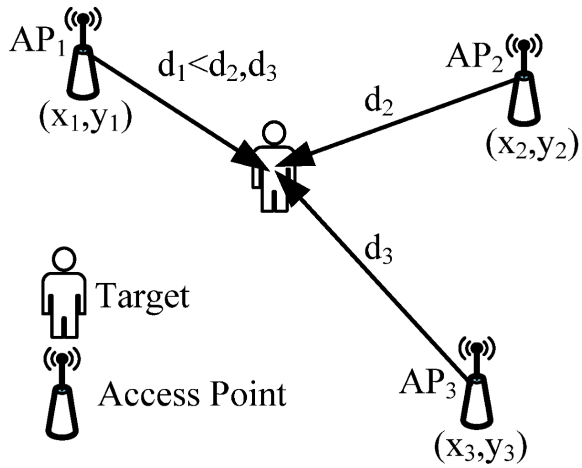


Figure 2.8: Proximite measurments(Source[17]).

2.5.6 Scene Analysis

In the field of scene analysis, particularly regarding the electromagnetic characteristics of a target, pattern recognition or fingerprinting is essential. This technique involves recording and comparing these characteristics with an existing dataset to map features to specific locations [119]. For example, wireless signal characteristics at specific points can be used to create a radio map, which facilitates the determination of a mobile device's location by matching its signal data to the map. This methodology, known as fingerprinting, is recognised for its simplicity but requires the collection of significant data. In addition, environmental changes can affect the characteristics of features, requiring periodic updates to the dataset.

Fingerprinting localisation, widely recognised for IPS development due to its high localisation accuracy, is extensively discussed in the literature [20]. The fundamental concept of fingerprinting lies in the uniqueness of the RF signature at a specific location over time [120]. Wi-Fi fingerprinting, a common indoor positioning technique, capitalises on the availability of infrastructure in urban areas [121]. The effectiveness of fingerprint-based methods is closely linked to the number of reference points in a given space [122]. Furthermore, the fingerprinting database serves as a crucial component of Wi-Fi positioning systems [66].

2.6 Wi-Fi Fingerprinting

Wi-Fi fingerprinting has become a prominent technology for indoor positioning systems due to its ability to provide accurate location information without requiring line-of-sight measurements of access points. This method has gained significant attention in recent years due to its high applicability in complex indoor environments [108], [123]. Wi-Fi technology combined with fingerprinting techniques has emerged as a prominent method for indoor positioning systems. This technique involves estimating a user's location by analysing the signals propagated from several Wi-Fi access points. In contrast to previous RF measurement methods for determining position, Wi-Fi uses RSSI, Channel State Information (CSI), and Round-Trip Time (RTT) as the main approaches in conjunction with fingerprinting techniques to achieve position estimation results.

2.6.1 Received Signal Strength(RSS)

RSS is the actual signal strength measured in units such as dBm at the receiver, typically as a negative value in RF contexts. In contrast, RSSI is a simplified, positive, unitless value that represents signal strength for easier understanding. RSSI scales the negative RF values into positive ones to enhance clarity. For example, 0 dBm (maximum) corresponds to 100 RSSI, and -100 dBm (minimum) corresponds to 0 RSSI.

The RSS data collected from different access points at specific locations is then processed into RSSI to form a database used for positioning. When using fingerprinting technology, this database matches unique RSSI patterns at specific locations to predict user positions. This approach is popular, particularly when combined with machine learning methods [22]. However, trilateration and proximity perception are straightforward methods that do not involve machine learning and rely on distances between access points and the user to calculate the user's location.

The relationship between RSS and distance follows a decay law but is affected by non-linearity due to factors such as multipath interference in indoor environments [21]. Movement and environmental conditions can cause fluctuations in RSS, impacting localisation accuracy [124]. RSS-based systems, such as WLAN, offer advantages such as continuous monitoring and low cost but may face interference issues [21]. These systems excel in short-range distances but may lack accuracy in long-range applications compared to time-of-arrival systems [21]. Various RSS-based localisation algorithms exist, including range-based and fingerprinting techniques, each with its strengths and limitations [21].

2.6.2 Channel State Information (CSI)

The use of orthogonal frequency division multiplexing (OFDM) enables the extraction of CSI, which is crucial for gaining a deeper understanding of the characteristics of the Wi-Fi channel, particularly between access points and receivers. Unlike RSS, which provides only limited information, CSI incorporates significant details such as fading, scattering, multipath effects, and the decay of power with distance during signal transmission.

However, obtaining CSI is more challenging than RSS because it requires extraction from the Wi-Fi receiver driver on mobile devices, making it difficult to utilise in smartphone-based indoor positioning systems [125], [126]. Since 2010, researchers have been able to extract Channel Frequency Response (CFR) as CSI using modified firmware on standard wireless network cards. This has overcome limitations in measuring the precision of wireless channels. CSI provides fine-grained information by offering amplitude and phase details of each subcarrier. This enables indoor positioning systems to achieve sub-metre-level accuracy. The amplitude and phase components of the frequency response of each subcarrier in CSI serve as valuable inputs for indoor positioning systems [125].

2.6.3 Round-trip time (RTT)

Wi-Fi round-trip time information is derived from the fine time measurement (FTM) protocol for ranging proposed by IEEE 802.11-2016. It is a new protocol used to directly calculate the time duration for a Wi-Fi signal to travel from the transmitter to the receiver. This information can be used for various purposes, including indoor positioning, device localisation, and distance estimation. The FTM protocol uses a series of messages between the transmitter and receiver to measure the round-trip time. The protocol can achieve sub-microsecond accuracy in measuring RTT, making it suitable for high-precision applications. The RTT Wi-Fi information is expected to enable new use cases and applications of Wi-Fi networks, especially in indoor environments where GPS signals are not available or accurate enough for positioning [21].

Wi-Fi signals can be used for indoor localisation through various methods such as fingerprint-based, range-based, and angle-based positioning. RSS, CSI, and RTT can all be extracted from these signals. Among these methods, RSS-based positioning has gained popularity due to its easy accessibility and versatility. Although RTT and CSI can offer better accuracy than RSS, they require specialised equipment for precise measurement and are still in the experimental stage. Consequently, they are not widely available for use. On the other hand, RSS is coarse-grained data that contains the superposition of multipath signals at the same time. RSS-based positioning is thus the preferred choice for indoor positioning solutions due to its accessibility, simplicity, and adaptability.

Although other techniques, such as CSI and RTT, offer fine-grained information and higher accuracy, they are often limited by equipment requirements or experimental stages, making RSS the pragmatic choice for scalable indoor positioning solutions. This research will thus focus on utilising RSS for optimal results.

2.7 RSS-based Fingerprinting

From now on, in this research, when we refer to Wi-Fi Fingerprinting, we are talking about using RSSI-based Fingerprinting. Wi-Fi technology is commonly found in smartphones and laptops, and it makes it easy to measure RSS for indoor positioning. Using existing Wi-Fi infrastructure in buildings keeps implementation costs low, making Wi-Fi fingerprinting an economically viable option in various settings such as airports, universities, hospitals, and shopping centres.

Furthermore, Wi-Fi Fingerprinting does not require precise knowledge of AP positions, so it can adapt to different indoor layouts. This adaptability also extends to data processing techniques like machine learning, which helps mitigate noise and correct signal distortion. Fingerprinting localisation has been widely accepted for IPS development due to its superior accuracy, as discussed by (Subedi & Pyun, 2020)[127]. This method eliminates the need for line-of-sight from access points, making it a preferred choice in IPS design.

The RSSI-based Fingerprinting method relies on establishing a unique RSSI fingerprint for each location, which is then matched with real-time RSSI data for positioning [21]. Advanced algorithms like the weighted K-nearest neighbours (Wk-NN) based on RSSI similarity and position distance have shown significant improvements in positioning accuracy[128]. Deep learning approaches, such as convolutional neural networks (CNNs), have been successfully applied to Wi-Fi fingerprinting for indoor localisation, enhancing accuracy and performance[22], [129]. Techniques such as clustering, weighted fusion, and manifold learning have been proposed to enhance the accuracy of fingerprinting-based positioning systems[127], [130].

Within Wi-Fi indoor positioning, RSSI emerges as a cornerstone metric, offering multiple advantages. RSSI-based methods include straightforward approaches such as trilateration and proximity estimation, along with sophisticated fingerprinting techniques. These techniques, particularly when augmented with deep learning algorithms, capitalise on unique RSSI patterns to predict user positions with remarkable accuracy. However, it's essential to acknowledge the challenges associated with RSSI, including non-linear decay laws influenced by factors like multipath interference and environmental dynamics, which can introduce fluctuations impacting localisation precision[21].

RSSI fingerprinting techniques, in particular, have witnessed significant advances. Algorithms such as Wk-NN and deep learning-based approaches have demonstrated substantial improvements in positioning accuracy [128], [129].

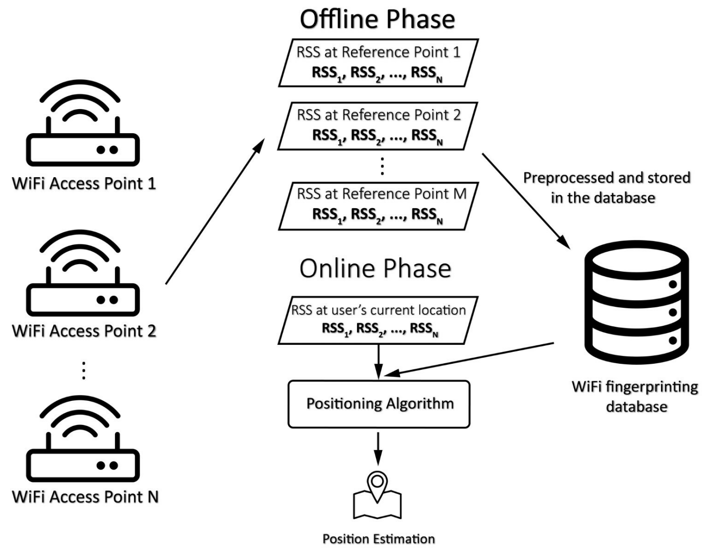


Figure 2.9: RSSI Fingerprinting Diagram (Source [126]).

2.7.1 RSSI Fingerprinting Construction

The RSSI fingerprinting process comprises two distinct phases: offline and online, as shown in Figure 2.9. During the offline phase, a set of Wi-Fi measurements is taken from various APs in a designated area to create a collection of Reference Points (RPs) at predetermined locations. These measurements are then stored in the positioning server database for future use. In the online phase, mobile devices in the same area sense RSSI measurements and send them to the positioning server. Using the RSSI values stored from the offline phase, the server can estimate the position of these mobile devices. The online phase typically employs various algorithms for positioning estimation, ranging from simpler ones like k-NN algorithms to more advanced machine learning and deep learning algorithms.[8], [131].

More details of RSSI Fingerprinting are dedicated to Chapter 3. The accuracy of RSSI fingerprinting depends on multiple factors, including the number of RPs, the distance between RPs, the number of APs used, the density of APs, the stability of the signal, and the interference from the environment. In general, a larger number of RPs and APs and a denser deployment of APs can improve the accuracy of the positioning system. However, this also increases the cost and complexity of the system. Therefore, a trade-off between accuracy and cost should be weighed when designing an RSSI fingerprinting system.

2.7.2 Position Estimation Algorithms

Estimating the location of a user based on RSSI values can be approached through two main methods: deterministic and probabilistic.

The deterministic method is a straightforward approach that relies on the average of the RSSI time samples to estimate the user's location within a building. This method assumes that the RSSI values are deterministic and that the signal propagation characteristics remain constant over time. By averaging the RSSI values collected at different time points, the deterministic method calculates a single-point estimate of the user's location. However, this approach may oversimplify the localisation process by not accounting for variability in RSSI measurements due to factors such as multipath propagation, signal attenuation, and environmental changes.

On the other hand, **the probabilistic method** offers a more sophisticated approach to user localisation, considering all RSSI time samples collected during the localisation process. Instead of relying on a single point estimate, the probabilistic method uses probabilistic algorithms such as Bayesian inference or particle filters to calculate the probability distribution for the user's position within the building. Taking into account the uncertainty associated with RSSI measurements, the probabilistic method provides a more robust and accurate estimate of the location of the user[132].

Deterministic methods include techniques such as nearest neighbour (NN), k-NN, and Wk-NN algorithms [133]. These methods typically involve comparing observed RSSI values with reference points in a radio map to determine the location of the user [134]. On the other hand, probabilistic methods involve models like Gaussian and lognormal distributions to represent RSSI randomness, enhancing the accuracy of location estimation [135]. Hence, the choice between deterministic and probabilistic methods for determining a user's location via RSSI values is contingent on the desired level of accuracy as well as the complexity of the surroundings.

Deterministic methods are straightforward and effective, while probabilistic methods offer a more in-depth analysis of RSSI data for a precise indoor location[136]. Our research aims to create a Wi-Fi Fingerprinting indoor positioning system that is efficient, affordable, and user-friendly, yet still delivers acceptable levels of accuracy, scalability, and dependability. Hence, we choose a deterministic approach for the research, where the Nearest Neighbour algorithm and its variations dominate.

2.8 System Performance Characteristics

The performance evaluation of an IPS encompasses various characteristics crucial for assessing its efficacy. While researchers have established numerous criteria, it is noteworthy that some of the items listed as metrics are better described as characteristics. Metrics ought to be measurable, whereas some of these characteristics lack quantifiability. Nonetheless, they are particularly relevant to the efficient design of scalable indoor positioning systems based on Wi-Fi fingerprinting.

2.8.1 Accuracy

This is the most important metric for any indoor positioning system. The accuracy of a Wi-Fi fingerprinting system is determined by how close the estimated position is to the actual position of the user. The accuracy can be measured in terms of metres or centimetres. The accuracy of an IPS is fundamental, especially for applications demanding precise location information. However, it's imperative to balance accuracy with factors like cost and real-time performance, tailoring the system's precision to meet specific requirements [67].

2.8.2 Scalability

The scalability of the system is determined by how well it performs in large and complex environments. A scalable system should be able to handle a large number of users and provide accurate positioning information even within complex indoor environments. Scalability is paramount for accommodating a growing user base without compromising system performance. Designing a scalable IPS ensures seamless expansion and adaptation to evolving demands, minimising additional costs and optimising performance [67].

2.8.3 Reliability

The reliability of an IPS is essential for consistent performance in diverse environments and conditions. Ensuring accuracy under adverse circumstances is crucial, particularly in critical scenarios where positioning accuracy is vital [137].

2.8.4 Cost

The cost of the system is an important factor to consider when designing and evaluating indoor positioning systems. Wi-Fi fingerprinting systems require hardware and software components, and the cost of these components should be reasonable and affordable. Cost-effectiveness plays an important role in the widespread adoption of indoor positioning technology. Considering cost implications throughout the design process is crucial, as it impacts deployment feasibility and popularity [138].

2.8.5 Precision

Precision, synonymous with consistency and repeatability, is vital for accurate positioning. It measures the closeness of measured positions to actual ones, critical for applications demanding high accuracy [139].

Other metrics such as availability, robustness, stability, latency, and power consumption are also essential considerations, but will be briefly mentioned here.

Availability: Availability refers to the percentage of time that a positioning system is operational and able to provide accurate results. High system availability ensures uninterrupted operation and user accessibility.

Robustness: The robustness of the system refers to how well it performs in the presence of interference and noise. Wi-Fi fingerprinting systems are susceptible to interference from other wireless devices and noise from environmental factors such as walls, furniture, and people. Robust systems can withstand environmental variations and disturbances, ensuring continued functionality in challenging conditions.

Stability: The stability of the system refers to how consistent the estimated position is over time. A stable system will provide the same estimated position for a user at a particular location over multiple measurements.

Latency: The latency of the system refers to the time it takes to provide a position estimate. A low-latency system is desirable for real-time applications such as indoor navigation. Minimising latency enhances real-time performance, benefiting applications requiring immediate positioning updates.

Power Consumption: Power consumption is an important criterion for evaluating positioning systems. Low power consumption is crucial for prolonging battery life, especially in portable devices such as smartphones.

By focusing on these key metrics, the design and evaluation of scalable indoor positioning systems based on Wi-Fi fingerprinting can be more effectively guided and assessed.

2.9 Cloud Technology in Indoor Positioning

The general concept of cloud-based IPS is visually depicted in Figure 2.10. These systems leverage Wi-Fi fingerprinting technology to deliver precise positioning services. Through the integration of cloud computing capabilities, these systems offer flexible and scalable solutions for indoor localisation. Incorporating machine learning algorithms enhances the accuracy and reliability of positioning prediction [68], [108]. Cloud computing significantly improves indoor positioning systems by providing scalable and efficient computational resources, addressing limitations such

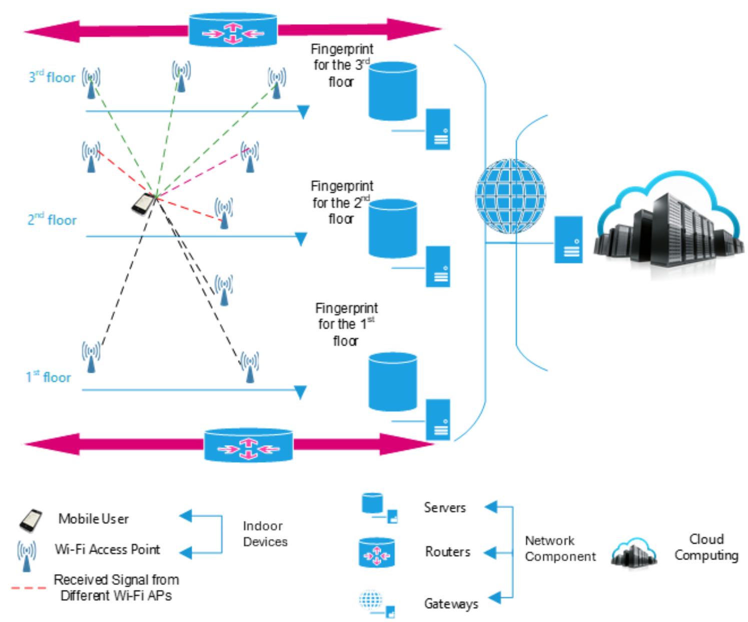


Figure 2.10: General Concept of Cloud-based IPS

as processing efficiency, fault tolerance, and privacy concerns [140].

The synergy between cloud and edge computing is essential for addressing challenges such as device diversity, data processing efficiency, and security. Cloud-based Wi-Fi indoor positioning systems integrate machine learning algorithms, hybrid positioning technologies, and advanced signal processing to deliver dependable location information across various applications. These systems offer scalable, precise, and efficient solutions for indoor positioning, catering to various needs [129].

Efficient indoor positioning systems must align with cloud computing paradigms to fully exploit their advantages, combining computational power, precise positioning techniques, and standardised methodologies. Achieving computational efficiency entails deploying lightweight algorithms capable of accurately determining user or device positions. However, the complexity of indoor environments poses challenges to achieving high positioning accuracy, especially in complex multi-building and multi-story settings.

Key attributes of cloud-based IPS include seamless integration of new positioning technologies, efficient algorithms, straightforward setup and maintenance, resilience to failures, adherence to privacy and security standards, compatibility with different computing paradigms, and real-time positioning tracking. Researchers are actively developing a range of software solutions, including proprietary and open-source options, to embody these characteristics effectively.

2.10 Related Work

This section categorises related studies into three thematic areas. First, IPS based on the Wi-Fi fingerprinting technique is examined, providing an overview of early and contemporary systems. Secondly, Efficient design approaches are presented, closely related to our method. Thirdly, the integration of Cloud-based Wi-Fi indoor positioning is addressed. Lastly, a summary of limitations in the existing literature is provided, discussing research gaps about the research questions.

2.10.1 IPS based on Wi-Fi Fingerprinting

Wi-Fi fingerprinting is a well-researched indoor positioning technique extensively discussed in the literature. Numerous Wi-Fi positioning systems have been proposed due to the proliferation of Wi-Fi hotspots, with early systems such as RADAR [141] and Horus [142] pioneering this field. RADAR, developed by Microsoft Research, was among the pioneering efforts to use Wi-Fi networks for generating location fingerprints. In the training phase, the area of interest is divided into a 1×1 metre grid, where signal strength measurements from access points are taken at each intersection. These measurements are then averaged to create a radio map for subsequent online phase use. However, challenges emerge as the stability of the radio map is not always guaranteed, impacting location estimation accuracy. Subsequent studies have explored various aspects of indoor Wi-Fi fingerprinting positioning systems, including positioning algorithms, feature extraction methods, and system architecture [5], [18], [58], [138], [143], [144].

Deterministic approaches to Wi-Fi fingerprinting research have focused on optimising the k-NN algorithm [145], [146]. Adaptive k-value methods have shown significant improvements in position accuracy [146], while algorithmic enhancements that incorporate signal propagation models have also been explored [147]. Quartile analysis and sparse learning techniques have been proposed for the preprocessing of RSSI data, leading to improved positioning accuracy [148], [149]. Various approaches have been used to select Wi-Fi signals that uniquely classify locations [108], [150]. The integration of Wi-Fi with other technologies such as BLE has been explored to improve indoor location systems [122]. Moreover, device-free localisation algorithms using RSSI metrics have become a promising research area [60].

Recent research has used machine learning algorithms to improve accuracy and efficiency, with deep learning models and support vector machines being applied to wireless fingerprinting [69], [151], [152]. Despite these advances, challenges remain in creating and maintaining radio maps, and in errors in measuring RSSI [153]. Various approaches, including SLAM, extrapolation/interpolation, and crowdsourcing, have been proposed to address these challenges [24], [154], [155]. Furthermore, [156] introduce a fingerprint-based indoor Wi-Fi localisation method that exceeds traditional algorithms. Collectively, these studies support the importance and progress in using Wi-Fi RSSI for indoor localisation applications.

2.10.2 Efficient Design of IPS

The design aspects of IPS have received relatively limited attention in the literature, with few studies addressing the essential considerations for scalable IPS design. Early research by Kaemarungsi (2005) presented a systematic study aimed at improving the design of indoor positioning systems based on location fingerprinting techniques [157]. Kaemarungsi proposed a modelling framework to facilitate the efficient design of such systems, quantifying the improvement in accuracy and precision resulting from adjustments in system parameters.

Similarly, [158] noted the absence of analytical models suitable for designing and deploying positioning systems. Their innovative approach modelled WLAN planning and positioning error reduction as an optimised solution to address indoor positioning challenges during WLAN planning. Subsequent studies investigated the optimisation of reference node (RN) placements to improve system performance [159].

Recent research has explored machine learning techniques to improve the accuracy of IPS, studies highlighting the importance of system scalability [160]. Researchers have proposed various scalable solutions, highlighting the importance of scalability for public applications [161]. Despite these advancements, challenges persist in highly dynamic and large-scale indoor environments, necessitating innovative approaches for accurate and reliable positioning services [162]. In the realm of fingerprinting methods, [127] introduced Affinity Propagation Clustering and Weighted Centroid Fingerprinting to improve the accuracy of location estimation. These advances aim to improve the accuracy and efficiency of indoor positioning systems.

Furthermore, studies by [71], [128] focused on refining positioning algorithms based on RSSI similarity and BLE devices, respectively. Despite advances in RSSI-based fingerprinting techniques, challenges persist, highlighting the need for comprehensive system design considerations.

2.10.3 Cloud-Based Wi-Fi Indoor Positioning

Cloud-based Wi-Fi indoor positioning systems have attracted considerable attention in recent years due to their potential to offer accurate and real-time location information in indoor environments. Leveraging cloud computing for indoor Wi-Fi positioning provides scalability, flexibility, and centralised data management capabilities. In addition, cloud-based Wi-Fi indoor positioning systems can benefit from advanced signal processing techniques to address challenges such as multipath propagation. By leveraging physical layer information and advanced algorithms such as extended Kalman filters [163], these systems can extract direct path signals and enhance localisation accuracy in complex indoor environments.

Researchers have explored the deployment of cloud-based architectures to manage the processing and storage requirements of large-scale Wi-Fi fingerprint datasets for indoor positioning [164]. Using cloud resources, these systems efficiently handle the collection, storage, and analysis of Wi-Fi fingerprints to improve the accuracy of the localisation [165]. Furthermore, cloud-based solutions enable real-time updates of radio maps and algorithms, ensuring the system's adaptability to changing indoor environments [166].

In [167] investigated latency issues in cloud-based IPS systems, particularly in scenarios where users are mobile, such as indoor navigation. They introduced the Location Retrospective Adjustment (LRA) method to address this issue, enhancing real-time navigation performance and reducing the computational burden on mobile devices. Notably, LRA operates solely through the mobile browser (HTML5), ensuring stable and accurate navigation without the need for additional application downloads.

In [168], a fixed-edge cloud-based design employing a single access point for guiding robots within buildings was proposed. While promising, this approach requires further refinement to work with multiple access points. In [169] presented a cloud-based indoor positioning service using Android devices to record RSSI and determine positions via server requests, subsequently transforming them into global coordinates. Despite demonstrating feasibility, their research highlighted average response times and performance issues, especially under high request loads. To address this, they suggested implementing a load balancing strategy to improve performance without compromising response times or throughput.

Additionally, [170] proposed a verifiable edge computing scheme for indoor positioning, integrating edge computing technologies to improve the reliability and security of location-based services. By leveraging edge computing capabilities, the system aims to tackle challenges related to device heterogeneity and data processing in indoor positioning applications. Moreover, [171] introduced a cloudlet-based cloud computing approach for Wi-Fi indoor positioning and navigation, utilising machine learning models to process RSSI data for accurate indoor localisation tasks. Using cloudlet resources, the system optimises data processing and analysis, leading to better indoor positioning accuracy.

Furthermore, [172] proposed an edge computing solution merging Wi-Fi and Bluetooth fingerprint data while prioritising user privacy, countering vulnerabilities inherent in cloud-based LBS solutions. Their approach advocates for a multi-tiered cloud-computing network based on a privacy-preserving framework to ensure location secrecy, accuracy, and reduced offline fingerprinting time. In [173] proposed a cloud-based machine learning mechanism for optimising fingerprint nodes in RSSI data, leveraging multidimensional spatial similarity to enhance Wi-Fi-based indoor positioning accuracy while mitigating computational complexity and estimation errors.

The integration of cloud computing with Wi-Fi fingerprinting technology allows for the development of innovative positioning algorithms capable of handling complex indoor environments [174]. By leveraging cloud resources for data processing and model training, these systems achieve high precision and reliability in indoor positioning applications [27]. Furthermore, cloud-based indoor positioning systems support multiple simultaneous measurements and collaborative feedback mechanisms to improve localisation accuracy [175].

The efficient design of a Wi-Fi fingerprinting indoor positioning system involves leveraging advanced techniques to enhance accuracy and scalability. To achieve this, integrating cloud computing capabilities can optimise system performance and flexibility [140]. Cloud platforms offer efficient computation, interoperability, and real-time updates, which are crucial for managing large-scale Wi-Fi fingerprinting datasets and ensuring accurate indoor localisation.

2.10.4 Gaps in the Existing Literature

Research on Wi-Fi fingerprinting-based IPS has predominantly focused on enhancing positioning accuracy. However, there has been limited attention to the design of scalable IPS architectures.

- Various approaches have been proposed to improve accuracy, but the role of well-designed Wi-Fi fingerprinting algorithms in enhancing both performance and scalability remains underexplored.
- Existing IPS studies often lack unified metrics and standardised datasets, hindering comprehensive evaluations.
- Previous research on IPS has mainly focused on achieving accurate indoor positioning by developing methods based on a single technology.
- These studies typically involved modifications to network access points and/or mobile nodes at the device level, necessitating signal processing measurements and hardware modifications.

Addressing these gaps in the literature is crucial for advancing cloud-based Wi-Fi indoor positioning systems. By focusing on standardised benchmark datasets, automatic radio map adaptation mechanisms, and exploring synergies with other sensor technologies, we can develop more accurate, reliable, and scalable indoor positioning systems.

2.11 Summary and Conclusions

2.11.1 Summary

The chapter provides an in-depth overview of IPS, highlighting its growing significance across various applications. IPS technologies such as Wi-Fi, BLE beacons, and UWB offer solutions tailored to diverse application needs. Understanding the strengths and limitations of these technologies is crucial for creating seamlessly connected indoor environments. Among these techniques, Wi-Fi Fingerprinting, particularly RSSI-based methods, stands out due to its ability to cope with indoor environmental challenges such as multi-path and obstacles. Wi-Fi fingerprinting utilises signal strength from multiple access points to estimate device locations within indoor environments.

The discussion delves into the stages of Wi-Fi fingerprinting, including training and positioning, and explores various matching approaches such as nearest-neighbour algorithms, deep neural networks (DNN), and recurrent neural networks (RNN). Despite potential inaccuracies in dynamic environments, Wi-Fi fingerprinting continues to be a preferred choice for indoor location tracking, providing precise location determination without the need for specialised equipment. By leveraging the ubiquity of Wi-Fi infrastructure, Wi-Fi fingerprinting offers a cost-effective and non-intrusive solution for indoor positioning.

The chapter examines key performance metrics for IPS, such as accuracy, scalability, reliability, and cost. Additionally, the chapter concludes by emphasising the role of cloud technology in indoor positioning and discussing advances in fingerprint-based indoor localisation, driven by developments in artificial intelligence and pattern recognition, particularly deep learning techniques. The chapter lays the foundation for subsequent research and development efforts by identifying gaps in the current literature.

2.11.2 Conclusions

The chapter concludes by emphasising the importance of IPS technologies in various applications and underscores the advantages of Wi-Fi fingerprinting, particularly its adaptability, scalability, and cost-effectiveness. Despite challenges such as dynamic environmental conditions, Wi-Fi fingerprinting remains a viable solution for indoor location tracking, offering precise results without the need for specialised hardware. The integration of cloud technology further enhances the capabilities of indoor positioning systems, paving the way for more efficient and reliable solutions.

Additionally, the chapter highlights the ongoing advancements in fingerprinting-based indoor localisation, driven by innovations in artificial intelligence and pattern recognition, including deep learning approaches. By addressing the existing gaps in the literature, future research endeavours can build upon these developments to further improve the accuracy and robustness of indoor positioning systems. In summary, the chapter provides a comprehensive understanding of IPS technologies and their potential for shaping the future of indoor navigation and location-based services.

Chapter 3

Wi-Fi Fingerprinting

This chapter presents the Wi-Fi fingerprinting method employed for indoor positioning estimation, with a primary focus on the Received Signal Strength Indicator (RSSI)-based approach. It introduces the fundamental fingerprinting techniques and discusses the baseline deterministic algorithms applied to RSSI-based fingerprinting.

3.1 Introduction

Wi-Fi technology is widely utilised for indoor positioning due to its broad availability and the absence of additional infrastructure requirements, as Wi-Fi access points (APs) are commonly installed in many buildings. Nevertheless, Wi-Fi-based positioning systems are often constrained by limited accuracy compared to alternatives such as UWB [176]. For instance, in [177] reported that Wi-Fi positioning typically achieves an accuracy of approximately 2 to 4 metres. This level of precision is generally considered sufficient for typical indoor applications, such as navigating large airport terminals or locating rooms in multi-floor hospital buildings.

Among indoor positioning techniques, Wi-Fi fingerprinting is one of the most prevalent due to its ease of implementation, lack of reliance on additional hardware, and capacity to deliver satisfactory accuracy. However, its performance is influenced by several factors, including the density and distribution of APs, the initial site survey required to construct the fingerprint database, and the periodic updates needed to maintain the database's accuracy. Furthermore, the precision of positioning depends on both the quality of data collection and the selection of appropriate estimation algorithms.

Wi-Fi fingerprinting operates by generating radio map fingerprints (RMFs) of specific areas through RSSI measurements from multiple APs. Nevertheless, incorporating all detected APs into fingerprint vectors can lead to increased computational complexity and reduced system efficiency [108]. Including all APs may introduce noise and redundancy, adversely affecting the accuracy of the IPS. Prior studies emphasise the importance of considering AP set similarity and RSSI distance to ensure that the fingerprint data accurately represents spatial proximity [178].

The RSSI fingerprinting technique constructs a radio map by collecting signal strength data throughout the environment and subsequently estimating a user's position by comparing real-time measurements with pre-recorded fingerprint values [58], [75]. RSSI is a standard metric in wireless communications that reflects the power level received by a device [17]. Modern single-chip transceivers typically include RSSI indicators to assess signal quality [179].

RSSI fingerprinting comprises two distinct phases: offline and online, as illustrated in Figure 3.1. During the offline phase, RSSI measurements are systematically collected at predefined locations—termed reference points (RPs), within the area of interest (AoI). These measurements are stored in a central database (DB) on the positioning server (PS), thereby constructing the radio map for the environment.

In the online phase, a mobile device (MD) operating within the same AoI measures RSSI values from nearby APs and transmits them to the positioning server. The PS then estimates the device's location by comparing these real-time measurements with the previously stored fingerprints, as shown in Figure 3.2.

Various algorithms can be employed for location estimation in the online phase, ranging from basic methods such as k-Nearest Neighbours (k-NN) to more sophisticated machine learning and deep learning approaches [8], [131].

Together, these processes facilitate accurate and reliable indoor positioning using Wi-Fi fingerprinting, highlighting the critical integration of hardware infrastructure and algorithmic methodologies in the design of indoor positioning systems.

Wi-Fi fingerprinting is considered a promising solution, but it has its drawbacks, such as the need for human effort to initially take measurements and to keep the fingerprint updated over time. The radio map (fingerprinting) can change due to alterations in furniture in the area [8], [157], [180].

3.2 System Model and Evaluation Framework

This section outlines the general system model for Wi-Fi fingerprinting based indoor positioning adopted in this thesis. It also defines the standard mathematical notation that will be used consistently across subsequent chapters and introduces the key performance evaluation metrics employed to assess the developed algorithms and systems. Establishing this unified framework aims to enhance clarity and ensure a consistent understanding of the methodologies and results presented.

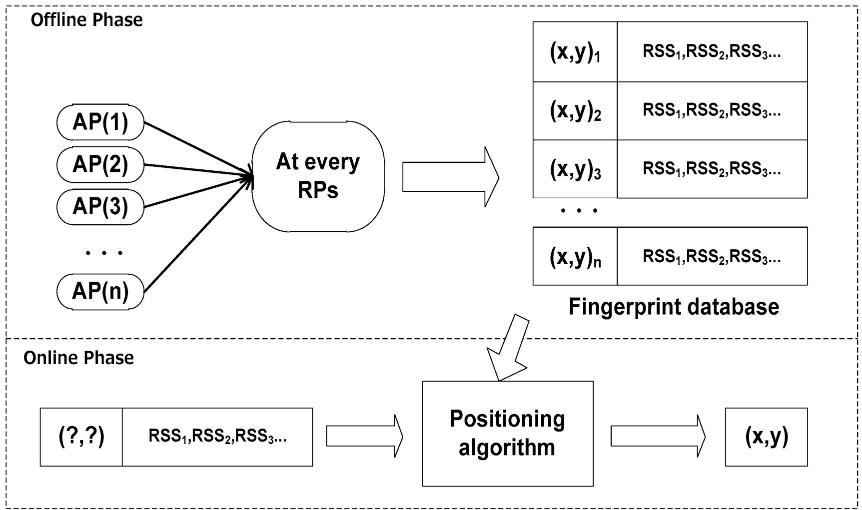


Figure 3.1: Wi-Fi Fingerprinting operation to estimate unknown position (Source [8])

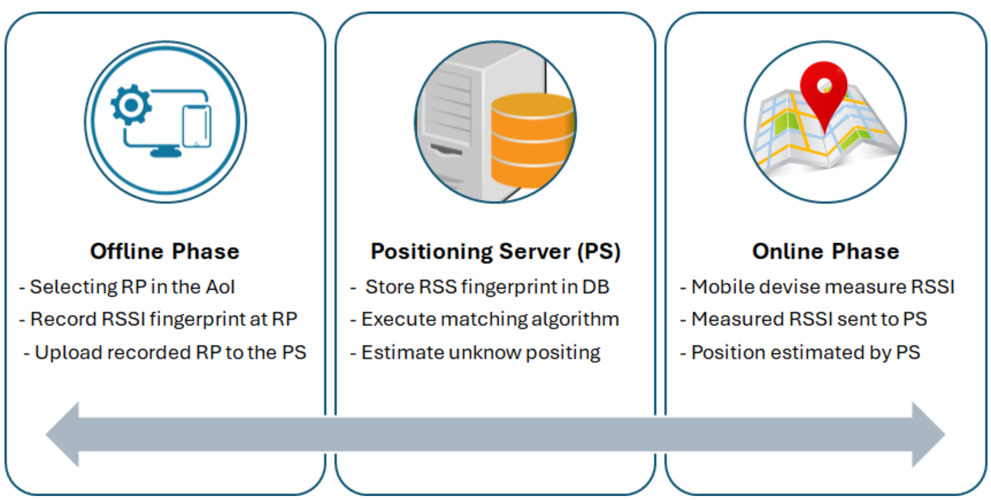


Figure 3.2: IPS Operation Phases

3.2.1 General System Model and Notation

The Wi-Fi fingerprinting technique for indoor positioning, as investigated in this research, operates based on a two-phase paradigm: an offline phase and an online phase.

Offline Phase (Radio Map Construction): During the offline phase, a detailed survey of the indoor environment is conducted. Received Signal Strength Indicator (RSSI) values are collected from M detectable Wireless Access Points (APs) at N predetermined discrete locations known as Reference Points (RPs). Each RP, denoted by index i (where $i = 1, \ldots, N$), has known true coordinates, which, for a 2D representation, can be expressed as $\boldsymbol{c}_i = (x_i, y_i)$. In multi-floor, multi-building scenarios, these coordinates can be extended to include floor and building identifiers, e.g., $\boldsymbol{c}_i = (Lon_i, Lat_i, Floor_i, BuildingID_i)$, as is the case with datasets like UJIIndoorLoc For each RP i, the set of RSSI values from all M APs forms a unique vector known as a fingerprint, denoted as $\boldsymbol{f}_i = [RSSI_{i1}, RSSI_{i2}, \ldots, RSSI_{iM}]$, where $RSSI_{ij}$ is the signal strength from AP j at RP i. The collection of all such fingerprint-coordinate pairs, $\{(\boldsymbol{f}_i, \boldsymbol{c}_i)\}_{i=1}^{N}$, constitutes the radio map or fingerprint database, denoted as $\mathcal{D}$.

Online Phase (Position Estimation): In the online phase, a Mobile Device (MD) at an unknown location measures its current RSSI values from the surrounding M APs. This results in an online RSSI vector, $\boldsymbol{z} = [RSSI_{z1}, RSSI_{z2}, \ldots, RSSI_{zM}]$. A positioning algorithm is then employed to compare this online vector $\boldsymbol{z}$ against the fingerprints $\{\boldsymbol{f}_i\}$ stored in the radio map $\mathcal{D}$. Based on this comparison, the algorithm estimates the MD's current coordinates, $\hat{\boldsymbol{c}}$.

Standard Mathematical Notation: To ensure consistency throughout this thesis, the following mathematical notations will be adopted:

- Scalars (e.g., individual RSSI values, coordinates, number of APs/RPs, error values, indices) will be denoted by normal italic letters: $x, y, k, M, N, d, RSSI, e, i, j$.
- Vectors (e.g., RSSI fingerprints, coordinate vectors) will be denoted by lowercase bold italic letters: $\boldsymbol{f}, \boldsymbol{z}, \boldsymbol{c}, \boldsymbol{p}, \boldsymbol{q}$. For instance, an RSSI vector $\boldsymbol{f} \in \mathbb{R}^M$.
- Matrices (e.g., data matrices, PCA projection matrix) will be denoted by uppercase bold italic letters: $\boldsymbol{X}, \boldsymbol{W}, \boldsymbol{Y}$.
- Sets (e.g., the radio map database) will be denoted by calligraphic uppercase letters: $\mathcal{D}$.
- Estimated values will be denoted with a hat accent: $\hat{\boldsymbol{c}}, \hat{x}$.

This notational convention is intended to provide clarity in all mathematical descriptions and algorithmic presentations. Table 3.1 and Table 3.2 provide mathematical symbols used throughout the thesis.

3.2.2 Performance Evaluation Metrics

The performance of the indoor positioning systems and algorithms developed and analysed in this thesis is assessed using a set of standard evaluation metrics. These metrics are chosen to provide a comprehensive understanding of the system's accuracy in both classification tasks (such as identifying the correct building or floor) and regression tasks (such as estimating precise 2D coordinates)[126].

3.2.2.1 Metrics for Classification Tasks

For tasks involving the prediction of categorical labels, such as the building or floor where a user is located, the primary metric used is the Hitting Rate.

Hitting Rate (HR): The Hitting Rate quantifies the percentage of test instances where the system correctly predicts the categorical label (e.g., building ID or floor number). It is a direct measure of classification accuracy.

$$HR = \frac{N_{correct}}{N_{total}} \times 100\% \tag{3.1}$$

where $N_{correct}$ is the number of correct predictions for a given category (e.g., correct floor identifications), and N_{total} is the total number of prediction attempts for that category. A higher HR indicates better classification performance.

3.2.2.2 Metrics for Regression Tasks (Coordinate Estimation)

For tasks involving the estimation of continuous values, such as the 2D geographical coordinates (e.g., Longitude and Latitude) of a user, the following metrics are employed, all based on the positioning error.

Positioning Error (e_k): For each test point k, the positioning error e_k is defined as the Euclidean distance between the true 2D coordinates (Lon_k, Lat_k) and the estimated 2D coordinates $(\widehat{Lon}_k, \widehat{Lat}_k)$ provided by the positioning system.

$$e_k = \sqrt{(\widehat{Lon}_k - Lon_k)^2 + (\widehat{Lat}_k - Lat_k)^2} \tag{3.2}$$

This error is typically measured in metres.

Mean Absolute Error (MAE): The MAE represents the average of the absolute positioning errors over all test instances. It provides a straightforward measure of the average prediction error magnitude.

$$MAE = \frac{1}{N_{test}} \sum_{k=1}^{N_{test}} e_k = \frac{1}{N_{test}} \sum_{k=1}^{N_{test}} \sqrt{(\widehat{Lon}_k - Lon_k)^2 + (\widehat{Lat}_k - Lat_k)^2} \tag{3.3}$$

where N_{test} is the total number of test points evaluated. A lower MAE indicates higher average accuracy.

Root Mean Squared Error (RMSE): The RMSE is the square root of the average of the squared positioning errors. It is sensitive to large errors, meaning outliers have a more significant impact on the RMSE value compared to MAE.

$$RMSE = \sqrt{\frac{1}{N_{test}} \sum_{k=1}^{N_{test}} e_k^2} = \sqrt{\frac{1}{N_{test}} \sum_{k=1}^{N_{test}} \left((\widehat{Lon}_k - Lon_k)^2 + (\widehat{Lat}_k - Lat_k)^2 \right)} \quad (3.4)$$

A lower RMSE indicates better positioning accuracy, with a greater penalty for large deviations.

Cumulative Distribution Function (CDF) of Positioning Errors: The CDF of positioning errors is a graphical representation that illustrates the probability $P(E \leq e)$ that the positioning error E is less than or equal to a specific value e. It provides a comprehensive overview of the error distribution, allowing for an understanding of, for example, the percentage of errors that fall below a certain acceptable threshold (e.g., 5 metres, 10 metres).

These unified metrics will be referenced in subsequent chapters when evaluating the performance of the proposed algorithms and system architectures.

Table 3.1: Summary of Scalar Mathematical Symbols and Notation

Symbol	**Description**
Scalars	
M	Total number of WAPs considered in a fingerprint.
N	Total number of Reference Points (RPs) in the radio map.
N_{test}	Total number of test queries used for performance evaluation.
$N_{correct}$	Number of correctly classified/localised test instances.
N_{total}	Total number of prediction attempts for a classification task.
$RSSI_{ij}$	Received Signal Strength Indicator from WAP j at RP i.
$RSSI_{R,j}$	RSSI from WAP j at a specific RP R.
$RSSI_{zj}$	RSSI from WAP j in an online (query) RSSI vector $\boldsymbol{z}$.
x_i, y_i	2D Cartesian coordinates of RP i.
Lon_i, Lat_i	Longitude and Latitude coordinates of RP i.
$Floor_i$	Floor identifier for RP i.
$BuildingID_i$	Building identifier for RP i.
$\hat{x}, \hat{y}$	Estimated 2D Cartesian coordinates.
$\widehat{Lon}_k, \widehat{Lat}_k$	Estimated Longitude and Latitude for test point k.
k	Number of nearest neighbours considered in the k-NN algorithm.
$d(\boldsymbol{p}, \boldsymbol{q})$	Distance (e.g., Euclidean) between RSSI vectors $\boldsymbol{p}$ and $\boldsymbol{q}$.
d_0	Reference distance in the path loss model (typically 1 metre).
$RSSI_0, RSSI_{0j}$	Reference RSSI value at distance d_0 (for WAP j).
n, n_j	Path loss exponent (for WAP j).
$X_{ij}, X_{j,target}$	Shadowing or large-scale fading component.
σ_j^2	Variance of the shadowing effect for WAP j.
p	Order parameter for Minkowski distance.
α	parameter in exponential RSSI representation.
γ	Weighting factor in auto-update.
β	Parameter in powered RSSI representation.
w_i	Weight assigned to the i-th nearest neighbour in Wk-NN.
e_k	Positioning error for the k-th test point.
μ	Mean (e.g., of RSSI values for normalisation).
σ	standard deviation (e.g., of RSSI values for normalisation).
$Pos(x)$	Positive representation of RSSI value x.
$Exp_i(x)$	Exponential representation of RSSI value x for WAP i.
$Pow_i(x)$	Powered representation of RSSI value x for WAP i.

