

Article

Experimental Study of Bluetooth Indoor Positioning Using RSS and Deep Learning Algorithms

Chunxiang Wu ^{1,2}, Ieok-Cheng Wong ¹, Yapeng Wang ^{1,*}, Wei Ke ¹ and Xu Yang ¹¹ Faculty of Applied Sciences, Macao Polytechnic University, Macao, China;

chunxiang.wu@mpu.edu.mo (C.W.); ieokcheng.wong@mpu.edu.mo (I.-C.W.); wke@mpu.edu.mo (W.K.); xuyang@mpu.edu.mo (X.Y.)

² School of Artificial Intelligence, Guangdong Mechanical & Electrical Polytechnic, Guangzhou 510515, China

* Correspondence: yapengwang@mpu.edu.mo

Abstract: Indoor wireless positioning has long been a dynamic field of research due to its broad application range. While many commercial products have been developed, they often are not open source or require substantial and costly infrastructure. Academically, research has extensively explored Bluetooth Low Energy (BLE) for positioning, yet there are a noticeable lack of studies that comprehensively compare traditional algorithms under these conditions. This research aims to fill this gap by evaluating classical positioning algorithms such as K-Nearest Neighbor (KNN), Weighted K-Nearest Neighbor (WKNN), Naïve Bayes (NB), and a Received Signal Strength-based Neural Network (RSS-NN) using BLE technology. We also introduce a novel method using Convolutional Neural Networks (CNN), specifically tailored to process RSS data structured in an image-like format. This approach helps overcome the limitations of traditional RSS fingerprinting by effectively managing the environmental dynamics within indoor settings. In our tests, all algorithms performed well, consistently achieving an average accuracy of less than two meters. Remarkably, the CNN method outperformed others, achieving an accuracy of 1.22 m. These results establish a solid basis for future research, particularly towards enhancing the precision of indoor positioning systems using deep learning for cost-effective, easy to set up applications.

Keywords: BLE indoor positioning; received signal strength; convolutional neural network**MSC:** 68T07

Citation: Wu, C.; Wong, I.-C.; Wang, Y.; Ke, W.; Yang, X. Experimental Study of Bluetooth Indoor Positioning Using RSS and Deep Learning Algorithms. *Mathematics* **2024**, *12*, 1386. <https://doi.org/10.3390/math12091386>

Academic Editor: Faheim Sufi

Received: 25 March 2024

Revised: 28 April 2024

Accepted: 29 April 2024

Published: 1 May 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The evolution of outdoor positioning technology, exemplified by the Global Positioning System (GPS), has established itself as a pivotal instrument for outdoor endeavors and spatial navigation. Likewise, indoor positioning assumes critical importance across diverse environments, including shopping malls, parking facilities, and industrial warehouses. In indoor environments, satellite positioning systems lose efficacy, resulting in hindered location detection and the inability to identify points of interest within facilities such as building complexes.

The most used indoor positioning technology is called RSS-based positioning, or Received Signal Strength based positioning, and operates by measuring the power level of received radio signals from known locations, typically from fixed transmitters like Wi-Fi access points or Bluetooth beacons. The core concept is that the signal strength decreases predictably as the distance from the source increases. By measuring the RSS values from multiple known points, the system can estimate the receiver's location through techniques such as trilateration or fingerprinting.

In trilateration, RSS values are converted into distance estimates between the receiver and multiple transmitters. Using these distances, the system calculates the receiver's location by finding where these distances intersect.

In fingerprinting, during a preliminary survey, RSS values at various known locations are recorded to create a “radio map”. Later, real-time RSS readings are matched against this map using algorithms like nearest neighbor or machine learning methods to estimate the user’s location based on the best match between the observed and recorded data.

Both methods rely on the propagation model of the environment, which can be influenced by factors such as obstacles, reflections, and interference, affecting the accuracy and reliability of the RSS-based positioning. Commonly employed techniques in indoor positioning include the use of Wi-Fi, RFID, and Bluetooth technologies, each having unique benefits and drawbacks. However, they often struggle with challenges such as signal interference, physical obstructions, and high power consumption, which can degrade positioning accuracy.

Historically, indoor positioning systems primarily relied on Wi-Fi because of its widespread availability. However, the massive deployment of Wi-Fi access points (APs) required to improve positioning accuracy can be costly in terms of power consumption and infrastructure. Extensive Wi-Fi deployments necessitate numerous APs to provide sufficient reference signals for precise positioning, which often prove impractical due to the high energy and operational costs involved.

In contrast, Bluetooth Low Energy (BLE) presents a more viable solution. BLE access points or beacons are significantly cheaper, have a smaller footprint, and are more power-efficient. With battery-powered units that can operate for over a year, BLE beacons can be easily deployed in indoor environments without the need for extensive cabling. This makes BLE an increasingly attractive option for efficient and cost-effective indoor positioning.

Angle of Arrival (AoA) positioning using Bluetooth Low Energy (BLE) is a technique where the direction of an incoming signal is determined using multiple antennas. This method enhances accuracy in determining locations, particularly useful in complex indoor environments for precise applications such as asset tracking. AoA offers superior accuracy and is less susceptible to multipath interference compared to other methods like RSSI, due to its focus on signal direction rather than strength.

However, implementing AoA can be complex and costly because it requires sophisticated antenna arrays and processing units. The setup also demands careful calibration and may need regular adjustments with changes in the environment. Moreover, the effectiveness of AoA decreases over longer distances, and the hardware required can increase device size and power consumption, which might not be ideal for smaller, battery-operated devices. Considering the real-world practical solution, in this study, we still focus on the RSS-based BLE positioning.

By using the Received Signal Strength from BLE beacons, we can determine the current location through positioning algorithms. This method is supplemented by an experimental approach involving five machine learning algorithms in this research: KNN, WKNN, NB, RSS-NN, and a new CNN based method. These algorithms are evaluated in terms of accuracy and our proposed CNN algorithm outperforms all other methods. Finally, we also investigated the Quuppa Intelligent Locating System, a commercial indoor positioning system, to gauge their effectiveness in a controlled indoor environment.

The main contributions of this study are below:

- **Enhanced Indoor Positioning Model:** We introduce a novel indoor positioning model that uses a CNN, optimized for processing the spatial dependencies in RSS data transformed into an image-like format. This model addresses the shortcomings of traditional RSS fingerprinting methods by effectively handling the environmental dynamics within indoor settings.
- **Comparative Analysis of Algorithms:** Our research systematically compares the performance of traditional algorithms with our CNN approach, providing insights into their practical applications and limitations in real-world indoor settings.
- **Experimental Validation:** The effectiveness of the proposed model and comparative algorithms is validated through rigorous field tests within a meticulously prepared

indoor environment. This includes the collection of a significant dataset that captures the complexity of real-world conditions.

- **Foundation for Future Research:** Our findings lay the groundwork for future research, particularly in enhancing the accuracy of indoor positioning systems using deep learning techniques and adapting to environmental changes more dynamically.

This paper is organized as follows: Section 2 presents an experimental overview of applicable research; Section 3 outlines the methodology employed for indoor positioning; Section 4 explains the design of positioning algorithms; and Section 5 encompasses the construction of the experimental site, thorough data analysis, rigorous evaluation of algorithms, and resolution of existing questions. Finally, in Section 6, we summarize the findings of our research, spotlighting achievements and limitations, and propose prospective research directions.

2. Related Works

With the advent of the Internet of Things (IoT) and the maturity of location-based technologies, the demand for indoor location services is growing.

Indoor positioning is in high demand across various domains, including enterprise management, security surveillance, emergency rescue, smart elderly care, and warehouse logistics. Nevertheless, existing solutions have proven inadequate in satisfying the user's need for heightened precision. In consequence, the widespread adoption of diverse indoor positioning technologies has been spurred by advancements in the field, leading to increased accuracy.

Lu Bai et al. [1] introduced a sensing system designed to capture raw Received Signal Strength Indicator (RSSI) data from BLE beacons. They proposed both a trilateration-based method and a fingerprinting-based method for indoor location determination and user tracking. Li et al. [2] employed virtual fingerprinting and two-way ranging techniques to address the four prominent limitations observed in traditional RSSI fingerprint-based methods. Subhan et al. [3] proposed a hybrid methodology that combines fingerprinting-based K-Nearest Neighbors (K-NN) with lateration-based MinMax position estimation. This approach utilizes a Euclidean distance formulation rather than relying on indoor radio channel modeling. Urano et al. [4] utilized an end-to-end Long Short-Term Memory (LSTM) neural network for indoor location estimation of BLE devices. Their approach demonstrates superior accuracy when compared to trilateration or fingerprint-based methods.

P. Weerasinghe et al. [5] introduced a Feed Forward Neural Network (FFNN) which utilizes Received Signal Strength Indicator (RSSI) values to accurately determine the location of a moving object or individual. This capability is essential for IoT-based Ambient Assisted Living (AAL) applications. Choi Jeongsik et al. [6] explored an unsupervised learning approach to predict the positions of unknown nodes using anchor nodes. They developed a calibration curve to rectify the distortion in raw distance measurements, enhancing the accuracy of node localization. Ding Hai-Lan et al. [7] introduced a hybrid algorithm merging Angle of Arrival (AOA) and Received Signal Strength Indication (RSSI) to mitigate the limitations inherent in RSSI alone. Through repeated testing, they achieved an average location measurement accuracy of approximately 35 to 36 centimeters. Chia-Hsin et al. [8] employed Angle of Arrival (AoA) and Received Signal Strength Indication (RSSI) techniques to mitigate Non-Line-of-Sight (NLOS) effects through appropriate range scaling and correction of AoA to the base station. This approach effectively addresses NLOS errors, rendering it appealing for location estimation within cellular networks. Le, Anh Tuyen, et al. [9] Le, Anh Tuyen, et al. introduced a range-based localization algorithm aimed at enhancing accuracy. The algorithm utilizes mathematical formulas combining Angle of Arrival (AoA) and Received Signal Strength Indication (RSSI) to achieve precise estimation of target nodes. Although several studies have concentrated on BLE-based positioning, there exists a significant gap in the literature concerning a thorough investigation that systematically compares traditional algorithms and their performance within this specific context.