Table 3.2: Summary of Vector, Matrix, and Set Mathematical Symbols

Symbol	**Description**
Vectors (lowercase bold italic)	
$\boldsymbol{c}_i$	Coordinate vector of RP i, e.g., (x_i, y_i).
$\hat{\boldsymbol{c}}$	Estimated coordinate vector for a query.
$\boldsymbol{f}_i, \boldsymbol{f}_R$	RSSI fingerprint vector for RP i or RP R; $\boldsymbol{f} \in \mathbb{R}^M$.
$\hat{\boldsymbol{f}}_i$	Estimated or updated fingerprint vector for location i.
$\boldsymbol{z}$	Online (query) RSSI vector measured by a Mobile Device; $\boldsymbol{z} \in \mathbb{R}^M$.
$\boldsymbol{p}, \boldsymbol{q}$	Generic RSSI vectors used in distance calculations.
Matrices (uppercase bold italic)	
$\boldsymbol{X}$	Radio map matrix, where rows are $\boldsymbol{f}_i$ and columns are WAPs.
$\boldsymbol{Y}$	Data matrix after transformation (e.g., projected data in PCA).
$\boldsymbol{W}$	Transformation matrix in PCA (matrix of principal components).
Sets (calligraphic)	
$\mathcal{D}$	The fingerprint database, typically a set of $(\boldsymbol{f}_i, \boldsymbol{c}_i)$ pairs.
$\mathcal{Q}_{data}$	Set of new fingerprint-location data derived from user queries.

3.3 RSSI Models

There exist two principal categories of RSS methodologies: range-based and range-free. The range-based technique entails the construction of a spatial map predicated on the physical attributes of the wireless signal, subsequently employed to ascertain the object's location through trilateration, min-max, or maximum likelihood algorithms. Nevertheless, this method exhibits potential limitations in precision and adaptability across diverse environments. Conversely, the range-free technique leverages a fingerprinting database (radio map) to deduce the object's position by juxtaposing the signal strength at the object's present location with a precompiled database of signal strengths at known locations.

The range-free methodology obviates the necessity for angular or distance measurements between nodes. A pivotal aspect of wireless signal transmission is RSSI, which quantifies the received signal's intensity at a receiver device, typically articulated in decibel-milliwatts (dBm) or milliwatts (mW). The RSSI value facilitates the estimation of the distance between transmitter (Tx) and receiver (Rx) devices, with proximity inversely correlated to RSSI magnitude. Consequently, signal strength attenuates with increasing distance, as illustrated in Figure 3.3.

The figure delineates the relationship between signal strength and distance,

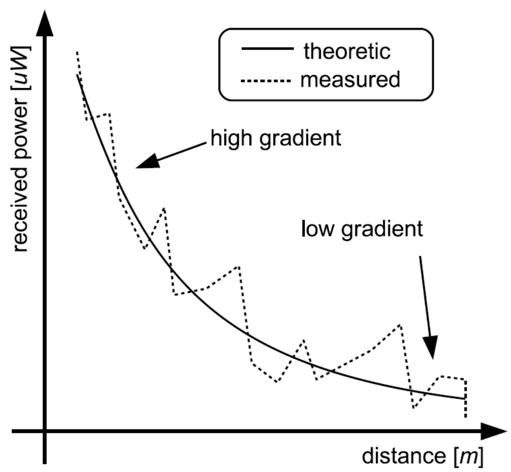


Figure 3.3: Signal characteristic: relation between signal strength and distance

encompassing both the transmitter (Tx) and the receiver (Rx). Although theoretical predictions of signal strength exhibit consistency, empirical measurements often demonstrate variability attributed to fading instigated by multipath propagation. In confined environments, signal propagation from assorted access points engenders a distinctive representation of received signals, influenced by multipath, attenuation, and spatial obstructions, such as furniture. This attribute proves advantageous for RSSI-based fingerprinting methodologies but may impede other range-based indoor positioning techniques, such as AoA or ToA.

A radio map at each designated position, known as a Reference Point (RP), is meticulously constructed using this technique. The fingerprinting methodology offers enhanced accuracy and applicability across various indoor settings; however, RSSI measurements are susceptible to environmental perturbations. Indoor environments, replete with obstacles, significantly affect radio signal propagation. Noise and multipath effects notably degrade RSSI localisation precision. Nonetheless, the accuracy of RSSI computations can be ameliorated through meticulous calibration and analysis of signal propagation [17].

RSSI represents a measure of RSS, as typically defined by individual chip vendors and quantified in arbitrary units per the RSSI [179]. By relying solely on received signal strength, the RSSI-based algorithm can efficaciously pinpoint an object's position without necessitating supplementary hardware or time synchronisation, thereby surpassing the accuracy of alternative methodologies.

The RSSI propagation model used in fingerprinting for indoor positioning is often based on the logarithmic distance path loss model. This model represents the relationship between the received signal strength and the distance between the transmitter AP and the target device.

The signal propagation characteristics from any Access Point (AP) to a specific location within an indoor environment can be described by the log-distance path loss model. Let $RSSI_{loc,j}$ be RSSI value received from the j^{th} AP at a generic location denoted by 'loc', and $d_{loc,j}$ be the distance from this location to the j^{th} AP. The model is expressed as:

$$RSSI_{loc,j} = RSSI_{0j} - 10n_j \log_{10}\left(\frac{d_{loc,j}}{d_0}\right) + X_{loc,j}, \tag{3.5}$$

where:

- $RSSI_{0j}$ is the reference RSSI value at a short reference distance d_0 (e.g., 1 metre) from the j^{th} AP. This accounts for the transmission power and antenna characteristics of AP j.
- n_j is the path loss exponent associated with the j^{th} AP, indicating the rate of signal strength decay with distance within the specific indoor environment.
- $X_{loc,j}$ represents the large-scale fading (shadowing) component for the signal path between the location 'loc' and the j^{th} AP, often modelled as a zero-mean Gaussian random variable in decibels, i.e., $X_{loc,j} \sim \mathcal{N}(0, \sigma_j^2)$, where σ_j^2 is the variance of the shadowing for AP j.

This general model (Equation 3.5) applies to both phases of the Wi-Fi fingerprinting process, with differences in context and known parameters:

1. Offline Phase Application (Radio Map Construction): During the offline phase, a radio map is constructed by collecting RSSI measurements at N known reference positions (RPs). For the i^{th} RP, 'loc' in Equation 3.5 corresponds to this RP i. Thus, $RSSI_{loc,j}$ becomes $RSSI_{ij}$, which is the signal strength value stored in the fingerprint database (often an average of multiple readings at RP i from AP j). The distance $d_{loc,j}$ becomes d_{ij}, the *known* Euclidean distance between RP i and AP j. The model in this context helps understand the signal characteristics that form the fingerprints stored in the database.

2. Online Phase Application (Position Estimation): In the online phase, a target device at an *unknown* location measures current RSSI values from surrounding APs. Here, 'loc' in Equation 3.5 corresponds to the target device's location. Thus, $RSSI_{loc,j}$ becomes $RSSI_{target,j}$, the live RSSI value measured by the target device from the j^{th} AP. The distance $d_{loc,j}$ becomes $d_{target,j}$, which is the *unknown* distance. The model in this context describes the physical generation of the RSSI values that the target device measures in real-time.

It is crucial to note that in the Wi-Fi fingerprinting technique, rather than directly solving Equation 3.5 for individual unknown distances $d_{target,j}$ (which is characteristic of range-based methods), the entire vector of measured $RSSI_{target,j}$ values is compared against the pre-calculated fingerprints in the radio map. The goal is to find the stored fingerprint (and thus its known location) that is most similar to the currently observed RSSI vector. This approach leverages the site-specific uniqueness of the overall radio signature.

The Wi-Fi fingerprinting technique leverages the distinct radio signature present at each location within an indoor environment. This uniqueness arises from the complex propagation of radio waves, which are affected by physical obstacles such as walls, doors, furnishings, and floors, leading to reflection, diffraction, and scattering. Consequently, the pattern of RSSI values from multiple detectable APs at a given location can serve as a distinguishing fingerprint for that location.

The core of RSSI fingerprinting involves two main phases:

1. Offline Radio Map Construction: In this phase, a database of RSSI fingerprints is constructed. At a series of pre-defined reference positions (RPs) within the indoor environment, RSSI values from all detectable APs are measured and recorded. For a given reference position R, the RSSI fingerprint, denoted as $\boldsymbol{f}_R$, is a vector containing these measurements:

$$\boldsymbol{f}_R = [RSSI_{R,1}, RSSI_{R,2}, \ldots, RSSI_{R,M}], \tag{3.6}$$

where M is the total number of APs considered, and $RSSI_{R,j}$ is the RSSI value measured at reference position R from AP j. Each such $RSSI_{R,j}$ value is an instance of the signal behaviour described by Equation 3.5 applied to the offline context. This database of fingerprints $\{\boldsymbol{f}_R\}$ paired with their known locations constitutes the radio map.

2. Online Position Estimation: When a target device requires its position, it performs a Wi-Fi scan to measure the current RSSI values from detectable APs. This results in an observed RSSI vector, $\boldsymbol{z}$:

$$\boldsymbol{z} = [RSSI_{target,1}, RSSI_{target,2}, \ldots, RSSI_{target,M}], \tag{3.7}$$

where $RSSI_{target,j}$ is the RSSI value currently measured by the target device from AP j. Each $RSSI_{target,j}$ is an instance of the signal behaviour described by Equation 3.5 applied to the online context.

The system then estimates the target device's position by comparing the observed vector $\boldsymbol{z}$ with the fingerprints $\boldsymbol{f}_R$ stored in the radio map.

3.4 Application of RSSI Models in This Research

The RSSI propagation models presented in Section 3.3, which describe the relationship between RSSI and distance (Equations 3.5), serve as the theoretical underpinning for understanding signal behaviour in the Wi-Fi fingerprinting techniques developed throughout this thesis. While these mathematical models articulate the expected signal characteristics, their direct application for positioning (e.g., by inverting them to calculate distances) is not the primary focus of our work due to real-world complexities.

Instead, this research adopts an *empirical fingerprinting methodology*. We leverage a database of collected RSSI measurements (the radio map) where each fingerprint is a vector of signal strengths from multiple APs at a known location. This approach inherently captures the complex multipath propagation, attenuation, and shadowing effects characteristic of indoor environments, which are often challenging to encapsulate exhaustively with purely theoretical propagation models. Specifically, while the logarithmic distance path loss model (as detailed in Section 3.3) informs our understanding of general signal decay and variability, our radio map construction relies on actual measured RSSI values.

The *principles of fingerprinting*, which are based on the unique RSSI signatures described by the models in Section 3.3, are directly realised in the positioning algorithms presented in Chapter 4. Particularly, the understanding derived from these models influences:

- The structure of our RSSI fingerprint database (Section 3.5), which stores vectors of measured RSSI values. Each element in these vectors is an instance of an RSSI governed by the principles laid out in Equations 3.5.

- The choice and application of distance metrics within our nearest-neighbour algorithms (Section 3.11.1 and Chapter 4), which operate by comparing these empirically collected RSSI vectors.

- The strategies for radio map optimisation are detailed in Chapter 5. These techniques, such as dimensionality reduction and auto-update mechanisms, aim to enhance the efficiency and robustness of the fingerprinting process, which fundamentally relies on the stability and distinctiveness of RSSI patterns.

Our implementation strategy acknowledges the practical limitations of relying solely on theoretical propagation models for precise positioning in varied indoor settings. By grounding our system in empirical measurements, while being informed by the theoretical behaviour of RSSI, we aim to develop robust and accurate positioning solutions. This allows our system to build upon established theoretical foundations while pragmatically addressing the real-world complexities of indoor signal propagation.

The specific algorithms, optimisations, and their performance evaluations, which embody this application of fingerprinting principles, are detailed in Chapters 4 and 5.

3.5 Fingerprint Database Constructing

Fingerprinting Databases can be categorised based on their coverage into:

- **Single-floor databases**
- **Single-building and multi-floor databases**
- **Multi-building and multi-floor databases**

While RSSI collection methods can be divided into:

- **Crowdsourcing:** Based on volunteers, resulting in random RSSI records at various RPs but requiring minimal effort.
- **Insourcing:** Purposefully designed and systematically carried out by project participants, resulting in a structured and organised RSSI database, but demanding considerable labour during construction.

In the offline stage, constructing the RSSI fingerprinting database requires a team of labour if the area to be covered is substantially large. The procedure begins with collecting RSSI data from various APs within a building or indoor space. This data includes information such as signal strength, MAC address, and position. The researchers then use this data to create a database that maps the RSSI readings to specific positions within the building. This database can then be used by indoor positioning systems to accurately locate users within the building based on RSSI readings received from their devices. To model this process, consider the following:

- N as the total number of reference positions.
- M as the total number of APs in the environment.
- $RSSI_{ij}$ as the RSSI measurement obtained from the i^{th} reference position for the j^{th} AP.

Then, the RSSI fingerprint database $\boldsymbol{X}$ can be mathematically represented as:

$$\boldsymbol{X} = \begin{bmatrix} RSSI_{11} & RSSI_{12} & \dots & RSSI_{1M} \\ RSSI_{21} & RSSI_{22} & \dots & RSSI_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ RSSI_{N1} & RSSI_{N2} & \dots & RSSI_{NM} \end{bmatrix}$$

Each row of this matrix represents the RSSI measurements from all APs at a specific reference position, while each column represents the RSSI measurements

Table 3.3: Datasets Characteristics

Dataset	Year	D_{tr}	D_{te}	#APs	FP/RP	Area (m^2)	#Bld	#Flr	Ref.
TUT 1	2013	1476	490	309	1	9000	1	4	[184]
TUT 2	2013	584	176	354	1	14000	1	3	[184]
UJI 1	2014	19861	1111	520	21	108703	3	4-5	[15]
UJI 2	2017	20972	5179	520	11	108730	3	4-5	[185]
TUT 3	2017	697	3951	992	1	8000	1	5	[68]
TUT 4	2017	3951	697	992	1	8000	1	5	[68]
LIB 1	2018	576	3120	174	12	308.4	1	2	[21]
LIB 2	2018	576	3120	197	12	308.4	1	2	[21]
UTS 1	2019	9108	388	589	6	44000	1	16	[22]
SOD 1	2022	21205	2720	105	1/30	8000	3	1-3	[125]

from a particular AP across all reference positions. To populate this matrix, RSSI measurements are collected at each RP from all available APs using a Wi-Fi scanning device, such as a mobile phone or laptop. These measurements are then organised into the matrix format as described above, forming the RSSI fingerprinting database used to locate and position devices within the indoor environment. This model matrix describes the spatial distribution of RF fingerprints within the target area, known as the radio map fingerprint (RMF).

3.6 Fingerprinting Datasets

Numerous experiments have been undertaken to propose Wi-Fi indoor positioning systems, primarily consisting of software simulations, with only a few executed within dedicated testbeds [181]–[183]. However, conducting empirical experiments may not always be financially viable or logistically feasible due to the considerable time and cost involved. Therefore, in our research, we focus on modelling indoor positioning and evaluating various algorithms for system performance using publicly available datasets from online repositories such as IndoorLoc[16]. These repositories offer diverse datasets, among which we have chosen UJIIndoorLoc [15]. The Table 3.3 illustrates a comparison of these Wi-Fi Fingerprinting datasets.

The data presented in Table 3.3 unmistakably highlight the UJI dataset as the largest in terms of covered area, spanning three buildings and five floors. Additionally, it boasts an impressive number of fingerprints per RP, totalling 21. It is noteworthy that the UJI dataset stands out as the most frequently used dataset in indoor positioning systems, having been referenced in more than 300 research articles [125]. In the following, we introduce the UJI dataset and its characteristics.

3.6.1 UJIIndoorLoc

The UJIIndoorLoc database serves as a widely adopted resource for evaluating and developing machine learning algorithms in indoor positioning, particularly those utilising Wi-Fi fingerprinting technology [15]. This database, which includes training and validation data, plays a crucial role in the development and validation of machine learning algorithms for indoor positioning systems. Machine learning algorithms, trained on datasets like UJIIndoorLoc, play a pivotal role in indoor positioning systems, especially when utilising Wi-Fi fingerprinting methods. These algorithms utilise RSSI values to accurately predict indoor locations.

The UJIIndoorLoc database has emerged as a widely utilised resource within the academic community for the refinement and evaluation of machine learning algorithms related to indoor positioning. Numerous scholarly works have leveraged this repository to introduce and validate a diverse array of machine learning algorithms specifically tailored for indoor positioning systems, with a pronounced emphasis on WLAN/Wi-Fi fingerprinting technology. This database, encompassing both training and validation datasets, plays a crucial role in the development and validation of custom matching algorithms for indoor positioning systems [15]. The UJIIndoorLoc dataset is a publicly available collection of data points intended to evaluate indoor positioning systems based on Wi-Fi fingerprinting. The dataset comprises 21,048 records (consisting of 19,937 training samples and 1,111 validation samples), each corresponding to a unique capture event and containing 529 numerical elements. These elements are categorised as follows:

- **RSSI (elements 001-520):** Signal strengths from 520 distinct wireless access points (WAPs) detected at the capture location.
- **Real-World Coordinates (elements 521-523):** X, Y, and Z coordinates of the capture point within the building, where Z refers to the floor.
- **Metadata:**
  - **BuildingID (element 524):** Identifier for the building where the capture occurred (one of three buildings).
  - **SpaceID (element 525):** Identifier for the specific space within the building (933 unique spaces).
  - **Relative Position (element 526):** Relative position of the capture point within the designated space (potentially useful for specific applications).
  - **UserID (element 527):** Identifier for the user who collected the data point.

- **PhoneID (element 528):** Identifier for the specific mobile device used for data collection (25 different models used).
- **Timestamp (element 529):** Date and time of the capture event.

3.6.2 Key Characteristics

- **Coverage Area:** The dataset encompasses an area of $108,703$ square meters across three buildings, with each building having $4-5$ floors.
- **Training and Validation Sets:** The data is divided into 19,937 samples for training/learning and 1,111 samples for validation/testing. To ensure dataset independence, validation samples were collected four months after the training samples.
- **User Diversity:** Data collection involved more than 20 users using 25 different mobile device models (some users used multiple models).
- **Data Collection Tools:** Two Android applications, CaptureLoc and ValidationLoc, were used for data collection. These applications referenced map services published on an ArcGIS server, providing users with visual aids during data capture (building interiors and reference point locations) to enhance positioning accuracy.

Originally conceived for indoor localisation within a university campus in Spain, the UJIIndoorLoc database stands out as a multi-building, multi-floor repository based on Wi-Fi fingerprinting technology, offering a realistic depiction of diverse indoor environments across three heterogeneous buildings. During its inception, the database detected 520 different WAPs, resulting in Wi-Fi fingerprints comprised of 520 intensity values per Reference Point (RP), thus forming a unique reference point characterised by a 520-element vector. These intensity values are represented as negative integer values ranging from -104 dBm (indicative of extremely poor signal) to 0 dBm, with a default value of +100 dBm assigned to undetected WAPs (as illustrated in Table 3.4). Furthermore, each reference point is accompanied by a set of attributes including longitude, latitude, floor, building, space, relevant position, user ID, phone ID, and time of records (as depicted in Table 3.5).

Leveraging this publicly available database facilitates the examination of novel indoor localisation algorithms and enables comparative analyses across different algorithms. This public dataset serves as a standardised means to assess the accuracy of localisation algorithms reliant on RSSI levels and facilitates comparative evaluations of localisation algorithms within a standardised experimental framework. Additionally, the accessibility of proposed algorithms for scrutiny is heightened by the public availability of the database, distinguishing it from other research endeavours confined to specific environments [15].

Table 3.4: WAPs measurement of RSS at one of the RP

WAP_{001}	...	WAP_{031}	WAP_{032}	WAP_{033}	WAP_{034}	WAP_{035}	WAP_{036}	...	WAP_{520}
-97	...	+100	-97	+100	+100	-65	-65	...	+100

Table 3.5: Example of a data entry at a RP with its attributes

WAP_{001}	...	WAP_{520}	Longitude	Latitude	Floor	BuildingID	SpaceID	Rel.Pos	UserID	PhoneID	Time
-97	...	+100	-7594.7	4864983.9	3	0	111	2	11	13	1370340142

3.6.3 Data preparation

The UJIIndoorLoc database comprises fixed-size vectors, with each index corresponding to a WAP. These vectors encapsulate original intensity values ranging from 0 (indicating the highest signal strength) to -104 (representing the lowest signal strength) in decibel milliwatts (dBm), with undetected WAPs denoted by a default value of 100 dBm. To ensure data consistency, any rows or columns containing this default value were omitted. Consequently, a total of 55 columns and 76 rows were removed from the original dataset, resizing it from (19,937x529) to (19,861x474), constituting approximately a 10.7% reduction in the original dataset size.

Additionally, negative intensity values were converted into their positive counterparts, facilitating a more intuitive interpretation of measurements and aiding in subsequent calculations, particularly for operations such as square-root transformations or logarithmic computations. Although some researchers [186] may advocate for additional normalisation techniques, such as converting values to a zero-to-one scale, in our investigation, the conversion of all RSSI values to positive values suffices for the baseline algorithm, particularly in the context of the k-NN algorithm.

3.6.4 RSSI Data Representation

The analysis of RSSI data, which constitutes a fundamental element of Wi-Fi fingerprinting, is subject to exploration through various representations. These representations encompass both raw signal strengths and transformed values, the assessment of which serves to elucidate their respective impacts on positioning accuracy. The discernment of the most effective representation therein serves as a critical endeavour in refining the feature space, thereby potentially enhancing algorithmic performance.

In our research, we explore three distinct representations for RSSI values: positive, exponential, and powered. We follow the methodology proposed in [186]. This preprocessing step has demonstrably enhanced model performance, primarily by effectively representing RSSI measurements for algorithmic calculations and concurrently reducing training time. The following delineation provides a detailed overview of these representations:

- Positive representation

$$Pos_i(x) = \begin{cases} (RSSI_i - min), & \text{if } WAP_i \text{is detected,} \\ 0, & \text{Otherwise.} \end{cases} \tag{3.8}$$

- Exponential representation

$$Exp_i(x) = \frac{exp(\frac{RSSI_i - min}{\alpha})}{exp(\frac{-min}{\alpha})}. \tag{3.9}$$

- Powered representation

$$Pow_i(x) = \frac{(RSSI_i - min)^{\beta}}{(-min)^{\beta}}, \tag{3.10}$$

where $RSSI_i$ is a received signal strength measurement, min represents the minimum value of $RSSI_i$ in the datasets. Lastly, α and β are mathematical constants with values of 24 and 2, respectively.

The parameters α and β control the scaling and non-linear transformation of RSSI values in the exponential and powered representations, respectively. Following the methodology proposed by [186], the values $\alpha = 24$ and $\beta = 2$ were selected based on their extensive empirical analysis of the UJIIndoorLoc dataset.

Specifically, $\alpha = 24$ was determined to be optimal for the exponential representation as it appropriately scales the exponential function to capture the logarithmic nature of signal propagation. The value preserves the relative importance of strong signals while preventing weaker signals from being completely diminished in the transformed space. The parameter $\beta = 2$ in the powered representation effectively implements a quadratic transformation that has been shown to enhance the discriminative power of RSSI fingerprints by amplifying the differences between stronger signals while moderating the impact of weaker, potentially less reliable signals. Our own empirical evaluations on the dataset further confirmed the suitability of these values.

These specific values represent a balance between enhancing signal differentiation and maintaining robustness against noise, which is crucial in indoor environments where signal propagation is affected by various factors such as multipath fading, reflection, and absorption. In [186] demonstrated through extensive experimentation that these parameter values yielded superior positioning accuracy compared to alternative configurations.

The parameters $\alpha = 24$ and $\beta = 2$ were adopted from [186], who established these values through empirical optimisation. While the specific process of determining these exact values is not elaborated in their work, these parameters serve important mathematical functions in transforming RSSI values. The parameter α in the exponential representation controls the scaling factor that determines how quickly the exponential transformation amplifies differences between signal strengths. Similarly, β in the powered representation determines the degree of the power function applied to signal strengths, with $\beta = 2$ implementing a quadratic transformation that accentuates stronger signals while diminishing the influence of weaker ones.

It is worth noting that these parameter values may be optimised for the specific characteristics of the UJIIndoorLoc dataset and the environmental conditions under which it was collected.

3.7 Dataset Profiling and Analysis

This section presents a comprehensive analysis and profiling of the indoor positioning dataset, with a focus on understanding the spatial and statistical characteristics of the RSSI values. The objective is to extract meaningful insights that can inform the selection and optimisation of algorithmic components in later stages. Both two-dimensional and three-dimensional visualisations are employed to explore the distribution of RSSI values across different reference points and access points. Additionally, this section examines signal variability, sparsity, and environmental noise, all of which significantly influence the performance of positioning algorithms. By establishing a clear understanding of the dataset's structure and signal behaviour, this foundational analysis supports the rational design and evaluation of the proposed methods.

In addition, this section presents the initial simulation results using the standard k-NN approach with default parameters such as $k = 1$. It establishes the baseline performance using Euclidean distance metrics and analyses their impacts on positioning accuracy. This baseline assessment is crucial for understanding the performance limitations of conventional approaches.

To model a simple system for IPS, the model was implemented using MATLAB R2023 (a), executed and tested on a Lenovo Laptop with an Intel(R) Core(TM) i5-8265u CPU @ 1.60 GHz, 1.80 GHz processor, 8 GB of RAM, and running

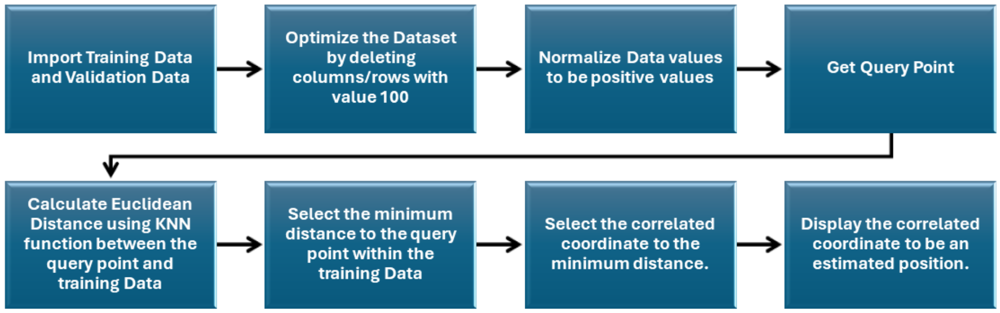


Figure 3.4: Modelling Process Flow

Windows 10, 64-bit. The modelling programming procedure, which processes the UJIIndoorLoc dataset detailed in Section 3.6 (comprising 19,861 training samples and 1,111 validation samples), consists of four main steps as follows:

1. Import Training and Validation Datasets
2. Pre-process Dataset
3. Implement Algorithm
4. Output the results

These steps were converted to a process flow diagram as shown in Figure 3.4. The flow diagram was then transformed into a model using MATLAB code, following the same sequence outlined in Algorithm 1.

Algorithm 1 K-NN

```
Load fingerprint dataset
Load validation dataset
Remove columns and rows with values equal to 100 from both datasets
Normalise the data to absolute values
Plot 3D scatter plot of fingerprint dataset and validation dataset
for each query point in the validation dataset do
    Find k nearest neighbours in the fingerprint data set using Euclidean distance
    Compute the mean position of k closest neighbors as estimated position
end for
Plot query points and estimated positions on 3D scatter plot
```

Database Exploration

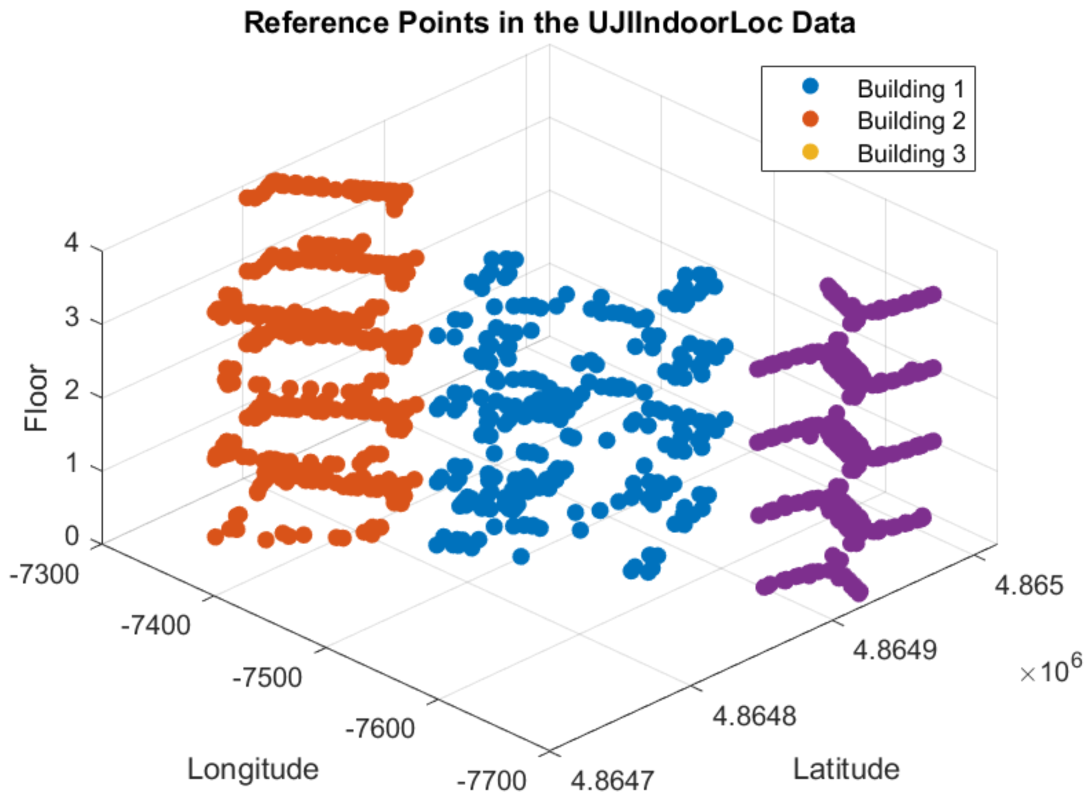


Figure 3.5: UJIIndoorLoc Training Datasets plotted in 3D for the three buildings

Figure 3.5 displays the coordinates of the entire dataset in a 3D plot, showcasing the layout across the three buildings. These coordinates, representing of RPs, are further visualised in 2D in Figure 3.6, providing insight into the spatial distribution and shape of the collected RPs. In particular, the arrangement of these RPs reflects the architectural structure of Jaume I University (Universitat Jaume I), as illustrated in Figure 3.7, where the dataset was originally collected.

Figure 3.8 highlights the distinctive characteristics of WAP fingerprinting at each reference point, illustrating the unique vector of values associated with each RP. These vectors serve as the basis for comparison against query points to compute the MD estimation position. Moreover, Figure 3.8 highlights the distinctive characteristics of WAP fingerprinting at each reference point. For each WAP shown (e.g., WAP1, WAP2), the y-axis depicts the processed positive RSSI value (representing signal strength magnitude as detailed in Section 3.6.4) recorded at different Reference Points (RPs, shown along the x-axis). The collection of these RSSI values from all WAPs at a

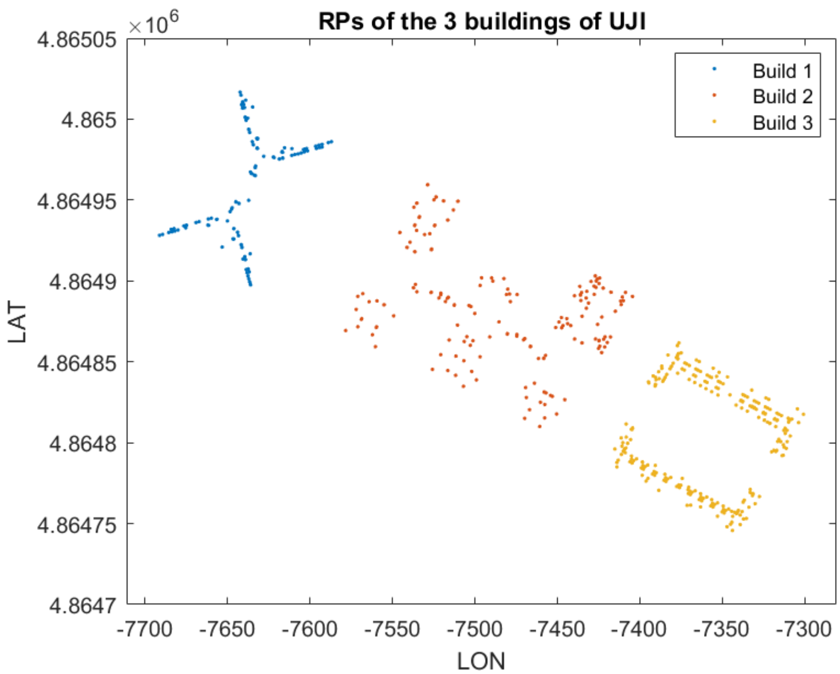


Figure 3.6: UJIIndoorLoc Training Datasets plotted in 2D for the three buildings

Figure 3.7: Jaume I University campus on Google Map

single RP forms its unique fingerprint vector, which serves as the basis for comparison against query points to compute the MD estimation position.

Figure 3.9 presents the signal distribution for both the dataset and the validation dataset, providing a visual representation of the RSSI values. In particular, the validation data set exhibits significantly fewer signal records compared to the original data set. The distribution of the dataset also reveals certain impractical values, such as those falling between 0 and -20, as well as values exceeding -100, indicative of very poor signal strength.

Figure 3.10 further elucidates additional features observed within both datasets, particularly focusing on the pattern of attributes by:

1. Provides a more comprehensive view of the overall distribution patterns across the entire dataset, while previous figures focused on specific aspects or subsets.

2. Illustrates how the ground truth labels (Longitude, Latitude, Floor, BuildingID) vary across the sequence of collected reference points. Understanding this label distribution is important for assessing the coverage and balance of the dataset used for training and validating the positioning algorithms.

3. Helps identify potential outliers, clusters, and density variations in the reference data that impact positioning accuracy in different regions of the environment.

4. Supports the analysis of data quality and consistency between training and validation sets, which is crucial for ensuring reliable algorithm performance.

Baseline Performance

We developed separate models for each attribute in the database to evaluate the baseline performance. The baseline performance is presented in Table 3.6:

Table 3.6: Baseline kNN Algorithm Results on UJI Datasets

Metrics	**BLD**	**FLO**	**Success**	**LAT**	**LON**	**MAE**
kNN	94.06	98.83	89.29	18.5163	25.2991	11.5182

3.8 Positioning Estimation

During the online positioning estimation stage, the algorithm estimates the position of an unknown RF fingerprint. The algorithm takes the RF fingerprint sampled at an unknown position as input and outputs the most probable position of the unknown fingerprint. To achieve this, the algorithm treats position estimation as

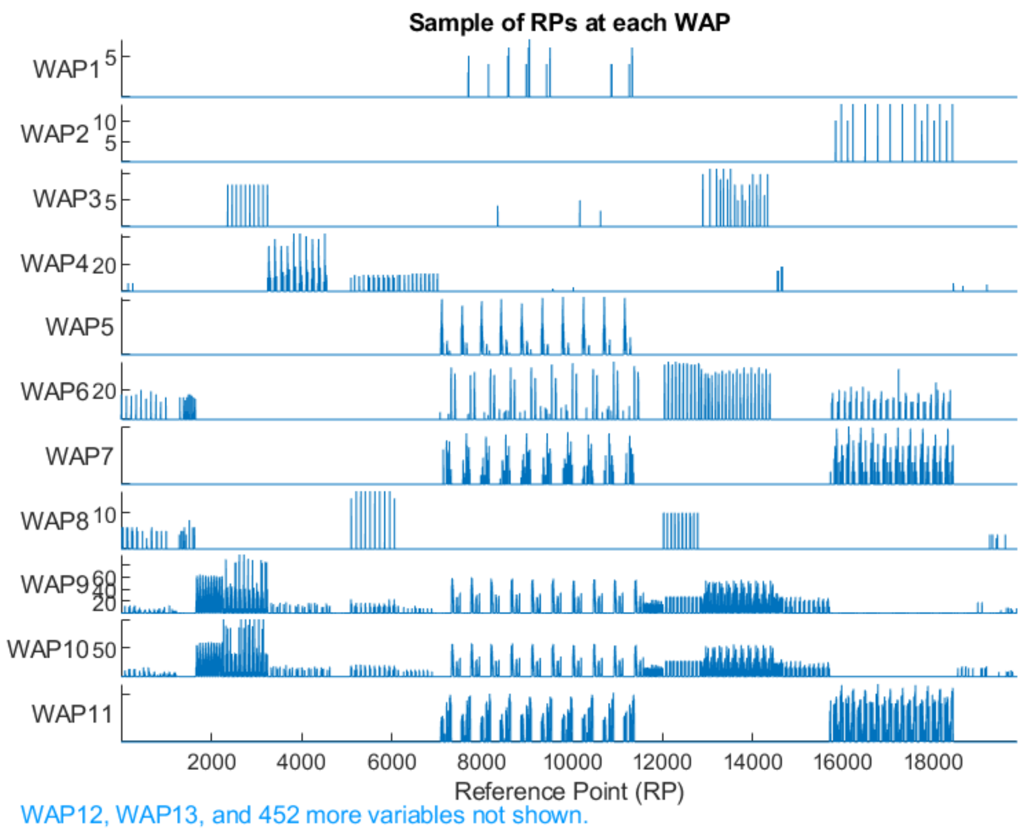


Figure 3.8: Sample of RPs shows RSSI at each particular WAP

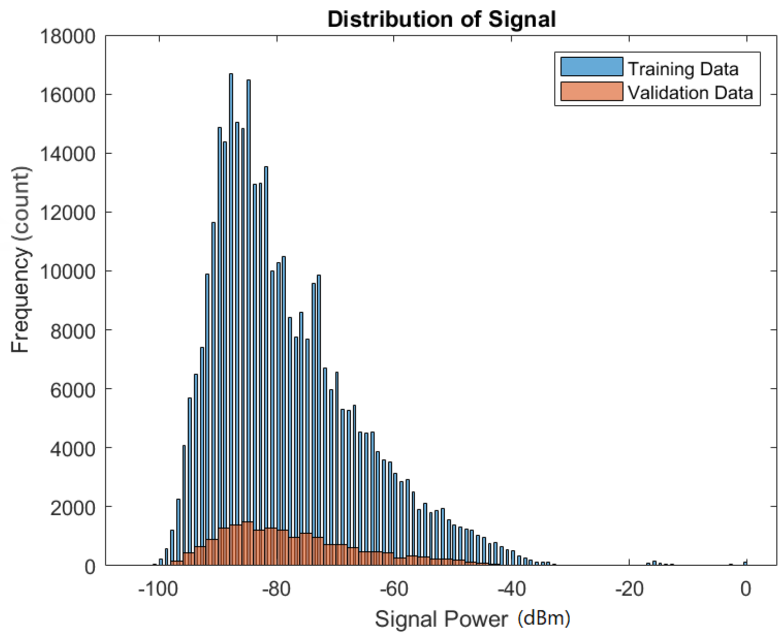


Figure 3.9: Distribution of Signal in Training Dataset and Validation Dataset

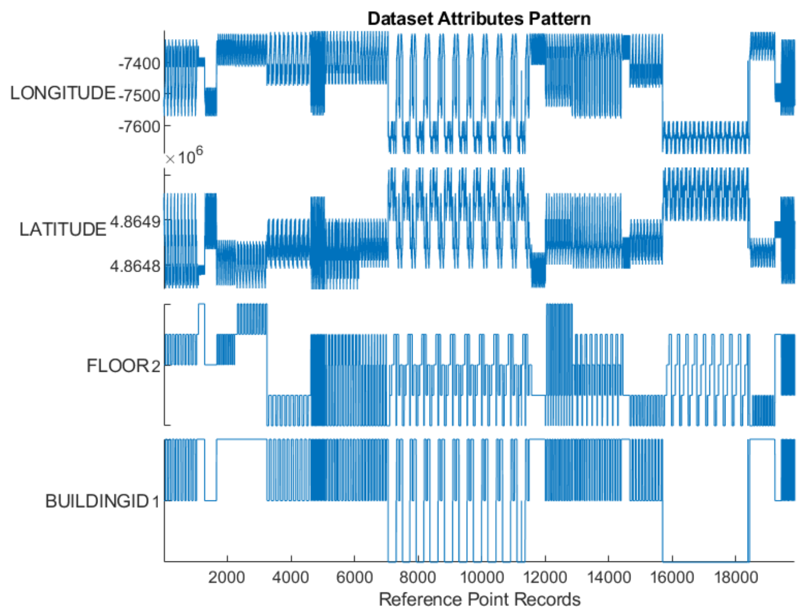
a classification problem. Each entry in the RMF represents a unique class of RF fingerprint that describes a particular position. The algorithm's objective is to classify the unknown RF fingerprint into one of these predefined classes based on specific optimisation criteria [157]. The choice of classifier and optimisation scheme depends on the information represented by RF fingerprints [157].

The architecture of the fingerprint database is quite straightforward, with the characteristic of the Reference Point based solely on the average Received Signal Strengths from each Access Point. Various algorithms are available for estimating the location of the MD. The fundamental technique among these is known as the Nearest Neighbour (NN) [141].

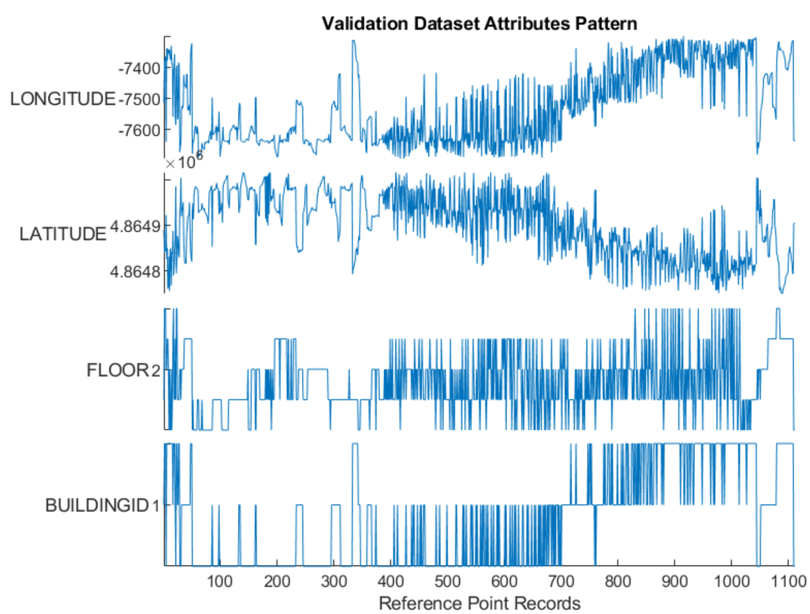
3.9 Wi-Fi Fingerprinting Technique

The Wi-Fi fingerprinting technique in indoor positioning uses the RSSI values of Wi-Fi APs to determine the position of a device. The process depicted in Figure 3.11 typically involves the following steps:

- **Data collection:** Collect Wi-Fi fingerprints from multiple positions in the indoor environment. The fingerprints should include the RSSI values of the APs and the corresponding position of the device that collects the data.



(a) Training Dataset Distribu



(b) Validation Datasets

Figure 3.10: Distribution of Fingerprinting

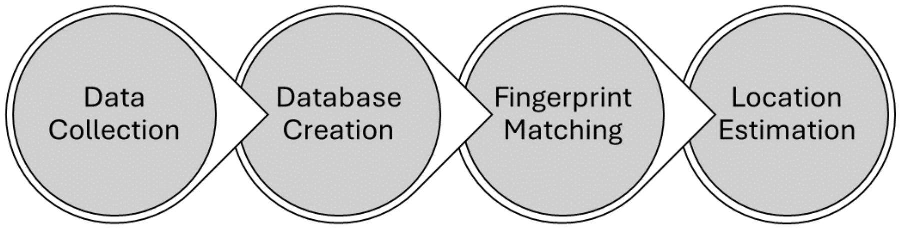


Figure 3.11: Fingerprinting Process

- **Database creation:** Create a Wi-Fi fingerprint database that contains the RSSI values of APs along with their corresponding positions.
- **Fingerprint matching:** When a device needs to determine its position, it scans nearby APs and measures their RSSI values. These values are then compared with the fingerprints in the database to find the best match.
- **Position estimation:** The position of the device is estimated based on the closest matching fingerprint.

After completing the offline phase of data collection, the online phase begins, creating a model for indoor positioning, as depicted in Figure 3.12. The basic steps of this process are as follows:

- **Feature selection:** Select the most relevant features from the collected fingerprints for use in the positioning algorithm. Common features include the mean, median, and standard deviation of the RSSI values of the APs.
- **Algorithm selection:** Select the appropriate positioning algorithm for the task. Common algorithms include k-NN, support vector machines (SVMs), and neural networks.
- **Model training:** Train the positioning algorithm with the collected fingerprints and selected features.
- **Model evaluation:** Evaluate the performance of the positioning algorithm using metrics such as accuracy, precision, and recall.
- **Model optimisation:** Optimise the algorithm by fine-tuning the parameters, selecting different features, or using alternative algorithms.

After the model is optimised, it can be deployed to estimate the position of devices based on their RSSI measurements in real-time.

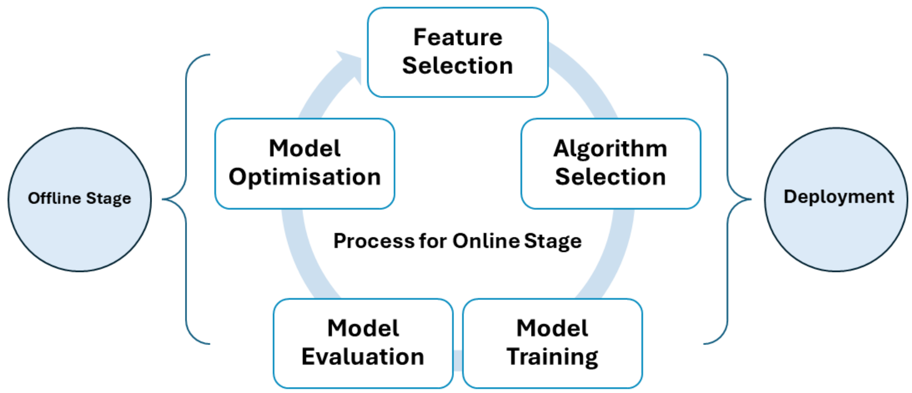


Figure 3.12: Process for the online stage

3.10 Wi-Fi Fingerprinting Accuracy

Under optimal conditions, certain Wi-Fi fingerprinting systems may achieve an average accuracy level of around 2-5 metres, and most Wi-Fi fingerprinting systems can accomplish this level of accuracy or better [187]. Wi-Fi fingerprinting has been proven to be highly accurate in indoor environments, achieving a point accuracy of 2 metres and a room accuracy of 98% [164]. However, accuracy may be affected by human behaviour and the distribution of RSSI. Studies have shown Wi-Fi fingerprinting accuracy between 2.0 and 2.5 metres [188]. When dealing with incomplete fingerprint databases, interpolation and extrapolation can be used to improve positioning accuracy. However, the best performance depends on more than just the average interpolation accuracy [189]. Large Wi-Fi fingerprinting positioning errors can be reduced by identifying and avoiding poor estimations. Hybrid methods that combine multiple technologies can be employed [123]. By incorporating weighted fusion and error-handling techniques during data acquisition, algorithms for Wi-Fi indoor positioning have shown greater accuracy and precision [190]. The accuracy of Wi-Fi fingerprinting can be reduced by the orientation of the devices during signal measurements. However, constructing RMFs considering various angular directions has been found to improve positioning accuracy [191].