3. Indoor Positioning Method

3.1. Wireless Communication Methods

Indoor positioning technology can be generally classified into two primary categories: wireless communication-based and physical-based methods. Each of these technologies possesses distinct characteristics, and their respective accuracies can be compared to evaluate their performance.

This categorization is based on the principles used to achieve positioning. Wireless communication technology, such as Wi-Fi, RFID, UWB, ZigBee, and Bluetooth Low Energy, is one such category.

3.1.1. Wi-Fi

Indoor Wi-Fi networks possess the capability to serve not only as conventional network infrastructures but also as tools for modeling path loss. This is achieved by leveraging the spatially propagated path loss phenomenon to interpret location data. Wi-Fi technology is frequently utilized for indoor positioning, with wireless local area networks (WLANs) comprising wireless access points, including wireless routers. These networks facilitate positioning, monitoring, and tracking endeavors within complex environments [10]. Nevertheless, Wi-Fi-based systems face susceptibility to interference from other signals, compromising their accuracy, while concurrently exhibiting high energy consumption in positioner devices. In Wi-Fi positioning, reliance is often placed on Received Signal Strength Indication (RSSI) to estimate distance, subsequently employing trilateration based on multiple distance measurements from different reference points. Position determination hinges on the estimated values of detected and received signals, where signal strength serves as a determinant of signal quality. However, RSSI-based systems typically offer only coarse-grained position estimates and may fall short of providing measurements with sufficient precision to accurately determine position in all scenarios.

3.1.2. Bluetooth Low Energy

The efficient power consumption characteristics of Bluetooth Low Energy (BLE) beacons facilitate battery-powered operation for durations exceeding a year, thereby minimizing infrastructure expenses when juxtaposed with alternative technologies. Despite the shared utilization of Bluetooth technology, the implementation specifics diverge. Historically, indoor positioning systems have been constructed utilizing either Apple's iBeacon protocol or Google's Eddystone protocol. The iBeacon protocol mandates the deployment of a considerable quantity of beacon devices indoors to attain a positioning precision within the range of 1–2 m [11]. Bluetooth communication represents a short-range, low-power wireless transmission technology utilized in establishing multiple Bluetooth local area network (LAN) access points within indoor environments to configure a network based on a multi-user framework. This approach guarantees that Bluetooth LAN access points consistently function as the principal device within this micro-grid, enabling the determination of user location information through signal strength detection.

3.1.3. RFID

Radio frequency identification (RFID) technology operates on the principles of electromagnetic signal transmission to enable communication between RFID tags and readers. This technology presents numerous benefits, such as enhanced data rates, heightened security measures, and the capacity to mitigate challenges associated with non-line-of-sight communication [12]. Nonetheless, obstacles including the deployment of a substantial quantity of RFID devices and the necessity to track individuals carrying RFID tags have impeded its widespread commercial utilization within the realm of indoor positioning.

3.1.4. UWB

Ultra-wideband technology (UWB) is a short-range communication technology distinguished by its ability to transmit pulses with a bandwidth exceeding 500 MHz within

a 1 nanosecond timeframe, all while maintaining low power consumption. UWB signals possess unique characteristics in terms of signal type and spectrum, rendering them less susceptible to multipath effects and highly adept at penetrating obstacles. Consequently, UWB technology facilitates positioning accuracy at levels as precise as 10 cm, even in intricate indoor environments [13]. Nonetheless, the progression of UWB technical standards has been sluggish, curtailing its broad implementation in consumer electronics. Although acclaimed for its superior accuracy, UWB technology remains confined to select specialized industrial applications due to the lack of support for the UWB protocol in mainstream mobile smart devices. In contrast to conventional narrow-band systems, UWB systems offer several advantages, including robust penetration capabilities, minimal power consumption, effective mitigation of multipath effects, elevated safety standards, and reduced system complexity, all of which contribute to a notable enhancement in positioning accuracy.

3.1.5. ZigBee

ZigBee technology, founded on the IEEE 802.15.4 [14] protocol, is designed for low cost, low power, and low data wireless sensor networks. Nonetheless, ZigBee finds limited adoption in commercial indoor positioning primarily because the majority of consumer-oriented smart devices lack support for this protocol.

3.2. Physical Methods

Physical methods, on the other hand, use visible light, infrared, geomagnetism, and computer vision techniques to achieve positioning.

3.2.1. Infrared LED

Visible light communication technology harnesses high-frequency flashing signals emitted by LED light sources to encode positional information [15]. This encoded information is subsequently deciphered by light-sensitive sensors, enabling the determination of the mobile device's position. However, the efficacy of this technology relies on LED lights equipped with flashing coding functionality, and its accuracy may diminish due to challenges associated with non-line-of-sight communication.

3.2.2. Geomagnetic

Geomagnetic-based indoor location technology has emerged as an innovative positioning method increasingly adopted in commercial sectors in recent years. This approach capitalizes on the ubiquitous nature of the Earth's magnetic field. Geomagnetic field data at specific locations worldwide is derived using the International Geomagnetic Reference Field Model (IGRF), alongside latitude and longitude coordinates provided by the International Association for Geomagnetism and Upper Atmospheric Physics (IAGA) [16].