Indoor positioning systems based on Wi-Fi fingerprinting exhibit varying degrees of precision but can achieve high accuracy under specific conditions. Implementing techniques such as frequency hopping, optimised fingerprint collection, and advanced algorithms can enhance accuracy through weighted fusion. Additionally, considering device orientation and using interpolation to fill gaps in the database can further enhance the performance of these systems. Even with obstacles, Wi-Fi fingerprinting remains a promising solution for indoor positioning, particularly when practicality is vital.

3.11 Deterministic Positioning Algorithms

Deterministic positioning algorithms in Wi-Fi fingerprinting aim to determine the position of a target device based on the similarity between the observed RSSI fingerprints and the fingerprints stored in the database. Several common algorithms are used for this purpose.

3.11.1 Nearest Neighbour (NN) Algorithms

1. **Nearest neighbour (NN)**: The Nearest Neighbour algorithm estimates the position of the target device by identifying the reference position in the database with the most similar RSSI fingerprint to the observed RSSI measurements. Mathematically, the estimated position $(\hat{x}, \hat{y})$ is determined as:

$$(\hat{x}, \hat{y}) = \arg\min_{(x_i, y_i)} \sqrt{\sum_{j=1}^{M} (RSSI_{ij} - z_j)^2}, \tag{3.11}$$

   where (x_i, y_i) represents the coordinates of the i^{th} reference position in the database, $RSSI_{ij}$ is the stored RSSI value from the j^{th} AP for the i^{th} reference position, and z_j is the currently observed RSSI value from the j^{th} AP by the target device. The summation is performed over all M Access Points that constitute the fingerprint vector. This equation identifies the reference position (x_i, y_i) whose fingerprint vector is closest (in Euclidean distance) to the observed RSSI vector $\boldsymbol{z}$.

2. **K-Nearest neighbours (k-NN)**:

   The K-Nearest-Neighbours algorithm extends the NN algorithm by considering the RSSI fingerprints of several reference positions. Calculates the position of the target device by averaging the coordinates of the K reference positions with the most similar RSSI fingerprints. Mathematically, the estimated position $(\hat{x}, \hat{y})$ is determined as:

$$(\hat{x}, \hat{y}) = \frac{1}{K} \sum_{i=1}^{K} (x_i, y_i), \tag{3.12}$$

   where (x_i, y_i) are the coordinates of the i^{th} reference position selected from the K nearest neighbours.

3. **Weighted K-Nearest neighbours (Wk-NN)**:

The Wk-NN algorithm applies weights to the reference positions based on the similarity of their RSSI fingerprints to the observed RSSI measurements. Calculates the position of the target device by computing a weighted average of the coordinates of the K reference positions. Mathematically, the estimated position $(\hat{x}, \hat{y})$ is determined as:

$$(\hat{x}, \hat{y}) = \frac{\sum_{i=1}^{K} w_i \cdot (x_i, y_i)}{\sum_{i=1}^{K} w_i}, \tag{3.13}$$

where w_i represents the weight assigned to the i^{th} reference position. The weights w_i are typically selected to be inversely proportional to the distance between the observed fingerprint and the reference fingerprint, giving higher importance to closer reference points. A common approach is to calculate w_i as:

$$w_i = \frac{1}{d_i^p}, \tag{3.14}$$

where d_i is the Euclidean distance between the target fingerprint and the i^{th} reference fingerprint, and p is a power parameter (typically set to 1 or 2) that controls the influence of distance on the weighting. When $p = 1$, the weights are inversely proportional to distance, and when $p = 2$, they are inversely proportional to the squared distance.

An alternative approach is to use Gaussian kernel weights:

$$w_i = e^{-\frac{d_i^2}{2\sigma^2}}, \tag{3.15}$$

where σ is a parameter that controls the width of the Gaussian kernel.

The choice of weighting scheme significantly impacts the performance of the Wk-NN algorithm, with optimal values for p or σ often determined empirically based on the specific characteristics of the indoor environment and the distribution of reference points.

These deterministic positioning algorithms provide a straightforward approach to estimating the position of a target device based on the similarity of RSSI fingerprints. They are widely used in indoor Wi-Fi fingerprinting positioning systems because of their simplicity and effectiveness.

NN algorithms are among the most widely used techniques for indoor positioning. They rely on the principle that a device's position can be estimated by identifying the closest known RPs and considering their relative positions. This approach is particularly effective in indoor environments where obstructions make GPS signals unavailable or unreliable.

3.11.2 Working Principle of NN Algorithms

NN algorithms are a class of deterministic positioning algorithms commonly used in Wi-Fi fingerprinting for indoor positioning. The working principle of NN algorithms is straightforward and intuitive.

1. **Database Construction**:

   The first step in using NN algorithms is to construct a database of RSSI fingerprints. This database contains RSSI measurements from multiple reference positions within the indoor environment. Each entry in the database contains RSSI measurements from all available APs at a specific reference position.

2. **Observation**:

   When a target device needs to be localised, it takes measurements of the RSSI from nearby APs in its vicinity. These observed RSSI measurements serve as input to the NN algorithm for estimating the device's position.

3. **Nearest neighbour Search**:

   The NN algorithm identifies the reference position in the database with the RSSI fingerprint most similar to the RSSI measurements observed by the target device. This is typically done by calculating the distance function, such as Euclidean distance, between the observed RSSI measurements and the RSSI fingerprints stored in the database.

4. **Position Estimation**:

   Once the nearest-neighbour reference position is identified, the NN algorithm estimates the position of the target device as the coordinates of this reference position. In other words, the estimated position of the target device is assumed to be the same as the known position of the reference point with the most similar RSSI fingerprint.

The inherent simplicity and implementation efficiency of NN algorithms constitute a principal advantage, rendering them highly suitable for real-time applications within the domain of indoor positioning. These algorithms exhibit a robustness to noise present in RSSI measurements, thereby reducing susceptibility to variations in signal strength. Nevertheless, NN algorithms may encounter inaccuracies when the captured RSSI measurements do not precisely correspond to any fingerprints within the database or amidst significant environmental changes. The precision of NN algorithms is contingent upon the quality of the RMF, necessitating meticulous data collection and preprocessing. Consequently, NN algorithms may face challenges related to scalability when applied to extensive databases.

3.11.3 Addressing Scalability Issues in NN Algorithms

Scalability issues with large databases in NN algorithms can be mitigated through several approaches:

1. **Dimensionality Reduction**: High-dimensional feature spaces can increase computational complexity and memory requirements in k-NN algorithms. Dimensionality reduction techniques, such as Principal Component Analysis (PCA) or feature selection methods, can help reduce the number of features while preserving the most relevant information. These techniques can improve the scalability of k-NN algorithms by reducing the computational burden.

2. **Approximate Nearest Neighbour Search**: Traditional k-NN algorithms perform an exhaustive search over the entire database to find the nearest neighbours for a given query, which can be computationally expensive for large databases. Approximate nearest-neighbour search algorithms, such as locality-sensitive hashing (LSH) or tree-based methods (e.g., KD-trees), offer more efficient methods to search for nearest neighbours in high-dimensional spaces. These methods provide approximate solutions at a lower computational cost, making the k-NN algorithms more scalable.

3. **Data Partitioning and Indexing**: Dividing the database into smaller partitions or utilising indexing structures can help accelerate the search process in k-NN algorithms. Techniques like spatial indexing (e.g., R-tree, Quadtree) or hash-based indexing can organise the database into hierarchical structures, enabling faster retrieval of nearest neighbours. By partitioning and indexing the data effectively, k-NN algorithms can achieve better scalability with large databases.

4. **Parallel and Distributed Computing**: Leveraging parallel and distributed computing frameworks can distribute the computational workload across multiple processors or nodes, enabling faster processing of k-NN queries. Techniques like MapReduce or Spark can be used to parallelise the k-NN search process and scale it to large datasets. By harnessing the power of parallel and distributed computing, k-NN algorithms can handle large databases more efficiently.

By employing these strategies, scalability issues with large databases in k-NN algorithms can be mitigated, allowing for efficient and effective indoor positioning in Wi-Fi fingerprinting systems even with large datasets.

3.11.4 Distance Function

The choice of distance metric significantly influences the outcome of NN-based algorithms. In this work, we investigate multiple distance functions—including *Euclidean*, *Manhattan*, and *Minkowski* distances—to evaluate their impact on algorithm sensitivity to spatial variations in indoor Wi-Fi signal propagation.

Building on prior studies [186], [192], we assess the performance of common distance metrics, including *Cityblock*, *Euclidean*, *Minkowski*, *Cosine*, and *Correlation*, across several dataset configurations. Notably, *Cityblock*, *Euclidean*, and *Minkowski* distances belong to the Minkowski family, where the general form is given by:

$$D_p(\boldsymbol{p}, \boldsymbol{q}) = \left(\sum_{j=1}^{M} |p_j - q_j|^p \right)^{\frac{1}{p}}, \quad \forall p \in \mathbb{N}^+, \tag{3.16}$$

where $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^d$ are two signal strength vectors representing locations in the indoor environment. The parameter p determines the type of distance metric: when $p = 1$, the metric reduces to the *Cityblock* (Manhattan) distance; for $p = 2$, it becomes the standard *Euclidean* distance. Values $p = 3, 4, 5$ yield higher-order Minkowski distances.

Cosine Distance. This metric belongs to the inner product family and is based on the angular similarity between vectors. It is particularly useful when comparing patterns of signal strength rather than their absolute values. Given two vectors $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^M$, where $\boldsymbol{p} \cdot \boldsymbol{q} = \sum_{i=1}^{M} P_i Q_i$ is the dot product of the vectors, and $\|\boldsymbol{p}\|$ and $\|\boldsymbol{q}\|$ are their Euclidean norms. M is the number of WiFi features (e.g., APs) considered in the environment. The cosine distance is then derived as:

$$\cos(\boldsymbol{p}, \boldsymbol{q}) = 1 - \frac{\sum_{i=1}^{M} P_i Q_i}{\sqrt{\sum_{i=1}^{M} P_i^2} \cdot \sqrt{\sum_{i=1}^{M} Q_i^2}}. \tag{3.17}$$

This formulation measures the cosine of the angle between the two vectors. A value of 0 indicates identical directions (maximum similarity), while a value of 1 corresponds to orthogonal vectors (no similarity). In indoor positioning, cosine distance is advantageous because it captures the relative pattern of RSSI values rather than their magnitude, which may vary due to device or environmental differences.

Correlation Distance. This metric evaluates the linear correlation between two vectors by incorporating mean normalisation. It is derived from the Pearson correlation coefficient, rescaled to fall within the range [0,1]. Given vectors $\boldsymbol{p}, \boldsymbol{q} \in \mathbb{R}^M$, the correlation distance is computed as:

$$d_{corr}(\boldsymbol{p}, \boldsymbol{q}) = \frac{1}{2}\left(1 - \frac{\sum_{j=1}^{M}(p_j q_j) - M\bar{p}\bar{q}}{\sqrt{\sum_{j=1}^{M} p_j^2 - M\bar{p}^2} \cdot \sqrt{\sum_{j=1}^{M} q_j^2 - M\bar{q}^2}}\right), \tag{3.18}$$

where $\bar{p}$ and $\bar{q}$ are the means of vectors $\boldsymbol{p}$ and $\boldsymbol{q}$, respectively. The numerator represents the covariance between the vectors, and the denominator is the product of their standard deviations.

This metric is robust against linear shifts in signal strength, focusing on the degree to which the vectors change together. A correlation distance of 0 implies perfect positive correlation, while a value near 1 indicates a lack of correlation or inverse trends.

3.12 Challenges in Wi-Fi Fingerprinting

The challenges inherent in Wi-Fi fingerprinting are multifaceted and have consequently attracted considerable scholarly attention. A primary challenge lies in the variability of RSSI measurements, which significantly undermines the accuracy and reliability of Wi-Fi fingerprinting systems [27]. This variability emanates from fluctuations within the radio signal environment, resulting in discrepancies between online and offline RSSI measurements [27].

Moreover, the deployment of Wi-Fi fingerprinting systems in real-world environments introduces further challenges, including the installation of infrastructure, calibration procedures, and the complexity of modelling building floor plans [193]. These obstacles can hinder the seamless operation and accuracy of Wi-Fi fingerprinting systems in practical applications. Additionally, Wi-Fi fingerprinting's dependence on the correlation between online and offline RSSI measurements exacerbates the difficulties encountered [27]. The necessity for continuous updates and maintenance of Radio Map Files (RMFs) to ensure optimal performance adds another layer of complexity to these systems [193].

Furthermore, the development of highly accurate fingerprinting-only solutions remains challenging, underscoring the need for further advancements in algorithmic development and system optimisation [194]. Wi-Fi fingerprinting faces numerous challenges that impact the system's accuracy and reliability. A critical issue is the selection of an appropriate distance metric for similarity estimation. Previous studies predominantly relied on Euclidean distance with raw, unprocessed data, without thoroughly exploring optimal distance functions or data representation methods [186]. This lack of comprehensive analysis can lead to suboptimal performance in Wi-Fi fingerprinting systems. Additionally, the presence of noise and missing signals in RSSI measurements poses a significant challenge. Traditional distance metrics such as Euclidean distance may not be the most suitable measures of similarity due to these factors [195]. Furthermore, the calculation of distances using conventional norms like Manhattan, Euclidean, or Mahalanobis norms can be affected by inherent RSSI variability, further complicating the accuracy of these systems [71], [196].

Additionally, the maintenance and updating of RMFs pose practical challenges. The continual requirement for RMF updates to accommodate environmental changes and ensure system accuracy introduces complexity in the deployment and operation of Wi-Fi fingerprinting-based indoor positioning systems. The nonlinear relationship between RSSI and physical distance can cause positioning errors when using traditional metrics such as Euclidean or Manhattan distances [130]. Device heterogeneity in Wi-Fi fingerprinting constitutes another major drawback, presenting a significant challenge.

The diversity of devices used for signal collection can result in variations in RSSI measurements, affecting the accuracy and consistency of the fingerprinting process. As discussed in [108], they address the challenges of calibrating heterogeneous devices for signal collection, emphasising the importance of standardising RSSI measurements for accurate positioning. Moreover, the presence of various devices can cause inconsistencies in signal interpretation and processing. In [197], highlight the vulnerability of Wi-Fi fingerprinting systems to over-the-air adversarial attacks, underscoring the potential security breaches that malicious devices can exploit.

The use of different devices with varying capabilities and vulnerabilities can undermine the overall integrity and reliability of Wi-Fi fingerprint-based indoor positioning systems. In [165], explore the impact of device heterogeneity on the performance of Wi-Fi fingerprinting algorithms. Their research compares various methods, including Nearest-Neighbour algorithms, Gaussian kernels, Bayesian models, neural networks, and deep learning, to mitigate the challenges posed by device heterogeneity. Adapting algorithms to accommodate the diverse characteristics of devices used for signal collection is essential for achieving accurate and reliable indoor positioning. These challenges can be summarised as follows:

1. **Localisation Accuracy**: Achieving high levels of localisation accuracy in Wi-Fi fingerprinting systems remains challenging, especially in complex indoor environments with multipath propagation and signal attenuation. Enhancing the robustness and reliability of localisation algorithms is essential for real-world deployment.

2. **Scalability**: Managing large-scale fingerprint databases can be particularly

challenging, especially in environments with a high density of APs and reference positions. Scalability issues arise in database construction, storage, and retrieval, impacting the efficiency and performance of Wi-Fi fingerprinting systems.

3. **Device Heterogeneity**: Device heterogeneity presents a complex challenge in Wi-Fi fingerprinting systems. The use of multiple types of devices with different hardware and software configurations can further complicate the system's calibration and performance metrics. In this case, employing a multi-device calibration process and algorithms designed to adapt to different device types can enhance the overall performance and accuracy of the system.

4. **Variability in RSSI Measurements**: RSSI measurements can vary due to factors such as obstructions, multipath interference, and environmental changes. These variations can impact the accuracy of fingerprinting-based positioning systems, leading to localisation errors.

5. **Dynamic Environments**: The characteristics of the Wi-Fi signal can change over time due to factors such as device mobility, user activity, and RF interference. Dynamic environments introduce challenges for maintaining the accuracy of fingerprint databases and require adaptive algorithms to cope with changes.

Addressing these challenges requires interdisciplinary research efforts that involve signal processing, machine learning, networking, and privacy considerations. By overcoming these challenges, Wi-Fi fingerprinting can continue to evolve as a reliable and accurate indoor positioning technology. In this research, we address the three challenges of localisation accuracy, scalability, and device heterogeneity.

3.13 Selected Approaches for This Research

Based on the comprehensive review of Wi-Fi fingerprinting techniques presented in this chapter, this research adopts specific approaches for the positioning algorithms and system design presented in subsequent chapters. The selection of these methods is informed by the analysis of their strengths, limitations, and suitability for addressing the key challenges identified in Section 3.12.

For positioning estimation, this research primarily builds upon the k-Nearest Neighbours (k-NN) algorithm described in Section 3.11.1, due to its demonstrated balance between accuracy and computational efficiency. Specifically, we adopt the weighted k-NN variant (equation 3.13) as our baseline algorithm, which we extend and enhance in Chapter 4 with novel optimisations to improve both accuracy and scalability.

To address the challenge of radio map complexity, this research adopts selective optimisation strategies, with a particular focus on enhancing accuracy and mitigating device and environmental heterogeneity. Chapter 5 introduces a novel radio map optimisation technique that builds on these foundations, incorporating additional enhancements to improve scalability. Chapter 6 presents a cloud-based indoor positioning system (CB-IPS) framework, developed using MATLAB and the UJIIndoorLoc dataset, designed to enhance the scalability, efficiency, and accuracy of Wi-Fi fingerprinting-based IPS. By integrating cloud and edge computing, the proposed framework optimises computational resource usage and introduces a caching mechanism to reduce execution times, thereby further improving the system's scalability and responsiveness.

For evaluation metrics, we primarily utilise the hitting rate (equations 3.1) and RMSE (equation 3.4) as defined in Section 3.2, which allow for a comprehensive assessment of both classification accuracy and positioning precision. These metrics form the basis for the experimental evaluation presented in Chapters 4,5, and 6.

The subsequent chapters present our novel contributions that extend these foundational approaches to address the identified challenges of localisation accuracy, scalability, and device heterogeneity in Wi-Fi fingerprinting systems.

3.14 Summary and Conclusion

3.14.1 Summary

This chapter has provided a detailed examination of the RSSI-based Wi-Fi fingerprinting method for indoor positioning. A key contribution of this chapter was the establishment of a unified **System Model, Notation, and Evaluation Framework** (Section 3.2), which defines the general fingerprinting system concept, standardises mathematical notations, and outlines the performance evaluation metrics (such as Hitting Rate, MAE, and RMSE) that will be consistently applied throughout the subsequent chapters of this thesis.

Beyond this foundational framework, the chapter delineated the integral components of the Wi-Fi fingerprinting process itself, including radio map (RMF) construction from RSSI measurements and the principles of positioning algorithms. It underscored a deterministic methodology, chosen for its balance of simplicity and design efficiency. While acknowledging that Wi-Fi fingerprinting can achieve an average positioning accuracy in the range of 2-5 metres, the chapter also highlighted inherent challenges such as signal shadowing effects and environmental dynamics that can impact efficacy.

Furthermore, this chapter explored various deterministic positioning algorithms, with a primary focus on Nearest Neighbour (NN) techniques and their variants, while also briefly acknowledging alternative approaches to enhance accuracy. Despite its advantages, Wi-Fi fingerprinting faces limitations, notably the need for potentially intensive initial measurements and periodic updates to the RMF due to environmental changes.

The chapter concluded by accentuating the imperative to address ongoing challenges such as optimal distance metric selection, mitigation of noise in RSSI measurements, robust RMF maintenance strategies, and managing device heterogeneity, all of which are crucial for bolstering the reliability and accuracy of Wi-Fi fingerprinting systems in diverse indoor positioning contexts. It underscored the necessity for continued research to refine these methods, aiming for advancements that promote exceptional accuracy and dependability in real-world indoor localisation systems.

3.14.2 Conclusion

Indoor Wi-Fi fingerprinting, as detailed in this chapter, represents a promising and widely adopted technology for enhancing the accuracy and reliability of Indoor Positioning Systems (IPSs). This chapter has not only explored the operational principles and inherent challenges of this technique but has also established a crucial methodological foundation for this thesis by introducing a unified system model, consistent mathematical notation, and a standard set of evaluation metrics. This framework is essential for the rigorous development and assessment of the algorithmic and architectural contributions presented in subsequent chapters.

The chapter emphasised the application of Wi-Fi fingerprinting within indoor localisation contexts, accentuating the ongoing significance of addressing challenges related to distance metric selection, RSSI noise, RMF lifecycle management, and device heterogeneity to improve overall system reliability and accuracy. Moreover, the discussion highlighted the potential of leveraging machine learning algorithms and systematic design approaches to tackle these pressing issues, paving the way for more robust and efficient indoor positioning solutions.

Chapter 4

Positioning Estimation Algorithm

The chapter provides an overview of the Positioning Estimation Algorithm utilised within the Wi-Fi Fingerprinting technique. It begins with an introduction to the methodology of the research and introduces the UJIIndorLoc database of RSSI signals used for indoor positioning. Subsequently, it explores an examination of the k-NN algorithm employed as the baseline algorithm for comparison purposes, and finally, tuned k-NN and Wk-NN algorithms are examined and discussed.

4.1 Introduction

Wi-Fi fingerprinting is a prominent technique for indoor positioning, primarily due to its reliance on existing wireless infrastructure and the widespread availability of Wi-Fi enabled devices. As discussed in Chapter 3, deterministic algorithms, particularly the k-Nearest Neighbour (k-NN) algorithm and its variants, offer a balance of simplicity and effectiveness for position estimation. However, their performance is highly sensitive to parameter choices and the specific characteristics of the operational environment and dataset.

This chapter focuses on significantly enhancing the accuracy and reliability of Wi-Fi fingerprinting-based indoor positioning by systematically optimising existing k-NN and Weighted k-NN (Wk-NN) algorithms. The UJIIndoorLoc dataset [15], a comprehensive multi-building, multi-floor benchmark, serves as the empirical basis for this investigation.

The **primary contributions** of this chapter are:

- A rigorous empirical analysis of crucial k-NN/Wk-NN hyperparameters, including the number of neighbours (k), various distance metrics (e.g., Euclidean, Manhattan, Cosine, Correlation), and different RSSI data representation schemes (positive, exponential, powered).

- A detailed investigation into the impact of dataset configurations, such as evaluating performance on the complete dataset versus individual building subsets, to understand algorithm scalability and sensitivity.
- The systematic application of distance weighting schemes within the Wk-NN framework to improve position estimation by giving more influence to closer or more reliable neighbours.
- The development of an optimally tuned Wk-NN configuration, informed by these analyses, which demonstrates substantial improvements in positioning accuracy (Mean Absolute Error) and classification success (Building and Floor Hitting Rate) compared to baseline k-NN implementations and a range of existing studies utilising the UJIIndoorLoc dataset.

The chapter begins by briefly introducing the UJIIndoorLoc dataset (Section 4.2) and the baseline k-NN algorithm (Section 4.3). It then details the methodologies for enhancing these algorithms (Section 4.4) and presents a comprehensive experimental evaluation of the optimised approaches, including a comparative analysis with state-of-the-art results (Section 4.5). The findings provide valuable insights into achieving optimal performance from k-NN based fingerprinting systems.

4.2 Dataset Overview

Various studies in the literature consistently utilise the UJIIndoorLoc database for indoor positioning, highlighting its widespread adoption and significance in specifying the input and output structures of proposed models. Researchers have designed and validated their algorithms against this publicly available database, establishing it as a benchmark for indoor positioning systems.

Table 4.1 provides a summary of related work, showing the various configurations of the UJIIndoorLoc database used in different studies. Notably, [15], [186], who initiated the UJI datasets, highlight the effectiveness of the k-NN algorithm, with varying parameters yielding different success rates and errors. Subsequent work explores different algorithms and configurations, demonstrating the versatility of machine learning techniques on UJI datasets to solve indoor localisation problems. Therefore, Table 4.1 not only summarises previous findings but also establishes key performance benchmarks from the literature that will be used in Section 4.5.4 to evaluate the advancements offered by the optimised algorithms developed in this chapter.

However, it is worth mentioning that not all the studies mentioned here use the same dataset configurations and methods, which may affect comparisons of their findings with others. For example, [198] used the test data sets that were provided

Table 4.1: Comparison of UJI Dataset Results

Reference	**Success (%)**			**Error (m)**
	BLD	**FLO**	**Mean**	
Torres-Sospedra et al. (2014) [15]	-	-	89.92	7.90
Torres-Sospedra et al. (2015) [186]	-	-	95.2	6.19
RTLS@UM: Moreira et al. (2015) [198]	100	93.74	-	6.20
Nowicki & Wietrzykowski (2017) [199]	-	-	92	-
Ibrahim et al. (2018) [200]	100	100	-	2.77
Hybloc: Akram et al. (2018) [201]	-	-	85	6.29
Gan et al. (2019) [202]	100	95.41	-	6.40
CNNLoc: Song et al. (2019) [22]	100	96.03	-	11.78
Liu et al. (2021) [203]	99.64	91.18	-	8.39
CCpos: Qin et al. (2021) [204]	99.6	95.3	-	12.4
Cao et al. (2021) [205]	-	99.54	-	3.46
Elesawi et al. (2021) [206]	100	95.23	-	8.62
DeepLocBox: Laska & Blankenbach (2021) [207]	99.64	92.62	-	9.07
Tang et al. (2022) [208]	100	94.20	-	8.42
EA-CNN: Alitaleshi et al. (2023) [209]	-	96.31	-	8.34

as part of the UJI data sets during the EVAAL competition [210]. While the results obtained from [200], the datasets were manipulated to obtain RSSI time-series readings, and then the new dataset was split. In [201], a new generated attribute named Room ID consists of Building ID, Floor ID, and Space ID, was used. Furthermore, [208] splits the validation dataset into a new validation and test set. Therefore, in the absence of a standardised evaluation framework, direct comparison of results among these studies remains methodologically unreliable.

4.3 Baseline k-NN Algorithm

As outlined by [186], K-NN operates as a distance-based classifier, wherein a current sample is compared to all labelled samples stored within a database. This requires the establishment of a comprehensive database, commonly referred to as a training set, where all samples are appropriately labelled.

The parameter ‘k’ in k-NN represents the number of nearest neighbours considered when estimating the position of a query point. In the context of indoor positioning, these samples typically consist of Wi-Fi fingerprints, represented as vectors containing WAP intensities, while the labels correspond to numerical values associated with real-world coordinates such as longitude, latitude, altitude/floor, and building data.

The selection of an appropriate value for ‘k’ is critical to the performance of the algorithm. A small value of k (such as k=1) may lead to overfitting and vulnerability to noise in the training data, while a large value may result in oversmoothing and loss of local patterns in the signal space. As will be demonstrated in Section 4.4.1, the optimal value of k depends on various factors, including dataset characteristics, signal representation, and distance metrics employed.

The k-NN algorithm, with parameters such as k = 1 and the Manhattan distance metric known as City Block, is often chosen as a baseline for comparison due to its simplicity and effectiveness[185].

The k-NN algorithm operates by evaluating the distance between a query point or position and all stored fingerprints within the training set. Subsequently, the algorithm assigns the query position to the position, comprising the longitude, latitude, and altitude of the training fingerprint that demonstrates the shortest distance or the highest similarity, employing Equation 3.12.

4.3.1 Performance Evaluation

The performance of the k-NN algorithm and its variants is assessed using the standard metrics defined in Section 3.2.2. For classification tasks such as building and floor identification, the Hitting Rate (HR, Equation 3.1) is reported. For the regression task of estimating 2D coordinates (Longitude and Latitude), the Mean Absolute Error (MAE, Equation 3.3) and Root Mean Squared Error (RMSE, Equation 3.4) are primary indicators, calculated from individual positioning errors (Equation 3.2).

4.4 Enhancement Methodology

This section outlines our methodology for enhancing the performance of the k-NN and Wk-NN algorithms in the context of indoor positioning using Wi-Fi fingerprinting. Rather than proposing a novel algorithm, our contribution lies in the systematic tuning of existing algorithmic parameters—specifically the k-value and distance weighting schemes—and evaluating their impact on positioning accuracy using the UJIIndoorLoc dataset. This approach ensures a reproducible and data-driven enhancement of the baseline algorithms.

The machine learning approach utilised here is based on an enhanced k-Nearest Neighbours (k-NN) algorithm, incorporating systematic parameter optimisation. This approach belongs to the category of *instance-based learning* (or memory-based learning), as opposed to *model-based* methods.

Specifically, our approach:

- **Extends the basic k-NN algorithm**: It builds upon the traditional k-NN method, which estimates locations based on the k most similar fingerprints in the training database.

- **Incorporates weighted averaging (Wk-NN)**: Rather than assigning equal influence to all k neighbours, we apply weights according to their similarity to the query point, giving more prominence to closer neighbours.
- **Optimizes multiple parameters**, including:
  - The number of neighbours (k)
  - The weight function parameter (w)
  - The choice of distance metric (e.g., Euclidean, Manhattan, Cosine)

Although the core methodology remains grounded in instance-based k-NN, these enhancements yield a significantly more sophisticated and effective localisation framework. The principal innovation lies in the empirical optimisation of parameters and the strategic selection of distance metrics, validated through experiments conducted on the `UJIIndoorLoc` database.

In contrast to model-based techniques, such as neural networks or support vector machines, which construct explicit predictive models during training, our enhanced k-NN approach retains the training data and performs computations at the prediction stage. This design offers advantages in terms of interpretability, adaptability to newly acquired data, and improved handling of non-linear spatial relationships characteristic of indoor positioning environments.

In our endeavour to optimise k-NN and Wk-NN algorithms for Wi-Fi fingerprinting in indoor positioning, significant emphasis is placed on calibrating their parameters. Through the exploration of various tuning techniques, we strive to improve the accuracy and efficiency of these algorithms, thereby advancing Wi-Fi fingerprinting technology for indoor positioning.

The outlined procedure describes the key components of our methodology:

4.4.1 Tuning the k-value for k-NN & Wk-NN

In k-NN and Wk-NN, the parameter k denotes the number of samples from the fingerprint dataset. Setting a low k-value, such as 1, may be insufficient, as it relies on only a single sample to estimate the final position. In contrast, a high k-value can degrade the model performance. Previous studies have often employed a fixed k-value across all models evaluated. However, our observations indicate that the optimal k-value may vary for each class or model, rather than being universally applicable across algorithms.

Therefore, we adopt an approach where we test different k-values ranging from 1 to 25. For each experimental configuration, we generate a model using various k-values and evaluate its performance. This iterative process allows us to determine the most effective k-value for each specific model, refining the overall accuracy of the algorithms.

4.4.2 Dataset Size and Configuration Analysis

The size of the dataset emerges as a crucial factor influencing the generalisability of matching algorithms. To elucidate the impact of database size on the performance of k-NN and Wk-NN algorithms, we conduct comprehensive experiments with varied configurations, ranging from subsets of the UJIIndoorLoc dataset to its entirety. This investigation aims to assess the scalability of the algorithms concerning the volume of available training data.

The UJIIndoorLoc database includes data from three buildings, namely BLD0, BLD1, and BLD2, each featuring multiple floors. Specifically, BLD0 and BLD1 have four floors, while BLD2 has five floors. Our exploration entails experimentation with diverse configurations, including both individual buildings and the complete dataset.

The datasets are represented as fixed-size vectors, where each index corresponds to 520 WAPs deployed across the three buildings at Jaume I University, Spain. These vectors encapsulate the original RSSI intensity values, ranging from 0 (indicating the highest signal) to -104 (representing the lowest signal) in decibels-milliwatts (dBm), with a default value of 100 dBm assigned for undetected WAPs [15].

In configuring the dataset, we examine both the complete dataset and each building separately to evaluate the performance of the algorithms under varying data sizes and compositions.

4.4.3 Incorporating Distance Weighting Schemes

In the k-NN algorithm, the distance weight (w) is uniform across all neighbours considered. Conversely, in Wk-NN, two distinct distance weighting schemes are employed: inverse distance and squared inverse distance, which are commonly used to formulate the Weighted k-NN variant. While k-NN treats all neighbours with equal importance, Wk-NN introduces a weighting mechanism that assigns varying degrees of importance to neighbouring points based on their proximity in signal space.

Our investigation explores the impact of distance weighting within the k-NN algorithm by examining different weight functions and configurations. This process is crucial to understanding how such weighting improves the adaptability of the algorithm to fluctuations in signal strengths and enhances the precision of position estimation. These investigations are conducted using the optimal parameters identified in prior stages, ensuring a systematic and rigorous evaluation of algorithmic performance.

In the following equations, let $\boldsymbol{p} \in \mathbb{R}^m$ denote the fingerprint vector of the target sample (i.e., the point being localised), and $\boldsymbol{q} \in \mathbb{R}^m$ denote the fingerprint vector of one of its k nearest neighbours, where m is the number of signal features. The Euclidean distance d between $\boldsymbol{p}$ and $\boldsymbol{q}$ is computed as:

$$d = \|\boldsymbol{p} - \boldsymbol{q}\|_2 = \sqrt{\sum_{j=1}^{m} (p_j - q_j)^2}, \tag{4.1}$$

where p_j and q_j are the j-th signal strength values of vectors $\boldsymbol{p}$ and $\boldsymbol{q}$, respectively. The two weighting schemes are then defined as follows:

$$w_{\text{inverse}}(\boldsymbol{p}, \boldsymbol{q}) = \frac{1}{d}, \tag{4.2}$$

$$w_{\text{squared inverse}}(\boldsymbol{p}, \boldsymbol{q}) = \frac{1}{d^2}, \tag{4.3}$$

where w_{inverse} and $w_{\text{squared inverse}}$ represent the weights assigned to neighbour $\boldsymbol{q}$ based on its distance from the target point $\boldsymbol{p}$. These schemes ensure that closer neighbours contribute more significantly to the final estimated location than those further away.

The overall procedure for parameter tuning and algorithmic enhancement, as detailed throughout this Section, is summarised in the pseudo-code presented in Algorithm 2, which outlines the systematic exploration of k-values, distance weighting schemes, and dataset configurations. This structured approach ensures reproducibility and clarity in evaluating the impact of each optimisation step.

Integrating these methodological components provides a holistic understanding of the factors affecting the performance of k-NN and Wk-NN algorithms in Wi-Fi fingerprint-based indoor positioning. In the following sections, we will present the experimental results and engage in an in-depth discussion regarding the implications of our findings on the optimisation of these algorithms for real-world applications.

Algorithm 2 Parameter Tuning and Enhancement for k-NN and Wk-NN Algorithms

Input: UJIIndoorLoc dataset, k-values range $K = \{1, 2, \ldots, 25\}$, distance weight types $W = \{$None, Inverse, Squared Inverse$\}$
Output: Optimised model accuracy and parameter configuration

for each building or dataset configuration **do**
    Extract training and testing sets
    for each algorithm in {k-NN, Wk-NN} **do**
        for each $k \in K$ **do**
            for each weight function $w \in W$ applicable to algorithm **do**
                Train model with parameters (k, w)
                Evaluate model on test set
                Record accuracy and parameter setting
            end for
        end for
    end for
end for
return Best-performing model configuration(s) and corresponding evaluation metrics

4.5 Experiment and Results

The modelling and testing procedures were carried out using the same simulation setup in Section 3.7. The modelling applies the k-NN and Wk-NN algorithms to each model for each configuration. While Section 3.7 established baseline performance using the standard k-NN approach with default parameters, this section presents the results after implementing the parameter optimisation and algorithm enhancements described in Section 4.4. This structured presentation allows for a clear before-and-after comparison that highlights the specific contributions of our algorithmic improvements.

Following this procedure, rigorous tests were conducted to evaluate the performance of the k-NN and Wk-NN algorithms on the UJIIndoorLoc dataset.

Table 4.1 presents a comparative performance evaluation of different distance metrics used in the k-NN algorithm for indoor positioning on the complete UJI Datasets. The table contains the following key information: Distance Metrics, Data Representation, and Performance Metrics (Location Success and Error). This comprehensive comparison provides the empirical foundation for selecting the most appropriate distance metric in our enhanced positioning algorithm, balancing the trade-offs between accuracy and floor/building detection reliability.

4.5.1 K-Value on Complete Dataset

The performance results presented in Table 4.2 and Figure 4.1 demonstrate significant improvements over the previous findings. In particular, the use of the correlation distance with exponential representation achieved a remarkable success rate of 98.15%. Moreover, the use of the same distance metric, combined with exponential representation on the complete dataset, resulted in a notable reduction in the MAE, which was 7.64 metres. These results highlight a substantial performance improvement compared to previously reported findings presented in Table 4.1.

4.5.2 K-Values on Individual Building

Analysis of the findings presented in Table 4.3 reveals variations in MAEs among different buildings. This discrepancy prompted an examination of the signal distributions within each building, as shown in Figure 4.2. Specifically, BLD1 has fewer RPs than the other buildings, whereas BLD2 has a higher count of RPs and an associated error range of 9 metres. In contrast, BLD0 exhibits a lower MAE range of 5 metres. Further investigation revealed that only two mobile devices were used to survey BLD0, whereas a diverse array of devices was employed for the other buildings. The variability in device usage may contribute to the observed differences in MAEs. An overview of the optimal results is summarised in Table 4.4

4.5.3 Weighted k-Nearest Neighbours (Wk-NN)

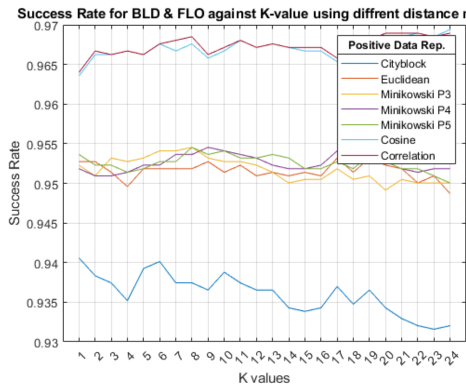
The transition from k-NN to Wk-NN involves incorporating distance weights (w). In this iteration, we use only the correlation distance and exponential data representation, as this combination demonstrated superior performance in previous experiments. The evaluation includes testing each distance weight, such as inverse and squared inverse, on the complete dataset and individual building configurations. Additionally, recalibration of the k-value was necessary due to the observed improvements following the application of distance weights. Consequently, we have increased the k-value from 24 to 50 and documented the optimal results where $k > 1$ in Table 4.5. Specifically, the most favourable outcome achieves a MAE of 7.39 metres, surpassing the performance of the k-NN algorithm.

4.5.4 Comparison with Other Studies

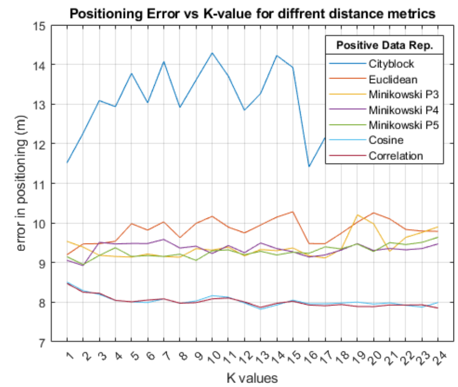
To ensure a fair and comprehensive comparison, we evaluated our optimally tuned Wk-NN algorithm using identical dataset configurations. Specifically, we selected studies with similar settings, using both training and validation datasets, regardless of the algorithm or methodology employed. Essentially, our comparison focuses on

Table 4.2: Best Performance of K Value on Different Distance Metrics and Data Representations using Complete UJI Datasets

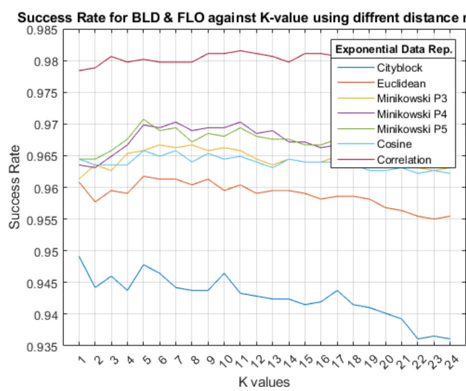
Distance Metrics	Data Rep.	Location Success (%)				Error (m)	
		k	BLD	FLO	Mean	k	MAE
Cityblock							
	Pos	1	98.88	89.28	94.05	16	11.41
	Exp	1	99.00	90.81	94.91	1	10.52
	Pow	1	99.36	90.90	95.13	1	9.89
Euclidean							
	Pos	17,19	99.36	91.26	95.31	1	9.19
	Exp	5	99.55	92.79	96.17	1	8.58
	Pow	6	99.73	93.15	96.44	2	8.76
Minkowski P3							
	Pos	8	99.91	90.99	95.45	5	9.13
	Exp	6,8	99.73	93.60	96.66	2	8.69
	Pow	5	99.73	93.87	96.80	1	8.65
Minkowski P4							
	Pos	9,19	99.82	91.08	95.45	2	8.92
	Exp	7,11	100	94.05	97.02	2	8.39
	Pow	6	99.82	94.05	96.93	2	8.42
Minkowski P5							
	Pos	8	100	90.90	95.45	2	8.95
	Exp	5	100	94.14	97.07	2	8.30
	Pow	11	99.82	94.23	97.02	2	8.57
Cosine							
	Pos	24	100	93.96	96.93	13	7.82
	Exp	5,7	99.55	93.60	96.57	1	8.52
	Pow	21	100	96.30	98.15	23	7.72
Correlation							
	Pos	22,24	100	93.87	96.89	24	7.85
	Exp	22,23	100	96.30	**98.15**	22	**7.64**
	Pow	20,22	100	96.21	98.10	23	7.69



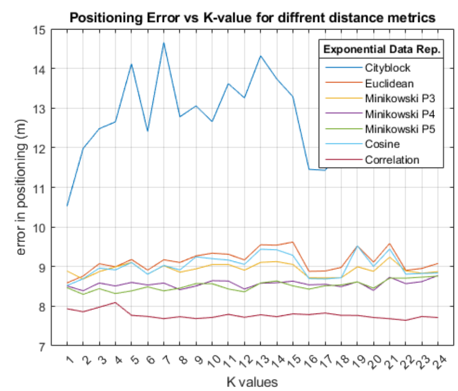
(a) Success: Positive Data Rep.



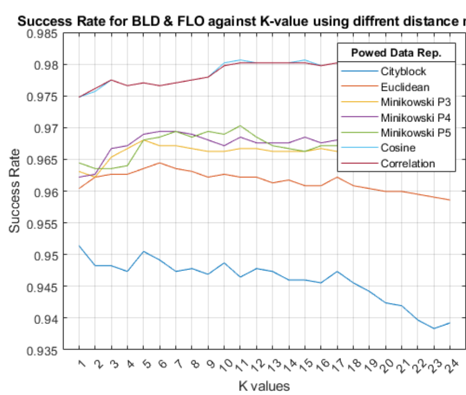
(b) Error: Positive Data Rep.



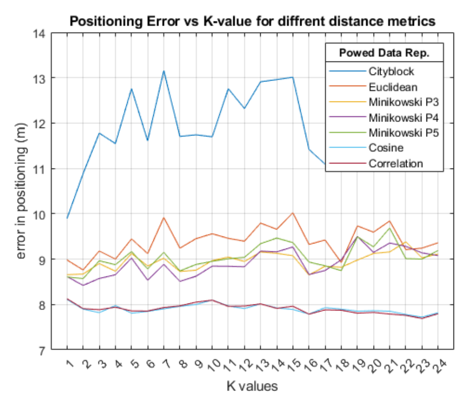
(c) Success: Exponential Data Rep.



(d) Error: Exponential Data Rep.



(e) Success: Powered Data Rep.



(f) Error: Powered Data Rep.

Figure 4.1: Results of K-value on Different Data Representation

Table 4.3: Performance on Floor Hitting Rate and Error for Each Building

Building No.	**Distance Metrics**	**Data Rep.**	**k**	**Hit Rate**	**k**	**MAE**
BLD0	Cityblock	Pos	9-11	97.20	10	5.47846
		Exp	3	97.01	4	5.29077
		Pow	4	97.38	4	5.42837
	Euclidean	Pos	19	97.76	3	5.49774
		Exp	9	97.38	1	5.57302
		Pow	4,7	97.01	5	5.25164
	Cosine	Pos	8, 9	97.76	8	5.74659
		Exp	1	97.57	4	**5.16713**
		Pow	1	97.57	6	5.39378
BLD1	Cityblock	Pos	2	77.85	3	11.9846
		Exp	6	79.47	3	11.8343
		Pow	6,10	79.47	2	11.2378
	Euclidean	Pos	18	78.50	23	11.1771
		Exp	18-24	81.75	14	10.3903
		Pow	4, 20	83.71	21	10.6452
	Cosine	Pos	24	85.34	21	9.54734
		Exp	24	85.66	15	9.98617
		Pow	21	**93.81**	3	9.57655
BLD2	Cityblock	Pos	1,2	90.29	9	12.9376
		Exp	1	94.02	1	11.3889
		Pow	1	94.77	1	11.0013
	Euclidean	Pos	2	95.52	1	11.1046
		Exp	4	**97.38**	6	10.0375
		Pow	3-5	97.01	5	10.3945
	Cosine	Pos	6,8	97.01	22	9.45524
		Exp	8-10	97.01	8	9.79197
		Pow	22, 24	**97.38**	9	9.72947

Table 4.4: Performance Summary

BLD No.	**Dist. Metrics**	**Data Rep.**	**Hit Rate**		**MAE**	
			k	**(%)**	**k**	**(m)**
BLD0	Exp	Correlation	1	97.94	5	5.17979
BLD1	Exp	Correlation	23	93.48	23	9.16244
BLD2	Exp	Correlation	22	97.01	22	9.24040
Average						**7.86087**

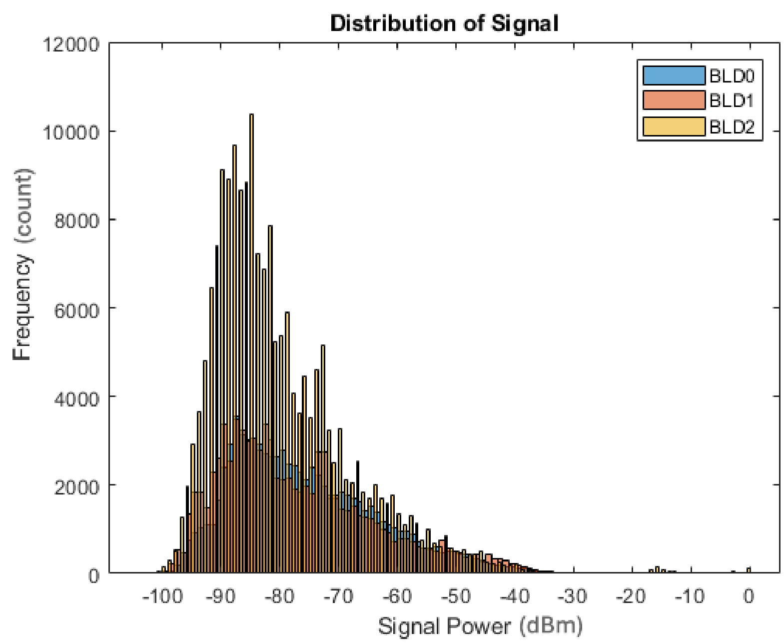


Figure 4.2: Distribution of Signals on Each Building

Table 4.5: Performance Summary of Distance Weight combined with Exp Data Representation and Correlation Distance

BLD No.	Weight	HR (%)				MAE	
		k	BLD	FLO	(Mean)	k	(m)
012	$1/d$	20,26	100	96.30	98.15	26	7.39643
	$1/d^2$	20,43	100	96.21	98.10	26	7.44725
0	$1/d$	2	-	97.94	-	2	5.57302
	$1/d^2$	2	-	97.94	-	24	5.52751
1	$1/d$	42,43	-	94.13	-	26	9.02968
	$1/d^2$	43,46	-	93.48	-	26	9.25612
2	$1/d$	2-5	-	97.01	-	29	9.03861
	$1/d^2$	4,5	-	97.01	-	35	8.95107

experiments conducted with the complete dataset, including validation datasets for testing where feasible. Table 4.6 presents our results alongside those of other studies.

The comparison clearly illustrates the substantial improvement in both the mean success rate and MAE achieved with our proposed approach. This enhancement highlights the innovative design and Wk-NN's superior performance compared to alternative methodologies.

4.6 Summary and Conclusion

4.6.1 Summary

The experimental investigations in this chapter systematically evaluated and optimised k-NN and Wk-NN algorithms for Wi-Fi fingerprinting using the UJIIndoorLoc dataset. A consistent k-value was employed for Longitude and Latitude when calculating the MAE to establish generalisable parameters. It was observed that the Correlation distance function, particularly when combined with exponential data representation, consistently yielded strong performance. For instance, optimal k-values (e.g., 22 and 24 for Correlation distance) were identified as well-suited for large, sparse datasets, with distinct optimal k-values emerging for different model configurations, though correlation and cosine distances showed robust performance across various k-values.

Our research highlights that the combination of the Correlation distance function

Table 4.6: Positioning Performance Comparison with other Studies

Reference	**Location HR (%)**			**MAE (m)**
	BLD	**FLO**	**Mean**	
Torres-Sospedra et al. (2014)[15]	-	-	89.92	7.90
Torres-Sospedra et al. (2015) [186]	-	-	95.20	6.19
Gan et al. 2019 [202]	100	95.41	-	6.40
CNNLoc: Song et al. 2019 [129]	100	96.03	-	11.78
Liu et al. 2021 [203]	99.64	91.18	-	8.39
CCpos: Qin et al. 2021 [204]	-	-	-	12.4
Elesawi et al. 2021 [206]	100	95.24	-	11.78
Tang et al. 2022 [208]	100	94.20	-	8.42
Proposed WKNN	100	**96.30**	**98.15**	**7.39**

and exponential data representation stands out as one of the most effective algorithmic choices for the UJIIndoorLoc dataset. The integration of inverse distance weighting into the Wk-NN algorithm led to the lowest Mean Absolute Error (MAE) for the complete multi-building dataset (BLD012), achieving 7.39 metres. While the classification success rates for building and floor identification remained comparable between the optimised k-NN and Wk-NN, the reduction in coordinate error by Wk-NN is a significant finding. The inverse distance weight function was generally superior for the entire dataset, though optimal configurations varied slightly for individual buildings.

Based on these comprehensive evaluations, the optimal configuration for k-NN was identified as using the Correlation distance function with exponential data representation and a k-value of 22. For the enhanced Wk-NN approach, the optimal setup involves the Correlation distance function, exponential data representation, an inverse distance weighting scheme, and a k-value of 26.

4.6.2 Conclusion

This chapter successfully addressed the objective of enhancing Wi-Fi fingerprinting-based indoor positioning through the systematic optimisation of k-NN and Wk-NN algorithms. The **primary achievement** lies in the development and empirical validation of an optimally tuned Wk-NN configuration that significantly improves positioning accuracy on the complex, multi-building UJIIndoorLoc dataset.

The rigorous analysis of hyperparameters confirmed that the choice of k-value, distance metric, and RSSI data representation profoundly impacts performance.

Specifically, the combination of the Correlation distance function with exponential data representation emerged as a highly effective strategy. Furthermore, the systematic application of inverse distance weighting in the Wk-NN algorithm was shown to reduce the Mean Absolute Error to 7.39 metres for the complete dataset, a notable improvement over baseline k-NN and competitive with existing literature (as detailed in Table 4.5.4). These findings underscore the value of careful parameter tuning and algorithmic refinement.

The investigation into different dataset configurations (complete dataset versus individual buildings) provided insights into the algorithms' behaviour under varying data distributions and environmental characteristics, contributing to a better understanding of their practical applicability.

A key contribution of this work is the provision of a comprehensive performance benchmark for k-NN and Wk-NN algorithms on the UJIIndoorLoc database, detailing optimal parameter settings. This framework and the identified optimal configurations (e.g., Wk-NN with Correlation distance, exponential representation, inverse weight, and k=26) offer a valuable resource for the research community. It facilitates more informed comparisons for future studies and aids in the design of machine learning-based Wi-Fi fingerprinting indoor positioning systems. The optimised Wk-NN algorithm developed in this chapter forms a robust positioning engine for the subsequent system-level investigations in this thesis.

Chapter 5

Radio Map Optimisation

This chapter explores the optimisation of radio map fingerprinting to address three significant optimisations: heterogeneity, dimensionality, and fingerprint updating. These challenges present considerable obstacles in Radio Map Fingerprinting (RMF) systems. The Radio Map Optimisation (RMO) in the context of Wi-Fi fingerprinting-based Indoor Positioning Systems refers to a comprehensive set of strategies aimed at enhancing the performance, scalability, and adaptability of the radio map. Specifically, RMO involves: (i) dimensionality reduction to compress and accelerate fingerprint matching processes; (ii) compensation for device heterogeneity in RSSI measurements to improve robustness; and (iii) the implementation of auto-update mechanisms that allow the radio map to evolve with minimal manual calibration. These strategies collectively address the critical challenges of accuracy, efficiency, and long-term maintenance in large-scale deployments.

Addressing heterogeneity in RSSI device measurements is crucial for enhancing system accuracy. Meanwhile, reducing dataset complexity through dimensionality reduction is essential for minimising computational time and managing large databases effectively. Additionally, implementing auto-update mechanisms ensures that each query entry in the matching process is saved for future reference, thus improving system adaptability to environmental changes and reducing human calibration costs. Collectively, these solutions contribute to the efficient management of large fingerprint databases.

5.1 Introduction

Radio Map Fingerprinting (RMF) has emerged as a promising technique for indoor positioning and localisation, especially in environments where GPS signals are unreliable or unavailable. The core concept behind RMF is to construct a database, the radio map, containing signal characteristics (typically RSSI) meticulously recorded at known locations within the target area. During the online positioning phase, real-time RSSI measurements from a mobile device are compared against this radio map to infer the device's location.

In the context of this thesis, Radio Map Optimisation refers to a multifaceted process of enhancing the radio map's efficacy and efficiency for indoor positioning. This involves the systematic application of techniques to improve the quality of the fingerprint data, reduce the resources required for storing and querying the map, and enhance the map's adaptability to dynamic environmental conditions and diverse user devices. The primary objectives of such optimisation are to achieve superior positioning accuracy, minimise computational overhead, reduce the human effort associated with initial calibration and ongoing maintenance, and ensure robust performance across heterogeneous scenarios.

This chapter delves into key aspects of RMF optimisation by addressing three significant challenges: signal measurement heterogeneity, high data dimensionality, and the imperative for continuous fingerprint updating. Heterogeneity in RSSI measurements, often stemming from device diversity and fluctuating environmental factors, can degrade system accuracy if not properly managed. Concurrently, the high dimensionality of fingerprint data, particularly in environments with numerous access points, can lead to increased computational burden and database management complexities. Effective dimensionality reduction is therefore essential. Furthermore, the dynamic nature of indoor environments necessitates mechanisms for automated radio map updates, ensuring the long-term viability and accuracy of the positioning system whilst minimising manual recalibration efforts. Addressing these challenges collectively contributes to the development of scalable and efficient RMF-based indoor positioning systems.

RMF is a crucial component of Wi-Fi fingerprinting-based IPS. The RSSI RMF technique involves creating a radio map of indoor environments based on RSSI values. This method relies on collecting RSSI fingerprints at various locations within a building to establish a database, enabling position estimation by matching real-time RSSI measurements with the closest fingerprints in the database [138].

Despite its advantages, RMF faces several challenges that can impact its accuracy and scalability. One primary challenge is dealing with heterogeneity in the radio environment, which arises from factors such as device diversity, signal propagation characteristics, and environmental dynamics. Heterogeneity can result in significant variations in RSSI measurements, making it difficult to achieve consistent and reliable positioning performance. Another challenge is the high dimensionality of RMF, which can lead to large database sizes and increased computational complexity during the matching process. As the number of WAPs or signal sources increases, the dimensionality of the fingerprints also increases, potentially causing overfitting, higher storage requirements, and longer processing times. Furthermore, constructing and maintaining RMF can be time-consuming and labour-intensive, often requiring manual calibration and periodic updates to account for environmental changes. This is particularly challenging in large-scale deployments or dynamic environments, where RMF may become outdated or inaccurate over time.

The widespread adoption of RSSI fingerprinting in IPS is due to its accessibility and high accuracy [200]. This technique involves collecting RSSI measurements at grid points in indoor environments during an offline phase to build a fingerprint database [45], which is then used for localisation during the online phase [24]. Numerous studies underscore the importance of RSSI fingerprinting in IPS, highlighting its ability to improve performance through techniques such as RSSI clustering [20]. Despite challenges such as maintaining RMFs through frequent surveys [211], RSSI fingerprinting remains effective for accurate indoor positioning based on radio signal characteristics. Furthermore, innovative methods, such as ViFi, which uses a Multi-Wall Multi-Floor (MWMF) propagation model for RSSI prediction, contribute to optimisation [24], as does leveraging spatial relationships between RSSI fingerprints [212]. To address computational challenges associated with k-NN and Wk-NN, the literature explores RMF data optimisation approaches, such as clustering and optimisation rules. These strategies involve grouping similar fingerprint locations and limiting distance calculations based on signal strength patterns [165].

The remainder of this chapter is dedicated to exploring specific strategies for RMF optimisation. We will propose and evaluate methods for mitigating device-induced heterogeneity, for prudent dimensionality reduction of the fingerprint dataset, and for implementing an adaptive auto-updating mechanism for the radio map. Furthermore, techniques for the efficient management of the resultant large-scale fingerprint databases will be considered. Through rigorous experimentation and detailed analysis, using the UJIIndoorLoc dataset as a benchmark, this chapter aims to provide valuable insights into the efficacy of these optimisation techniques and their collective impact on overall IPS performance, particularly in terms of accuracy, computational efficiency, and operational adaptability in diverse indoor settings.

5.2 Methodologies for Radio Map Optimisation

In line with the previously established definition of Radio Map Optimisation, this section presents specific methodologies and analyses aimed at achieving these optimisation goals. These methodologies are designed to address the critical challenges of heterogeneity in signal measurements, to reduce the dimensionality of the fingerprint data, and to incorporate an adaptive auto-update mechanism for the radio map. Additionally, strategies for managing the potentially large databases generated during the fingerprinting process are discussed.

In this section, we present different methodologies and analyses to optimise RMF.

These methodologies are designed to address heterogeneity, reduce dimensionality, and incorporate an auto-update fingerprint mechanism. It also includes strategies for managing the large databases generated during the fingerprinting process. RMF is a technique for indoor localisation that relies on a pre-collected database of RSSI from RPs within the environment. Given a user's device and its measured RSSI fingerprint (unique RSSI signature), the system estimates the user's location by finding the RP in the database with the most similar fingerprint. The similarity can be measured using various distance metrics, such as Euclidean distance.

5.2.1 Heterogeneity Issue

Heterogeneity in RMF can stem from various factors, such as device variation, signal transmission properties, and environmental changes. Heterogeneity poses a significant obstacle to achieving high accuracy with indoor Wi-Fi fingerprinting. It affects the system by causing fluctuations in RSSI. Ideally, RSSI should remain stable for a given location, but heterogeneity makes this more complex. There are several reasons for RSSI fluctuations, including:

- Device Hardware: Different devices (phones, tablets) have varying antenna designs and signal processing capabilities. This can lead to different RSSI readings even at the same spot.
- Environmental factors: Walls, furniture, and even people can absorb or reflect Wi-Fi signals, causing fluctuations in RSSI depending on the user's position and surroundings.
- Time-based variations: Signal strength can fluctuate due to network congestion or changes in WAP configurations.

These variations impact localisation accuracy in several ways:

- Fingerprint Database Mismatch: The fingerprint database, built with reference RSSI measurements from a specific device, might not match the RSSI readings from a different device used for localisation. This mismatch leads to inaccurate location estimates.
- Environmental Sensitivity: If the environment changes significantly between fingerprint collection and localisation (e.g., furniture rearrangement), RSSI variations become more pronounced, further reducing accuracy.

5.2.2 Heterogeneity Effect

To illustrate the effect of device heterogeneity on positioning estimation accuracy, we present Figure 5.1, which details the devices used to create the UJI dataset across three multi-floor buildings at Jaume I University (UJI). These are all Android devices from various manufacturers such as Samsung Galaxy Nexus, HTC One, and Nexus 4 [15]. Figure 5.2 highlights the extensive array of mobile devices employed throughout the data collection phase. This diversity in device models underscores the inherent heterogeneity within the dataset, a factor that can potentially influence the accuracy of positioning estimation algorithms.

Notably, our observations reveal that building 0 (BLD0) was surveyed using only two distinct devices. This is consistent with the findings detailed in Section 4.5.2 and supports the results presented in Table 4.3. Specifically, we note that the MAE for BLD0 falls within the range of 5 meters, whereas other buildings exhibit errors ranging from 9 to 11 meters. These discrepancies underscore the significant impact of device heterogeneity on the performance of positioning estimation algorithms. Figure 5.3 illustrates the variety of devices that construct the validation dataset. Furthermore, of the 25 devices used in the creation of the UJI database, 16 were utilised to construct the training dataset, while 11 devices contributed to the validation dataset; see Figure 5.4. In particular, only phone IDs 13 and 14 were used in both datasets. This diversity in device usage underscores the richness of the dataset, which, while beneficial, also introduces complexities that may contribute to positioning errors.

After conducting further tests using only devices (PHONEID) 13 and 14 from the validation dataset, notable improvements in accuracy were observed compared to the results obtained in Section 4.5 with our proposed algorithm, Wk-NN. The overall MAE rate for the complete datasets was reduced to 6.3940. Subsequently, we focused on evaluating Building 0 (BLD0), which had its training and validation datasets created using the same device IDs 13 and 14. Impressively, the results for BLD0 showed a significantly improved MAE of 4.6639. Conversely, Building 1 (BLD1) exhibited a MAE rate of 8.2010, while Building 2 (BLD2) displayed a MAE rate of 7.7291 metres.

5.2.3 Mitigate Heterogeneity

Several approaches exist to mitigate heterogeneity in Wi-Fi fingerprinting for indoor positioning:

- Calibration Techniques: Some methods involve calibrating the RSSI readings from different devices to a reference device or using signal processing techniques to normalise the data.

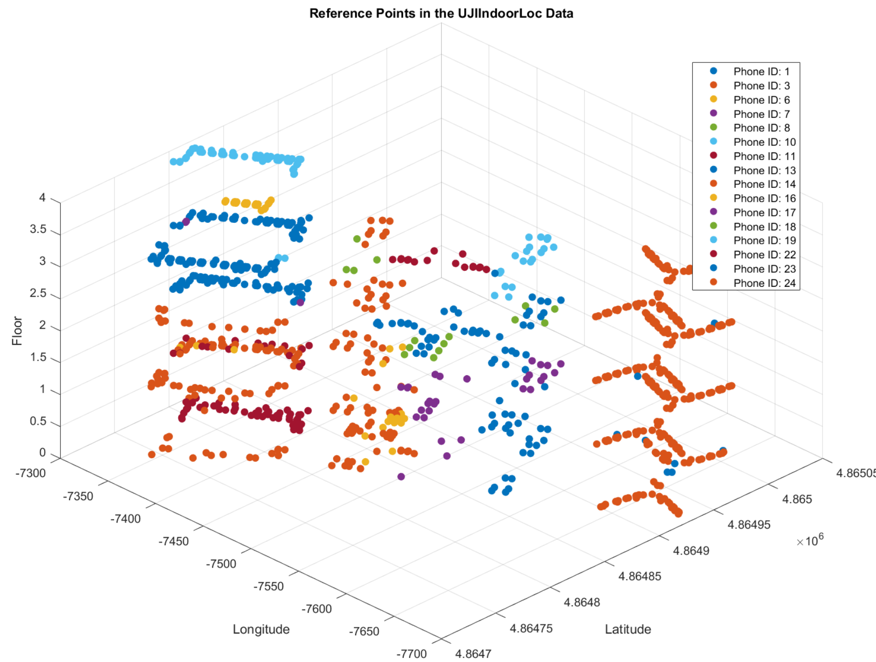


Figure 5.1: Phone ID plotted in 3D

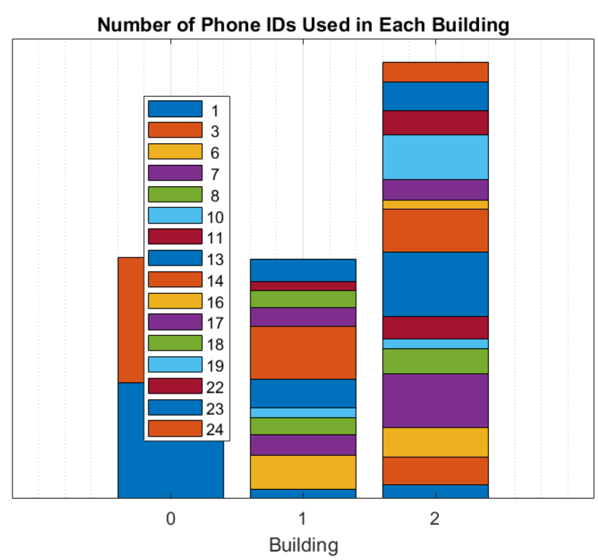


Figure 5.2: The Distribution and Quantity of Phones used at each Building

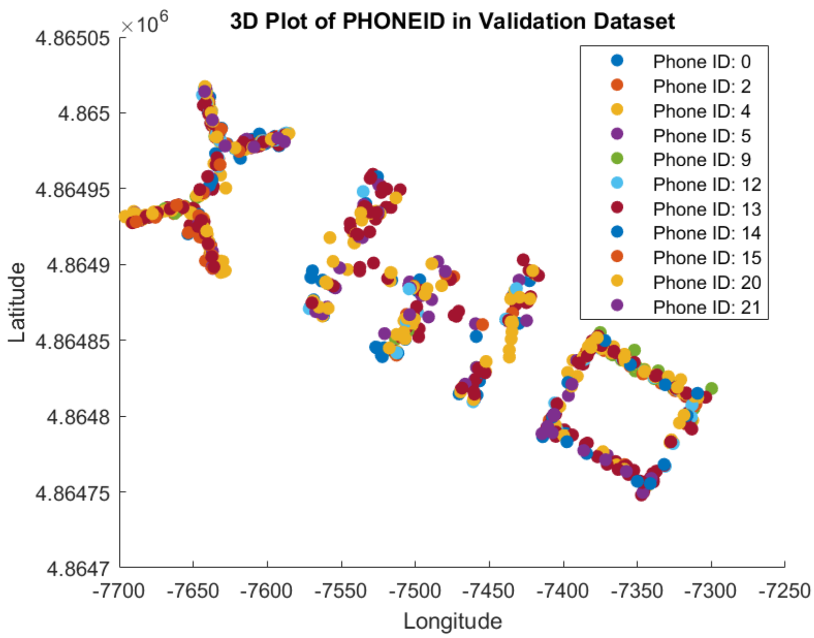
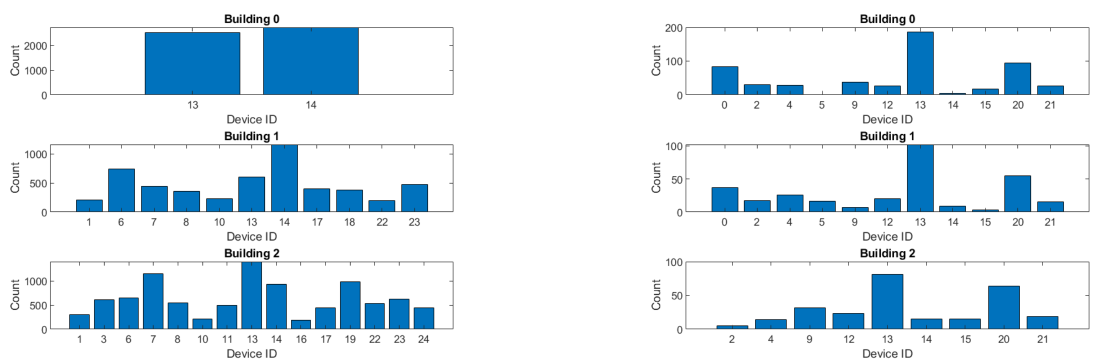


Figure 5.3: Phone ID used to build Validation Dataset



(a) Training Dataset (b) Validation Dataset

Figure 5.4: Devices used to create the UJI data sets for each building

- Data Preprocessing: Fingerprint preprocessing can be applied to account for variations by considering factors such as the standard deviation of RSSI readings or using probabilistic approaches.
- Machine Learning: Machine learning models can be trained on data with inherent heterogeneity to recognise patterns and improve the matching process between reference and live RSSI measurements.

As calibration techniques are implemented exclusively during the offline phase, they are not included in our research. Instead, our focus is on leveraging advanced preprocessing and machine learning techniques to mitigate heterogeneity in RSSI variations. Heterogeneity in signal strength can be represented as a vector of signal strengths from different access points. If there are n access points, the signal strength can be represented as a vector $S = [s_1, s_2, ..., s_n]$. Heterogeneity in Wi-Fi fingerprinting presents a challenge, but researchers are actively developing methods to mitigate it. By incorporating calibration techniques, advanced data preprocessing, and machine learning, the accuracy of Wi-Fi fingerprinting for indoor localisation can be significantly improved.

In Section 5.2.2, we explored the inherent variability in RSSI readings, a crucial component derived from Wi-Fi chipsets embedded in mobile and laptop devices or WAPs. This variability stems from the diverse manufacturing processes of these chipsets, resulting in inconsistencies in RSSI measurements. Such discrepancies contribute to positioning errors, particularly evident during both the site survey phase, as exemplified by the UJI dataset collected from 25 different devices, and the online phase, where slight discrepancies may arise between the RSSI data in the RMF and those from the querying device.

The challenge of heterogeneity in RSSI fingerprinting poses a significant obstacle for indoor positioning systems, necessitating the development of advanced techniques and methodologies for effective mitigation. ML algorithms, which include both supervised and unsupervised approaches, offer promising avenues to address device heterogeneity in indoor positioning systems [213]. By leveraging ML techniques, such systems can adapt to the diverse RSSI patterns generated by different devices, thereby enhancing positioning accuracy amidst heterogeneity.

Furthermore, adopting hybrid combinations of signal measurement principles, such as RSSI, CSI, and TOA, presents a comprehensive approach to tackling heterogeneity in RSSI fingerprinting [20]. Integrating multiple signal measurement techniques broadens the scope of information captured, leading to more robust and accurate indoor localisation results. Leveraging distance and similarity measures within Wi-Fi fingerprinting systems also holds promise in addressing heterogeneity challenges [186]. By incorporating both natural RSSI data and architectural information, these systems achieve a nuanced understanding of the indoor environment, accounting for variations in signal strength as well as structural constraints. Moreover, techniques such as normalisation, filtering, or outlier removal can be applied to the collected RSSI data to mitigate noise and reduce the impact of outliers. This, in turn, enhances the system's capacity to manage heterogeneity in RSSI fingerprinting effectively.

Overall, mitigating heterogeneity in RSSI fingerprinting for indoor positioning systems requires a multifaceted approach, integrating machine learning algorithms, hybrid signal measurement principles, distance and similarity measures, and advanced propagation models. By synergistically deploying these techniques, indoor localisation systems can overcome the challenges posed by device heterogeneity and environmental variations, thereby achieving more reliable and accurate positioning results.

5.2.4 Dimensionality Issue

The proliferation of access points and the accumulation of signal measurements contribute to the dimensionality of fingerprint databases, thus increasing computational complexity and storage requirements. High-dimensional RMF can lead to overfitting, increased storage requirements, and longer processing times during matching. Large fingerprint databases can be computationally expensive to manage and compare. The collected data are high-dimensional, with each access point representing a dimension. To manage this high dimensionality, researchers use dimensionality reduction techniques such as PCA. These techniques transform high-dimensional data into a lower-dimensional space, thereby reducing system complexity and improving efficiency.

5.2.4.1 Principal Component Analysis (PCA)

PCA is employed to transform high-dimensional signal data into a lower-dimensional subspace while maximising variance retention, thus simplifying subsequent processing tasks. It is widely used for feature extraction, where principal components are derived from the eigenvectors and eigenvalues of the covariance matrix of the data.

This technique identifies a set of orthogonal principal components (PCs) that capture the most significant variance in the data. Fingerprint data can be projected onto a subspace spanned by these PCs, achieving dimensionality reduction without compromising accuracy. The projected data $\boldsymbol{Y}$ can be obtained using:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}, \tag{5.1}$$

where:

- $\boldsymbol{X} \in \mathbb{R}^{M \times N}$ represents the original fingerprint data matrix with dimensions $M \times N$, where M is the number of samples (reference points) and N is the number of features (Wi-Fi access points).

- $\boldsymbol{W} \in \mathbb{R}^{N \times K}$ is the transformation matrix composed of the top K principal components (i.e., the leading eigenvectors of the covariance matrix of $\boldsymbol{X}$), where $K < N$.
- $\boldsymbol{Y} \in \mathbb{R}^{M \times K}$ represents the projected data matrix in the reduced K-dimensional subspace.

This projection preserves the directions of maximal variance in the original dataset, allowing the indoor positioning model to operate with fewer features while maintaining robustness and scalability.

5.2.5 Auto-Update Mechanism

One effective strategy involves implementing machine learning algorithms capable of continuous learning from new queries and subsequently updating the fingerprint database [69]. By using machine learning techniques, the system can adapt to environmental changes and improve its accuracy over time without requiring manual intervention.

Furthermore, integrating deep learning models, such as CNNs, can enhance the system's self-learning capabilities by enabling it to extract intricate patterns from RSSI data [22]. These models facilitate efficient processing of incoming queries, pattern identification, and updating of the fingerprint database to enhance localisation accuracy.

Crowdsourcing data for RMF construction can also significantly contribute to the self-learning process of fingerprinting systems [211]. By amalgamating data from multiple sources, the system can continuously refine and update its fingerprint database based on real-time information, thereby reducing reliance on manual calibration and lowering maintenance costs.

Moreover, integrating Bayesian probability and online sequential learning algorithms can further enhance the system's adaptability to environmental dynamics and improve localisation precision [100]. These algorithms enable real-time updating of the fingerprint database based on new queries, ensuring that the RMF remains accurate and up-to-date.

Traditional RMF involves creating a static RMF used for localisation without updates. However, this approach proves inadequate for dynamic indoor environments where signal strength can fluctuate over time due to various factors. Therefore, a mechanism is necessary to update the RMF to reflect these changes.

Maintaining and updating the RMF can be costly and time-consuming, often necessitating manual calibration. To address this challenge, we propose an auto-update fingerprint mechanism in our framework. This mechanism learns from user queries and utilises this information to update the fingerprinting.

Our proposed auto-update mechanism addresses this need by saving each incoming query for future reference. This data is then used to update the fingerprinting, allowing the system to learn and adapt over time. This mechanism not only improves the accuracy of the system but also reduces the need for manual calibration, thereby lowering costs.

Implementing the auto-update mechanism involves several steps. First, when a query enters the system, it is matched with the existing fingerprints in the RMF. The result of this matching process is saved along with the query, forming a repository of query-result pairs.

Next, this repository is used to update the fingerprints. The system learns from the query-result pairs and updates the fingerprints in the RMF accordingly. This update can be performed periodically or whenever a significant amount of new data is added to the repository.

The following steps outline the operation of the auto-update mechanism:

- Incremental Learning: New signal measurements obtained during user queries are incrementally integrated into the fingerprint database, ensuring that the system remains up-to-date and reflective of real-time conditions.

- Feedback Loop: Establishing a feedback loop mechanism to capture user feedback on localisation accuracy and adjust fingerprinting parameters accordingly facilitates continuous improvement and refinement of the system.

- Anomaly Detection: Employing anomaly detection techniques to identify and flag anomalous signal measurements enables the system to distinguish between genuine environmental changes and spurious deviations.

To address the challenges associated with manual calibration and updating RMF, we propose an auto-update mechanism that leverages user queries to continuously refine and update the fingerprint database.

The auto-update mechanism can be represented as a function f that updates the fingerprint database $\mathcal{D}$ based on a new query q and its result r:

$$\mathcal{D}' = f(\mathcal{D}, \boldsymbol{q}, \boldsymbol{r}), \tag{5.2}$$

where $\mathcal{D}'$ represents the updated database.

5.2.5.1 User Query Incorporation

During the positioning phase, when a user queries the system with real-time RSSI measurements, the system estimates the user's location based on the current fingerprint database. The user's actual location, if available (e.g., through manual input or additional sensors), can be incorporated into the fingerprint database as a new data point, effectively updating the RMF. Let $\mathcal{D}_{old}$ be the fingerprint database and $\mathcal{Q}_{data}$ be the set of user queries. The updated fingerprint database $\mathcal{D}_{new}$, after incorporating user queries, can be represented as:

$$\mathcal{D}_{new} = \mathcal{D}_{old} \cup \mathcal{Q}_{data}, \tag{5.3}$$

where $\cup$ is the set-union operation, incorporating user queries $\mathcal{Q}_{data}$ into the existing fingerprint database $\mathcal{D}_{old}$.

5.2.5.2 Active Learning

Active learning techniques can be used to selectively query users for their actual locations when the system is uncertain or when the potential information gain is high. This approach can help prioritise the acquisition of valuable data points for updating the fingerprint database while minimising the burden on users.

By incorporating machine learning, deep learning, crowd-sourcing, and advanced algorithms, indoor positioning systems can achieve auto-update capabilities in the RMF. These technologies enable the system to continuously learn from new queries, adapt to environmental changes, and improve localisation accuracy over time, thereby overcoming the limitations of human calibration and reducing long-term costs.

5.2.6 Database Management Strategies

The fingerprinting process generates large databases that need to be efficiently managed. This is a significant challenge in the field of RMF. In this section, we discuss the strategies for managing these databases in our proposed methodology.

The RMF process involves collecting signal strength information from multiple access points within an indoor environment. This information is then used to create an RMF, which is essentially a database of fingerprints. As the number of access points and the size of the environment increase, the size of this database can become quite large. Efficient management of this large database is a significant challenge.

To address this challenge, several strategies for managing the databases generated during the fingerprinting process have been developed. These strategies include the use of efficient data structures for storing the data and algorithms for the quick retrieval of relevant information.

5.2.6.1 Efficient Data Structures

The choice of data structure for storing fingerprints can significantly impact the efficiency of the system. Using a tree-based data structure, such as a k-d tree or a ball tree, to store the fingerprints can be an effective strategy. These data structures facilitate the efficient storage of high-dimensional data and enable rapid nearest-neighbour searches, which are crucial for the fingerprint-matching process.

5.2.6.2 Fast Retrieval Algorithms

In addition to efficient data structures, there a rapid retrieval algorithms for managing databases. These algorithms, such as approximate nearest-neighbour search methods, enable the swift retrieval of relevant fingerprints from the database. Rapid retrieval is essential for ensuring the real-time performance of the system.

5.2.6.3 Indexing and Data Structures

Utilising effective indexing techniques, such as spatial indexing (e.g., R-trees, quadtrees) or inverted indexing, can improve efficient fingerprint retrieval and matching. Moreover, employing appropriate data structures, such as hash tables or binary search trees, facilitates organised and rapid access to fingerprint data.

5.2.6.4 Clustering and Compression

Clustering algorithms can group similar fingerprints, thereby reducing the overall database size by storing representative fingerprints (cluster centroids) instead of individual data points. Moreover, compression techniques, such as quantisation or dimensionality reduction, can be applied to further decrease storage requirements. However, it is important to note that clustering is not a foolproof solution, as it does not eliminate heterogeneity. Poor feature selection can also lead to less effective clusters, and clustering algorithms may be computationally intensive.

5.3 Proposed Approach

To address the challenges of heterogeneity, dimensionality, and auto-update in the RSSI fingerprinting technique, we propose a comprehensive approach that incorporates various methods to optimise RMF. Our approach has been tested using the UJIIndoorLoc database, which is detailed in Section 3.6. We utilised the UJIIndoorLoc dataset to explore heterogeneity, auto-update, and the efficient management of large fingerprint databases. We employed the following combination of techniques:

5.3.1 Handling Heterogeneity

To mitigate the effects of heterogeneity, we employed the following techniques.

5.3.1.1 Data Preprocessing

The initial offline phase involves collecting RSSI RMF data by gathering signal strength information from multiple access points within the indoor environment. Due to factors such as missing information and inherent heterogeneity in RSSI data, stemming from different types of access points, devices used for collecting RSSI, and varying signal strengths, preprocessing is essential. This preprocessing stage aims to address heterogeneity by employing techniques to normalise the data and standardise it to a common scale.

As discussed in Section 3.6.3, we preprocess the datasets, handling missing values, filtering irrelevant data, and normalising RSSI values.

5.3.1.2 Feature Normalization

Feature normalisation aims to bring RSSI measurements from different devices to a common scale. This ensures that the positioning system treats measurements from various devices uniformly, regardless of their original scale. A common technique for normalisation is Z-score normalisation, also known as standardisation. The implementation is as follows:

1. **Calculate the mean and standard deviation:** Compute these statistics for RSSI measurements from each Wi-Fi WAP across all devices during the calibration phase.

2. **Normalise RSSI Measurements:** For each RSSI scalar measurement $x_{ij} \in \mathbb{R}$ — where i denotes the WAP index and j denotes the device or sample index — apply the Z-score transformation:

$$\acute{x}_{ij} = \frac{x_{ij} - \mu_i}{\sigma_i}, \tag{5.4}$$

   where μ_i and σ_i are the mean and standard deviation of RSSI values for WAP i across all samples, respectively.

3. **Store Normalised Data:** Store the normalised RSSI values $\acute{x}_{ij}$ along with their corresponding device and WAP identifiers in the database or RMF.

As discussed in Section 3.6.4, we adopt exponential data representation, which has demonstrated suitability for RSSI RMF.

5.3.2 Reduce Dimensionality

Dimensionality reduction is achieved via Principal Component Analysis (PCA), as detailed in the previous Section 5.2.4.1. PCA is a widely used technique that projects high-dimensional data into a lower-dimensional subspace while retaining the most significant variance. This transformation is particularly beneficial for large-scale Wi-Fi fingerprinting, as it reduces computational cost and storage without compromising localisation accuracy.

The integration of PCA using equation 5.1 with our approach proceeds as follows:

1. Apply PCA to the preprocessed fingerprint data to extract principal components.

2. Select a subset of components that capture a predefined percentage of total variance (e.g., 90%).

3. Project both the fingerprint data and user query vectors onto the subspace spanned by the selected components.

This approach reduces storage requirements for the fingerprint database and enhances the efficiency of similarity calculations during localisation. PCA focuses on reducing the dimensionality of data by transforming it into a new set of features called PCs. These PCs capture the most significant variations in the original data. In Wi-Fi fingerprinting, PCA can be used to reduce the number of RSSI measurements considered, which is beneficial when dealing with a large number of access points.

5.3.3 Auto-Update

Maintaining the accuracy of a Wi-Fi fingerprinting system necessitates manual updates following environmental changes, which can be inefficient and time-consuming. Our solution introduces an auto-update mechanism, detailed in Algorithm 3, to automatically update the RMF based on user queries and feedback, as illustrated in Figure 5.5.

5.3.3.1 User Query Integration

Let $\boldsymbol{Q} = \boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_n$ denote a set of user queries, where each query $\boldsymbol{q}_i$ is a vector containing the RSSI measurements from nearby WAPs for user i.

Algorithm 3 Auto-Update Algorithm

Initialise $\boldsymbol{Q}$ as a set of user queries
Initialise the model parameters using the initial radio map or a subset of calibration data
for each query $\boldsymbol{q}_i$ in $\boldsymbol{Q}$ **do**
  Estimate the user's position $\hat{x}_i$ by finding the fingerprint f_j in the database that minimizes a distance metric between $\boldsymbol{q}_i$ and f_j
  Calculate the position error e_i as the difference between the estimated position $\hat{x}_i$ and the actual user position x_i
  if $\boldsymbol{q}_i$ matches an existing fingerprint f_i in the database **then**
    Calculate δ based on the estimated error e_i
    Update the fingerprint f_i using the weighted averaging formula: $f_{\text{updated}_i} = \delta \cdot f_i + (1 - \delta) \cdot \boldsymbol{q}_i$
  else
    Estimate the fingerprint $\hat{f}_i$ for the queried location using predictive models or interpolation techniques
    Calculate δ based on the estimated error e_i
    Update the fingerprint using the weighted averaging formula: $f_{\text{updated}_i} = \delta \cdot \hat{f}_i + (1 - \delta) \cdot \boldsymbol{q}_i$
  end if
  Update the model parameters using the learning algorithm chosen based on $(\boldsymbol{q}_i, x_i)$ and e_i
end for
Periodically evaluate the system's performance using metrics like MAE and adjust the learning rate to control the adaptation speed

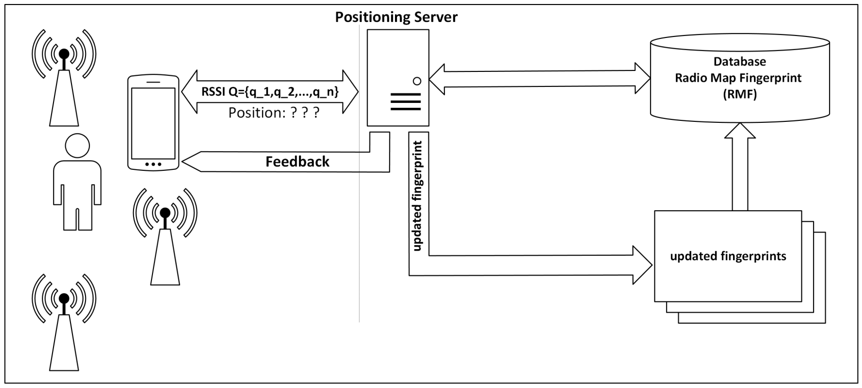


Figure 5.5: RMF Auto-update Mechanism

5.3.3.2 Localisation and Error Estimation

1. For each query $\boldsymbol{q}_i$, the system estimates the user's location ($\hat{x_i}$) by identifying the fingerprint $\boldsymbol{f}_j$ in the database that minimises a distance metric (e.g., Euclidean distance) between $\boldsymbol{q}_i$ and $\boldsymbol{f}_j$.

2. The localisation error e_i is computed as the difference between the estimated location ($\hat{x_i}$) and the actual user location (x_i), which is obtained through user feedback or external sources (e.g., GPS).

5.3.3.3 Fingerprint Database Update

As illustrated in Figure 5.6, the fingerprint database is updated using a weighted averaging approach. This method updates or incorporates fingerprints based on new user queries and their proximity to existing or estimated reference points. The general formula for updating or creating a fingerprint entry using a user query $\boldsymbol{q}_i$ is:

$$\boldsymbol{f}_i^{new} = \delta \boldsymbol{f}_i^{current} + (1 - \delta)\boldsymbol{q}_i, \tag{5.5}$$

where:

- $\boldsymbol{f}_i^{new}$ is the newly updated or incorporated fingerprint associated with a location i.

- $\boldsymbol{q}_i$ is the user's currently measured RSSI vector (query).

- δ is a weighting factor ($0 < \delta < 1$). Its value is typically determined based on the estimated error e_i of the initial position estimate derived from $\boldsymbol{q}_i$. A higher error e_i might lead to a lower δ, thus giving more weight to the new user query

$\boldsymbol{q}_i$ for a more substantial update or for creating a new fingerprint if $f_i^{current}$ is an estimation.

- $\boldsymbol{f}_i^{current}$ represents the existing information in the radio map for location i. This term can take two forms:
  - If query $\boldsymbol{q}_i$ is deemed sufficiently close to an existing fingerprint in the database (denoted as $\boldsymbol{f}_i$), then $\boldsymbol{f}_i^{current} = \boldsymbol{f}_i$. The formula then updates this existing fingerprint.
  - If query $\boldsymbol{q}_i$ does not closely match any existing fingerprint, but its location can be reliably estimated (e.g., through user feedback or interpolation from nearby RPs), an estimated fingerprint for that location, $\hat{\boldsymbol{f}}_i$, might be generated. In this scenario, $\boldsymbol{f}_i^{current} = \hat{\boldsymbol{f}}_i$, and the formula helps to refine this estimated fingerprint with the actual user measurement $\boldsymbol{q}_i$. This is particularly relevant when incorporating new data points into sparse areas of the radio map.

This weighted averaging mechanism allows the system to autonomously update the RMF by combining existing or estimated fingerprint information with new user queries, with the weighting factor δ balancing the confidence between the established radio map data and the new observations.

This approach enables the system to adapt to new queries and update the RMF even when an exact match is absent in the database, leveraging predictive models or interpolation to estimate the fingerprint for the queried location.

5.3.3.4 Implementation Steps

1. Prepare the datasets and select algorithm models for classification and regression tasks.

2. Initialise the model parameters using the initial RMF or a subset of calibration data.

3. As new user queries $(\boldsymbol{q}_i, \boldsymbol{x}_i)$ arrive:
   - Calculate the localisation error e_i.
   - Update the model parameters using the selected learning algorithm based on $(\boldsymbol{q}_i, \boldsymbol{x}_i)$ and e_i.

4. Periodically evaluate the system's performance using metrics such as MAE and adjust the learning rate to control the adaptation speed.

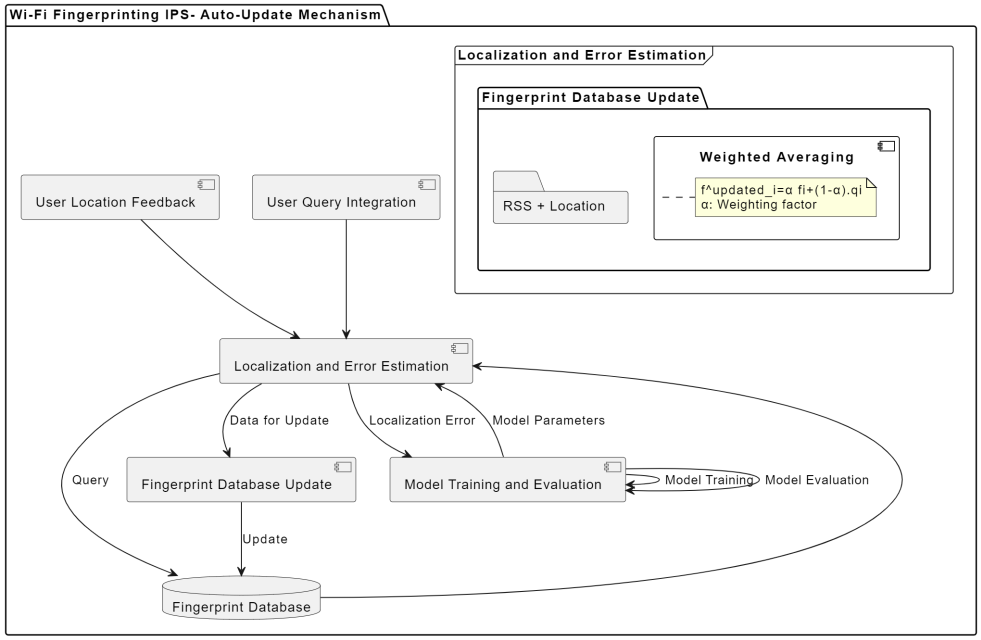


Figure 5.6: Implementation Steps of Auto-Update Mechanism

This proposed method offers several benefits, including reduced manual calibration efforts, adaptation of the fingerprint database to dynamic environments and user behaviour, and improved accuracy and robustness of the positioning system over time.

5.3.4 Efficient Management of Large Databases

The incremental growth of RMF requires efficient database management strategies over time to accommodate this expansion. Implementing auto-updates leads to an increase in the size of the RMF.