3.2.3. Computer Vision

The computer vision-based indoor positioning system represents a well-established method wherein a computer vision algorithm is employed to ascertain the location coordinates of an individual. This is achieved by comparing captured image data during the positioning process with a pre-existing visual database of the venue [17]. In recent years, propelled by rapid advancements in artificial intelligence technology, the accuracy of computer vision-based indoor positioning systems has notably enhanced, achieving precision levels of approximately 1–2 meters.

3.3. Comparison of Different Indoor Positioning Technologies

By consolidating the distinctive characteristics of various indoor positioning technologies discussed earlier, we can present an overview in the form of Table 1. This table provides the key properties and attributes associated with each technology, their respective strengths and limitations.

Table 1. Contrast of indoor positioning technologies.

	Wi-Fi [10]	BLE [11]	RFID [12]	ZigBee [13]	UWB [14]	Infrared LED [15]	Geomagnetic [16]	Computer Vision [17]
Typical positioning accuracy(m)	15–20	2–3	1–5	1–5	0.1	1–2	1–2	1
Hardware Deployments	Large	Large	Large	Large	Large	Large	Few	Large
Collection Efforts	High	High	None	None	None	None	More	Very High
System Calculations	Moderate	Moderate	High	High	High	High	High	Very High
System Network Bandwidth	Moderate	Low	Low	Low	High	Low	High	Very High

When considering the diverse properties of various indoor positioning technologies, it becomes evident that they exhibit distinct characteristics. Consequently, when deploying these technologies in commercial environments, it is essential to consider not only the inherent accuracy of the technology, but also factors such as the specific environment, construction aspects, deployment strategies, rollout processes, and other relevant considerations. In this context, BLE indoor positioning technology emerges as a highly suitable solution, even in the prevalent era of iBeacon adoption. This is primarily due to the widespread prevalence of Bluetooth-enabled devices, making it a feasible choice for a broad range of commercial applications. Furthermore, it is important to acknowledge that besides the accuracy of the positioning device itself, the effectiveness of the algorithm in correcting measurement inaccuracies also plays a vital role.

A very traditional approach, the computer vision-based indoor positioning system uses a computer vision algorithm to determine the coordinates of a person's location by comparing the image information captured by the camera during the positioning process with a database of pre-collected visual photos of the venue. With the rapid development of artificial intelligence technology in recent years, the accuracy of computer vision-based indoor positioning systems has been greatly improved, generally in 1 to 2 m.

Integrating all of the above properties of the different indoor positioning technologies, Table 1 can be visualized as follows.

Thus, different indoor positioning technologies are very different from each other, and if they are to be applied in a commercial environment, not only the accuracy of the technology itself, but also the environment, construction, deployment, rollout, and other factors should be taken into consideration.

Based on BLE's indoor positioning technology, even in the era of widespread use of iBeacon, it is a very suitable solution when all factors are taken into consideration, because almost everyone has a Bluetooth device. In addition to the measurement accuracy of the device itself, how the accuracy is corrected by the algorithm is also very important.

3.4. Comparing Wi-Fi and BLE for Practical Mobile Positioning

For practical and easy deployment within existing environments, Wi-Fi and BLE are the most commonly used technologies for indoor positioning, as most modern smartphones are equipped with these capabilities. However, these two technologies are not initially designed to consider the position functions. Here, we discuss the practical issues related to the two technologies in terms of MAC mechanism and scanning capabilities.

3.4.1. MAC Mechanism in Wi-Fi and BLE

The Medium Access Control (MAC) mechanisms [18,19] of Wi-Fi and BLE are designed to optimize their specific wireless environments and usage models. Wi-Fi utilizes Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), optionally enhanced with Request to Send/Clear to Send (RTS/CTS) protocols and acknowledgments (ACKs) to manage data traffic in dense and potentially congested environments, ensuring robust high-speed data transmission by avoiding collisions. In contrast, BLE employs a simpler,

less resource-intensive approach using Adaptive Frequency Hopping (AFH) to mitigate interference and enhance connection stability in a primarily low power, low data rate environment. BLE's method involves less overhead and is geared towards maximizing energy efficiency, which is crucial for battery-operated devices like wearables and IoT sensors. In terms of positioning scenario which require long-time continuous operating, BLE is more favorable as its low power consumption.

3.4.2. Scanning Capabilities in Wi-Fi and BLE

As RSS-based positioning needs the mobile to constantly scan the surrounding Wi-Fi APs or BLE beacons for real time update of user's location. The scanning capability of both technologies is crucial for positioning usage.

Wi-Fi devices can perform network scans using either passive or active methods, with active scanning being quicker. Typically, a Wi-Fi device actively scans by spending about 50 milliseconds on each channel, allowing it to theoretically scan all 13 channels of the 2.4 GHz band roughly 92 times per minute. However, such frequent scanning is generally impractical as it would severely impact battery life and network efficiency. In reality, devices adjust their scanning frequency based on context and necessity, employing a mix of passive and active scanning to optimize both power consumption and connection stability. In our own test, a successful scanning of APs can take up to several seconds in smartphones.