The challenges associated with managing large fingerprint databases include storage requirements, retrieval efficiency, and computational complexity. To address these challenges, we use the following strategies:

- Employing techniques for efficiently storing and retrieving fingerprint data, such as indexing methods, data compression algorithms, and dimensionality reduction techniques.
- Evaluating the performance of database management strategies in terms of storage requirements, retrieval time, and positioning accuracy.

By employing these strategies, we aim to optimise the management of large fingerprint databases and ensure scalability and efficiency in indoor positioning systems.

5.4 Experimental Setup

This section describes the experimental setup used to evaluate the effectiveness of our proposed RMF optimisation approach.

To assess the effectiveness of the proposed methodologies, we conducted experiments using the UJIIndoorLoc dataset, a widely used benchmark for indoor localisation research.

5.4.1 UJIIndoorLoc dataset

The experimental evaluation of the proposed methodology uses the UJIIndoorLoc dataset, a widely recognised benchmark dataset in the field of indoor localisation. The UJIIndoorLoc dataset comprises Wi-Fi fingerprint measurements collected from multiple buildings on a university campus, encompassing diverse environmental conditions and architectural layouts. Each fingerprint measurement includes the signal strength readings from multiple WAPs observed at a specific location within the indoor environment, along with the corresponding ground truth coordinates. The dataset provides a rich source of real-world data for evaluating the performance of indoor localisation algorithms, making it well-suited for our experimental purposes. Full details of UJIIndoorLoc have been discussed in Section 3.6. This dataset is preprocessed using the techniques discussed in Section 3.6.3 to handle heterogeneity and reduce dimensionality.

5.4.2 Experimental Environment

The experiments were conducted using MATLAB, a powerful numerical computing environment widely used for data analysis, algorithm development, and visualisation. The MATLAB code developed for the experiments utilises various libraries and functions for data preprocessing, feature extraction, machine learning, and performance evaluation.

We utilised several MATLAB toolboxes and libraries for implementing the proposed methodologies, including:

- Signal Processing Toolbox: For preprocessing and filtering the RSSI measurements.

- Statistics and Machine Learning Toolbox: For implementing dimensionality reduction techniques (e.g., PCA), feature selection methods, and machine learning algorithms (e.g., Wk-NN).
- Optimisation Toolbox: For optimising various parameters and hyperparameters within the proposed methodologies.
- Mapping Toolbox: For visualising and analysing the spatial distribution of fingerprints and localisation results.
- Image Processing Toolbox: For assisting in managing the storage requirements and computational complexity when applying a data compression strategy using DCT and incremental updates.

5.4.3 Evaluation Metrics

To assess the performance of the proposed methodologies, we used several evaluation metrics described in Sections 3.2.2.

5.4.4 Experimental Procedure

The experimental procedure involves the following steps:

1. Data Preprocessing: The UJIIndoorLoc dataset is preprocessed to clean and normalise the Wi-Fi signal strength measurements, remove outliers, and extract relevant features for fingerprinting.
2. Model Training: Baseline and optimised RMF models are trained using the preprocessed dataset. The baseline model does not incorporate any optimisation techniques, while the optimised models leverage the proposed methodologies for handling signal heterogeneity, dimensionality reduction, and auto-update RMF.
3. Evaluation: The trained models are evaluated using the test portion of the UJIIndoorLoc dataset. The localisation accuracy and mean positioning error are calculated to assess the performance of each model.
4. Results Analysis: The experimental results are analysed to evaluate the effectiveness of the proposed methodology in improving the accuracy, robustness, and efficiency of RMF systems. Comparative analyses are conducted to contrast the performance of baseline and optimised models.

By analysing the localisation accuracy metrics under different scenarios, we can evaluate the effectiveness of our proposed approach compared to a baseline RMF method without optimisation techniques.

This section outlines the experimental setup for evaluating the proposed methodology using the UJIIndoorLoc dataset in the MATLAB environment. The subsequent section will present the experimental results and provide information on the performance of the optimised RMF system.

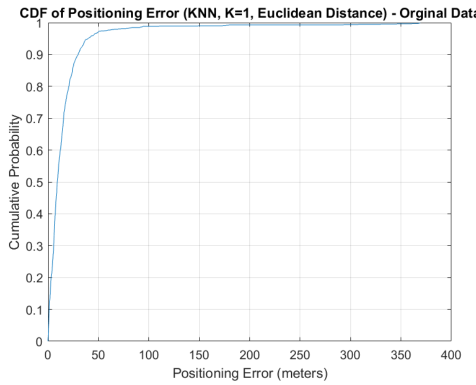
5.5 Results and Discussion

5.5.1 Heterogeneity Mitigation

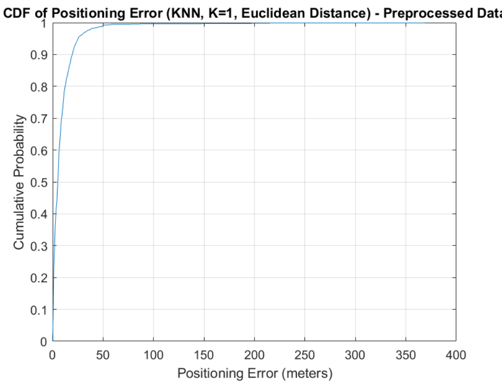
To mitigate the effects of heterogeneity in RMF, we evaluated the performance of the handling techniques through data preprocessing. Figure 5.7 shows the CDF of positioning errors before and after the implementation of data preprocessing techniques, including standardisation, missing value imputation, and exponential data representation. The preprocessing significantly reduces positioning errors, as evidenced by the mean positioning error decreasing from 15.84 metres to 8.58 metres (see Table 5.1). This improvement can be attributed to the mitigation of hardware differences between the mobile devices used for fingerprinting, as well as the enhancement of weak signal strengths, thereby improving classification and regression performance. The RSSI data undergoes preprocessing, including exponential transformation and standardisation of signal strength values. This preprocessing step plays a crucial role in normalising the data, mitigating device heterogeneity in RSSI measurements, and rendering it more suitable for utilisation within the k-NN algorithm.

5.5.2 Dimensionality reduction

RMF comprises 520 features of different WAPs in approximately 20,000 RPs, which can significantly increase processing time. The impact of dimensionality reduction on the number of principal components chosen (N) in PCA affects the accuracy of the localisation, such as the mean positioning error. Our main objective was to ensure that the positioning error levels remained similar or close to the condition without dimensionality reduction. We applied PCA to our baseline k-NN algorithm, aiming to reduce the dimensionality of the radio dataset. We generated a plot showing how the mean positioning error changes with different numbers of principal components. This can help determine the optimal number of components to use for our specific use case. Figure 5.8a illustrates the impact of PCA on the baseline k-NN algorithm. We observed the best positioning error at 120 principal components. Despite a slight



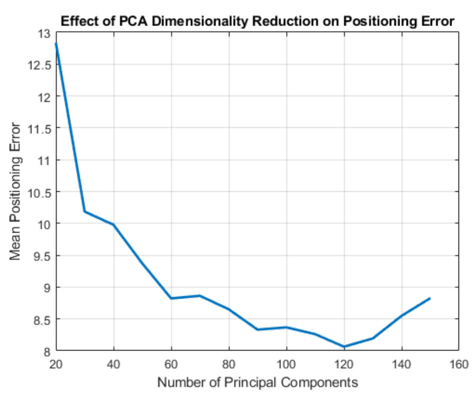
(a) Orginal Dataset



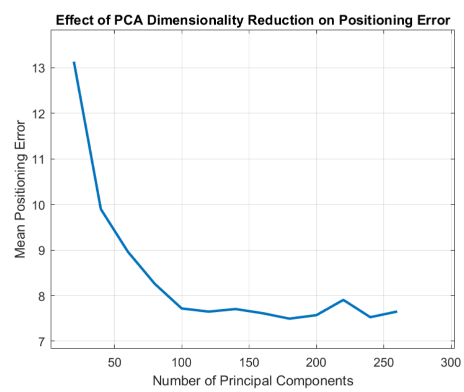
(b) Preprocessed Dataset

Figure 5.7: CDF of Positioning Errors with and without Data Preprocessing

increase in the positioning error, the reduction in algorithm calculation time was significant, decreasing from 31 seconds to 15 seconds, as shown in Table 5.1.



(a) Baseline k-NN: PE vs. Number of PC



(b) Tuned Wk-NN: PE vs. Number of PC

Figure 5.8: Effect of PCA Dimensionality Reduction on Positioning Error

Subsequently, we tested our tuned Wk-NN algorithm (k = 26, correlation distance + preprocessed data) to identify the optimal position error with PCA. Figure 5.8b displays the elbow curve, indicating that the lowest mean positioning error of 7.49 metres was achieved with 180 principal components. Although this result is approximately 0.1 metres higher than the best result obtained without PCA (7.39 metres), the percentage change from 7.39 to 7.49 metres is only approximately 1.35%. Furthermore, the computational time was halved, from 28 seconds to 14 seconds, as demonstrated in Table 5.1. This achievement represents a significant advance in RMF

Description	K	BLD	FLO	Success	MAE	Time
KNN BaseLine	1	98.83	75.33	87.08	15.84	35.74
KNN + Pre-processed Data	1	99.64	92.52	96.08	8.58	14.82
KNN + Preprocessed + PCA (120)	1	99.91	93.34	96.76	8.06	8.55
Tuned WKNN	26	100	96.31	98.15	7.39	28.19
Tuned WKNN + PCA (193)	26	99.82	95.77	97.79	7.49	12.50
Tuned WKNN + PCA (95%)	26	99.91	94.86	97.39	7.94	8.18
Tuned WKNN + PCA (99.5%)	26	99.91	95.68	97.79	7.49	12.18

Table 5.1: Algorithms performance in different RM Optimisation

optimisation and system efficiency. Given that indoor positioning systems operate in real-time, the rapid response of the matching algorithm during the operational phase improves the overall quality of service provided by the system.

After applying PCA to reduce the dimensionality of the UJIIndoorLoc data, which proved effective, we considered determining the number of principal components based on a certain threshold of explained variance (e.g. 95%) instead of a fixed number (193). We applied this approach and achieved optimal results of an ME 7.49 and a success rate of 97.79%, at a threshold of explained variance of 99.5%, as shown in Figure 5.9 and Table 5.1. This provided the fastest calculation time, about 12 seconds and the ME and success rate obtained are similar to the results for the fixed PC of 193, in our case. Notably, PCA leads to a reduction in data size, which helps reduce storage requirements in the long run while maintaining the efficiency of the matching algorithm.

PCA is a strong choice for reducing dimensionality while preserving most of the information. The optimal reduction in the number of principal components chosen (N) depends on the dataset used and the desired trade-off between accuracy and computation time.

5.5.3 Auto-update

A strategy has been developed to integrate new query data into the fingerprint database and update the positioning model accordingly. This auto-update mechanism is essential for adapting to changes in the environment or signal propagation characteristics over time. To evaluate its performance, we simulated changes by introducing new data points as new user queries.

For this evaluation, we preserved 30% of the validation dataset to be used as user queries. By comparing the MAE obtained with and without the user query-based fingerprint update mechanism, it is possible to assess how the auto-update approach improves accuracy over time as user data is integrated. Figure 5.10 illustrates this

Figure 5.9: MAE vs PCA Explained Variance Threshold

slight improvement. Although the success rate does not change before and after the update, it shows improvement compared to the result presented in Table 5.1 as 97.94%. This improvement can be attributed to the gradual refinement of the model's understanding of user queries over time. With the auto-update approach enabled, the model incorporates feedback from user queries, allowing adaptation and refinement of its predictions based on real-world usage patterns. As a result, although the success rate remains consistent, the ME decreases slightly, indicating a more accurate positioning system overall. This iterative learning process, depicted in Figures 5.12 and 5.13, showcases how the model's performance evolves as it continuously integrates new data and adjusts its parameters.

The core concept here is the weighted averaging method, where the delta is calculated based on the error. In our implementation, delta serves as a weighting factor based on the estimated error, ensuring that higher error values lead to lower delta, giving more weight to the user data for a more significant update. Thus, we compute delta as the inverse of the error, with a small constant added to the denominator to prevent division by zero and limit the maximum value of delta.

First, calculate the error:

$$error = |\boldsymbol{x}_i - \text{predictedLocation}|$$

Next, calculate the delta based on the error and add a small constant to prevent

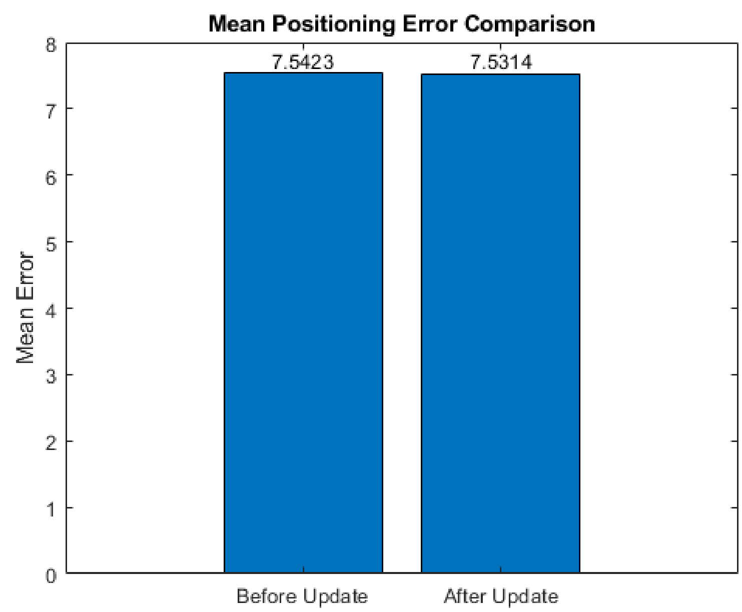


Figure 5.10: Mean Absolute Error Comparison

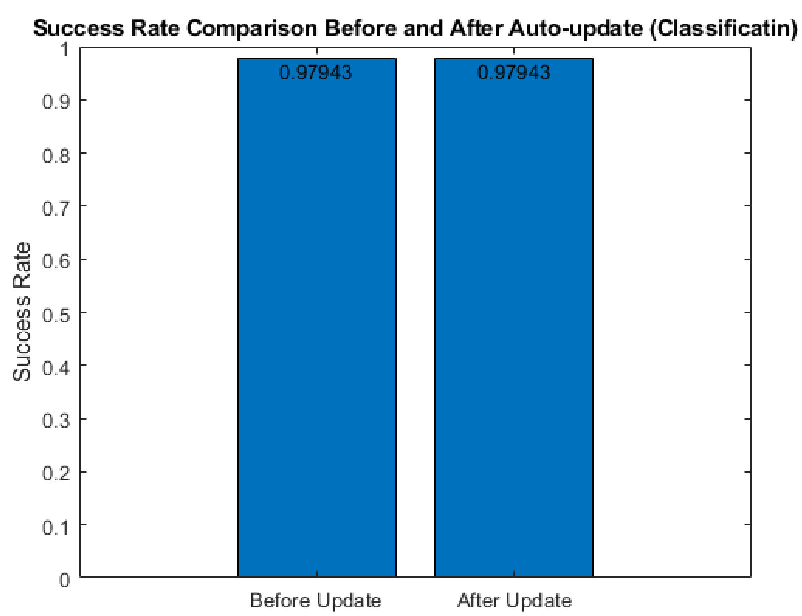


Figure 5.11: Success Rate Comparison

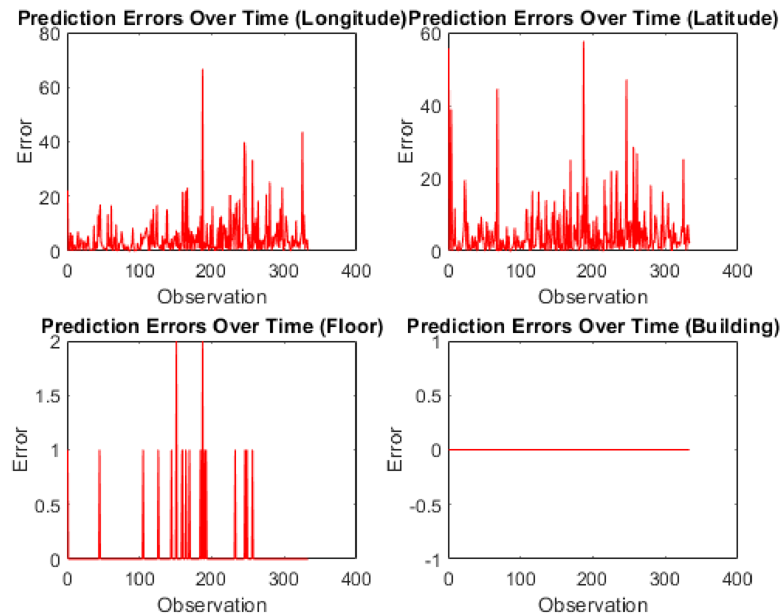


Figure 5.12: Predicted Locations Over Time

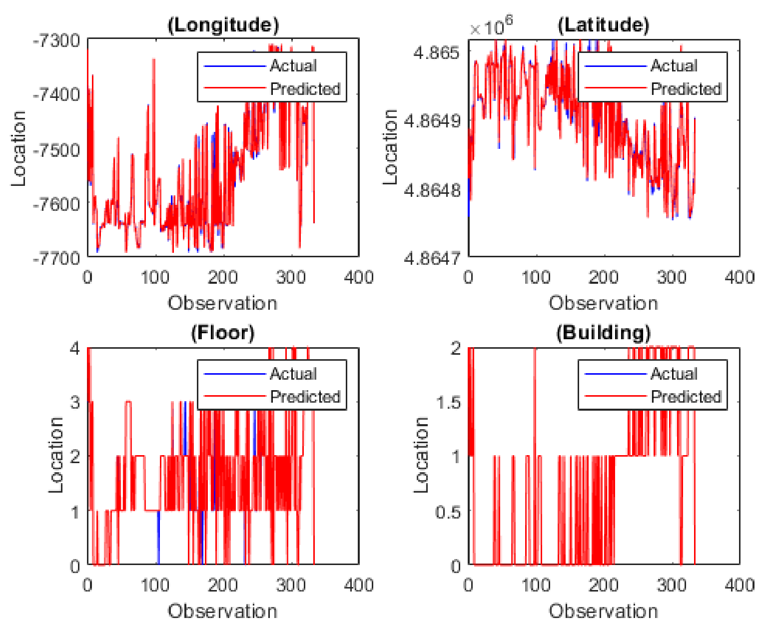


Figure 5.13: Actual vs Predicted Locations

division by zero:

$$delta = \frac{1}{(error + 0.01)}$$

.

Finally, limit the maximum value of delta to 1:

$$delta = \min(\delta, 1)$$

.

In this method, delta is calculated as the inverse of the error, with a small constant of 0.01 added to the error to prevent division by zero. The min function is then applied to limit the maximum value of delta to 1. However, it is important to note that this approach to calculating delta based on error may not be optimal for all use cases. Experimenting with different methods for calculating delta is advisable to determine the most effective approach for each specific dataset and model.

Following the auto-update process, the performance of the updated model is evaluated to assess the extent of improvement. Plots of actual vs. predicted locations, as depicted in Figure 5.13, are typically generated to visualise the model's performance.

The model is updated with the new fingerprint in each iteration of the loop. However, depending on the size of the user queries and the complexity of the model, this process could potentially be time-consuming. If performance becomes a concern, it may be necessary to reduce the frequency of model updates. To conserve computational resources, reducing the frequency of model updates is a common strategy. Batch updates are employed, where instead of updating the model with each new fingerprint, a collection of new fingerprints is accumulated, and the model is updated with the entire batch at once. In our experiment, we implemented batch processing by initialising a batch size (in our case, 43) and initialising arrays to store the batch of new fingerprints and their corresponding locations as shown in Algorithm 4. This approach allows the algorithm to update the models in batches, enhancing efficiency and potentially reducing memory usage.

5.5.4 Database Management

As the fingerprint database and user query data expand, it becomes imperative to consider computational efficiency by exploring techniques such as approximate nearest neighbour search, data compression, and distributed computing. Integrating these techniques with the strategies developed in the previous sections can significantly enhance the efficiency and scalability of RMF systems, particularly for resource-constrained and real-time applications of indoor positioning systems.

Following the implementation of optimisation strategies, including data preprocessing, dimensionality reduction, and auto-update mechanisms, we experimented

Algorithm 4 Auto-Update Mechanism in Batch Process

Initialise *predictedLocations*, *errors* as empty lists
Set *batchSize* to 43
Initialise *batch Fingerprints for Longitude, Latitude, Floor and Building* as empty lists
Set *yTrain for Longitude, Latitude, Floor, Building* to corresponding columns of *yTrain*
for i from 1 to height of *xUpdatePCA* **do**
    Predict location for each model using *xUpdatePCA*[i]
    Add predicted location to *predictedLocations*
    Calculate error for each model
    Add error to *errors*
    Calculate *delta* for each model
    Perform update for each model
    Add updated fingerprint and actual location to batch for each model
    if size of *batchFingerprintsLongitude* is greater than or equal to *batchSize* **then**
        Convert updated fingerprint to a table
        Set variable names of *batchFingerprintsTable* to those of *xTrainPCA*
        Add updated fingerprint and actual location to training set
        Update the models
        Clear the batch
    end if
end for

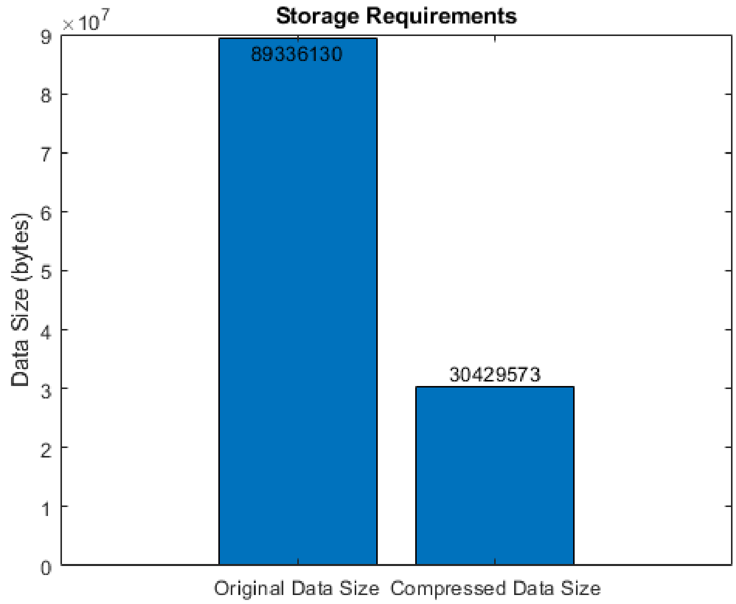


Figure 5.14: Database Storage Requirements

with various data compression strategies, such as data clustering and Discrete Cosine Transform (DCT), to address the challenges of managing large fingerprint databases. We integrated these database management strategies into our existing RMF optimisation framework. We employed a KD-tree-based nearest-neighbour searcher for efficient storage and retrieval of fingerprint data based on location coordinates. Additionally, we applied compression techniques such as gzip and DCT to further reduce storage requirements. Figure 5.14 shows the effect of storage before and after applying data compression. However, after evaluating their impact on performance, we chose to retain PCA as the most effective technique. PCA effectively reduces the dimensionality of our data, thereby lowering storage requirements, while maintaining an acceptable level of performance, as discussed in Section 5.5.2

We measured and compared storage size, retrieval time, and positioning accuracy before and after applying compression and indexing techniques, as depicted in Figure 5.15.

The integration of indexing and data compression strategies involved implementing spatial indexing using KD-tree and PCA. We evaluated the performance of these strategies by measuring storage requirements, retrieval time, and positioning accuracy before and after implementation. The results, shown in Figure 5.16, highlight the success in maintaining effective database operation after the auto-update, despite the high memory usage, as depicted in Figure 5.17.

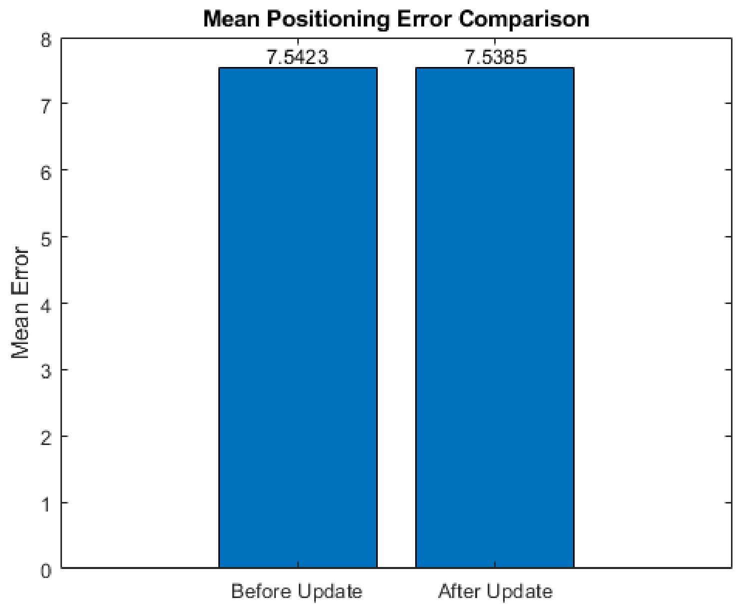


Figure 5.15: MAE Comparison Before and After the Update

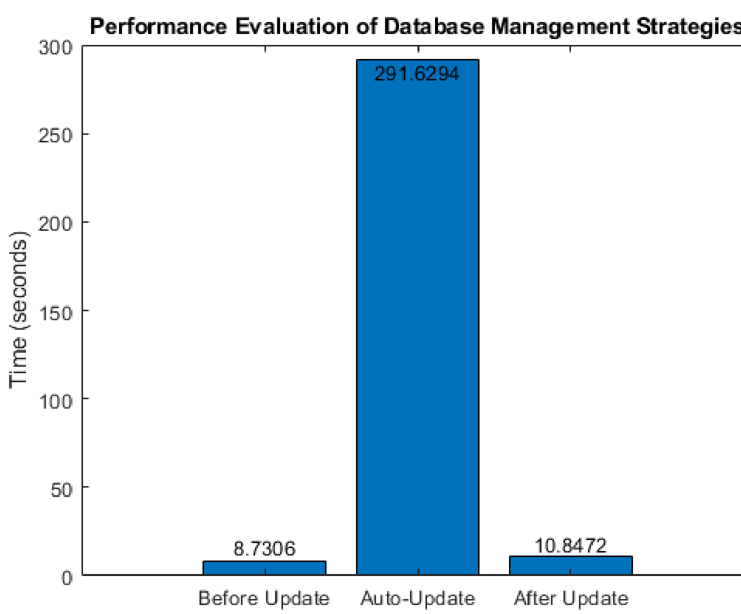


Figure 5.16: Performance Evaluation of Database Management Strategies

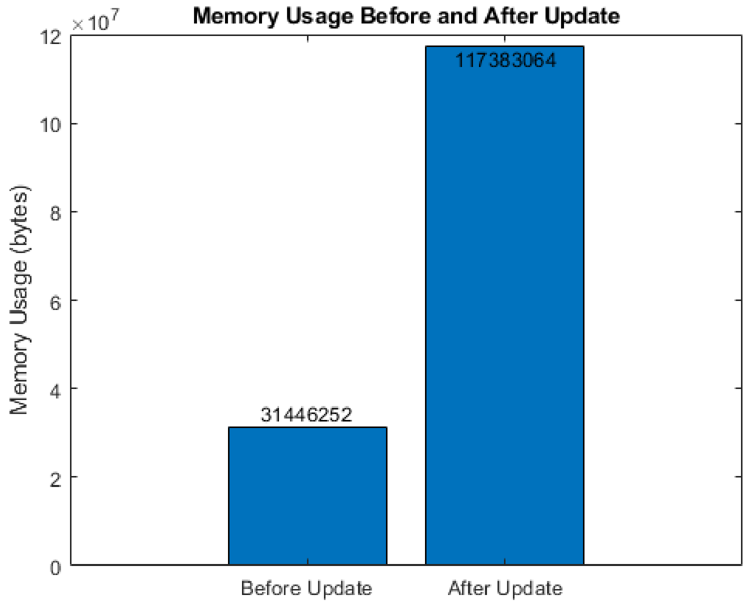


Figure 5.17: Memory Usage Before and After Update

Figures 5.15, 5.16, 5.17, 5.14 and the tabulated results in Table 5.2 provide a comprehensive evaluation of the performance improvements achieved through the integration of optimisation techniques, compression methods and database management strategies within an RMF system. These findings contribute significantly to the discourse surrounding the optimisation of indoor positioning systems in resource-constrained environments.

Beginning with an analysis of data size and compression ratio, the original dataset, comprising 89,336,130 bytes, was reduced substantially to 30,429,573 bytes after compression. This marked compression ratio of approximately 2.94 underscores the efficacy of the employed compression techniques in significantly reducing storage requirements.

The investigation of positioning error, a critical metric for assessing system accuracy, revealed only marginal changes between the mean positioning errors before and after updates. This consistency suggests that the optimisation and compression procedures did not adversely affect the system's accuracy in localising users within indoor environments.

Moreover, the constancy of the success rates both before and after updates, remaining at 97.9434%, underscores the robustness of the system's performance across different operational states. This steadfast accuracy in user localisation reinforces the system's reliability even during the implementation of optimisation measures.

However, the temporal analysis reveals notable shifts in processing time. While the system exhibited a slight increase in processing time from 6.60 seconds before updates to 8.66 seconds after updates, the auto-update mechanism caused a substantial rise in processing duration, escalating from 6.60 seconds to 299.73 seconds. This stark increase underscores the computational overhead associated with updating the database, necessitating a balance between system responsiveness and the frequency of auto-updates.

Memory utilisation emerged as another critical consideration, with a significant increase in memory usage from 31,446,252 bytes before updates to 117,383,064 bytes after updates. This marked increase of 85,936,812 bytes highlights the resource implications of implementing optimisation and compression techniques, particularly concerning memory consumption.

Overall, the findings encapsulate a nuanced understanding of the interplay between optimisation strategies, compression methodologies, and database management techniques within RMF systems. While these interventions succeed in reducing data size and maintaining accuracy, they also introduce trade-offs in processing time and memory utilisation. Consequently, a sensible balance between performance optimisation and resource efficiency is imperative to ensure the seamless operation of indoor positioning systems in real-world contexts. These modifications significantly improve the efficiency and scalability of RMF systems to handle large fingerprint databases, particularly in resource-constrained and real-time scenarios of indoor positioning systems.

Metric	**Value**
Original data size	89,336,130 bytes
Compressed data size	30,429,573 bytes
Compression ratio	2.9358
MAE before update	7.5423 meters
MAE after update	7.5385 meters
Success rate before update	97.9434%
Success rate after update	97.9434%
Time before update	6.60 seconds
Auto-update time	299.73 seconds
Time after update	8.66 seconds
Memory usage before update	31,446,252 bytes
Memory usage after update	117,383,064 bytes
Increase in memory usage due to update	85,936,812 bytes

Table 5.2: Performance Metrics

.

As part of future research directions, we could explore additional techniques, such as approximate nearest-neighbour search, data compression, and distributed computing, to further enhance the efficiency of RMF systems. Investigating the impact of various compression algorithms and indexing techniques on model performance could provide valuable insights into designing efficient RMF systems.

Examining the influence of various compression algorithms and indexing techniques on model performance could offer valuable insight into designing efficient RMF systems. Additionally, exploring the feasibility of re-compressing data for online operation and its impact on real-time performance could be considered in future studies.

While data compression techniques can help manage large datasets in indoor positioning systems, it is essential to carefully consider the choice of compression algorithm and its impact on model performance. Experimenting with various approaches and thoroughly evaluating their impact on system performance are crucial steps toward achieving an optimal balance between storage efficiency, computational efficiency, and model performance.

As observed during the experiments, one of the primary challenges encountered by the fingerprint-based method is the substantial data size of the RMF, wherein each fingerprint sample comprises the RSSI of surrounding WAPs. In practical terms, this means the widespread deployment of Wi-Fi networks results in the detection of a large number of WAPs, many of which are non-informative and redundant. Additionally, some WAPs may have weak signals due to the considerable distance between the user and the WAPs, leading to inconsistencies within the RM. These inconsistencies complicate not only the classification process but also escalate the computational costs during the online phase.

The classification process's reliance on a vast database renders it impractical for real-time systems due to the associated computational burden. Consequently, there is a pressing need to mitigate computational costs to reduce time delays and conserve memory resources in real-time systems. Strategies aimed at optimising the RMF and reducing computational costs are imperative for enhancing the efficiency and feasibility of real-time fingerprinting-based methods.

The framework we present demonstrates an auto-update system that updates its models with new data, which can be useful in scenarios where the environment changes over time. The models are evaluated in each iteration, allowing for the monitoring of their performance as they update. We selected the Wk-NN algorithm for its simplicity and effectiveness in many classification and regression tasks. The use of PCA helps to reduce the dimensionality of the data, which can improve the efficiency and performance of the k-NN models. The proposed framework also demonstrates good practices in data preprocessing, such as the handling of missing values and transforming the data to a suitable scale. Utilising correlation-based distance and inverse distance weighting in the k-NN models can improve their accuracy by giving more importance to closer neighbours and features more correlated with the response.

Unfortunately, MATLAB lacks support for some advanced database management strategies, such as distributed database management, caching, and prefetching. These strategies are typically implemented at the database management system (DBMS) level and would require a DBMS supporting these features, such as MySQL, PostgreSQL, or MongoDB.

Nevertheless, we confidently provide a framework that effectively guides the efficient design of Wi-Fi Fingerprinting IPS systems. Our approach has efficiently optimised the RMF database, resulting in an overall improvement in system performance when compared with baseline methods and state-of-the-art techniques from the literature. Although using the updated dataset in a small amount may not fully reveal future performance implications, it provides a good starting point for exploring the best available options for designing an efficient IPS system.

5.6 Summary and Conclusion

5.6.1 Summary

This chapter investigated critical strategies for Radio Map Fingerprint (RMF) optimisation within Wi-Fi based Indoor Positioning Systems (IPSs), focusing on enhancing system accuracy, efficiency, and adaptability. The research specifically addressed the challenges of signal heterogeneity due to device variability, the high dimensionality of fingerprint data, the need for dynamic RMF updates, and the efficient management of large fingerprint databases.

A quantitative, experimental approach was adopted using the UJIIndoorLoc dataset to evaluate a proposed suite of optimisation techniques. Key findings from this investigation include:

- **Heterogeneity Mitigation:** The application of data preprocessing techniques, notably signal standardisation and the use of an exponential data representation, proved effective in mitigating RSSI variations arising from device differences. This contributed to a significant reduction in mean positioning error from 15.84 metres (baseline) to 8.58 metres after preprocessing.

- **Dimensionality Reduction:** Principal Component Analysis (PCA) was successfully employed to reduce the dimensionality of the RMF. It was demonstrated that with an optimal number of principal components (e.g., 180 components for the tuned Wk-NN, or by retaining 99.5% explained variance), the computational time for the positioning algorithm could be approximately halved (e.g., from 28 seconds to around 12-14 seconds) while maintaining a mean positioning error (7.49 metres) very close to that achieved with the full-dimension dataset (7.39 metres). This also contributes to reduced data storage requirements.

- **Auto-Update Mechanism:** An auto-update mechanism for the RMF, leveraging user queries and a weighted averaging approach, was proposed and evaluated. Simulations indicated that this mechanism can lead to incremental improvements in positioning accuracy over time (e.g., MAE slightly improving from 7.5423m to 7.5385m after updates) by adapting the radio map to new data, thereby reducing the reliance on manual recalibration.

- **Database Management:** Strategies for managing the incrementally growing RMF, including the use of KD-trees for efficient nearest-neighbour search and evaluation of data compression (though PCA was primarily adopted for its dual benefit of dimensionality reduction and implicit compression), were considered. The experiments showed that while auto-updates increase database size and memory usage, the system maintained effective operation.

Collectively, these findings support the hypothesis that a structured approach to RMF optimisation, incorporating data preprocessing, dimensionality reduction, and adaptive update mechanisms, can substantially enhance the overall efficiency, scalability, and long-term viability of Wi-Fi fingerprinting-based IPS.

5.6.2 Conclusion

The research presented in this chapter makes a significant technical contribution by demonstrating a practical and effective framework for optimising Radio Map Fingerprints in Wi-Fi based indoor positioning. The primary contribution is the **synergistic application and evaluation of a multi-faceted optimisation strategy**, encompassing heterogeneity mitigation, PCA-based dimensionality reduction, and an RMF auto-update mechanism, which collectively enhances system performance on a large-scale, real-world dataset.

It was concluded that effective data preprocessing is crucial for addressing device heterogeneity, leading to substantial initial gains in accuracy[cite: 1804]. The strategic use of PCA offers a compelling trade-off, drastically reducing computational load and storage needs with only a minimal impact on positioning accuracy, a key finding for developing scalable systems. Furthermore, the proposed auto-update mechanism provides a pathway towards more adaptive and self-maintaining IPS, reducing the lifecycle costs associated with manual RMF updates.

While the implemented database management techniques (KD-tree indexing and the benefits of PCA for data size reduction) proved effective, the chapter also acknowledges the computational overhead and increased memory usage associated with auto-updating large RMFs. These considerations highlight the practical challenges in deploying fully autonomous and continuously learning RMF systems.

The findings affirm that optimised RMFs are fundamental to achieving efficient and scalable indoor positioning. This chapter's work provides valuable insights and empirically validated techniques for addressing key limitations in Wi-Fi fingerprinting, thereby contributing to the advancement of more robust and practical indoor positioning solutions for applications such as navigation, asset tracking, and location-based services. The optimised RMF and insights gained herein also inform the design of the cloud-based architecture presented in Chapter 6.

Chapter 6

IPS Cloud-based System Design

In this chapter, we develop and design a framework for the cloud-based Indoor Positioning System (henceforth referred to as CB-IPS) to simulate a CB-IPS system. We conduct a thorough conceptual analysis and testing of the proposed design framework and cloud paradigm to achieve an efficient design. CB-IPS is proposed and tested to offer scalability, efficiency, and radio map storage management, where central cloud and edge cloud layers are used to distribute datasets.

6.1 Introduction

The use of wireless technology has grown significantly. Many smart devices are used indoors to foster intelligent environments, especially when there is a need for location awareness. Cloud computing facilitates seamless implementation of the IPS, particularly when scalability is required. The cloud infrastructure provides cost-effectiveness and efficient performance to support many on-premise services.

CB-IPS offers positioning, localisation, and navigation services over the Internet. By delivering these services, they eliminate the need for complex installations and administration, leveraging cloud and other computing paradigms to ensure high availability, computational power, storage capabilities, and ubiquitous computing, thereby avoiding overloading the user's device. However, the efficient design and management of such systems present significant challenges due to the generation and storage of large databases during the fingerprinting process.

The adoption of CB-IPSs has gained popularity due to their high availability, computational and storage capabilities, and ubiquitous computing. Offering a range of benefits, including preventing overloading of the user device [169], [214].

In this research, we explore the design considerations of indoor positioning systems within the cloud environment. We also summarise the challenges of this implementation and define the perspective of future indoor positioning and navigation systems through the cloud infrastructure. Then, the cloud architecture using an IPS is described and envisioned.

We explore the use of cloud computing and edge cloud technologies in the indoor positioning system to enhance performance and more effectively manage large databases of fingerprinting data. In addition, we review current technologies used for indoor positioning systems and survey ongoing research within this field. Although many new positioning methods have been developed in recent years, little has been done to integrate them into a robust, reliable, and cost-effective system, especially when using cloud computing. To address this need, the chapter presents a clear step-by-step approach that avoids key challenges in developing such a system, including complexity, context, ambiguity, and data handling.

In this chapter, we develop and design an indoor positioning system using MATLAB and the UJIIndoorLoc Dataset for system modelling and simulation. We explore various design frameworks and algorithms to achieve an efficient design, supported by conceptual analysis and system modelling, to manage large-scale fingerprint databases. This exploration aims to provide a comprehensive understanding of the system's functionality and identify potential areas for improvement.

6.2 System Architecture

Let us first outline a conceptual model to offer a high-level overview of a Wi-Fi fingerprint indoor positioning system. It is important to note that the actual implementation of such a system may vary according to the specific requirements and constraints of the indoor environment. Moreover, the system performance can be further enhanced by integrating additional information, such as signal propagation models, building layouts, or user movement patterns.

An IPS based on Wi-Fi fingerprinting typically comprises three main components, as illustrated in Figure 6.1. Firstly, there is the Wi-Fi infrastructure, which encompasses a set of WAPs deployed throughout the indoor space. This infrastructure is typically connected to an Internet Service Provider (ISP) for Internet access and communication purposes.

Within this network, there is a Positioning Server (PS), which may be located either locally on-premise or remotely in the cloud. This server facilitates the processing and analysis of Wi-Fi signals to estimate the positions of mobile devices within the indoor environment. Finally, the system enables IPS operations on mobile devices, such as smartphones or laptops. These mobile devices use the Wi-Fi infrastructure to communicate with the Positioning Server, allowing users to access location-based services and applications.

This conceptual model provides a foundational framework for understanding

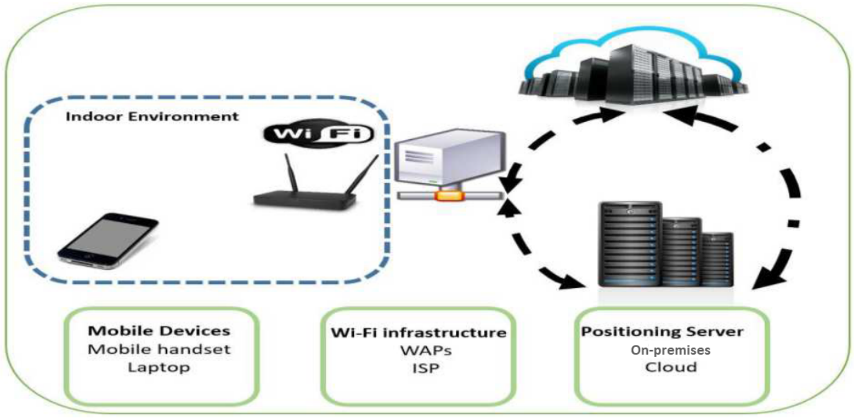


Figure 6.1: Schematic of a Generic Wi-Fi IPS

the key components and functions of a Wi-Fi fingerprinting indoor positioning system. However, it is essential to acknowledge that actual implementation may involve additional complexities and considerations tailored to specific use cases and environments.

The indoor Wi-Fi fingerprinting positioning system operates through two distinct phases, offline and online, as discussed in Chapter 3. However, in our research, conducted as a software-based simulation using an open-source Wi-Fi fingerprint dataset (UJIIndoorLoc), we omit the offline phase. Therefore, our research focuses exclusively on the online phase, and our modelling efforts do not involve direct collection of RSSI data from WAPs. Instead, we rely on the available RSSI data from the UJIIndoorLoc dataset, processed through a range of algorithms. This methodology, in conjunction with the intricate interplay between system components (e.g., operation, matching, and data management), allows us to conduct system-level modelling, facilitating the simulation and evaluation of diverse algorithms and frameworks.

6.3 System Model

Our system model assumes an indoor area covered by a Wi-Fi-based indoor positioning system featuring a WLAN deployed on a single floor of a building. We assume that M WAPs are distributed throughout the area, ensuring comprehensive coverage. To facilitate location estimation, we define a square grid on the two-dimensional floor plan, limiting the estimated positions of MDs to points on this grid.

With the grid structured to include L points along both the x and y axes, we derive $L \times L = L^2$ potential positions within the area. Each location can be represented by a label (x, y), signifying its 2D coordinates (usually latitude and longitude) on the floor plane, with all coordinates maintaining a zero height $(z = 0)$. During the offline phase, a total of $N = L^2$ RSSI vectors are collected through site surveys conducted at predetermined grid points or RP. These RSSI measurements are meticulously recorded in an RMF database, with each entry mapping the grid coordinates (x, y) to the corresponding RSSI values from all WAPs in the area. In the online phase, a sample RSSI vector is obtained from all WAPs at the current MD's position. This vector is then compared against all existing entries in the database, with the fingerprint entry exhibiting the closest match used as an estimate of the user's current location.

To formalise our system modelling, we utilise mathematical constructs, employing sample and fingerprint vectors in estimating MD locations. The sample vector $\boldsymbol{r}$ comprises RSSI samples measured at the MD from the M WAPs in the area, with each r_i (in dBm) treated as a Gaussian random variable. We assume mutual independence among all r_i, with a known standard deviation σ (in dB) and the means $E[r_i] = p_i$. Conversely, the fingerprint vector $\boldsymbol{f}$ in the RMF comprises a means of all RSSI random variables at a specific location from the M WAPs. To determine the MD's position, we calculate the Euclidean distance $(d(\boldsymbol{r}, \boldsymbol{f}))$ between the RSSI vector sample $\boldsymbol{r}$ and the fingerprint $\boldsymbol{f}$ (as per the distance definitions in Section 3.2.2).

The diagram in Figure 6.2 illustrates a platform-independent system architecture based on a server. In this scenario, the MD (typically a mobile phone) has no software installed and only participates by periodically sending packets over Wi-Fi. This requirement is typically satisfied by most Wi-Fi-enabled devices, including mobile phones, as probe requests are sent at set intervals to search for new networks in the area [215]. As no additional software can be added to the device, the handling of RSSI data and positioning must be performed externally on a server.

In such a system, signals emitted by an MD are received by nearby WAPs and forwarded to a central server, where a PS processes the data and provides access to the RMF database. An MD in the area can then access the positioning system through its browser or app using the user interface (UI) and Transmission Control Protocol (TCP) applications to communicate with the server. The MDs in this system typically run on Android operating systems, such as smartphones or tablets, because of their open-source nature and large user base.

Additionally, in our system modelling, we incorporate the following constraints:

- The user or device is static with no movement to provide positioning, not a trajectory.
- 2-D model coordinates (X, Y) are used instead of 3-D model (X, Y, Z) in room space, although elevator dimensions (Z) are considered for predicting the floor level.

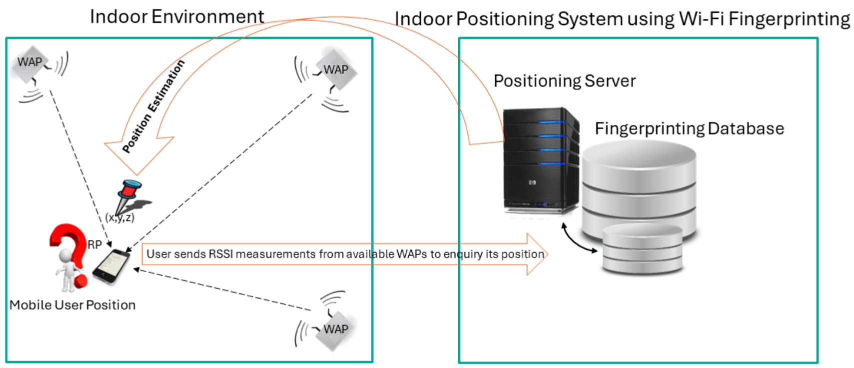


Figure 6.2: Wi-Fi Fingerprinting Technique

6.4 Cloud Architecture

The design and structure of a cloud computing environment encompass the various components and layers that make up the cloud architecture. According to the deployment model or the service level agreement (SLA)[216], the cloud architecture can be provided as:

- **Infrastructure as a Service (IaaS):** At the base level of the cloud architecture lies Infrastructure as a Service (IaaS), which provides virtualised computing resources over the Internet. This includes virtual machines, storage, and networking components, allowing users to deploy and manage their applications without having to invest in physical hardware.

- **Platform as a Service (PaaS):** PaaS builds on top of IaaS, offering a platform for developers to build, deploy, and manage applications without worrying about the underlying infrastructure. PaaS providers offer tools and services such as development frameworks, databases, and application runtime environments, streamlining the application development process.

- **Software as a Service (SaaS):** SaaS delivers software applications over the Internet on a subscription basis. Users can access these applications through a web browser without having to install or maintain any software locally. SaaS offerings cover a wide range of applications, including email, web applications, and more.

Cloud architecture can be deployed in various availability models[216], including public, private, and hybrid clouds. Public clouds are hosted and managed by third-party providers, offering resources to multiple tenants over the Internet. Private clouds, on the other hand, are dedicated to a single organisation and hosted either on-premise or by a third-party provider. Hybrid clouds combine elements of both public and private clouds, allowing organisations to leverage the benefits of both deployment models. One of the key advantages of cloud architecture is its ability to scale resources up or down based on demand. Scalability refers to the ability to add or remove resources dynamically to handle fluctuations in workload, while elasticity refers to the automatic scaling of resources in response to changes in demand[216].

Overall, cloud architecture plays a crucial role in enabling organisations to harness the power of cloud computing, providing a scalable, flexible, and cost-effective platform for deploying and managing their IT resources and applications.

6.4.1 Cloud Paradigms

Cloud computing is a key technology that enables large-scale technological innovation. As emerging technologies like IPS continue to evolve, cloud platforms offer a diverse array of computing paradigms[140], known as:

- **Cloud Computing (CC):** Cloud computing refers to the delivery of computing services, including servers, storage, databases, networking, software, and more, over the Internet (the cloud). Cloud computing enables users to access resources and services on demand, without the need for on-site infrastructure or management. It offers scalability, flexibility, and cost efficiency to organisations of all sizes.
- **Edge Computing (EG):** Edge computing extends the capabilities of cloud computing to the edge of the network, closer to the source of data generation or consumption. In edge computing, data processing and analysis are performed locally on edge devices, such as routers, gateways, or IoT devices, rather than relying solely on centralised cloud servers. This reduces latency, bandwidth usage, and dependency on cloud infrastructure, making it ideal for real-time applications and use cases where data needs to be processed quickly.
- **Fog Computing (FC):** Fog computing is an extension of edge computing that emphasises the distribution of computing resources and services between the cloud and the edge of the network. In fog computing, intermediate computing nodes, called fog nodes, are deployed at various points in the network to provide computing, storage, and networking services closer to the edge devices. Fog

computing enables efficient data processing, analysis, and storage at the network edge, while still leveraging the scalability and resources of the cloud.

- **Mist Computing (MC):** Mist computing is a newer paradigm that focuses on pushing computing and data processing even closer to the data source than edge computing. Mist computing involves deploying lightweight computing resources, such as micro data centres or edge servers, directly on or near the edge devices themselves. This allows for ultra-low latency and real-time processing of data at the device level, without the need to transmit data to centralised cloud or edge servers. Mist computing is particularly suitable for highly distributed and latency-sensitive applications, such as industrial IoT, smart cities, and autonomous vehicles.

Although cloud computing provides centralised computing resources over the Internet, edge computing, fog computing, and mist computing aim to distribute computing resources closer to the edge of the network or data source, allowing faster response times, improved scalability, and better support for real-time applications [140]. The advancement of computing paradigms of using Edge and Fog computing in IPS is quite good, as a network of peripheral devices such as mobile phones and WAPs, which are connected to the Fog layer Routers, gateways, and servers to perform indoor position estimates. Unlike CC, which processes data in a central public cloud, connected devices on the EC and FC paradigms send data to the cloud or receive data from the cloud as and when needed. Such an architecture reduces latency and enables energy-efficient data processing. EC is best suited for real-time sensitive systems such as IPS with strict deadlines.

However, because of resource limitations in computing power and memory, EC is insufficient for meeting some of the system requirements. Therefore, the FC architecture concept is the paradigm that can process data faster than CC and closer to EC performance. Both EC and FC can provide a decentralised service across a large area, which serves the purpose of scalability for the IPS system that a cloud-based solution can offer. Thus, the IPS service would greatly benefit from real-time processing and lower latency on a larger scale. Additionally, the service can be geographically distributed for coverage scalability or in the event of instabilities in the central data processor [140], [216], [217].

Furthermore, cloud platforms offer versatile deployment options, including private, public, community, or hybrid clouds, thereby catering to a variety of organisational requirements and preferences [216]. This strategic shift towards cloud-based architectures not only aligns with the evolving needs of indoor positioning services but also underscores the importance of leveraging cloud technologies in enhancing system efficiency and scalability.

6.4.2 IPS Cloud Challenges

Numerous challenges associated with the deployment of IPS on the cloud have been documented [140], [216]–[219]. Common specific challenges in CB-IPS include the following:

- **Privacy and Security:** Security remains a fundamental concern within the cloud computing paradigm, encompassing issues such as the loss of confidential user data, data leakage, and breaches of personal privacy [218]. However, these concerns can be mitigated through data encryption, adherence to standards, and service level agreements [216].
- **Protocols and Standards:** CB-IPS may employ diverse protocols to establish and maintain communication between various cloud-computing paradigms and edge devices. Given the heterogeneity of devices used by users and service providers, adherence to protocols is crucial to mitigate communication issues and ensure a common language [216].
- **Real-time (Latency-sensitive) Services:** Positioning and indoor navigation services require real-time responses with minimal latency to ensure a seamless user experience. The inherent distance between cloud computing location services and end-users, coupled with communication and processing overheads, introduces latency in service delivery [167]. However, cloud models like MC and FC may help reduce latency to some extent as they are closer to the end-user.

Furthermore, the main limitation of the cloud-based Wi-Fi fingerprinting approach, as proposed by contemporary systems such as [167], [220], [221], is its dependence on a server-centric system architecture. While this architecture enables the system's functionality, it introduces several critical issues. The addition of a positioning server increases hardware costs and creates a single point of failure. If the server encounters hardware failures, power outages, or security breaches, the availability of the location service is compromised [216].

In addition, server access involves communication costs and requires continuous network connectivity, which may not be feasible during network downtimes due to maintenance or disasters. This compromises the overall robustness of the positioning system. As a result, system administrators must implement strategies such as backup servers, uninterrupted power supplies, encryption, and authentication protocols to ensure service availability and security. However, these measures inevitably increase the deployment and operational costs of the positioning system.

6.4.3 CB-IPS Design

The initial design phase involved a comprehensive review of theoretical work on CB-IPS [140], [169], [170], [173], [216]–[218], [221]–[223], analysing various methods and systems employed. Subsequently, conclusions drawn from these studies were synthesised to develop an efficient design for CB-IPS.

Traditionally, indoor positioning services have been based primarily on on-premises platforms such as LAN and WAN. However, the growing interest in indoor positioning has spurred the development of methods and techniques for cloud-based indoor location services. This transition extends from back-end services to end-to-end cloud solutions, accommodating various service models, such as SaaS, PaaS, or IaaS [217].

From a software and system development perspective, the design of CB-IPS can be segmented into three key stages: UI Design, Data Design, and Process Design [224]. Although software developers typically prioritise user interfaces, this research concentrates on the data and process design aspects of the system, as depicted in Figure 6.3, to meet the user's specific functional and non-functional requirements.

Functional requirements (FRs) govern the execution of the system and encompass behavioural specifications regarding inputs and outputs. In the context of CB-IPS, these may include ensuring resource and service availability, responding to user requests, calculating position estimations, and transmitting user location estimations. These functional requirements, often supported by non-functional requirements (NFRs), are also known as quality requirements. NFRs establish criteria to support the operation of the system, focusing on aspects such as scalability, reliability, and security. Unlike functional requirements, NFRs delineate the desired behaviour of functions, also referred to as performance requirements. Each utilisation of these NFRs delineates behavioural scenarios through one or more functional requirements [225]. For example, an NFR for scalability may dictate that the system should be able to handle a tenfold increase in user traffic without significant performance degradation. Similarly, a reliability NFR may specify that the system must maintain an uptime of at least 99.9% over a specified period. Lastly, a security NFR may require that user data be encrypted both in transit and at rest to mitigate the risk of unauthorised access. Each of these NFRs influences the functional requirements and design decisions within the IPS framework.

Furthermore, within FRs, achieving varying levels of accuracy in indoor positioning depends on the level of service and the nature of the activities that are being performed. For example, while precise positioning is critical for robot navigation, a university environment may not require high positioning accuracy to locate classrooms. Consequently, algorithms executed on High-Performance Computing (HPC) platforms are likely to yield superior position estimation and accuracy compared to those on edge or fog computing. The emphasis on functional and non-functional requirements in CB-IPS design underscores the importance of addressing

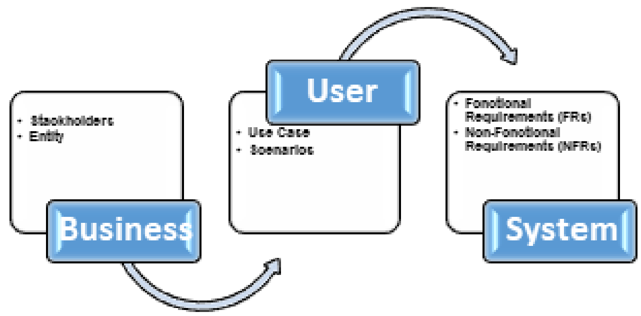


Figure 6.3: Design Requirements Process for Cloud-based IPS

current research focuses in fingerprint positioning. A primary objective in this domain is to reduce computational complexity while simultaneously improving positioning accuracy. This goal is particularly significant in Wi-Fi positioning, where the positioning terminal typically comprises a battery-powered device with limited computational capabilities. Challenges further arise with the management of large fingerprint databases, as exhaustive searching of all fingerprints demands significant computational resources, thereby compromising positioning efficiency.

Given an understanding of the capabilities and limitations of cloud computing, IPS stands to benefit significantly from a hybrid combination of cloud paradigms. This approach aligns with user non-functional requirements, such as fast response time and privacy preservation, which can be effectively met through edge computing. Meanwhile, central cloud resources can handle high-performance computational operations. Thus, cloud computing emerges as a strategic solution, with PaaS offering a particularly appealing option due to its flexibility and control. In our system design case, PaaS proved to be the most suitable option.

By harnessing the computational power of cloud servers and the flexibility of computing paradigms, the burden on the positioning terminal can be mitigated, leading to improved efficiency and scalability in positioning services. This strategic shift towards cloud-based architecture not only enhances computational capabilities for fingerprint positioning but also holds promise for addressing the evolving needs and complexities of indoor positioning systems.

6.5 Proposed CB-IPS Architecture

Based on the insights gleaned from our preceding sections and the outcomes of our testing results, particularly in Chapters (4 and 5) where notable achievements were observed in the classification task, achieving a commendable 100% building hit rate and approximately 96% floor hit rate. However, despite these successes, the regression tasks concerning longitude and latitude did not show significant improvement. This leads to the following conclusion:

Initial Assumptions:

- $BLD = 100\%$ represents the certainty of being within a particular building.
- $FLO \approx 96\%$ represents the confidence level in floor estimating.

Identifying the Building (BLD):

$$BLD = 100\%.$$

Estimating the Floor (FLO):

$$FLO \approx 96\%.$$
$$\text{Estimated Floor: } F\hat{L}O.$$

Utilising Floor Fingerprint Data:
The pseudocode of the algorithm is presented in the Algorithm Appendix, Algorithm 5 as a function. We incorporate both above-floor and below-floor data, if available, to refine the estimation of the target floor. For example, if BLD0 is identified with FLO1, this forces the model to prepare floor data, including the current floor FLO1, the below floor FLO0 and the above floor FLO2.

Enhancing Computational Algorithm:
By integrating the estimated floor $F\hat{L}O$ and complementary data from adjacent floors, our goal is to improve the precision of regression tasks to determine longitude and latitude.
Regression Task for Longitude and Latitude:

$$\text{Algorithm}(BLD, F\hat{L}O, \text{Above-Floor Data}, \text{Below-Floor Data})$$

6.5.1 Model Description

Figure 6.4 illustrates the high-level architecture of the proposed IPS fingerprint indoor positioning system based on the cloud. The proposed CB-IPS architecture presents an innovative approach to serving multi-floor buildings by integrating central cloud computing and edge computing paradigms. To mitigate computational complexity, we propose utilising cloud-based distributed fingerprinting methods tailored for multi-building and multi-floor settings. In this architecture, each floor is assigned a dedicated PS operating on the PaaS model, facilitating online positioning using

Algorithm 5 loadFloorData(buildingID, floorID)

```
floorsLoaded ← empty cell array
% Load current floor
filename ← constructFilename(buildingID, floorID)
if filename exists as a file then
    load data from filename
    floorData ← data
    append filename to floorsLoaded
    display "Current Loaded floor: filename"
else
    raise error "File filename does not exist."
end if
% Load adjacent floor above if it exists
filename_adj_above ← constructFilename(buildingID, floorID + 1)
if filename_adj_above exists as a file then
    load data from filename_adj_above
    append data to floorData
    append filename_adj_above to floorsLoaded
    display "Above Floor Loaded: filename_adj_above"
end if
% Load adjacent floor below if it exists
if floorID > 0 then
    filename_adj_below ← constructFilename(buildingID, floorID - 1)
    if filename_adj_below exists as a file then
        load data from filename_adj_below
        append data to floorData
        append filename_adj_below to floorsLoaded
        display "Below Floor Loaded: filename_adj_below"
    end if
end if
return floorData
```

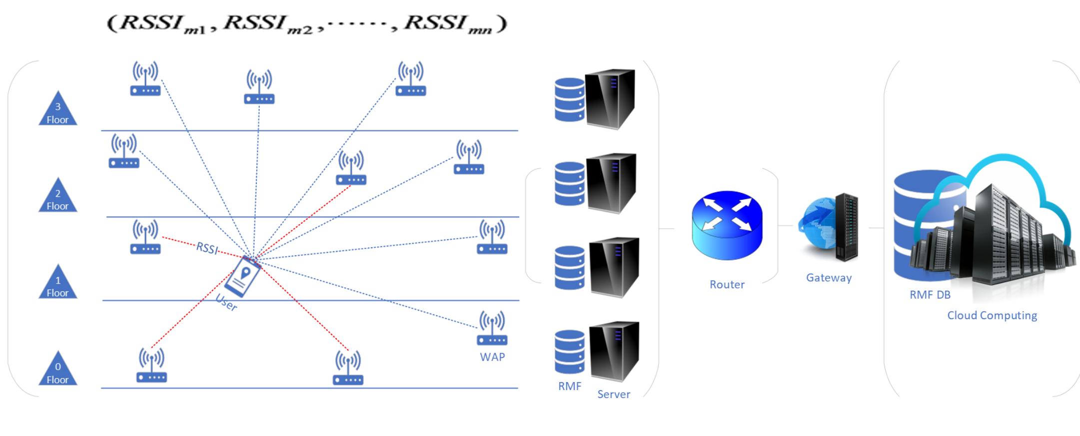


Figure 6.4: Diagram of Proposed CB-IPS

current floor data alongside existing data from adjacent floors. This allocation minimises computational costs and enhances system efficiency.

For simulation and evaluation purposes, we used the UJIIndoorLoc dataset, which encompasses three buildings with varying floor structures. Virtual provision servers at the edge layer serve as virtual machines (VMs) for online floor operations. During positioning operations, two options are considered: either preloading prepared RMFs onto each floor server or dynamically loading required floor data during simulation operations to streamline system complexity. We opt for the latter option as expressed in the Algorithm 5, to simplify system operations, avoiding the need for establishing 21 edge servers and database partitions as required by the former option.

The partitioning technique is employed to utilise only relevant floor data for positioning operations on each specific floor, incorporating data from adjacent floors if applicable. This clustering criterion minimises noise from distant sources of RSSI in the RMF, enhancing the accuracy of position estimations by ensuring a more accurate reflection of RP distributions.

The data centre or central cloud serves as the system's hub, overseeing data flow and operations, as shown in Figure 6.5. Securely hosts accumulated big data from the fingerprint database and identifies the correct building and floor for communication with edge servers, facilitating the loading of specific floor data.

Database partitioning and clustering strategies enhance retrieval efficiency and reduce computational complexity by segmenting radio map fingerprinting data based on building ID and floor number before regression model training. This approach facilitates the training of separate models for each floor, further optimising retrieval efficiency.

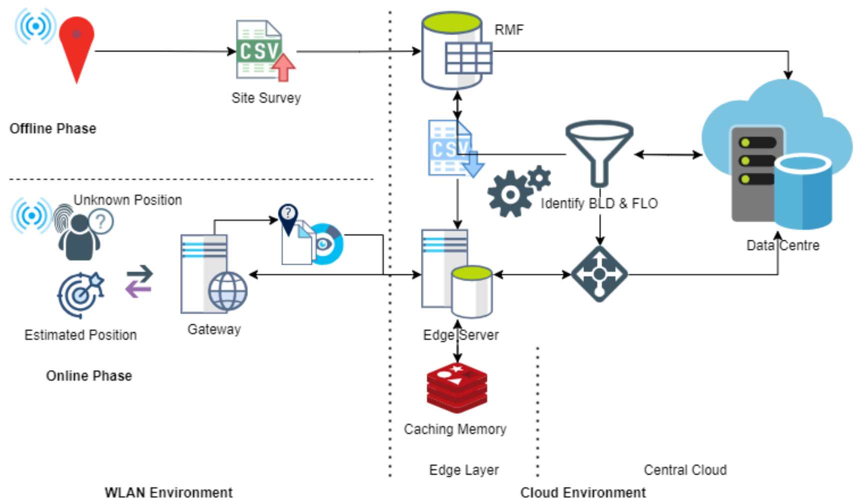


Figure 6.5: Schematic CB-IPS System Design

Our proposed design adopts a caching mechanism and a distributed approach to construct the fingerprint database, facilitating efficient and scalable database management. We also employ the proposed Wk-NN algorithm described in 4.5 for location estimation that balances simplicity and efficiency, making it ideal for real-time IPS operation, leveraging the scalability and elasticity of cloud computing to manage the fingerprint database and perform location estimation tasks effectively.

6.5.2 Implementation Environment

The CB-IPS prototype was fully implemented and evaluated in MATLAB using the experimental setup discussed in Section 6.5.7. Key steps were as follows:

- **Data Import & Preprocessing:**
  - The UJIIndoorLoc dataset (three buildings, 1,174,692 RSS records) was imported using built-in MATLAB functions (`readtable`, `table2array`).
  - Outliers and missing RSS values were handled via median filtering and simple imputation.
  - Z-score normalisation (Section 5.3) and PCA-based dimensionality reduction (Section 5.4) were applied using `zscore` and `pca` functions.
- **Cloud–Edge Architecture Simulation:**
  - The central cloud node was emulated as a MATLAB class (`CloudServer`) managing the global fingerprint database.

  - Edge nodes were instantiated as separate MATLAB objects (`EdgeNode`), each loading floor-specific RMFs via the `loadFloorData` function (Algorithm 5).
  - Communication between cloud and edge was simulated through function calls and shared MATLAB workspaces rather than network sockets.

- **Positioning & Evaluation:**
  - The weighted k-NN algorithm (Section 4.5) was executed at the edge node level via a MATLAB script (`runPositioning.m`).
  - Batch queries were simulated by reading a subset of the dataset and issuing concurrent calls (via `parfor` loops) to emulate up to 500 simultaneous requests.
  - Response times and localisation errors were recorded using `tic`/`toc` and aggregated for performance plots (Figs 6.6–6.9).

This setup ensured that all components of the proposed CB-IPS architecture (Fig. 6.4) were faithfully represented within a single MATLAB environment, allowing reproducible and controlled experimentation on a standard laptop platform.

6.5.3 Scalability Measures in CB-IPS

The scalability of our proposed system is intrinsic to its design, leveraging the capabilities of cloud computing to accommodate any building or floor within the system. Regardless of the number of users or the geographic area covered, our CB-IPS system can dynamically adjust to meet demand fluctuations, benefiting from the scalability features offered by cloud technology.

6.5.4 Privacy Measures in CB-IPS

The proposed system prioritises user privacy by storing only a dataset in the central cloud database for the detection of buildings and floors. This database is used only for the classification task when identifying the correct building and floor for the system. Other responses for the regression task (e.g. longitude and latitude) that reveal the estimated position are hosted in the edge layer by floor servers. Thus, user-sensitive information regarding positioning is kept separate in the edge layer, not in the central cloud.

6.5.5 Response Time Optimisation Strategies

In our proposed system, the implementation of the edge-cloud paradigm ensures that computations are performed closer to the user, thereby guaranteeing quick response times. To mitigate inherent latency in a cloud environment, where operations are conducted remotely from the user's physical location, our system employs two levels of online operation, each optimised for efficient response times:

1. **Cache Mechanism:** The system incorporates a cache mechanism that instantly responds if the user's query is present in the system memory. This mechanism enhances response times by retrieving pre-stored data from memory, eliminating the need for time-consuming computations.

2. **Floor Servers on Edge Layer:** The second level of operation involves floor servers situated at the edge layer. As seen in Figure 6.4, by decentralising processing power closer to the user, these servers facilitate faster responses to online operations, significantly reducing latency.

6.5.6 Resource Management Strategies

Our system model incorporates several resource management strategies to optimise memory usage and processing efficiency:

- **Caching Memory-Efficient:** We implement a Least Recently Used (LRU) caching strategy within the cache mechanism. The LRU cache removes the least recently used items first when the cache is full and a new item needs to be added. This strategy ensures that the cache size never exceeds the specified maximum size, helping to control memory usage efficiently. The cache size can be adjusted based on specific requirements and available system memory. Our caching implementation is presented in Algorithms Appendix, Algorithms 6, 7 and 8

- **Edge-Layer Storage Optimisation :** Floor data is loaded only for floor-specific operations, conserving system storage space and processing resources, as presented in our Algorithm 5. Additionally, Compression and decompression strategies can be implemented to further optimise storage at the edge layer.

In this system, the data for each floor is stored separately in the local or remote edge server. When a user's query is received, the system first checks the cache memory for an exact match. If an exact match is found, the system reveals the position. Otherwise, the query is sent to the central cloud to identify the building and floor. After identifying the building and floor, the system retrieves the relevant floor data from the edge server, estimates the position using the floor storage data, sends the estimated position to the user, and saves a copy in the cache data file for future use.

Algorithm 6 LRUCache Implementation

```
procedure PUT(key, value)
    if obj.cache.isKey(key) then
        node ← obj.cache(key)
        node.value ← value
        obj.lruList.moveToHead(node)
    else
        if obj.cache.Count ≥ obj.maxSize then
            node ← obj.lruList.removeFromTail()
            obj.cache.remove(node.key)
        end if
        newNode ← obj.lruList.insertAtHead(key)
        newNode.value ← value
        obj.cache(key) ← newNode
    end if
end procedure
procedure GET(key)
    if obj.cache.isKey(key) then
        node ← obj.cache(key)
        obj.lruList.moveToHead(node)
        return node.value
    else
        return []
    end if
end procedure
procedure isKey(key)
    return obj.cache.isKey(key)
end procedure
```

Algorithm 7 Doubly Linked List

```
procedure insertAtHead(key)
    newNode ← Node(key)
    if isempty(obj.head) then
        obj.head ← newNode
        obj.tail ← newNode
    else
        newNode.next ← obj.head
        obj.head.prev ← newNode
        obj.head ← newNode
    end if
    obj.numMisses ← obj.numMisses + 1
    return newNode
end procedure
procedure removeFromTail
    node ← obj.tail
    if ¬isempty(obj.tail) then
        if obj.tail.prev ≠ [] then
            obj.tail ← obj.tail.prev
            obj.tail.next ← []
        else
            obj.head ← []
            obj.tail ← []
        end if
    end if
    return node
end procedure
procedure moveToHead(node)
    if node ≠ obj.head then
        if node == obj.tail then
            obj.tail ← node.prev
            obj.tail.next ← []
        else
            node.prev.next ← node.next
            node.next.prev ← node.prev
        end if
        node.prev ← []
    end if
end procedure
```

Algorithm 8 Node Class

procedure NODE(key)
  obj.key ← key
  obj.value ← []
  obj.prev ← []
  obj.next ← []
end procedure

6.5.7 Experimental Setup

The proposed CB-IPS was implemented and tested using MATLAB R2023a on a Lenovo laptop (Intel(R) Core(TM) i5-8265u CPU@ 1.60GHz, 1.80 GHz, 8 GB of RAM, Windows 10, 64-bit). All components of the system—including the central cloud server, edge floor nodes, and caching mechanism—were implemented as modular simulation scripts within MATLAB. Although simulated on a single device, the architecture was logically partitioned to emulate distributed cloud-edge operations, thereby maintaining architectural fidelity for performance evaluation.

The experimental setup involved the following steps:

- **Dataset Import**: UJIIndoorLoc Training and Validation datasets were imported into the simulation environment.
- **System Implementation**: The proposed CB-IPS architecture, incorporating an edge computing layer for floor-specific positioning and a central cloud server for building and floor detection, was implemented.
- **User Query Representation**: Each test scenario was represented by a user query, consisting of RSSI values at RPs, from the Validation dataset at 100% for baseline and with 30% used for caching scenarios.
- **Positioning Operation**: User queries were sent to the system to initiate the positioning operation.
- **Cache Memory Check**: Upon receiving a user query, the system first checks the cache memory for an exact match. If found, the system returned the corresponding position; otherwise, the query was forwarded to the central cloud.
- **Building and Floor Identification**: The central cloud identified the building and floor associated with the user query and communicated with the respective edge floor server to estimate the position using stored floor data.

- **Position Estimation**: A fingerprinting algorithm, such as weighted k-NN, was implemented to estimate the user's location on the identified floor.
- **Result Delivery**: The estimated position was sent to the user, and a copy was stored in the cache data file for future reference.

This experimental setup allowed a thorough evaluation of the proposed CB-IPS architecture in various scenarios, ensuring robustness and scalability in indoor positioning applications.

6.5.8 Experimental Conditions

To experiment, it is necessary to determine the floor on which the user is located so that only the related fingerprint dataset for that particular floor is run in the edge layer. This can be achieved in two ways:

- Firstly, by establishing a physical connection link to the edge layer that only checks the user inquiries on the local edge storage of that particular floor. However, this topology is beyond our scope.
- The second, to analyse the user queries, which are RSSI measurements, to determine which floor the user is on. This process needs to be done on the central cloud, which hosts the entire dataset of buildings/areas, to infer which floor the user is on and then redirect to the corresponding floor edge layer for processing the position estimation.

Simulating cloud computing will involve integrating cloud services into the proposed IPS model. This includes simulating data transmission from the user's mobile device to the cloud, cloud-based processing, and result retrieval. Synthetic data generation will not be employed; instead, existing datasets will be used to simulate users' RSSI readings and environmental conditions, which are essential for testing algorithms in realistic scenarios.

6.5.9 Evaluations and Testing Scenarios

Positioning accuracy is evaluated using the metrics described in 3.2.2. For evaluation purposes, we identified two dataset configurations and different scenarios to conduct our experiments:

1. **First**: Using complete Validation dataset at 100% without caching implementation. This is the baseline to evaluate the proposed system. The following testing scenarios were implemented:

- **Scalability Testing**: Simulates increasing numbers of concurrent user queries to measure response times and throughput under load. The system's scalability in accommodating a high demand of concurrent user queries is a good measure to test how the system is capable of delivering its services under different loads.
- **Fault Tolerance Testing**: Evaluates the system's resilience to failures in cloud components, edge components, and data storage by simulating failure scenarios and measuring recovery time and availability. For the CC component, we simulate a failure by introducing delays of 300 milliseconds as errors and measure the time taken to recover and process the queries correctly. The Edge component and data storage failures are simulated with delays of 200 milliseconds and 100 milliseconds, respectively.
- **Latency Testing**: Assesses the system's response time under different scenarios, including normal load, high load, and poor network conditions. Under normal load, queries are processed without any artificial delays; under high load, a 100 millisecond delay was introduced, and under poor network conditions, a 500 millisecond delay was introduced due to packet loss, for example.

2. **Second**: Using a copy of 30% from the validation dataset to evaluate the caching mechanism, and to evaluate the proposed system. To ensure a systematic approach to testing and validating the effectiveness of our caching mechanism, the following scenarios were applied:

    - Cold Start Scenario: Test performance when the cache is empty.
    - Repeated Query Scenario: Test repeated queries to ensure the cache hits are being leveraged.
    - Mixed Training and Testing: Ensure caching works during both phases.
    - Measure Performance: To ensure measurements for all metrics are included.
    - Analyse Results: Compare the metrics for different scenarios to the baseline.

To gauge the effectiveness of our system, evaluations were conducted across diverse scenarios and parameters to thoroughly assess the system's performance under different conditions. We meticulously analysed the experimental results, aiming to understand how our proposed system improved accuracy, scalability, and efficiency compared to traditional IPS systems.

Our evaluation procedure is robust, providing valuable information on the strengths and limitations of our proposed system. This rigorous assessment framework

Metrics	Value (m)
MAE	7.3893
RMSE	11.0641

Table 6.1: Positioning Error Metrics

facilitated informed decision-making and laid the foundation for further optimisations and advancements in indoor positioning technology.

6.6 Results and Discussion

In this chapter, our main focus is on improving the regression tasks; hence, the accuracy of the system in identifying the correct building and floor is not a concern, as presented in our proposal assumptions.

6.6.1 Dataset Configuration (1)

6.6.1.1 Positioning Accuracy

We test the positioning accuracy of the system by comparing the estimated positions with the ground truth positions using the complete validation dataset. Table 6.1 shows different evaluation metrics. The MAE of the proposed system is 7.3893, which is slightly improved compared to our previous results of 7.39.

As we can see in Figure 6.7, the 2D plot visualises the true vs. estimated positions when using the full validation dataset. Figure 6.6 shows the empirical cumulative distribution function (ECDF) of positioning errors. The plot provides a comprehensive look at the distribution of positioning errors. Key observations include:

- Approximately 70% of errors are less than 10 m.
- 90% of errors are less than 20 m.
- The maximum error is bounded at approximately 60 m.

Figure 6.8 shows a histogram representing the distribution of positioning errors for the proposed cloud-based indoor positioning system. The x-axis represents the positioning error in metres, while the y-axis represents the probability density. From the histogram, we can observe that the highest bar corresponds to positioning errors between 0–10 m, indicating that a significant portion of the estimated positions have relatively small errors within this range. The probability density gradually

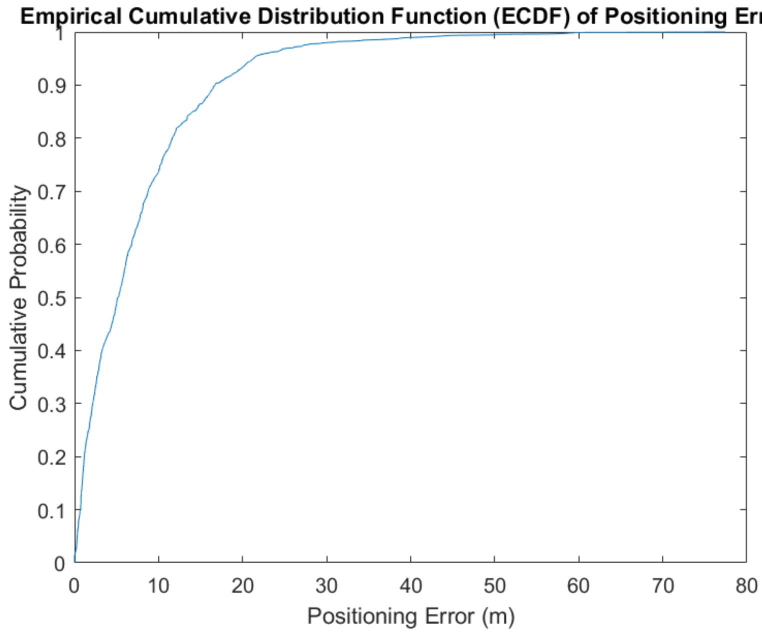


Figure 6.6: ECDF of Positioning Errors using full validation dataset

decreases as the positioning error increases, with some smaller bars representing larger positioning errors of up to approximately 50 - 80 m.

This distribution of positioning errors suggests that the system is capable of providing reasonably accurate position estimates for most cases, with a clustering of errors within the lower range. However, it also highlights the presence of some larger errors, which could be attributed to various factors such as signal interference, environmental conditions, or limitations in positioning algorithms.

The proposed CB-IPS system achieves 90% accuracy within a 20 m error range. While this level of granularity may not be sufficient for applications requiring room-level accuracy (e.g., surgical navigation or AR gaming), it is acceptable for floor-level navigation in large public buildings, rough indoor asset tracking, emergency personnel guidance, or visitor assistance systems in hospitals or universities. These use cases tolerate higher localisation uncertainty and prioritise scalable deployment and reliability.

Overall, the distribution of positioning errors provides a comprehensive view of the system's performance and serves as a starting point for further analysis, benchmarking, and optimisation efforts. By combining these results with domain knowledge and application requirements, researchers can refine the proposed system and contribute to the advancement of indoor positioning technologies.

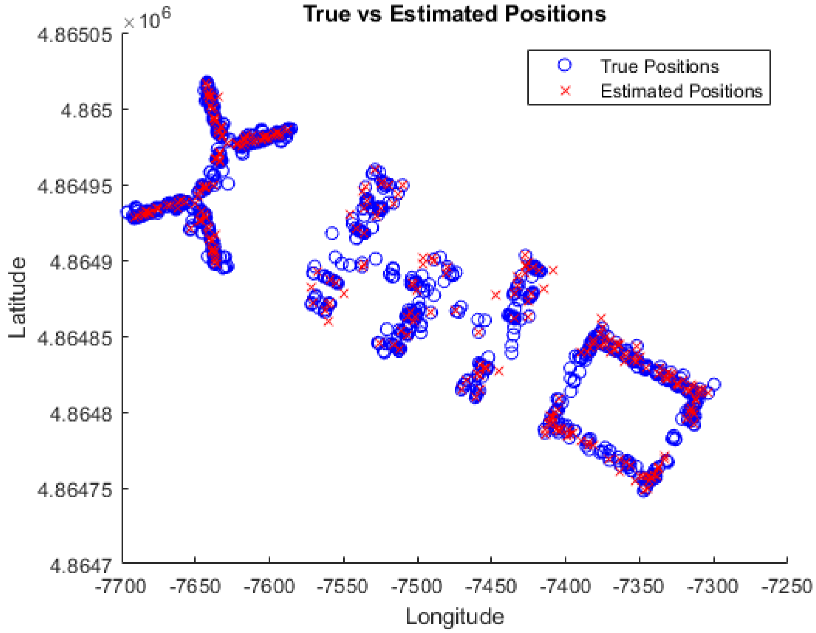


Figure 6.7: 2D Plot Visualising of True vs. Estimated Positions on Validation Dataset

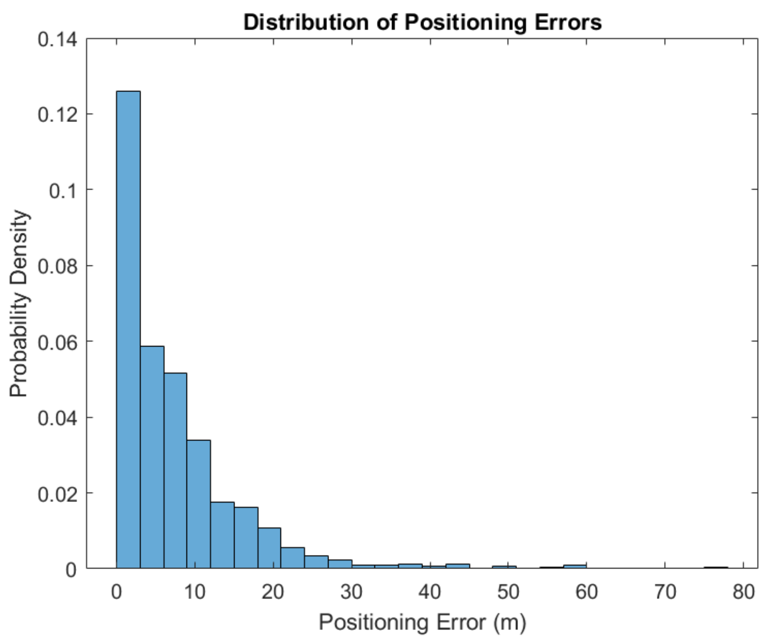


Figure 6.8: Distribution of Positioning Errors

6.6.1.2 Scalability

We evaluated the system's ability to handle an increasing number of concurrent user queries. By simulating multiple users requesting position estimates simultaneously. We obtained the following outcome. The line plot in Figure 6.9 shows the system's response times under different numbers of concurrent user queries. As shown in Figure 6.9, the system exhibits an initially high response time at very low concurrency levels, which then sharply decreases and stabilises as the number of concurrent queries increases. However, the response time appears to grow linearly, suggesting that the system can handle higher loads without experiencing exponential growth in response times.

The observed linear growth in response times is a positive indication of the system's scalability. However, it is important to note that the maximum number of concurrent queries tested is 1,000, which may not be representative of the actual load the system might encounter in real-world scenarios. Further testing with higher loads or stress tests could provide more insight into the system's scalability limits.

In addition, Figure 6.9 shows a response time anomaly, where the initial rise in response time for a small number of concurrent queries (e.g., 2–10) is attributed to system initialisation overhead and MATLAB interpreter latency. As concurrency increases, resource utilisation improves, and latency is amortised across multiple threads, reducing the average response time. Although not shown in Figure 6.9, it is anticipated that system saturation could occur at concurrency levels beyond those tested, potentially resulting in increased response times.

Furthermore, Figure 6.10 displays the system's throughput (number of queries processed per second) under different numbers of concurrent user queries. The throughput increases linearly with the number of concurrent queries, which aligns with the linear growth observed in the response times in Figure 6.9. The linear increase in throughput suggests that the system can effectively utilise additional resources (e.g., cloud computing resources) to handle higher loads. However, it is important to consider the potential trade-off between throughput and response times, as higher throughput may come at the cost of increased response times.

6.6.1.3 Fault Tolerance

We test the system's ability to handle failures or errors in the cloud component, edge component, and data storage, introducing simulated failures, and observe the system's behaviour. Figure 6.11 shows a bar chart that represents the availability of the system in a different single failure scenario. The availability is approximately 97% across the three failure scenarios, indicating that the system can recover from a cloud component failure relatively quickly and maintain a high level of availability.

Figure 6.12 displays the recovery time for the three different scenarios (cloud

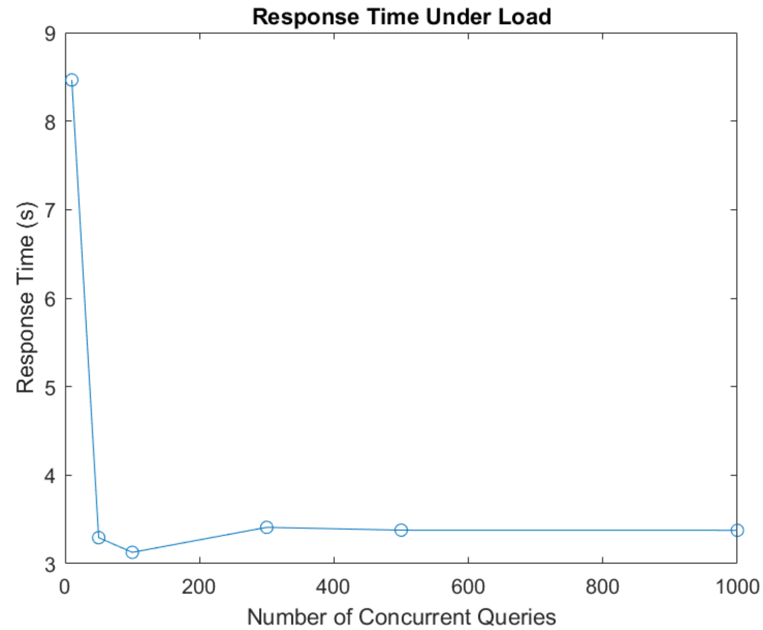


Figure 6.9: Response Time Under Load

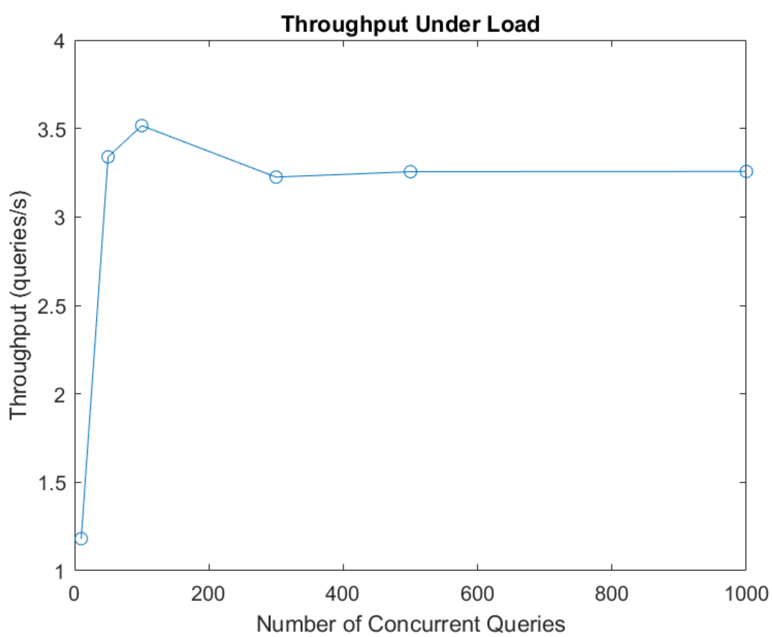


Figure 6.10: Throughput Under Load

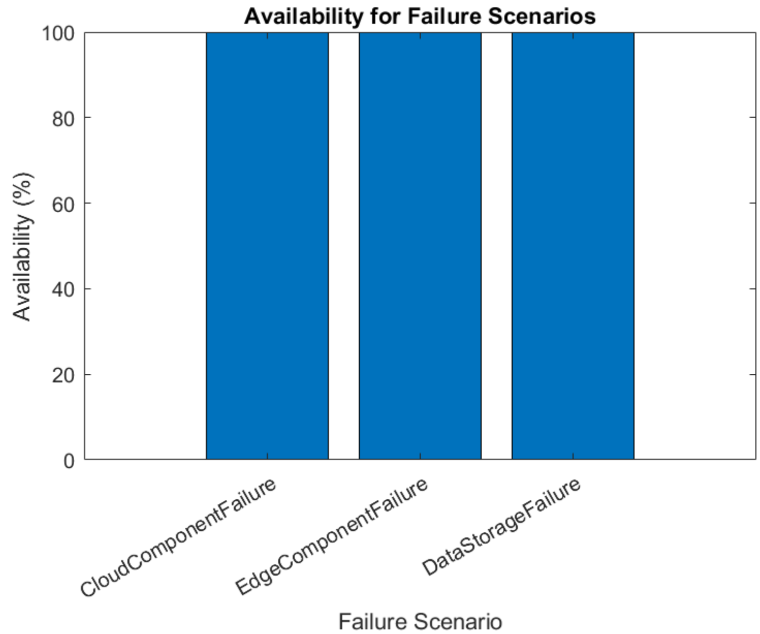


Figure 6.11: Availability for Failure Scenarios

component, edge component, and data storage). The recovery time represents the duration it takes for the system to recover from the failure and resume normal operations. The recovery time for the cloud component is around 17 seconds, making it the longest recovery duration. In contrast, the data storage component takes about 15 seconds, and the edge component has the shortest recovery time of roughly 9 seconds.

While the availability result in Figure 6.11 seems promising, the recovery time of 17 seconds for a cloud component failure may be considered relatively high for a real-time positioning system. In practical scenarios, users might require faster recovery times to ensure uninterrupted service. Potential solutions could include implementing redundancy or failover mechanisms for critical cloud components, as well as improving failure detection and recovery processes.

Overall, the results demonstrate the system's ability to handle failures, scale to manage increased loads, and maintain reasonable response times and throughput under varying conditions. However, there are areas for improvement, such as reducing recovery times for critical failures and further optimising the system's performance under extreme loads.

It should be noted that these results are based on simulations, and real-world performance may differ due to various factors, such as network conditions, hardware specifications, and the complexity of positioning algorithms. Continuous monitoring,

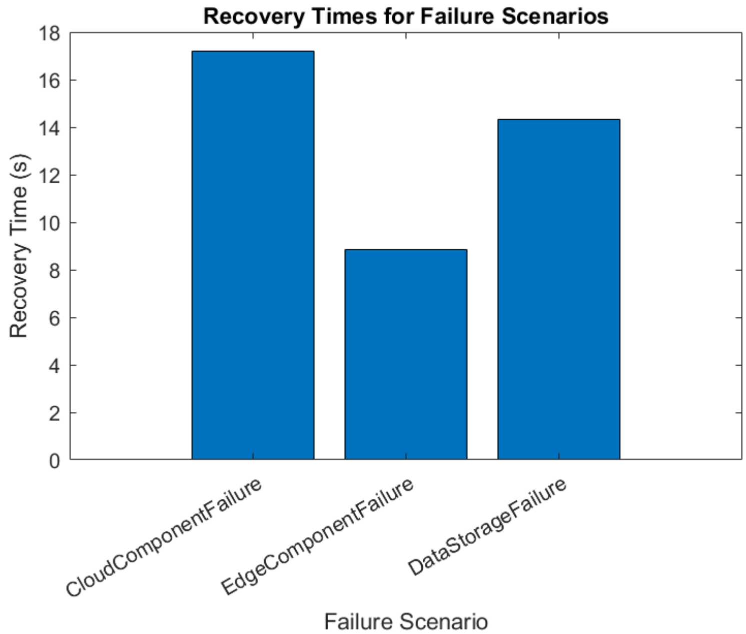


Figure 6.12: Recovery Times for Failure Scenarios

optimisation, and load testing in real-world scenarios would be necessary to ensure the system's reliability and performance in production environments.