The speed at which a mobile phone can scan for BLE (Bluetooth Low Energy) beacons varies based on the scan settings—specifically, the scan interval and window. Typically, a mobile phone can alternate between fast and slow scanning modes depending on the application's needs and battery life considerations. Fast scanning might involve a scan window of about 30 to 100 milliseconds and an interval of 100 to 500 milliseconds, allowing for the quick discovery of nearby devices. Slow scanning, used mainly to conserve battery when the app is running in the background or when high responsiveness is not critical, could extend the interval to several seconds. In our testing, smartphones can take several scans within one second; therefore, real-time position updates become possible.

3.5. RSS-Based BLE Method

Apple and Google have engaged in collaborative efforts towards the development of standards for BLE beacons [20]. Apple unveiled its iBeacon technology during the 2013 Apple Worldwide Developers Conference [21], while Google introduced its Eddystone technology in July 2015. These small wireless sensors transmit data and enable communication with other smart devices, making them a viable option for the Internet of Things (IoT) [22].

BLE beacons function on Bluetooth Low Energy technology, offering advantages such as low cost, minimal power consumption, and high scanning/broadcasting speed. These devices can broadcasting/transmit radio signals with data, which can be received by nearby devices. Signal strengths can be analyzed to obtain additional information. BLE beacons can also be configured to transmit packets for positioning at high rates (e.g., every 100 ms), rendering them suitable for real-time continuous positioning scenarios.

Additionally, these devices can be deployed in a dense manner, making them particularly effective for accurate and real-time indoor positioning. The BLE positioning technique utilizes RSS to obtain an approximation of the distance between the transmitter and receiver. This distance information is then used in trilateration to determine the position of the receiver based on multiple distance measurements from different points. The quality of the detected signal is estimated based on the strength of the received signal, which is represented by a numerical value. However, using RSS alone can only provide a rough estimation of the position, and its accuracy may not always be sufficient for precise localization.

Wireless-based positioning algorithms have been extensively studied for various wireless technologies over the years. Numerous RSS-based systems rely on free space radio

propagation loss equations. For instance, the propagation loss in free space is proportional to $1/r^2$, where r denotes the distance between the transmitter and receiver.

The distance between the transmitter and receiver can be estimated from the RSS values by using the path loss model [23]. For outdoor locations, the relationship between distance d and path loss P_L in dB is given with Equation (1):

$$P_L(d) = P_L(d_0) + 10\gamma \log_{10} \left(\frac{d}{d_0} \right) \quad (1)$$

where d_0 is the reference distance, $P_L(d_0)$ is the respective path loss in dB and γ is the path loss exponent, typically around 2 in free space.

In indoor settings, the presence of multiple obstacles, such as walls, furniture, and human bodies, between a transmitter and receiver result in significant propagation loss due to radio signal absorption and diffraction. The Log-Distance Path Loss extends the free-space model by considering additional losses caused by obstacles and multipath propagation, where signals take multiple paths to the receiver due to reflections, as given with Equation (2):

$$P_L(d) = P_L(d_0) + 10\gamma \log_{10} \left(\frac{d}{d_0} \right) + X_\sigma \quad (2)$$

where X_σ is a Gaussian random variable representing the additional loss caused by obstacles and signal reflections, and n is usually greater than 2, often ranging from 2.7 to 3.5 depending on the density of obstacles.

However, for an accurate calculation of path loss in complex indoor environment, ray tracing may be used to simulate every potential path that a signal might travel, including reflections, diffraction, and scattering, which provides a more accurate depiction of signal behavior in complex indoor environments.

In real devices, signal strength is usually measured and reflected with Radio Signal Strength Indicator (RSSI). RSSI is measured in decibels relative to a milliwatt (dBm). The value is typically acquired directly from the hardware of the receiving device. It represents the power level of the received signal as detected by the receiver's RF front end. Most Bluetooth devices can provide the RSSI value of a connection or of nearby devices in discoverable mode. The RSSI value can be approximately linked to the distance between devices using the logarithmic path loss models discussed previously. The formula to relate RSSI to distance is derived from the general path loss model as shown in Equation (3):

$$RSSI = -(10n \log_{10} d + A) \quad (3)$$

where A is defined as the absolute energy expressed in dB m at a distance of 1 m from the transmitter, which corresponds to the RSS reading at 1 m from the transmitter; n is the signal transmission constant, which is influenced by the signal transmission environment; and d is the distance between the transmitter node and the receiver node. Although RSSI can indicate the distance between the transmitter and receiver, however, the RSSI value can vary significantly depending on environmental factors such as obstacles, walls, and interference from other wireless devices. Especially in indoor environments, RSSI values can fluctuate due to multipath effects where the signal bounces off objects, arriving at the receiver multiple times from different paths. In addition, different devices might have slightly different ways of measuring and reporting RSSI, which can lead to inconsistencies across different types of Bluetooth hardware. So, for location tracking, RSSI should be used carefully, as it is susceptible to various sources of error and environmental effects.