6.6.1.4 Latency

Figure 6.13 shows the response times for position estimation in the proposed cloud-based indoor positioning system under different load and network conditions. It compares the system's latency in three scenarios:

- **Normal Load**: The system operates under typical computational load and network conditions. The response time is approximately 1.2 seconds.
- **High Load**: The system is subjected to increased computational load, simulating a scenario with many concurrent users or complex positioning calculations. Despite the high load, the response time remains relatively stable at around 1.4 seconds, showing only a slight increase compared to the normal load scenario.
- **Poor Network Condition**: This scenario tests the system's performance when network connectivity is suboptimal, such as in areas with weak signal strength or high network congestion. The response time increases to about 1.8

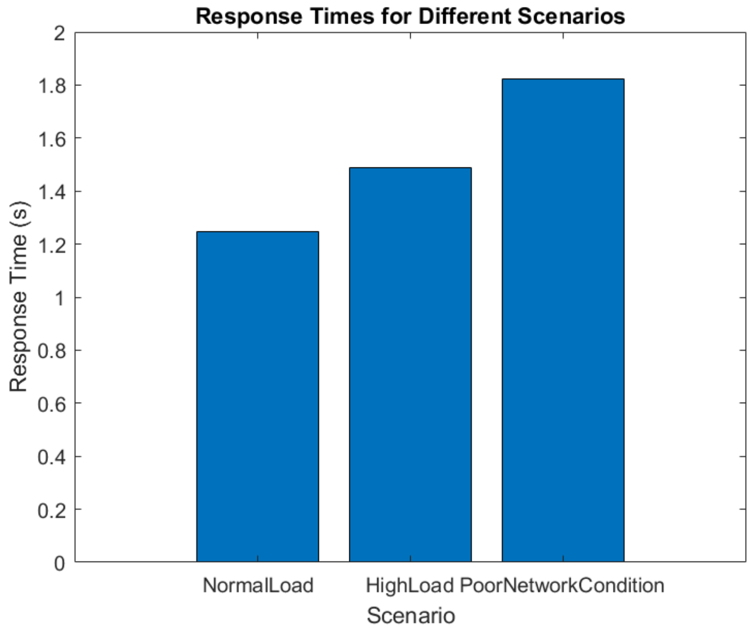


Figure 6.13: Response Times for Different Scenarios

seconds, indicating that network conditions have a more significant impact on the system's latency compared to computational load.

The results demonstrate that our CB-IPS maintains a reasonable response time of under 2 seconds across all tested scenarios. The system architecture and design seem to handle variations in computational load effectively, as evident from the minimal difference in response times between the normal and high load scenarios. However, the system's performance is more sensitive to network conditions, with poor connectivity resulting in a noticeable increase in response time. This highlights the importance of a reliable and efficient network infrastructure for the optimal functioning of our cloud-based positioning system.

Our proposed CB-IPS using edge computing improves the system's resilience against poor network conditions, where some positioning calculations are performed closer to the user devices. This helps reduce the dependency on network connectivity and potentially improves response times in challenging network environments.

6.6.2 Dataset Configuration (2)

In this test, we used a 30% copy of the validation dataset to test caching performance. When a new user query comes in, the system will first check the cache memory. If there is an exact match, it will return the position. If not, the query will be sent to the central cloud to identify the building and floor, then communicate with the respective edge-floor server to estimate the position using the stored floor data. The estimated position will then be sent to the user, and a copy will be stored in the cache data file for future use.

6.6.2.1 Caching Performance

We tested the effectiveness of the caching strategy by evaluating the cache hit rate, memory usage, and overall performance across different scenarios. The results demonstrate varying impacts on performance metrics when introducing caching into the proposed cloud-based indoor positioning system. As we can see in Figure 6.14, the baseline scenario is compared with a cold start scenario. The MAE and RMSE values are identical between the two scenarios (7.39 m for MAE and 11.06 m for RMSE), indicating that the introduction of caching does not impact the positioning accuracy when the cache is empty. While in Figure 6.15 introduces a mixed scenario in which some queries can be served from the cache, while others require processing by the central cloud and edge floor servers. Compared to the baseline, the mixed scenario shows a slight increase in RMSE (from 11.06 m to 11.14 m) and MAE (from 7.39 m to 7.42 m). This suggests a minor degradation in positioning accuracy in the mixed scenario. Figure 6.16 focuses on the impact of repeated queries. Contrary to the initial analysis, the repeated queries scenario shows a significant increase in both RMSE and MAE compared to the baseline. RMSE increased from 11.06 to 23.36 m, while MAE increased from 7.39 to 23.36 m. This substantial increase in error metrics suggests that the caching mechanism may introduce errors when handling repeated queries, possibly due to outdated or imprecise cached positions.

Figure 6.17 provides a comprehensive view of the system's performance metrics across all scenarios. The MAE values confirm the observations from the previous figures. Regarding execution time, the cold start scenario shows a slight improvement over the baseline. The mixed scenario demonstrates an increase in execution time, likely due to both training and testing samples being processed for cache testing. Unsurprisingly, the repeated queries scenario exhibits the lowest execution time, only 1.82 seconds, which is expected due to a small number of queries repeated to the system, which can easily provide instant position estimation without involving the CC and other resources, such as processing and calling the RMF. Therefore, in the repeated queries scenario, the MAE has the highest error because the positioning accuracy is calculated against 30% of the datasets that do not adequately represent the entire dataset, resulting in a high error rate.

In conclusion, while caching demonstrates potential benefits by optimising system resource usage, the observed increase in positioning errors for repeated queries and the rise in execution time in the mixed scenario highlight areas needing further

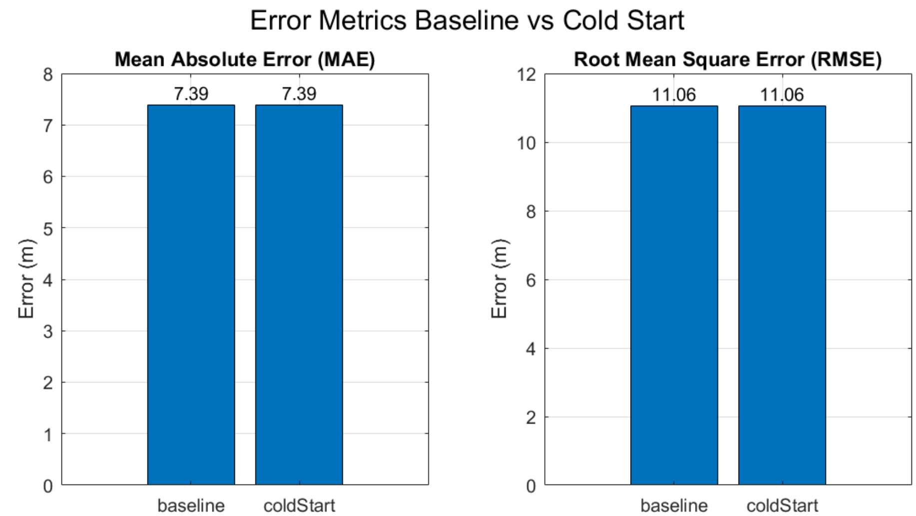


Figure 6.14: Baseline vs. Cold Start

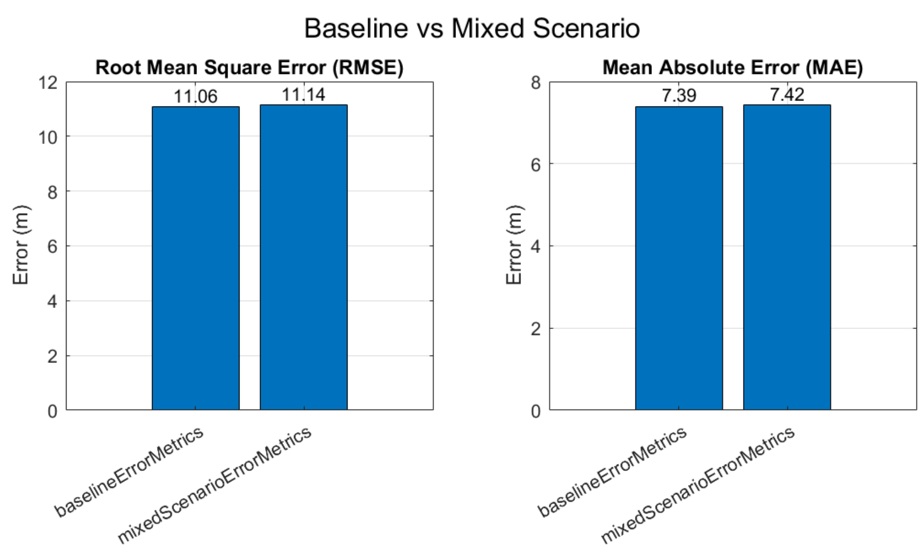


Figure 6.15: Baseline vs. Mixed Scenario

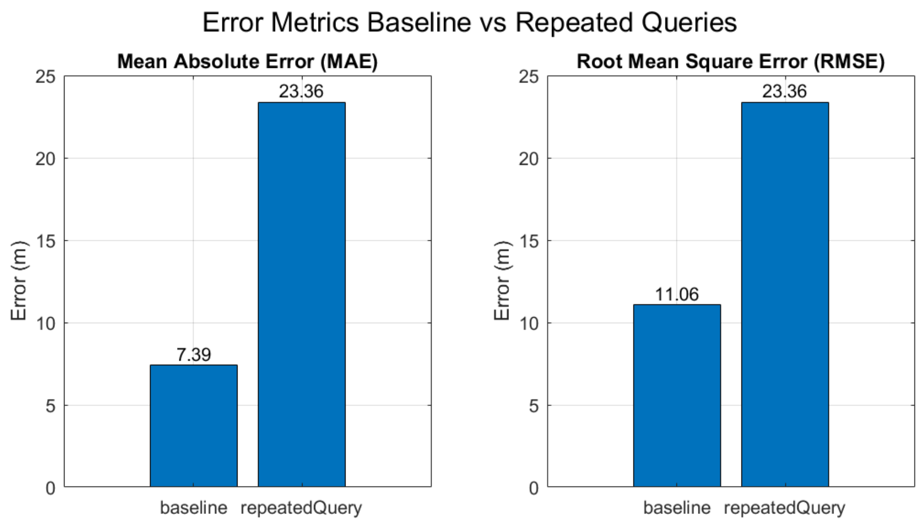


Figure 6.16: Baseline vs. Repeated Queries

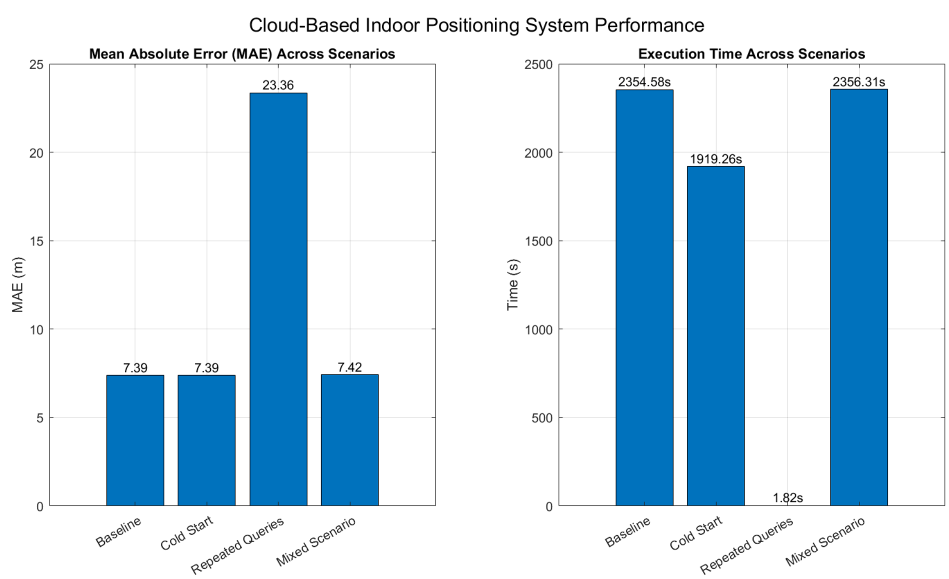


Figure 6.17: System Performance Metrics

investigation. We suggest that the caching strategy may require refinement to handle mixed scenario queries more effectively and to minimise execution time. Future work should focus on optimising the caching mechanism to address these issues and enhance overall system performance in various applications.

6.7 Summary and Conclusion

6.7.1 Summary

This chapter addressed the challenges of designing scalable and efficient Indoor Positioning Systems (IPS) by proposing and evaluating a novel **Cloud-Based Indoor Positioning System (CB-IPS) framework**. The research utilised MATLAB simulations and the UJIIndoorLoc dataset to model and assess this framework, which integrates central cloud computing with edge computing paradigms to enhance performance and manage large Wi-Fi fingerprinting radio maps. The core objective was to improve system scalability, processing efficiency, radio map storage management, and user privacy, while maintaining competitive positioning accuracy.

The proposed CB-IPS architecture features a distributed model where building and floor identification (classification tasks) are handled by a central cloud component, while fine-grained coordinate estimation (regression tasks) occurs at edge servers dedicated to specific floors. This partitioning, combined with a strategy of loading only relevant floor-specific (and adjacent-floor) RMF data at the edge, was designed to minimise computational load and improve response times. A cache mechanism was also incorporated to further optimise response times for repeated queries.

Key findings from the experimental evaluation include:

- **Positioning Accuracy:** The CB-IPS framework, leveraging the floor-specific data distribution and the optimised Wk-NN algorithm from Chapter 4, achieved a Mean Absolute Error (MAE) of 7.3893 metres on the complete UJIIndoorLoc validation dataset. This result is competitive with, and in some cases surpasses, those reported in existing literature for the same dataset reported in Table 4.6.

- **Scalability and Fault Tolerance:** The system demonstrated an ability to handle increasing numbers of concurrent user queries with linear growth in response times and throughput. It also showed resilience, maintaining high availability (approx. 97%) during simulated failures of cloud, edge, and data storage components, though recovery times for cloud failures (approx. 17 seconds) suggest areas for further optimisation.

- **Latency:** Response times for position estimation were maintained under 2 seconds across normal load, high load, and simulated poor network conditions, indicating effective handling of computational load by the architecture.

- **Caching Performance:** The LRU caching mechanism showed potential in reducing execution time for repeated queries (1.82s). However, it also led to a degradation in positioning accuracy (MAE increasing to 23.36m) when tested with a limited subset of data for repeated queries, indicating that the caching strategy requires further refinement to balance speed with accuracy, particularly concerning cache representativeness and update policies.

- **Resource Management and Privacy:** The distributed architecture inherently supports privacy by processing detailed coordinate estimation at the edge layer, with only building/floor level data managed centrally. Loading floor data on demand at edge nodes also optimises memory and storage.

These findings collectively underscore the viability of the proposed CB-IPS framework for creating more efficient, scalable, and manageable indoor positioning solutions.

6.7.2 Conclusion

This chapter makes a key technical contribution by presenting the **design, implementation (via simulation), and comprehensive evaluation of a novel Cloud-Based Indoor Positioning System (CB-IPS) architecture tailored for Wi-Fi fingerprinting**. The core novelty of this architecture lies in its hybrid approach, strategically partitioning responsibilities between central cloud resources and distributed edge computing nodes to optimise both large-scale data management and real-time positioning performance.

The research successfully demonstrated that such a CB-IPS framework can achieve competitive positioning accuracy (MAE of 7.3893m on the UJIIndoorLoc dataset) while offering significant advantages in scalability, fault tolerance, and efficient resource utilisation. The specific design choice of using the central cloud for coarse localisation (building/floor detection) and edge servers for fine-grained, floor-specific coordinate estimation (utilising optimised RMFs from Chapter 5 and algorithms from Chapter 4) proved effective in balancing computational loads and enhancing user privacy.

The integration and evaluation of a caching mechanism highlighted its potential for improving response times but also revealed challenges in maintaining accuracy with cached data, pointing to necessary future refinements. The system's robust performance under simulated load and failure scenarios further validates the architectural design choices.

Ultimately, this chapter contributes a well-defined and empirically evaluated blueprint for developing scalable and efficient cloud-enhanced Wi-Fi fingerprinting

IPS. The findings illustrate the pivotal role of edge computing in managing real-time processing demands and distributed radio map data. The proposed CB-IPS is well-suited for deployment in large-scale indoor environments like hospitals, university campuses, and shopping malls, providing a foundation for diverse location-based services and contributing to the advancement of intelligent indoor environments. The work presented herein directly addresses the research questions concerning the utilisation of cloud architectures for accurate, private, scalable, and responsive IPS.

Chapter 7

Conclusions and Future Work

7.1 Conclusion

This thesis has presented a comprehensive investigation into the efficient design and implementation of scalable Wi-Fi fingerprinting-based Indoor Positioning Systems (IPS). The research has systematically addressed key challenges in the field, spanning position estimation algorithms, Radio Map Fingerprint (RMF) optimisation, and the strategic use of cloud-based architectures. The work culminates in a robust framework and a set of empirically validated techniques that significantly enhance the accuracy, efficiency, and scalability of Wi-Fi fingerprinting IPS.

The major technical contributions and findings of this research are summarised as follows:

Firstly, a rigorous optimisation of k-Nearest Neighbour (k-NN) and Weighted k-NN (Wk-NN) position estimation algorithms was conducted (Chapter 4). This empirical study on the UJIIndoorLoc dataset identified that the Correlation distance function, when combined with an exponential data representation and an optimised k-value (e.g., k=26 for Wk-NN with inverse distance weighting), yields superior performance. Specifically, the optimally tuned Wk-NN algorithm achieved a Mean Absolute Error (MAE) of 7.39 metres, providing a highly efficient and relatively low-complexity solution for accurate position estimation. This work also established a valuable performance benchmark for these foundational algorithms on a standard dataset.

Secondly, this research developed and validated a multi-faceted RMF optimisation framework (Chapter 5) to tackle critical issues of signal heterogeneity, data dimensionality, and RMF maintenance. Key achievements include:

- Effective mitigation of device-induced RSSI heterogeneity through data preprocessing techniques, significantly reducing initial positioning errors.

- Successful application of Principal Component Analysis (PCA) for dimensionality reduction, which was shown to approximately halve computational time for positioning algorithms with minimal degradation in accuracy (e.g., maintaining an MAE around 7.49 metres while using significantly fewer features).
- Proposal and initial validation of an auto-update mechanism for the RMF, demonstrating the potential for adaptive learning from user queries to improve accuracy over time and reduce manual recalibration efforts.

These strategies collectively enhance the practicality and reduce the operational overhead of maintaining Wi-Fi fingerprint databases.

Thirdly, the thesis introduced and evaluated a novel Cloud-Based Indoor Positioning System (CB-IPS) architecture (Chapter 6) that strategically leverages cloud computing, edge computing, and cache technologies. This framework offers a scalable and reliable solution by:

- Partitioning tasks: Utilising the central cloud for coarse localisation (building and floor identification, achieving up to 100% building and 96.3% floor hit rates) and edge servers for fine-grained, floor-specific coordinate estimation (achieving an MAE of 7.3893 metres).
- Enhancing resource management through distributed RMF storage and on-demand data loading at the edge.
- Addressing privacy concerns by processing sensitive location data primarily at the edge layer.
- Demonstrating robustness through fault tolerance and latency tests, maintaining response times under 2 seconds even under varying load conditions.

The CB-IPS framework provides a blueprint for developing efficient and resilient large-scale indoor positioning services.

This research was initiated to address several critical questions concerning the design and implementation of efficient and scalable Wi-Fi fingerprinting-based IPS. The work presented provides substantial insights into these questions:

1. *What is the most efficient and yet least complex position estimation algorithm suitable for the IPS based on Wi-Fi Fingerprinting techniques?* This thesis demonstrated (Chapter 4) that an optimally tuned Weighted k-Nearest Neighbour (Wk-NN) algorithm, specifically employing the Correlation distance function with exponential data representation and inverse distance weighting, provides an excellent balance of high accuracy (MAE of 7.39m on UJIIndoorLoc) and computational efficiency, outperforming baseline k-NN without resorting to overly complex models.

2. *How can a Wi-Fi Fingerprinting-based IPS achieve scalable and reliable performance, making it simple, efficient, and competitive with state-of-the-art systems while ensuring acceptable positioning services?* Scalability and reliability were addressed through the RMF optimisation techniques (Chapter 5), such as PCA for reduced data handling, and the CB-IPS architecture (Chapter 6). The CB-IPS framework, with its distributed cloud-edge design, efficient data management, and caching, demonstrates a clear pathway to achieving scalable, reliable, and efficient performance competitive with existing systems, delivering acceptable positioning accuracy for many indoor applications.

3. *How can cloud architectures be utilised to maintain the required accuracy, privacy, and response time while providing scalability to the IPS?* Chapter 6 detailed how a hybrid cloud-edge architecture can be strategically employed. The central cloud manages overall coordination and coarse localisation, while edge nodes handle computationally intensive, latency-sensitive fine localisation for specific floors. This distribution maintains accuracy (MAE 7.3893m), enhances privacy by processing detailed location data locally at the edge, ensures low response times via edge processing and caching, and provides inherent scalability through the cloud model.

4. *What approaches can be employed to enhance the positioning accuracy and scalability of Wi-Fi RSSI-based systems, with a focus on simplifying database fingerprinting complexity using an edge-computing architecture?* This research proposed several approaches: RMF optimisation techniques (Chapter 5) such as data preprocessing to handle heterogeneity and PCA to reduce database dimensionality, significantly simplifying complexity. The auto-update mechanism further aids in managing RMF evolution. The edge computing aspect of the CB-IPS architecture (Chapter 6) directly simplifies database handling by partitioning the RMF and processing data locally, enhancing both accuracy (by using focused data) and scalability.

In summary, this thesis has successfully demonstrated that through systematic algorithm optimisation, intelligent radio map management, and a well-designed cloud-edge architecture, Wi-Fi fingerprinting can be a highly effective, scalable, and efficient technology for a wide range of indoor positioning applications. The findings emphasise that an integrated approach, considering all aspects of the IPS from signal processing to system architecture, is crucial for advancing the field and overcoming its inherent challenges.

7.2 Future Work

While this thesis has made significant contributions to the efficient design of scalable Wi-Fi fingerprinting IPS, several avenues for future research and development emerge from this work, aiming to further enhance performance, adaptability, and practical applicability:

- **Advanced Machine Learning Integration:** Explore the integration of more sophisticated machine learning models, particularly deep learning architectures (e.g., Convolutional Neural Networks, Recurrent Neural Networks, Transformers), for both fingerprint feature extraction and position estimation. These models may offer improved handling of complex signal patterns and environmental dynamics, potentially leading to higher accuracy, especially in challenging scenarios.

- **Dynamic RMF Maintenance and Self-Adaptation:** Further develop the auto-update RMF mechanisms (explored in Chapter 5) into fully autonomous, self-adapting systems. This could involve incorporating unsupervised or semi-supervised learning for detecting environmental changes and updating the radio map without requiring explicit user feedback, leveraging techniques like active learning or reinforcement learning to intelligently query for updates when uncertainty is high.

- **Robustness to Environmental and Device Dynamics:** Investigate advanced techniques to improve system robustness against severe environmental changes and greater device heterogeneity. This includes research into domain adaptation methods and transfer learning to enable radio maps created with one set of devices or in one state of the environment to be effectively used with others or as the environment evolves.

- **Optimisation of Caching Strategies for CB-IPS:** The caching mechanism introduced in Chapter 6 showed promise but also areas for improvement regarding accuracy. Future work should focus on more intelligent caching strategies, such as predictive caching based on user movement patterns, adaptive cache eviction policies, and ensuring the freshness and representativeness of cached RMF data segments.

- **Real-World Deployment and Large-Scale Trials:** While simulations on datasets like UJIIndoorLoc provide valuable insights, deploying the proposed CB-IPS framework in diverse, large-scale real-world environments (e.g., entire university campuses, large hospitals, multi-story shopping complexes) is crucial. Such trials would validate performance under operational conditions and reveal practical implementation challenges related to real-time data ingestion, network latency, and system administration.

- **Enhanced Sensor Fusion:** Explore tighter integration of Wi-Fi fingerprinting with other available sensors on mobile devices (e.g., IMUs for Pedestrian Dead Reckoning, magnetometers, barometers for floor detection) within the proposed cloud-edge framework. Fusion at different levels (data, feature, or decision) could enhance accuracy, provide smoother tracking, and improve robustness in areas with sparse Wi-Fi coverage.

- **Energy Efficiency for Mobile Devices:** Investigate methods to reduce the energy consumption of the positioning process on mobile devices, particularly in the context of continuous tracking within the CB-IPS framework. This could involve optimising the frequency of Wi-Fi scans and data transmission to the edge/cloud.

- **Security and Privacy in CB-IPS:** While the proposed CB-IPS architecture considers privacy through data partitioning, further research into advanced privacy-preserving techniques (e.g., federated learning for RMF updates, differential privacy for query submissions) specifically for cloud-based positioning services is warranted.

Addressing these future research directions will be essential for pushing the boundaries of Wi-Fi fingerprinting technology, leading to even more precise, reliable, adaptable, and user-friendly indoor positioning systems capable of meeting the evolving demands of modern smart environments.

References

[1] E. Ebaid and K. Navaie, “Optimum NN Algorithms Parameters on the UJI-IndoorLoc for Wi-Fi Fingerprinting Indoor Positioning Systems,” Wellington, New Zealand: IEEE, Dec. 2022, pp. 280–286, ISBN: 978-1-6654-7104-6. DOI: 10.1109/ITNAC55475.2022.9998385.

[2] E. Ebaid and K. Navaie, “Efficient Design of Scalable Indoor Positioning System Based on Wi-Fi Fingerprinting,” in *Proceedings of the CPS Summer School PhD Workshop 2022 co-located with 4th Edition of the CPS Summer School (CPS 2022), Pula, Sardinia (Italy), September 19-23, 2022*, R. L. López, D. M. Quintín, F. Palumbo, C. Pilato, and A. Tacchella, Eds., ser. CEUR Workshop Proceedings, vol. 3252, CEUR-WS.org, Sep. 2022. [Online]. Available: https://ceur-ws.org/Vol-3252/short1.pdf.

[3] O. U. g. i. a. t. G. P. S. (GPS), *Gps accuracy*, 2020. [Online]. Available: https://www.gps.gov/systems/gps/performance/accuracy/#:~:text=As%20with%20positioning%2C%20the%20speed,interval%2C%20with%2095%25%20probability..

[4] L. Qi, Q. Jiahui, and C. Yi, “Research and development of indoor positioning,” *China Communications*, vol. 13, no. Supplement2, pp. 67–79, 2016, ISSN: 1673-5447. DOI: 10.1109/CC.2016.7833461.

[5] R. F. Brena, J. P. García-Vázquez, C. E. Galván-Tejada, D. Muñoz-Rodriguez, C. Vargas-Rosales, and J. Fangmeyer, “Evolution of indoor positioning technologies: A survey,” *Journal of sensors*, vol. 2017, pp. 1–21, 2017, ISSN: 1687-7268 1687-725X. DOI: 10.1155/2017/2630413.

[6] W. Sakpere, M. A. Oshin, and N. B. Mlitwa, “A state-of-the-art survey of indoor positioning and navigation systems and technologies,” *South African Computer Journal*, vol. 29, Not available, Dec. 2017, ISSN: 2313-7835, 1015-7999. DOI: 10.18489/sacj.v29i3.452.

[7] J. Kárník and J. Streit, “Summary of available indoor location techniques,” *IFAC PapersOnLine*, vol. 49, no. 25, pp. 311–317, 2016, ISSN: 2405-8963. DOI: 10.1016/j.ifacol.2016.12.055.

[8] S. Xia, Y. Liu, G. Yuan, M. Zhu, and Z. Wang, "Indoor fingerprint positioning based on wi-fi: An overview," *ISPRS international journal of geo-information*, vol. 6, no. 5, p. 135, 2017, ISSN: 2220-9964. DOI: 10.3390/ijgi6050135.

[9] D. Giovanelli, E. Farella, D. Fontanelli, and D. Macii, "Bluetooth-based indoor positioning through tof and rssi data fusion," Nantes, France: IEEE, Sep. 2018, pp. 1–8, ISBN: 978-1-5386-5636-5. DOI: 10.1109/IPIN.2018.8533853.

[10] P. Dabove, V. D. Pietra, M. Piras, A. A. Jabbar, and S. A. Kazim, "Indoor positioning using ultra-wide band (uwb) technologies: Positioning accuracies and sensors' performances," *2018 IEEE/ION Position, Location and Navigation Symposium (PLANS)*, vol. Not available, Not available, Apr. 2018, ISSN: Not available. DOI: 10.1109/plans.2018.8373379.

[11] J. Luo, L. Fan, and H. Li, "Indoor positioning systems based on visible light communication: State of the art," *IEEE Communications surveys and tutorials*, vol. 19, no. 4, pp. 2871–2893, 2017, ISSN: 1553-877X. DOI: 10.1109/comst.2017.2743228.

[12] M. Merenda, L. Catarinucci, R. Colella, F. G. D. Corte, and R. Carotenuto, "Exploiting rfid technology for indoor positioning," Bol and Split, Croatia: IEEE, Sep. 2021, pp. 1–5, ISBN: 978-1-6654-4202-2. DOI: 10.23919/SpliTech52315.2021.9566443.

[13] M. Uradzinski, H. Guo, X. Liu, and M. Yu, "Advanced indoor positioning using zigbee wireless technology," *Wireless personal communications*, vol. 97, no. 4, pp. 6509–6518, 2017, ISSN: 1572-834X 0929-6212. DOI: 10.1007/s11277-017-4852-5.

[14] A. F. G. Ferreira, D. M. A. Fernandes, A. P. Catarino, and J. L. Monteiro, "Localization and positioning systems for emergency responders: A survey," *IEEE Communications Surveys & Tutorials*, vol. 19, pp. 2836–2870, May 2017, ISSN: 1553-877X. DOI: 10.1109/comst.2017.2703620.

[15] J. Torres-Sospedra, R. Montoliu, Martinez-Uso, T. J. Arnau, and Benedito-Bordonau, "Ujiindoorloc: A new multi-building and multi-floor database for wlan fingerprint-based indoor localization problems," in *2014 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, Oct. 2014. DOI: 10.1109/ipin.2014.7275492.

[16] R. Montoliu, E. Sansano, J. Torres-Sospedra, and O. Belmonte, "Indoorloc platform: A public repository for comparing and evaluating indoor positioning systems," *2017 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, vol. Not available, Not available, Sep. 2017, ISSN: Not available. DOI: 10.1109/ipin.2017.8115940.

[17] T. K. Geok, K. Z. Aung, M. S. Aung, *et al.*, "Review of indoor positioning: Radio wave technology," *Applied Sciences*, vol. 11, p. 279, Dec. 2020, ISSN: 2076-3417. DOI: `10.3390/app11010279`.

[18] M.-S. Germán Martín, M. M.-S. Germán, T.-S. Joaquín, and H. Joaquín, "A meta-review of indoor positioning systems," *Sensors*, vol. 19, no. 20, p. 4507, 2019, 31627331 2981223452, ISSN: 1424-8220. DOI: `10.3390/s19204507`.

[19] E. G. Zimbelman and R. F. Keefe, "Real-time positioning in logging: Effects of forest stand characteristics, topography, and line-of-sight obstructions on gnss-rf transponder accuracy and radio signal propagation," *PLOS ONE*, vol. 13, e0191017, Jan. 2018, ISSN: 1932-6203. DOI: `10.1371/journal.pone.0191017`.

[20] S. Santosh and P. Jae-Young, "A survey of smartphone-based indoor positioning system using rf-based wireless technologies," *Sensors*, Dec. 2020, 33348701 3111367006. DOI: `10.3390/s20247230`.

[21] G. Mendoza-Silva, P. Richter, J. Torres-Sospedra, E. Lohan, and J. Huerta, "Long-term wifi fingerprinting dataset for research on robust indoor positioning," *Data*, vol. 3, no. 1, p. 3, 2018, ISSN: 2306-5729. DOI: `10.3390/data3010003`.

[22] X. Song, X. Fan, C. Xiang, *et al.*, "A novel convolutional neural network based indoor localization framework with wifi fingerprinting," *IEEE Access*, vol. 7, pp. 110 698–110 709, 2019. DOI: `10.1109/ACCESS.2019.2933921`.

[23] H. Minh Tu, Z. Yizhou, Y. Brosnan, *et al.*, "A soft range limited k-nearest neighbors algorithm for indoor localization enhancement," *IEEE Sensors Journal*, 2018. DOI: `10.1109/jsen.2018.2874453`.

[24] G. Caso, L. D. Nardis, F. Lemic, V. Handziski, A. Wolisz, and M.-G. D. Benedetto, "Vifi: Virtual fingerprinting wifi-based indoor positioning via multi-wall multi-floor propagation model," *IEEE Transactions on Mobile Computing*, vol. 19, pp. 1478–1491, Jun. 2020, ISSN: 1536-1233, 1558-0660, 2161-9875. DOI: `10.1109/tmc.2019.2908865`.

[25] M. Md Abdulla Al, M. Md. Abdulla Al, and Y. Mehmet Rasit, "Map-aided fusion of imu pdr and rssi fingerprinting for improved indoor positioning," *Italian National Conference on Sensors*, 2021, 4200432850. DOI: `10.1109/sensors47087.2021.9639778`.

[26] X. Tao, T. Shi, X. Ma, H. Zhang, and Z. Pei, "An improved indoor pedestrian dead reckoning algorithm using ambient light and sensors," *Multimedia Tools and Applications*, vol. 81, pp. 32 581–32 592, Apr. 2022, ISSN: 1380-7501, 1573-7721. DOI: `10.1007/s11042-022-13072-y`.

[27] T. D. Vy, T. L. N. Nguyen, and Y. Shin, "A precise tracking algorithm using pdr and wi-fi/ibeacon corrections for smartphones," *IEEE Access*, vol. 9, pp. 49 522–49 536, 2021, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2021.3069261.

[28] X. Li, J. Wang, and C. Liu, "A bluetooth/pdr integration algorithm for an indoor positioning system," *Sensors*, vol. 15, pp. 24 862–24 885, Sep. 2015, ISSN: 1424-8220. DOI: 10.3390/s151024862.

[29] Y. Zhu, X. Luo, S. Guan, and Z. Wang, "Indoor positioning method based on wifi/bluetooth and PDR fusion positioning," in *13th International Conference on Advanced Computational Intelligence, ICACI 2021, Wanzhou, China, May 14-16, 2021*, IEEE, 2021, pp. 233–238. DOI: 10.1109/ICACI52617.2021.9435887.

[30] S. Mstislav, F. Grigoriy, D. Pavel, *et al.*, "Wi-fi based indoor positioning system using inertial measurements," *Next Generation Teletraffic and Wired/Wireless Advanced Networking*, 2017, 2754522657. DOI: 10.1007/978-3-319-67380-6_69.

[31] N. Yu, X. Ma, X. Chen, R. Feng, and Y. Wu, "High-precision indoor positioning method based on multifeature fusion of inertial sensor network," *IEEE Transactions on Instrumentation and Measurement*, vol. 73, pp. 1–16, 2024, ISSN: 1557-9662. DOI: 10.1109/TIM.2023.3345905.

[32] N. El-Sheimy and Y. Li, "Indoor navigation: State of the art and future trends," *Satellite Navigation*, vol. 2, Not available, May 2021, ISSN: 2662-1363. DOI: 10.1186/s43020-021-00041-3.

[33] Z. Yang, C. Wu, Z. Zhou, X. Zhang, X. Wang, and Y. Liu, "Mobility increases localizability: A survey on wireless indoor localization using inertial sensors," *ACM Computing Surveys (CSUR)*, vol. 47, no. 3, pp. 1–34, 2015, ISSN: 0360-0300. DOI: 10.1145/2676430.

[34] G. Ouyang, K. Abed-Meraim, and Z. Ouyang, "Magnetic-field-based indoor positioning using temporal convolutional networks," *Sensors*, vol. 23, p. 1514, Jan. 2023, ISSN: 1424-8220. DOI: 10.3390/s23031514.

[35] R. Montoliu, J. Torres-Sospedra, and O. Belmonte, "Magnetic field based indoor positioning using the bag of words paradigm," Alcala de Henares, Spain: IEEE, Oct. 2016, pp. 1–7, ISBN: 978-1-5090-2426-1. DOI: 10.1109/IPIN.2016.7743596.

[36] B. Li, T. Gallagher, A. G. Dempster, and C. Rizos, "How feasible is the use of magnetic field alone for indoor positioning?" *2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, vol. Not available, Not available, Nov. 2012. DOI: 10.1109/ipin.2012.6418880.

[37] V. Pasku, A. D. Angelis, M. Dionigi, G. D. Angelis, A. Moschitta, and P. Carbone, "A positioning system based on low-frequency magnetic fields," *IEEE Transactions on Industrial Electronics*, vol. 63, pp. 2457–2468, Apr. 2016, ISSN: 0278-0046, 1557-9948. DOI: `10.1109/tie.2015.2499251`.

[38] J. Zhu, X. Liu, Q. Shi, *et al.*, "Development trends and perspectives of future sensors and mems/nems," *Micromachines*, vol. 11, p. 7, Dec. 2019, ISSN: 2072-666X. DOI: `10.3390/mi11010007`.

[39] S. Holm and C.-I. C. Nilsen, "Robust ultrasonic indoor positioning using transmitter arrays," Zurich, Switzerland: IEEE, Sep. 2010, pp. 1–5, ISBN: 978-1-4244-5862-2. DOI: `10.1109/IPIN.2010.5646198`.

[40] G. Feferman, M. Blatt, and A. Eilam, "Indoor positioning with unsynchronized sound sources," Eilat, Israel: IEEE, Dec. 2018, pp. 1–4, ISBN: 978-1-5386-6379-0. DOI: `10.1109/ICSEE.2018.8646313`.

[41] Q. Shi, Z. Sun, X. Le, J. Xie, and C. Lee, "Soft robotic perception system with ultrasonic auto-positioning and multimodal sensory intelligence," *ACS Nano*, vol. 17, pp. 4985–4998, Mar. 2023, ISSN: 1936-0851, 1936-086X. DOI: `10.1021/acsnano.2c12592`.

[42] R. Ma, W. Kong, T. Chen, R. Shu, and G. Huang, "KNN based denoising algorithm for photon-counting lidar: Numerical simulation and parameter optimization design," *Remote. Sens.*, vol. 14, no. 24, p. 6236, Dec. 2022. DOI: `10.3390/RS14246236`.

[43] Z. Liu, L. Zhang, Q. Liu, Y. Yin, L. Cheng, and R. Zimmermann, "Fusion of magnetic and visual sensors for indoor localization: Infrastructure-free and more effective," *IEEE Transactions on Multimedia*, vol. 19, pp. 874–888, Apr. 2017, ISSN: 1520-9210, 1941-0077. DOI: `10.1109/tmm.2016.2636750`.

[44] X. Zhang, F. He, Q. Chen, *et al.*, "A differentially private indoor localization scheme with fusion of wifi and bluetooth fingerprints in edge computing," *Neural Computing and Applications*, vol. 34, no. 6, pp. 4111–4132, 2022, ISSN: 0941-0643. DOI: `10.1007/s00521-021-06815-9`.

[45] G. Xiansheng, A. Nirwan, H. Fangzi, S. Yuan, E. Nkrow Raphael, and L. Lin, "A survey on fusion-based indoor positioning," *IEEE Communications Surveys and Tutorials*, 2020, 2985031189. DOI: `10.1109/comst.2019.2951036`.

[46] A. Yuri, A. B. F. O. Horácio, A. B. F. d. O. Horacio, *et al.*, "Indoor positioning system using synthetic training and data fusion," *IEEE Access*, 2021, 3194106925. DOI: `10.1109/access.2021.3105188`.

[47] J. Liu, J. Luo, J. Hou, D. Wen, G. Feng, and X. Zhang, "A BIM based hybrid 3d indoor map model for indoor positioning and navigation," *ISPRS Int. J. Geo Inf.*, vol. 9, no. 12, p. 747, 2020. DOI: 10.3390/IJGI9120747.

[48] J. Suhardi Azliy, S. Santosh, and P. Jae-Young, "Floor map-aware particle filtering based indoor navigation system," *IEEE Access*, 2021, 3194690436. DOI: 10.1109/access.2021.3102992.

[49] A. Basiri, E. S. Lohan, T. Moore, *et al.*, "Indoor location based services challenges, requirements and usability of current solutions," *Computer Science Review*, vol. 24, pp. 1–12, 2017, ISSN: 1574-0137. DOI: 10.1016/j.cosrev.2017.03.002.

[50] H. Yunus, A. Gülsüm Zeynep Gürkaş, and T. Zeynep, "Hybrid indoor positioning for smart homes using wifi and bluetooth low energy technologies," *Journal of Ambient Intelligence and Smart Environments*, 2023, 4324138409. DOI: 10.3233/ais-220484.

[51] Y. Yue, C. Ruizhi, C. Liang, L. Wei, W. Yuan, and Z. Haitao, "H-wps: Hybrid wireless positioning system using an enhanced wi-fi ftm/rssi/mems sensors integration approach," *IEEE Internet of Things Journal*, 2021, 4205834222. DOI: 10.1109/jiot.2021.3132023.

[52] S. Santosh, H. Suk-Seung, and P. Jae-Young, "Hybrid wireless indoor positioning system combining ble beacons and wi-fi aps," *Information and Communication Technology Convergence*, Dec. 2020, 3116179002. DOI: 10.1109/ictc49870.2020.9289235.

[53] S. Leonardo, R. Ohara Kerusauskas, and L. Paulo, "Hybrid indoor localization system combining multilateration and fingerprinting," 2022, 4310972266. DOI: 10.1109/iecon49645.2022.9968816.

[54] Y.-W. Ma, J.-L. Chen, Y.-T. Chen, C.-C. Hsu, and C.-H. Chuang, "Hybrid vlc-based indoor positioning system for future productivity 4.0," Toyama, Japan: IEEE, Jul. 2019, pp. 158–161, ISBN: 978-1-7281-1652-5. DOI: 10.1109/SNPD.2019.8935876.

[55] A. N. N. Hisham, Y. H. Ng, C. K. Tan, and D. Chieng, "Hybrid wi-fi and ble fingerprinting dataset for multi-floor indoor environments with different layouts," *Data*, vol. 7, no. 11, p. 156, 2022. DOI: 10.3390/DATA7110156.

[56] Z. Zhang, Y. Yu, L. Chen, and R. Chen, "Hybrid indoor positioning system based on acoustic ranging and wi-fi fingerprinting under nlos environments," *Remote. Sens.*, vol. 15, no. 14, p. 3520, 2023. DOI: 10.3390/RS15143520.

[57] R. Mautz, "Indoor positioning technologies," 2012.

[58] F. Zafari, A. Gkelias, and K. K. Leung, "A survey of indoor localization systems and technologies," *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2568–2599, 2019. DOI: 10.1109/COMST.2019.2911558.

[59] N. Pirzada, M. Y. Nayan, F. Subhan, M. F. Hassan, and M. A. Khan, "Comparative analysis of active and passive indoor localization systems," *AASRI Procedia*, vol. 5, pp. 92–97, Nov. 2013, ISSN: 2212-6716. DOI: 10.1016/j.aasri.2013.10.063.

[60] D. Konings, F. Alam, F. Noble, and E. M. K. Lai, "Springloc: A device-free localization technique for indoor positioning and tracking using adaptive rssi spring relaxation," *IEEE access*, vol. 7, pp. 56 960–56 973, 2019, ISSN: 2169-3536. DOI: 10.1109/access.2019.2913910.

[61] M. Chen, K. Liu, J. Ma, *et al.*, "Moloc: Unsupervised fingerprint roaming for device-free indoor localization in a mobile ship environment," *IEEE Internet of Things Journal*, vol. 7, no. 12, pp. 11 851–11 862, 2020, ISSN: 2327-4662. DOI: 10.1109/jiot.2020.3004240.

[62] K. M. Chen, R. Y. Chang, and S. Liu, "Interpreting Convolutional Neural Networks for Device-Free Wi-Fi Fingerprinting Indoor Localization via Information Visualization," *IEEE Access*, vol. 7, pp. 172 156–172 166, 2019. DOI: 10.1109/ACCESS.2019.2956187.

[63] D. Shubham, S. Khedkar, A. Shaikh, and N. Anisaara, "Wi fi based indoor positioning system," 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:212582244.

[64] Y. Wang, M. Li, and M. Li, "The statistical analysis of ieee 802.11 wireless local area network–based received signal strength indicator in indoor location sensing systems," *International Journal of Distributed Sensor Networks*, vol. 13, p. 155 014 771 774 785, Dec. 2017, ISSN: 1550-1477, 1550-1477. DOI: 10.1177/1550147717747858.

[65] F. Xanar, D. R, R. Mendes, E. Mendes, and J. Basil, "Indoor wi-fi based positioning system using signal fingerprinting," 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:21648035.

[66] M. U. Ali, S. Hur, and Y. Park, "Wi-fi-based effortless indoor positioning system using iot sensors," *Sensors (Basel)*, vol. 19, no. 7, p. 1496, 2019, ISSN: 1424-8220. DOI: 10.3390/s19071496.

[67] F. Liu, J. Liu, Y. Yin, *et al.*, "Survey on wifi-based indoor positioning techniques," *IET Communications*, vol. 14, pp. 1372–1383, Jun. 2020, ISSN: 1751-8636, 1751-8636. DOI: 10.1049/iet-com.2019.1059.

[68] E. Lohan, J. Torres-Sospedra, H. Leppäkoski, P. Richter, Z. Peng, and J. Huerta, "Wi-fi crowdsourced fingerprinting dataset for indoor positioning," *Data*, vol. 2, p. 32, Oct. 2017, ISSN: 2306-5729. DOI: 10.3390/data2040032.

[69] S. Navneet, C. Sangho, and P. Rajiv, "Machine learning based indoor localization using wi-fi rssi fingerprints: An overview," *IEEE Access*, vol. 9, pp. 127 150–127 174, 2021, ISSN: 2169-3536. DOI: 10.1109/access.2021.3111083.

[70] P. Chen, J. Shang, and F. Gu, "Learning rssi feature via ranking model for wi-fi fingerprinting localization," *IEEE Trans. Veh. Technol.*, vol. 69, no. 2, pp. 1695–1705, 2020. DOI: 10.1109/TVT.2019.2959308.

[71] K. Loizos, K. Akis, L. Antonio, and S. Stavros, "Fusing bluetooth beacon data with wi-fi radiomaps for improved indoor localization," *Sensors*, 2017, 28394268 2606067159. DOI: 10.3390/s17040812.

[72] F. Zhang, "Fusion positioning algorithm of indoor wifi and bluetooth based on discrete mathematical model," *Journal of Ambient Intelligence and Humanized Computing*, vol. Not available, Not available, Nov. 2020, ISSN: 1868-5137, 1868-5145. DOI: 10.1007/s12652-020-02635-2.

[73] B. Karamat and S.-M. Hamed, "An intelligent indoor positioning algorithm based on wi-fi and bluetooth low energy," *arXiv.org*, 2023. DOI: 10.48550/arxiv.2312.15744.

[74] S. Yu, S. Jan, and D. S. D. Lorenzo, "Indoor navigation using Wi-Fi fingerprinting combined with pedestrian dead reckoning," in *IEEE/ION Position, Location and Navigation Symposium, PLANS 2018, Monterey, CA, USA, April 23-26, 2018*, IEEE, 2018, pp. 246–253. DOI: 10.1109/PLANS.2018.8373387.

[75] R. C. Luo and T. J. Hsiao, "Indoor localization system based on hybrid wi-fi/ble and hierarchical topological fingerprinting approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 10 791–10 806, 2019. DOI: 10.1109/TVT.2019.2938893.

[76] S. Xu, R. Chen, Y. Yu, G. Guo, and L. Huang, "Locating smartphones indoors using built-in sensors and wi-fi ranging with an enhanced particle filter," *IEEE Access*, vol. 7, pp. 95 140–95 153, 2019, ISSN: 2169-3536. DOI: 10.1109/access.2019.2927387.

[77] V. Bellavista-Parent, J. Torres-Sospedra, and A. Pérez-Navarro, "Comprehensive analysis of applied machine learning in indoor positioning based on wi-fi: An extended systematic review," *Sensors*, vol. 22, no. 12, p. 4622, 2022, ISSN: 1424-8220. DOI: 10.3390/s22124622.

[78] J. Yan, G. Qi, B. Kang, X. Wu, and H. Liu, "Extreme learning machine for accurate indoor localization using rssi fingerprints in multifloor environments," *IEEE Internet of Things Journal*, vol. 8, no. 19, pp. 14 623–14 637, 2021, ISSN: 2327-4662. DOI: 10.1109/jiot.2021.3071152.

[79] P. Koovimol and S. Pattaramalai, "Experimental machine learning for rssi fingerprint in indoor wifi localization," in *2021 18th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, IEEE, May 2021. DOI: 10.1109/ecti-con51831.2021.9454865.

[80] I. Neupane, B. Alsinglawi, and K. Rabie, "Indoor positioning using wi-fi and machine learning for industry 5.0," Atlanta, GA, USA: IEEE, Mar. 2023, pp. 359–362, ISBN: 978-1-6654-5382-0. DOI: 10.1109/PerComWorkshops56833.2023.10150346.

[81] D. TAŞKIN, "Design of bluetooth low energy based indoor positioning system," *Balkan Journal of Electrical and Computer Engineering*, vol. Not available, pp. 60–65, Sep. 2017, ISSN: 2147-284X. DOI: 10.17694/bajece.334388.

[82] J. Huang, S. Chai, N. Yang, and L. Liu, "A novel distance estimation algorithm for bluetooth devices using rssi," *Proceedings of the 2017 2nd International Conference on Control, Automation and Artificial Intelligence (CAAI 2017)*, vol. Not available, Not available, 2017. DOI: 10.2991/caai-17.2017.86.

[83] S. Chai, R. An, and Z. Du, "An indoor positioning algorithm using bluetooth low energy rssi," *Proceedings of the 2016 International Conference on Advanced Materials Science and Environmental Engineering*, vol. Not available, Not available, 2016. DOI: 10.2991/amsee-16.2016.72.

[84] Z. Deng, X. Fu, and H. Wang, "An imu-aided body-shadowing error compensation method for indoor bluetooth positioning," *Sensors*, vol. 18, p. 304, Jan. 2018, ISSN: 1424-8220. DOI: 10.3390/s18010304.

[85] M. S. Bargh and R. de Groote, "Indoor localization based on response rate of bluetooth inquiries," *Proceedings of the first ACM international workshop on Mobile entity localization and tracking in GPS-less environments*, vol. Not available, Not available, Sep. 2008. DOI: 10.1145/1410012.1410024.

[86] X. Wu, R. Shen, L. Fu, X. Tian, P. Liu, and X. Wang, "Ibill: Using ibeacon and inertial sensors for accurate indoor localization in large open areas," *IEEE Access*, vol. 5, pp. 14 589–14 599, 2017, ISSN: 2169-3536. DOI: 10.1109/access.2017.2726088.

[87] M. Charlier, R.-A. Koutsiamanis, and B. Quoitin, "Scheduling uwb ranging and backbone communications in a pure wireless indoor positioning system," *IoT*, vol. 3, pp. 219–258, Mar. 2022, ISSN: 2624-831X. DOI: 10.3390/iot3010013.

[88] M. R. Mahfouz, M. J. Kuhn, Y. Wang, J. Turnmire, and A. E. Fathy, "Towards sub-millimeter accuracy in uwb positioning for indoor medical environments," *2011 IEEE Topical Conference on Biomedical Wireless Technologies, Networks, and Sensing Systems*, vol. Not available, Not available, Jan. 2011. DOI: 10.1109/biowireless.2011.5724360.

[89] L. Cheng, A. Zhao, K. Wang, H. Li, Y. Wang, and R. Chang, "Activity recognition and localization based on uwb indoor positioning system and machine learning," Vancouver, BC, Canada: IEEE, Nov. 2020, pp. 0528–0533, ISBN: 978-1-7281-8417-3. DOI: 10.1109/IEMCON51383.2020.9284937.

[90] S. Sung, H. Kim, and J.-I. Jung, "Accurate indoor positioning for uwb-based personal devices using deep learning," *IEEE Access*, vol. 11, pp. 20 095–20 113, 2023, ISSN: 2169-3536. DOI: 10.1109/access.2023.3250180.

[91] K. Minne, N. Macoir, J. Rossey, *et al.*, "Experimental evaluation of uwb indoor positioning for indoor track cycling," *Sensors*, vol. 19, p. 2041, May 2019, ISSN: 1424-8220. DOI: 10.3390/s19092041.

[92] M. Sakr, A. Masiero, and N. El-Sheimy, "Evaluation of dynamic ad-hoc uwb indoor positioning system," *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. XLII-1, pp. 379–385, Sep. 2018, ISSN: 2194-9034. DOI: 10.5194/isprs-archives-xlii-1-379-2018.

[93] G. Schroeer, "A real-time uwb multi-channel indoor positioning system for industrial scenarios," *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, vol. Not available, Not available, Sep. 2018, ISSN: Not available. DOI: 10.1109/ipin.2018.8533792.

[94] A. Poulose, O. S. Eyobu, M. Kim, and D. S. Han, "Localization error analysis of indoor positioning system based on uwb measurements," in *2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN)*, IEEE, Jul. 2019. DOI: 10.1109/icufn.2019.8806041.

[95] X. Cai, L. Ye, and Q. Zhang, "Ensemble learning particle swarm optimization for real-time uwb indoor localization," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, Not available, May 2018, ISSN: 1687-1499. DOI: 10.1186/s13638-018-1135-0.

[96] Q. Ma, X. Li, G. Li, B. Ning, M. Bai, and X. Wang, "Mrliht: Mobile rfid-based localization for indoor human tracking," *Sensors*, vol. 20, p. 1711, Mar. 2020, ISSN: 1424-8220. DOI: 10.3390/s20061711.

[97] M. Andrea, B. Alice, and N. Paolo, "A survey on indoor vehicle localization through rfid technology," *IEEE Access*, 2021, 3126380805. DOI: 10.1109/access.2021.3052316.

[98] M. Hasani, J. Talvitie, L. Sydanheimo, E.-S. Lohan, and L. Ukkonen, "Hybrid wlan-rfid indoor localization solution utilizing textile tag," *IEEE Antennas and Wireless Propagation Letters*, vol. 14, pp. 1358–1361, 2015, ISSN: 1536-1225, 1548-5757. DOI: 10.1109/lawp.2015.2406951.

[99] F. Wang, T. Su, X. Jin, Y. Zheng, J. Kong, and Y. Bai, "Indoor tracking by rfid fusion with imu data," *Asian Journal of Control*, vol. 21, pp. 1768–1777, Dec. 2018, ISSN: 1561-8625, 1934-6093. DOI: 10.1002/asjc.1954.

[100] H. Xu, Y. Ding, P. Li, R. Wang, and Y. Li, "An rfid indoor positioning algorithm based on bayesian probability and k-nearest neighbor," *Sensors*, vol. 17, p. 1806, Aug. 2017, ISSN: 1424-8220. DOI: 10.3390/s17081806.

[101] J. Liu and R. Rao, "A coordinator loss-tolerant zigbee mesh network with region detection using rfid for the real-time indoor positioning system," Shanghai, China: IEEE, Nov. 2011, pp. 472–476, ISBN: 978-0-7695-4559-2. DOI: 10.1109/MINES.2011.55.

[102] F. Hassoon Salman, A.-A. Karim, S. K. Yousef, and S. Mehdi Naderi, "An experimental comparison of rssi-based indoor localization techniques using zigbee technology," *IEEE Access*, 2023, 4385834303. DOI: 10.1109/access.2023.3305396.

[103] W. Wang, Q. Zhu, Z. Wang, X. Zhao, and Y. Yang, "Research on indoor positioning algorithm based on saga-bp neural network," *IEEE Sensors Journal*, vol. 22, pp. 3736–3744, Feb. 2022, ISSN: 1530-437X, 1558-1748, 2379-9153. DOI: 10.1109/jsen.2021.3120882.

[104] H. Pujiharsono, D. Utami, and R. D. Ainul, "Trilateration method for estimating location in rssi-based indoor positioning system using zigbee protocol," *JURNAL INFOTEL*, vol. 12, Not available, Apr. 2020, ISSN: 2460-0997, 2085-3688. DOI: 10.20895/infotel.v12i1.380.

[105] R. D. Ainul, "An enhanced trilateration algorithm for indoor rssi based positioning system using zigbee protocol," *JURNAL INFOTEL*, vol. 14, pp. 301–306, Nov. 2022, ISSN: 2460-0997, 2085-3688. DOI: 10.20895/infotel.v14i4.822.

[106] R. M. Dellosa, A. C. Fajardo, and R. P. Medina, "Modified fingerprinting localization technique of indoor positioning system based on coordinates," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 15, p. 1345, Sep. 2019, ISSN: 2502-4760, 2502-4752. DOI: 10.11591/ijeecs.v15.i3.pp1345-1355.

[107] B. K. Dash and J. Peng, "Zigbee wireless sensor networks: Performance study in an apartment-based indoor environment," *Journal of Computer Networks and Communications*, vol. 2022, pp. 1–14, Aug. 2022, ISSN: 2090-715X, 2090-7141. DOI: 10.1155/2022/2144702.

[108] S. He and S.-H. G. Chan, "Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 18, pp. 466–490, 2016, ISSN: 1553-877X. DOI: 10.1109/comst.2015.2464084.

[109] H.-H. Liu and C. Liu, "Implementation of wi-fi signal sampling on an android smartphone for indoor positioning systems," *Sensors (Basel)*, vol. 18, no. 1, p. 3, 2017, ISSN: 1424-8220. DOI: 10.3390/s18010003.

[110] T.-y. Kim and S.-s. Hwang, "Cascade aoa estimation algorithm based on flexible massive antenna array," *Sensors*, vol. 20, p. 6797, Nov. 2020, ISSN: 1424-8220. DOI: 10.3390/s20236797.

[111] H.-C. Chen, T.-H. Lin, H. T. Kung, C.-K. Lin, and Y. Gwon, "Determining rf angle of arrival using cots antenna arrays: A field evaluation," *MILCOM 2012 - 2012 IEEE Military Communications Conference*, vol. Not available, Not available, Oct. 2012. DOI: 10.1109/milcom.2012.6415851.

[112] N. Patwari, J. Ash, S. Kyperountas, A. Hero, R. Moses, and N. Correal, "Locating the nodes: Cooperative localization in wireless sensor networks," *IEEE Signal Processing Magazine*, vol. 22, pp. 54–69, Jul. 2005, ISSN: 1053-5888. DOI: 10.1109/msp.2005.1458287.

[113] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," *Proceedings of the 7th annual international conference on Mobile computing and networking*, vol. Not available, Not available, Jul. 2001. DOI: 10.1145/381677.381693.

[114] S. Go, S. Kim, and J.-W. Chong, "An efficient non-line-of-sight error mitigation method for toa measurement in indoor environments," *Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication*, vol. Not available, Not available, Jan. 2014. DOI: 10.1145/2557977.2558011.

[115] I. Guvenc and C.-C. Chong, "A survey on toa based wireless localization and nlos mitigation techniques," *IEEE Communications Surveys & Tutorials*, vol. 11, pp. 107–124, 2009, ISSN: 1553-877X. DOI: 10.1109/surv.2009.090308.

[116] D.-H. Shin and T.-K. Sung, "Comparisons of error characteristics between toa and tdoa positioning," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 38, pp. 307–311, 2002, ISSN: 0018-9251. DOI: 10.1109/7.993253.

[117] H. Wang, Y. Jiang, D. Gao, and Y. Xue, "Optimization of indoor positioning algorithm based on asynchronous tdoa model," Changchun, China: IEEE, Dec. 2019, pp. 33–37, ISBN: 978-1-7281-4399-6. DOI: 10.1109/CCIOT48581.2019.8980301.

[118] P. Grasso and M. S. Innocente, "Theoretical study of signal and geometrical properties of two-dimensional uwb-based indoor positioning systems using tdoa," Barcelona, Spain: IEEE, Feb. 2020, pp. 130–134, ISBN: 978-1-7281-5740-5. DOI: 10.1109/ICMRE49073.2020.9065121.

[119] H. Obeidat, W. Shuaieb, O. Obeidat, and R. Abd-Alhameed, "A review of indoor localization techniques and wireless technologies," *Wireless Personal Communications*, vol. 119, no. 1, pp. 289–327, Feb. 2021, ISSN: 0929-6212. DOI: 10.1007/s11277-021-08209-5.

[120] M. Ramires, J. Torres-Sospedra, and A. J. C. Moreira, "Accurate and Efficient Wi-Fi Fingerprinting-Based Indoor Positioning in Large Areas," in *96th Vehicular Technology Conference, VTC Fall 2022, London, United Kingdom, September 26-29, 2022*, IEEE, 2022, pp. 1–6. DOI: 10.1109/VTC2022-FALL57202.2022.10012985.

[121] H. Jing, J. Pinchin, C. Hill, and T. Moore, "Wi-Fi fingerprinting based on collaborative confidence level training," *Pervasive Mob. Comput.*, vol. 30, pp. 32–44, 2016. DOI: 10.1016/J.PMCJ.2015.10.005.

[122] X.-M. Yu, H.-Q. Wang, and J.-Q. Wu, "A method of fingerprint indoor localization based on received signal strength difference by using compressive sensing," *EURASIP Journal on Wireless Communications and Networking*, vol. 2020, no. 1, 2020, ISSN: 1687-1499. DOI: 10.1186/s13638-020-01683-8.

[123] J. Torres-Sospedra and A. Moreira, "Analysis of sources of large positioning errors in deterministic fingerprinting," *Sensors*, vol. 17, no. 12, p. 2736, 2017, ISSN: 1424-8220. DOI: 10.3390/s17122736.

[124] R. Ma, Q. Guo, C. Hu, and J. Xue, "An improved wifi indoor positioning algorithm by weighted fusion," *Sensors*, vol. 15, no. 9, pp. 21 824–21 843, 2015, ISSN: 1424-8220. DOI: 10.3390/s150921824.

[125] J. Bi, Y. Wang, B. Yu, H. Cao, T. Shi, and L. Huang, "Supplementary open dataset for wifi indoor localization based on received signal strength," *Satellite Navigation*, vol. 3, no. 1, pp. 25–15, 2022, ISSN: 2662-1363 2662-9291. DOI: 10.1186/s43020-022-00086-y.

[126] X. Feng, K. A. Nguyen, and Z. Luo, "A survey of deep learning approaches for wifi-based indoor positioning," *Journal of Information and Telecommunication*, pp. 1–54, 2021, ISSN: 2475-1839. DOI: 10.1080/24751839.2021.1975425.

[127] S. Subedi, H.-S. Gang, N. Y. Ko, and Hwang, "Improving indoor fingerprinting positioning with affinity propagation clustering and weighted centroid fingerprint," *IEEE access*, vol. 7, pp. 31 738–31 750, 2019, ISSN: 2169-3536. DOI: `10.1109/access.2019.2902564`.

[128] B. Wang, X. Gan, X. Liu, *et al.*, "A novel weighted knn algorithm based on rss similarity and position distance for wi-fi fingerprint positioning," *IEEE Access*, vol. 8, pp. 30 591–30 602, 2020, ISSN: 2169-3536. DOI: `10.1109/ACCESS.2020.2973212`.

[129] X. Song, X. Fan, X. He, *et al.*, "Cnnloc: Deep-learning based indoor localization with wifi fingerprinting," Leicester, UK: IEEE, Aug. 2019, pp. 589–595, ISBN: 978-1-7281-4035-3. DOI: `10.1109/SmartWorld-UIC-ATC-SCALCOM-IOP-SCI.2019.00139`.

[130] B. Wang, X. Liu, B. Yu, R. Jia, and X. Gan, "An improved wifi positioning method based on fingerprint clustering and signal weighted euclidean distance," *Sensors*, vol. 19, no. 10, p. 2300, 2019, ISSN: 1424-8220. DOI: `10.3390/s19102300`.

[131] L. Xiao, A. Behboodi, and R. Mathar, "Learning the Localization Function: Machine Learning Approach to Fingerprinting Localization," *arXiv.org*, 2018, accessed 2024-02-03].

[132] X. Wen, W. Tao, C.-M. Own, and Z. Pan, "On the dynamic rss feedbacks of indoor fingerprinting databases for localization reliability improvement," *Sensors*, vol. 16, p. 1278, Aug. 2016, ISSN: 1424-8220. DOI: `10.3390/s16081278`. [Online]. Available: `https://dx.doi.org/10.3390/s16081278`.

[133] D. Li and J. Huang, "Rss based method for sensor localization with unknown transmit power and uncertainty in path loss exponent," *2016 8th IEEE International Conference on Communication Software and Networks (ICCSN)*, vol. Not available, Not available, Jun. 2016. DOI: `10.1109/iccsn.2016.7586668`.

[134] B. Wu, X. Shi, and J. Huang, "Analysis of frequency uncertainty on positioning accuracy based on winner ii channel," *Proceedings of the 2016 3rd International Conference on Materials Engineering, Manufacturing Technology and Control*, vol. Not available, Not available, 2016. DOI: `10.2991/icmemtc-16.2016.215`.

[135] H. Zhang, H. Zhang, B. Di, K. Bian, Z. Han, and L. Song, "Towards ubiquitous positioning by leveraging reconfigurable intelligent surface," *IEEE Communications Letters*, vol. 25, pp. 284–288, Jan. 2021, ISSN: 1089-7798, 1558-2558, 2373-7891. DOI: `10.1109/lcomm.2020.3023130`.

[136] C. Song, J. Wang, and G. Yuan, "Hidden naive bayes indoor fingerprinting localization based on best-discriminating ap selection," *ISPRS International Journal of Geo-Information*, vol. 5, p. 189, Oct. 2016, ISSN: 2220-9964. DOI: 10.3390/ijgi5100189.

[137] G. M. Mendoza-Silva, J. Torres-Sospedra, and J. Huerta, "A more realistic error distance calculation for indoor positioning systems accuracy evaluation," Sapporo, Japan: IEEE, Sep. 2017, pp. 1–8, ISBN: 978-1-5090-6300-0. DOI: 10.1109/IPIN.2017.8115950.

[138] Y. Ali, N. Youssef, A. Mariette, *et al.*, "Recent advances in indoor localization: A survey on theoretical approaches and applications," *IEEE Communications Surveys and Tutorials*, 2017, 2557998141. DOI: 10.1109/comst.2016.2632427.

[139] K. A. Nguyen, "A performance guaranteed indoor positioning system using conformal prediction and the wifi signal strength," *Journal of Information and Telecommunication*, vol. 1, no. 1, pp. 41–65, 2017, ISSN: 2475-1839, 2475-1847. DOI: 10.1080/24751839.2017.1295659.

[140] D. Quezada-Gaibor, J. Torres-Sospedra, J. Nurmi, Y. Koucheryavy, and J. Huerta, "Cloud platforms for context-adaptive positioning and localisation in gnss-denied scenarios—a systematic review," *Sensors*, vol. 22, no. 1, p. 110, 2021, ISSN: 1424-8220. DOI: 10.3390/s22010110.

[141] P. Bahl and V. N. Padmanabhan, *Radar: An in-building rf-based user location and tracking system*, 2000. DOI: 10.1109/INFCOM.2000.832252.

[142] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services - MobiSys '05*, ser. MobiSys05, ACM Press, Jun. 2005. DOI: 10.1145/1067170.1067193.

[143] L. Hui, D. Houshang, B. Prashant, B. Pat, and L. Jing, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems Man and Cybernetics Part C (Applications and Reviews)*, vol. 37, no. 6, pp. 1067–1080, 2007, 2100989187, ISSN: 1094-6977. DOI: 10.1109/tsmcc.2007.905750.

[144] K. Al Nuaimi and H. Kamel, *A survey of indoor positioning systems and algorithms*, 2011. DOI: 10.1109/INNOVATIONS.2011.5893813.

[145] N. Alsindi, Z. Chaloupka, N. AlKhanbashi, and J. Aweya, "An empirical evaluation of a probabilistic rf signature for wlan location fingerprinting," *IEEE Transactions on Wireless Communications*, vol. 13, pp. 3257–3268, Jun. 2014, ISSN: 1536-1276. DOI: 10.1109/twc.2014.041714.131113.

[146] J. Oh and J. Kim, "Adaptive k-nearest neighbour algorithm for wifi fingerprint positioning," *ICT Express*, vol. 4, no. 2, pp. 91–94, 2018, ISSN: 2405-9595. DOI: 10.1016/j.icte.2018.04.004.

[147] X. Ge and Z. Qu, "Optimization wifi indoor positioning knn algorithm location-based fingerprint," in *2016 7th IEEE International Conference on Software Engineering and Service Science (ICSESS)*, IEEE, Aug. 2016. DOI: 10.1109/icsess.2016.7883033.

[148] D. Ferreira, R. Souza, and C. Carvalho, "Qa-knn: Indoor localization based on quartile analysis and the knn classifier for wireless networks," *Sensors*, vol. 20, no. 17, p. 4714, 2020, ISSN: 1424-8220. DOI: 10.3390/s20174714.

[149] S. Zhang, D. Cheng, Z. Deng, M. Zong, and X. Deng, "A novel k nn algorithm with data-driven k parameter computation," *Pattern Recognition Letters*, vol. 109, pp. 44–54, 2018, ISSN: 0167-8655. DOI: 10.1016/j.patrec.2017.09.036.

[150] E. Martin, O. Vinyals, G. Friedland, and R. Bajcsy, "Precise indoor localization using smart phones," *Proceedings of the 18th ACM international conference on Multimedia*, vol. Not available, Not available, Oct. 2010. DOI: 10.1145/1873951.1874078.

[151] W. Zhang, X. Hua, K. Yu, W. Qiu, S. Zhang, and X. He, "A novel wifi indoor positioning strategy based on weighted squared euclidean distance and local principal gradient direction," *Sensor Review*, vol. 39, no. 1, pp. 99–106, 2019, ISSN: 0260-2288. DOI: 10.1108/SR-06-2017-0109.

[152] M. Alfakih and M. Keche, "An enhanced indoor positioning method based on wi-fi rss fingerprinting," *Journal of Communications Software and Systems*, vol. 15, no. 1, pp. 18–25, 2019, ISSN: 1845-6421. DOI: 10.24138/jcomss.v15i1.612.

[153] B. Jang and H. Kim, "Indoor positioning technologies without offline fingerprinting map: A survey," *IEEE Communications surveys and tutorials*, vol. 21, no. 1, pp. 508–525, 2019, ISSN: 1553-877X. DOI: 10.1109/comst.2018.2867935.

[154] Ó. Belmonte-Fernández, R. Montoliu, Torres-Sospedra, and D. Chia-Aguilar, "A radiosity-based method to avoid calibration for indoor positioning systems," *Expert systems with applications*, vol. 105, pp. 89–101, 2018, ISSN: 0957-4174. DOI: 10.1016/j.eswa.2018.03.054.

[155] Y. Zhao, W.-C. Wong, T. Feng, and H. K. Garg, "Efficient and scalable calibration-free indoor positioning using crowdsourced data," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 160–175, 2020. DOI: 10.1109/JIOT.2019.2944929.

[156] J. Zheng, K. Li, and X. Zhang, "Wi-fi fingerprint-based indoor localization method via standard particle swarm optimization," *Sensors*, vol. 22, p. 5051, Jul. 2022, ISSN: 1424-8220. DOI: `10.3390/s22135051`.

[157] K. Kaemarungsi, "Efficient design of indoor positioning systems based on location fingerprinting," in *2005 International Conference on Wireless Networks, Communications and Mobile Computing*, vol. 1, IEEE, 2005, pp. 181–186. DOI: `10.1109/wirles.2005.1549406`.

[158] Z. You, O. Baala, and A. Caminada, *Efficient design of indoor positioning systems based on optimization model*, 2010. DOI: `10.1109/WD.2010.5657722`.

[159] S. Aomumpai, K. Kondee, C. Prommak, and K. Kaemarungsi, *Optimal placement of reference nodes for wireless indoor positioning systems*, 2014. DOI: `10.1109/ECTICon.2014.6839894`.

[160] A. Islam, M. T. Hossan, M. Z. Chowdhury, and Y. M. Jang, *Design of an indoor positioning scheme using artificial intelligence algorithms*, 2018. DOI: `10.1109/ICOIN.2018.8343265`.

[161] H. Shuai, Z. Kun, Z. Zhengqi, J. Wenqing, L. Tianyi, and L. Xiaofei, "An optimized fingerprinting-based indoor positioning with kalman filter and universal kriging for 5g internet of things," *Wireless Communications and Mobile Computing*, 2021, 3175140309. DOI: `10.1155/2021/9936706`.

[162] F. Jin, K. Liu, H. Zhang, *et al.*, "Toward scalable and robust indoor tracking: Design, implementation, and evaluation," *IEEE Internet of Things Journal*, vol. 7, no. 2, pp. 1192–1204, 2020. DOI: `10.1109/JIOT.2019.2953376`.

[163] G. Guo, K. Yan, Z. Liu, L. Chen, and R. Chen, "Virtual wireless device-constrained robust extended kalman filters for smartphone positioning in indoor corridor environment," *IEEE Sensors Journal*, vol. 23, no. 3, pp. 2815–2822, Feb. 2023. DOI: `10.1109/jsen.2022.3232332`.

[164] C. Poh Yuen, C. Ju-Chin, and W. Ruey-Beei, "A wi-fi-based passive indoor positioning system via entropy-enhanced deployment of wi-fi sniffers," *Italian National Conference on Sensors*, 2023, 36772416 4318200389. DOI: `10.3390/s23031376`.

[165] J. Torres-Sospedra, P. Richter, A. J. C. Moreira, *et al.*, "A comprehensive and reproducible comparison of clustering and optimization rules in wi-fi fingerprinting," *IEEE Trans. Mob. Comput.*, vol. 21, no. 3, pp. 769–782, 2022. DOI: `10.1109/TMC.2020.3017176`.

[166] S. Ivo, P. Cristiano, P. Cristiano, and M. Adriano, "Real-world deployment of low-cost indoor positioning systems for industrial applications," *IEEE Sensors Journal*, 2021, 3192255671. DOI: `10.1109/jsen.2021.3103662`.

[167] X. Wang, X. Zhang, C. Zu, *et al.*, "An accurate cloud-based indoor localization system with low latency," *International Journal of Intelligent Systems*, vol. 37, pp. 4794–4809, Nov. 2021, ISSN: 0884-8173, 1098-111X. DOI: `10.1002/int.22740`. [Online]. Available: `https://dx.doi.org/10.1002/int.22740`.

[168] T. T. Khanh, T. H. Hai, V. Nguyen, T. D. Nguyen, N. T. Thu, and E.-N. Huh, "The practice of cloud-based navigation system for indoor robot," *2020 14th International Conference on Ubiquitous Information Management and Communication (IMCOM)*, vol. Not available, Not available, Jan. 2020, ISSN: Not available. DOI: `10.1109/imcom48794.2020.9001709`.

[169] D. William, F. A. Nugroho, and G. P. Kusuma, "Cloud-based indoor positioning service for indoor navigation system," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 2, pp. 1774–1781, 2020, ISSN: 2278-3091. DOI: `10.30534/ijatcse/2020/133922020`.

[170] S. Liu and Z. Yan, "Verifiable edge computing for indoor positioning," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, IEEE, Jun. 2020. DOI: `10.1109/icc40277.2020.9148819`.

[171] T. T. Khanh, V. Nguyen, X.-Q. Pham, and E.-N. Huh, "Wi-fi indoor positioning and navigation: A cloudlet-based cloud computing approach," *Human-centric Computing and Information Sciences*, vol. 10, no. 1, 2020, ISSN: 2192-1962. DOI: `10.1186/s13673-020-00236-8`.

[172] X. Zhang, F. He, Q. Chen, *et al.*, "A differentially private indoor localization scheme with fusion of wifi and bluetooth fingerprints in edge computing," *Neural Computing and Applications*, vol. 34, pp. 4111–4132, Jan. 2022, ISSN: 0941-0643, 1433-3058. DOI: `10.1007/s00521-021-06815-9`.

[173] N. Jiang, G. Dong, Y. Hu, L. Gao, and J. Chen, "Toward efficient indoor positioning for cloud services in siot," in *2020 IEEE 13th International Conference on Cloud Computing (CLOUD)*, IEEE, Oct. 2020. DOI: `10.1109/cloud49709.2020.00030`.

[174] L. Yin, P. Ma, and Z. Deng, "Jlgbmloc—a novel high-precision indoor localization method based on lightgbm," *Sensors*, vol. 21, no. 8, p. 2722, 2021, ISSN: 1424-8220. DOI: `10.3390/s21082722`.

[175] A. Md. Hossen, K. Kamilia, M. Siti Nurhafizza, *et al.*, "Wi-fi fingerprinting for indoor positioning," *International Journal of Integrated Engineering*, 2022, 4316362993. DOI: `10.30880/ijie.2022.14.06.020`.

[176] G. Caso, M. Le, L. De Nardis, and M.-G. Di Benedetto, "Performance comparison of wifi and uwb fingerprinting indoor positioning systems," *Technologies (Basel)*, vol. 6, no. 1, p. 14, 2018, ISSN: 2227-7080. DOI: `10.3390/technologies6010014`.

[177] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "Spotfi: Decimeter level localization using wifi," *Computer communication review*, vol. 45, no. 4, pp. 269–282, 2015, ISSN: 0146-4833. DOI: 10.1145/2829988.2787487.

[178] X. Hu, J. Shang, F. Gu, and Q. Han, "Improving wi-fi indoor positioning via ap sets similarity and semi-supervised affinity propagation clustering," *International Journal of Distributed Sensor Networks*, vol. 11, no. 1, p. 109 642, 2015, ISSN: 1550-1477. DOI: 10.1155/2015/109642.

[179] *Received signal strength indicator - wikipedia*, https://en.wikipedia.org/wiki/Received_signal_strength_indicator, Accessed: 2023-03-05, 2023.

[180] C. Qiang, S. Van de Velde, and H. Steendam, "How to get the best out of your fingerprint database: Hierarchical fingerprint indoor positioning for databases with variable density," *IEEE Access*, pp. 1–1, 2019. DOI: 10.1109/ACCESS.2019.2939545.

[181] G. Retscher, M. Eva, V. Dennis, and H. Dirk, "Performance and accuracy test of the wlan indoor positioning system ipos," 2020.

[182] S. Schmitt, H. Will, B. Aschenbrenner, T. Hillebrandt, and M. Kyas, *A reference system for indoor localization testbeds*, 2012. DOI: 10.1109/IPIN.2012.6418865.

[183] S.-S. Jan, L.-T. Hsu, and W.-M. Tsai, "Development of an indoor location based service test bed and geographic information system with a wireless sensor network," *Sensors (Basel)*, vol. 10, no. 4, pp. 2957–2974, 2010, ISSN: 1424-8220. DOI: 10.3390/s100402957.

[184] S. Shrestha, J. Talvitie, and E. S. Lohan, "Deconvolution-based indoor localization with wlan signals and unknown access point locations," *2013 International Conference on Localization and GNSS (ICL-GNSS)*, vol. Not available, Not available, Jun. 2013, ISSN: Not available. DOI: 10.1109/icl-gnss.2013.6577256. [Online]. Available: https://dx.doi.org/10.1109/icl-gnss.2013.6577256.

[185] J. Torres-Sospedra, A. J. C. Moreira, S. Knauth, *et al.*, "A realistic evaluation of indoor positioning systems based on wi-fi fingerprinting: The 2015 evaal-etri competition," *J. Ambient Intell. Smart Environ.*, vol. 9, no. 2, pp. 263–279, 2017. DOI: 10.3233/AIS-170421.

[186] J. Torres-Sospedra, R. Montoliu, S. Trilles, Ó. Belmonte, and J. Huerta, "Comprehensive analysis of distance and similarity measures for Wi-Fi fingerprinting indoor positioning systems," *Expert Syst. Appl.*, vol. 42, no. 23, pp. 9263–9278, 2015. DOI: 10.1016/J.ESWA.2015.08.013.

[187] C. Chen, Y. Chen, Y. Han, H.-Q. Lai, and K. J. R. Liu, "Achieving centimeter accuracy indoor localization on wifi platforms: A frequency hopping approach," *IEEE Internet of Things Journal*, vol. Not available, pp. 1–1, 2016, ISSN: 2327-4662. DOI: `10.1109/jiot.2016.2628701`.

[188] N. T. Thuong, H. T. Phong, D.-T. Do, P. V. Hieu, and D. T. Loc, "Android application for wifi based indoor position: System design and performance analysis," *2016 International Conference on Information Networking (ICOIN)*, vol. Not available, Not available, Jan. 2016. DOI: `10.1109/icoin.2016.7427147`.

[189] J. Talvitie, M. Renfors, and E. S. Lohan, "Distance-based interpolation and extrapolation methods for rss-based localization with indoor wireless signals," *IEEE Transactions on Vehicular Technology*, vol. 64, pp. 1340–1353, Apr. 2015, ISSN: 0018-9545, 1939-9359. DOI: `10.1109/tvt.2015.2397598`.

[190] M. Sivers, G. Fokin, P. Dmitriev, A. Kireev, D. Volgushev, and A.-o. A. H. Ali, "Indoor positioning in wifi and nanoloc networks," *Lecture Notes in Computer Science,Internet of Things, Smart Spaces, and Next Generation Networks and Systems*, vol. Not available, pp. 465–476, 2016, ISSN: 0302-9743, 1611-3349. DOI: `10.1007/978-3-319-46301-8_39`.

[191] K. Chen, C. Wang, Z. Yin, H. Jiang, and G. Tan, "Slide: Towards fast and accurate mobile fingerprinting for wi-fi indoor positioning systems," *IEEE Sensors Journal*, vol. 18, pp. 1213–1223, Feb. 2018, ISSN: 1530-437X, 1558-1748, 2379-9153. DOI: `10.1109/jsen.2017.2778082`.

[192] H. A. A. Alfeilat and et al., "Effects of distance measure choice on k-nearest neighbor classifier performance: A review," *Big Data*, vol. 7, no. 4, p. 221, 2019, ISSN: 2167-6461. DOI: `10.1089/big.2018.0175info:doi/10.1089/big.2018.0175`.

[193] I. Silva, C. G. Pendão, J. Torres-Sospedra, and A. J. C. Moreira, "Quantifying the Degradation of Radio Maps in Wi-Fi Fingerprinting," in *International Conference on Indoor Positioning and Indoor Navigation, IPIN 2021, Lloret de Mar, Spain, November 29 - Dec. 2, 2021*, IEEE, 2021, pp. 1–8. DOI: `10.1109/IPIN51156.2021.9662558`.

[194] A. Mittal, S. Tiku, and S. Pasricha, "Adapting convolutional neural networks for indoor localization with smart mobile devices," in *Proceedings of the 2018 on Great Lakes Symposium on VLSI*, ser. GLSVLSI '18, Chicago, IL, USA: Association for Computing Machinery, May 2018, pp. 117–122, ISBN: 9781450357241. DOI: `10.1145/3194554.3194594`.

[195] Z. Rong, Y. Yexi, and C. Puchun, "An rss transform-based wknn for indoor positioning," *Sensors*, vol. 21, no. 17, p. 5685, 2021, 34502577 3195612747, ISSN: 1424-8220. DOI: 10.3390/s21175685.

[196] A. Kokkinis, L. Kanaris, A. Liotta, and S. Stavrou, "Rss indoor localization based on a single access point," *Sensors*, vol. 19, no. 17, p. 3711, 2019, ISSN: 1424-8220. DOI: 10.3390/s19173711.

[197] F. Xiao, Y. Huang, Y. Zuo, W. Kuang, and W. Wang, "Over-the-air adversarial attacks on deep learning wi-fi fingerprinting," *IEEE Internet Things J.*, vol. 10, no. 11, pp. 9823–9835, 2023. DOI: 10.1109/JIOT.2023.3236314.

[198] A. J. C. Moreira, M. J. Nicolau, F. Meneses, and A. Costa, "Wi-Fi fingerprinting in the real world - RTLS@UM at the EvAAL competition," in *2015 International Conference on Indoor Positioning and Indoor Navigation, IPIN 2015, Banff, AB, Canada, October 13-16, 2015*, IEEE, 2015, pp. 1–10. DOI: 10.1109/IPIN.2015.7346967.

[199] M. Nowicki and J. Wietrzykowski, "Low-effort place recognition with wifi fingerprints using deep learning," in *Automation 2017*. Springer International Publishing, 2017, pp. 575–584. DOI: 10.1007/978-3-319-54042-9_57. arXiv: 1611.02049.

[200] M. Ibrahim, M. Torki, and M. ElNainay, "Cnn based indoor localization using rss time-series," in *2018 IEEE Symposium on Computers and Communications (ISCC)*, 2018, pp. 01 044–01 049. DOI: 10.1109/ISCC.2018.8538530.

[201] B. A. Akram, A. H. Akbar, and O. Shafiq, "Hybloc: Hybrid indoor wi-fi localization using soft clustering-based random decision forest ensembles," *IEEE Access*, vol. 6, pp. 38 251–38 272, 2018, ISSN: 2169-3536. DOI: 10.1109/access.2018.2852658.

[202] H. Gan, M. H. B. M. Khir, G. Witjaksono Bin Djaswadi, and N. Ramli, "A hybrid model based on constraint oselm, adaptive weighted src and knn for large-scale indoor localization," *IEEE Access*, vol. 7, pp. 6971–6989, 2019. DOI: 10.1109/ACCESS.2018.2890111.

[203] S. Liu, R. De Lacerda, and J. Fiorina, "Wknn indoor wi-fi localization method using k-means clustering based radio mapping," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*, IEEE, Apr. 2021. DOI: 10.1109/vtc2021-spring51267.2021.9448961.

[204] F. Qin, T. Zuo, and X. Wang, "Ccpos: Wifi fingerprint indoor positioning system based on cdae-cnn," *Sensors*, vol. 21, no. 4, p. 1114, 2021, ISSN: 1424-8220. DOI: 10.3390/s21041114.

[205] X. Cao, Y. Zhuang, X. Yang, X. Sun, and X. Wang, "A universal wi-fi fingerprint localization method based on machine learning and sample differences," *Satellite Navigation*, vol. 2, no. 1, 2021, ISSN: 2662-1363. DOI: 10.1186/s43020-021-00058-8.

[206] A. E. Ahmed Elesawi and K. S. Kim, "Hierarchical multi-building and multi-floor indoor localization based on recurrent neural networks," in *2021 Ninth International Symposium on Computing and Networking Workshops (CANDARW)*, 2021, pp. 193–196. DOI: 10.1109/CANDARW53999.2021.00038.

[207] M. Laska and J. Blankenbach, "Deeplocbox: Reliable fingerprinting-based indoor area localization," *Sensors (Basel)*, vol. 21, no. 6, p. 2000, 2021, ISSN: 1424-8220. DOI: 10.3390/s21062000.

[208] Z. Tang, S. Li, K. S. Kim, and J. Smith, "Multi-output gaussian process-based data augmentation for multi-building and multi-floor indoor localization," *arXiv:2202.01980*, May 2022. DOI: 10.1109/iccworkshops53468.2022.9814616.

[209] A. Alitaleshi, H. Jazayeriy, and J. Kazemitabar, "Ea-cnn: A smart indoor 3d positioning scheme based on wi-fi fingerprinting and deep learning," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105 509, Jan. 2023, ISSN: 0952-1976. DOI: 10.1016/j.engappai.2022.105509.

[210] F. Potorti, P. Barsocchi, Girolami, and R. Montoliu, "Evaluating indoor localization solutions in large environments through competitive benchmarking: The evaal-etri competition," in *2015 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, IEEE, Oct. 2015. DOI: 10.1109/ipin.2015.7346970.

[211] G. Ran and H. Robert, "Towards a crowdsourced radio map for indoor positioning system," *2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, 2017, 2611631089. DOI: 10.1109/percomw.2017.7917559.

[212] K. Jayakanth, K. AbdelGhani, A.-M. Somaya, A.-M. Somaya, and A.-A. Abdulla, "Indoor positioning and wayfinding systems: A survey," *Human-centric Computing and Information Sciences*, vol. 10, no. 1, 2020, 3030402745, ISSN: 2192-1962. DOI: 10.1186/s13673-020-00222-0.

[213] N. Ahasanun, A. Bhagawat, H. Fatima, and F. Xavier, "A survey of machine learning for indoor positioning," *IEEE Access*, 2020, 3108914020. DOI: 10.1109/access.2020.3039271.

[214] C. Liu, H. Wang, M. Liu, and P. Li, "Research and analysis of indoor positioning technology," Changsha, China: IEEE, Mar. 2021, pp. 1212–1217, ISBN: 978-1-6654-4681-5. DOI: 10.1109/AEMCSE51986.2021.00248.

[215] L. Oliveira, D. Schneider, J. D. Souza, and W. Shen, “Mobile device detection through wifi probe request analysis,” *IEEE Access*, vol. 7, pp. 98 579–98 588, 2019, ISSN: 2169-3536. DOI: 10.1109/access.2019.2925406.

[216] A. Cardoso and P. Simões, “Cloud computing: Concepts, technologies and challenges,” in *Communications in Computer and Information Science*. Springer Berlin Heidelberg, 2012, pp. 127–136. DOI: 10.1007/978-3-642-31800-9_14.

[217] Y. Zhou, D. Zhang, and N. Xiong, “Post-cloud computing paradigms: A survey and comparison,” *Tsinghua Science and Technology*, vol. 22, no. 6, pp. 714–732, 2017, ISSN: 1007-0214. DOI: 10.23919/tst.2017.8195353.

[218] Y. Wei and M. B. Blake, “Service-oriented computing and cloud computing: Challenges and opportunities,” *IEEE Internet Computing*, vol. 14, no. 6, pp. 72–75, 2010, ISSN: 1089-7801. DOI: 10.1109/mic.2010.147.

[219] K. Sumalatha and M. S. Anbarasi, “A review on various optimization techniques of resource provisioning in cloud computing,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, p. 629, 2019, ISSN: 2088-8708. DOI: 10.11591/ijece.v9i1.pp629-634.

[220] P. Konstantopoulos, E. G. Petrakis, and S. Sotiriadis, “Inaas: Indoors navigation as a service on the cloud and smartphone application,” in *2018 IEEE 39th Sarnoff Symposium*, IEEE, Sep. 2018. DOI: 10.1109/sarnof.2018.8720502.

[221] A. Christy Jeba Malar, M. Deva Priya, J. Sengathir, N. Kiruthiga, R. Anitha, and T. Sangeetha, “An intelligent multi-floor indoor positioning system for cloud-based environment,” *International Journal of Computers and Applications*, pp. 1–8, 2019, ISSN: 1206-212X. DOI: 10.1080/1206212x.2019.1696447.

[222] K. Chen and K. R. Vadde, “Design and evaluation of an indoor positioning system framework,” *UC Berkeley course project for CS262A*, 2012.

[223] N. Min-Allah, M. B. Qureshi, F. Jan, S. Alrashed, and J. Taheri, “Deployment of real-time systems in the cloud environment,” *The Journal of Supercomputing*, vol. 77, no. 2, pp. 2069–2090, 2021, ISSN: 0920-8542. DOI: 10.1007/s11227-020-03334-7.

[224] O. Vogel, I. Arnold, A. Chughtai, and T. Kehrer, “Software architecture a comprehensive framework and guide for practitioners,” 2011.

[225] D. Ameller, X. Burgués, D. Costal, C. Farré, and X. Franch, “Non-functional requirements in model-driven development of service-oriented architectures,” *Science of Computer Programming*, vol. 168, pp. 18–37, 2018, ISSN: 0167-6423. DOI: 10.1016/j.scico.2018.08.001.