4. Algorithms Design and Analysis

4.1. Experiment Environment

In order to test and compare the performance of different positioning algorithms, we chose a large office to setup the experiment. The layout of the experimental environment is shown in Figure 1, with a size 12 m × 6 m; normal office furniture (office desks, chairs)

exist in the environment for a realistic complex indoor environment. There are eight BLE beacons are installed on the ceiling of the room and around the office. On average, each of the eight beacons is at a distance of around 1.5 m from the nearby beacons. The room space was divided in a 1.2 m × 1.2 m block. To compare the efficacy of the algorithms, a series of experiments were conducted within an office space equipped with eight BLE Beacons. The experimental data were acquired using a customized BLE scanner application running on an Android device, which was specifically configured to detect only the aforementioned eight BLE Beacons. Each data record consisted of eight RSS values derived from the detected BLE beacons, along with the corresponding location coordinates.



Figure 1. Experimental layout.

Figure 1 displays the arrangement of the Beacons, training points, and testing points within the office setting. The 28 green circles denote the training points, with the distance between each training point and its adjacent point set at 1.5 m. Meanwhile, the 12 red triangles represent the testing points, and the 8 blue diamonds indicate the location of the BLE beacons. For each training point, approximately 500 to 600 records were collected in each of the four different directions. As for each testing point, approximately 50 records were gathered. Upon filtering out any invalid records, the total number of valid data points was found to be 540 testing records and 67,589 training records.

The raw data collected are presented in Figure 2. At each data collection point, multiple entries are recorded. For each entry, the mobile phone’s received signal strength in RSSI was collected from eight beacons (BC1 to BC8), along with the original coordinates ranging from A–J and 01–05. To simplify location calculations, these coordinates have been remapped to integer values ranging from 1–10, both horizontally and vertically. These coordinates will be utilized for both training the model and subsequent positioning tasks.

	BC1	BC2	BC3	BC4	BC5	BC6	BC7	BC8	row	col	x	y
0	-62	-67	-85	-87	-67	-88	-74	-74	I	01	1	1
1	-61	-68	-83	-83	-86	-68	-91	-74	I	01	1	1
2	-61	-67	-86	-83	-88	-65	-88	-73	I	01	1	1
3	-62	-68	-81	-76	-83	-65	-91	-74	I	01	1	1
4	-58	-70	-82	-77	-79	-64	-84	-76	I	01	1	1
...
2694	-61	-76	-79	-79	-79	-78	-80	-68	H	05	2	5
2695	-62	-74	-77	-85	-97	-75	-81	-62	H	05	2	5
2696	-69	-72	-74	-73	-89	-71	-83	-78	I	05	1	5
2697	-67	-73	-73	-74	-87	-71	-83	-79	I	05	1	5
2698	-68	-74	-75	-77	-88	-73	-84	-74	I	05	1	5

[2699 rows x 12 columns]

Figure 2. Raw data collected.

The experimental data were utilized to empirically assess the performance of five distinct algorithms, namely KNN, WKNN, and CNN. To compare the efficacy of these algorithms, the error distance between the actual location and the location estimated by the respective algorithms was calculated, along with their associated coordinates.

4.2. K-Nearest Neighbor Algorithm

The K-Nearest Neighbors (KNN) algorithm is a well-established and straightforward method for addressing classification tasks, as indicated by its prominence in the literature. Its fundamental premise is to identify the closest neighbors within the entire training dataset by using the received data of the object of interest, and to subsequently assign the object to the class that is most frequently represented among the nearest neighbors.

To identify the k-nearest neighbors, the Euclidean distance between the current RSS observation and each of the pre-existing RSS records is computed, resulting in a distance object that includes the corresponding location coordinates. The Euclidean distance formula is utilized to compute the distance between every location as Equation (4).

$$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (4)$$

where x_i and y_i denote the i^{th} out of n values within each pre-recorded RSS and the currently observed RSS.

Once the distance calculations have been completed, the results are sorted in ascending order based on their distances. The coordinate of the current location is then determined by taking the average of the coordinates of the k-nearest neighbors, which are represented by the first k elements in the sorted list of distance objects, as expressed in Algorithm 1.

Algorithm 1 Proposed KNN Algorithm

```

1: Input: z: testing sample, X training data, L: class labels of X, k: number of nearest neighbors
2: Output: Class of the testing data z
3: Start
4:    $D \leftarrow \emptyset$ 
5:   for each x in X do
6:     Calculate the distance:  $d_i(z, x_i) = \sqrt{\sum_{i=1}^n (z - x_i)^2}$ 
7:      $D \leftarrow D \cup d_i$ 
8:   end
9:   end
10:  Sort  $D = \{d_i, \text{for } i = 1 \text{ to } k\}$  in ascending order;
11:  Select  $N \supseteq Z$ , based on  $D$ , be the set of testing data with the  $k$  small distances
12:  Classify z in the majority class  $C(z) = \operatorname{argmax}_k \sum_{X_j \in N} C(X_j, L_k)$ 
13: End

```

4.3. Weight K-Nearest Neighbor Algorithm

To enhance the accuracy of the KNN algorithm, which may be affected by the varying distances of the nearest neighbors from the actual location, the WKNN was developed. In WKNN, the nearest neighbors are assigned weights that are proportional to their respective distances, with closer neighbors receiving higher weights than those that are further away. This approach aims to reduce the impact of more distant neighbors on the final prediction. Following the sorting of the distance list, the coordinates of the current location in WKNN are computed by taking a weighted average of the coordinates of the k-nearest neighbors. The weight assigned to each neighbor is determined by its distance, with closer neighbors receiving higher weights than those that are further away, as presented in Algorithm 2.

Both KNN and WKNN face limitations due to their high sensitivity to environmental changes, such as physical barriers and signal interference, which affect signal strength and positioning accuracy. Both require a densely populated and precisely measured set of sample points for accurate functioning, making them vulnerable to errors in sparse or dynamically changing environments. Additionally, Weighted KNN's complexity increases with the need to fine-tune the weighting factors based on distance, and both algorithms struggle with scalability and need frequent updates of the RSS map to cope with environmental changes, impacting their efficacy in large-scale or real-time applications.

Algorithm 2 Proposed WKNN Algorithm

```

1: Input:  $z$ : testing sample,  $X$  training data,  $L$ : class labels of  $X$ ,  $k$ : number of nearest neighbors
2: Output: Class of the testing data  $z$ 
3: Start
4:    $D \leftarrow \emptyset$ 
5:   for each  $x$  in  $X$  do
6:     Calculate the distance:  $d_i(z, x_i) = \sqrt{\sum_{i=1}^n (z - x_i)^2}$ 
7:      $D \leftarrow D \cup d_i$ 
8:   end
9:   Sort  $D = \{d_i, \text{for } i = 1 \text{ to } k\}$  in ascending order;
10:  Select  $N \supseteq Z$ , based on  $D$ , be the set of testing data with the  $k$  small distances
11:  Calculate the weight:  $w_i = 1 / \sqrt{\sum_{i=1}^n (z - x_i)^2}, i \in K$ 
12:  Classify  $z$  in the majority class  $C(z) = \operatorname{argmax}_k \sum_{X_j \in N} w_i C(X_j, L_k)$ 
13: End

```

4.4. Naïve Bayes Algorithm

The Naïve Bayes Classifier is a widely used and effective algorithm for addressing classification problems. It utilizes the renowned Bayes Theorem to make predictions about the probability of a given data point belonging to a specific class. The classifier calculates the probability for each class and selects the class with the highest probability as the final result.

Bayes Theorem provides a mathematical framework for estimating the probability of an event, which can be expressed using Equation (5). By applying Bayes Theorem, the Naïve Bayes Classifier can make probabilistic inferences and classify data points based on their likelihood of belonging to different classes.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (5)$$

where $P(A)$ is the probability of class A being true, $P(B)$ is the probability of predictor B is true and $P(A | B)$ is the probability of class A given that predictor B is true. $P(B | A)$ is the probability of predictor B given that A is true.

The basic idea is train an NB classifier with the training data set first, and then predicts the location using the trained classifier with the current observed RSS values. For classifier training, the first step is converting each RSS sample in the training data set with the corresponding coordinate, and creates a classification dataset object with a list of data point objects. The second step is creating an NB classifier and trains it with the classification dataset object created in the first step.

For location prediction, convert the current observed RSS values to data point object first, and then use the trained classifier to classify the data point object. The result is a object which contains the probabilities of all locations in the training set, and the most likely function returns the index of the location with highest probability in the location list.

However, in the actual scenario, the current location will not always exactly be one of the training points, even if it is, the highest probability may not be actually high and several locations have the probability close to the highest probability.

Thus, the averaging concept in KNN and WKNN may be help to optimize the accuracy of NB. The probability values of each possible location with the same index of location are contained in the location list. The probability values are sorted by descending order, and the corresponding indexes of locations with high probability can be finding via probability value. The locations with probability different from the highest probability not greater than 10% are to be averaged. In addition, for the weighted averaging approach, the locations are weighted based on probability.

4.5. RSS-Based Neural Network Algorithm

Our RSS based Neural Network is computational models inspired by biological neural networks. They are composed of interconnected layers, including an input layer, an output

layer, and one or more hidden layers. Neurons within each layer are connected to neurons in the subsequent layer. Through the learning process using a given training set, the model assigns weights to these connections and adjusts them incrementally to improve the accuracy of predictions.

To implement the proposed RSS-based Deep Neural Network, the neural network architecture must be defined, specifying the number of neurons in the input, hidden, and output layers. Training and testing data can be imported from .csv files, and the number of input and output variables is provided to create the corresponding datasets. Once the neural network and datasets are established, the training process can commence.

Before training the neural network, parameters such as the maximum allowable error rate and the maximum number of iterations can be set. During the training phase, a graphical representation of the total network error is displayed, illustrating inputs, outputs, desired outputs, errors, and the total mean square error of the test dataset. This visualization aids in monitoring the network’s convergence and evaluating its performance.

4.6. Convolution Neural Network Algorithm

The Convolutional Neural Network (CNN) is a type of neural network that employs convolution as a distinct linear operation, as opposed to the conventional matrix multiplication, in at least one of its layers. This approach is particularly effective when processing data with a grid-like topology, such as image data or time series data. We utilized a pre-existing CNN model as a reference [24] and subsequently refined its parameters to enhance its performance, including the adjustment of filter sizes, feature maps, and number of CNN layers.

The room depicted in Figure 1 was partitioned into 1.2 m × 1.2 m blocks, with RSS collection being conducted at 28 designated training points. Multiple beacons typically detected each point, with all measurements captured. The complete labeled dataset comprised 2699 labeled data points, with 2159 data points utilized for training and 540 data points reserved for testing.

The original dataset consisted solely of RSSI readings from eight BLE beacons and the coordinates of each recording point. To facilitate the use of CNNs for feature extraction, we have transformed these data into an image-like format. Our testing area, originally divided into grids labeled A–J and 1–5, has been expanded to create a 10 × 10 grid representation, as illustrated in Figure 3. Notably, the original grid configuration was a 10 × 5 rectangle; for compatibility with CNN processing, we extended the Y-axis from 5 to 10 by padding the grid. The left side of Figure 3 displays this grid, with the RSSI values from each beacon plotted within the corresponding grid sections. To streamline processing with CNNs, the data have been normalized using a straightforward formula (as shown in Equation (6)) to scale the RSSI values to a real number between 0 and 1. In this scheme, 0 represents pure black and 1 represents pure white, with intermediate values depicted in varying shades of gray.

$$V_{pixel} = \begin{cases} \frac{RSSI}{-200} & \text{if RSSI is not empty} \\ 0 & \text{if RSSI is empty} \end{cases} \quad (6)$$

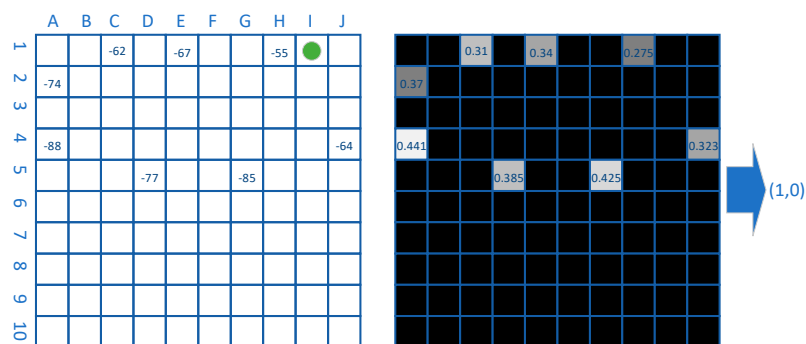


Figure 3. Mapping original data into a greyscale image.

Our CNN network architecture is illustrated in Figure 4.

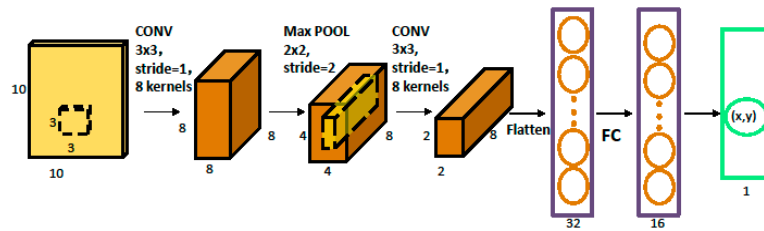


Figure 4. Proposed CNN Network Architecture.

The parameters of our CNN Architecture are set as follows: Input layer: [10, 10, 1] grayscale; 2 Convolution layers: number of the filters, respectively [8, 8], filter sizes: [3, 3] with valid padding; 1 Max-pooling Layer: number of filters [8, 8], filter sizes: [2, 2]; 1 Dense layer: 16 nodes; Output layer: 1 representing x and y coordinates; Activation function: ReLU. Additionally, other hyper-parameters, such as batch size, number of layers, and layer sizes, can be modified during the testing procedure to facilitate the attainment of optimal positioning performance.

The proposed algorithm encompasses distinct phases, namely the offline phase and the online phase. During the offline phase, RSSI measurements are acquired at various reference positions, yielding datasets denoted as $D_R = \{R_1; \dots; R_N\}$, where $R_i = \{r_{i1}; \dots; r_{iM}\}$, N represents the number of samples in the dataset, and M represents the number of BLE beacons. Subsequently, the dimensionality of the RSSI measurements undergoes reduction through the application of Principal Component Analysis (PCA) and Kalman filtering techniques for noise mitigation. Furthermore, a normalization process is conducted. Subsequent to these preprocessing steps, Convolutional Neural Network (CNN) architecture is employed for feature extraction, followed by the utilization of a Soft-max layer for classification-based learning. Consequently, a localization classification model is formulated. In the subsequent online phase, data preprocessing procedures, coupled with feature extraction techniques, pave the way for the application of the classification model to yield the ultimate estimation of the positional coordinates.

5. Experimental Analysis

5.1. Optimal Parameters Setting for Algorithms

5.1.1. Optimal K for KNN and WKNN

In the context of KNN and WKNN, there is no universal optimal value of K, as it is contingent upon a variety of factors, including minimizing the error rate while ensuring that all neighbors remain in proximity. Consequently, we tested several values of K, using the same training and testing dataset across all iterations. In Figure 5, the average distance across different K values ranging from 1 to 28 in KNN and WKNN. The outcomes indicate that the most proximal average distance was attained at $K = 10$. However, for WKNN, it is possible that a small K performs better than the K with minimum average distance in the case where not all K-nearest neighbors are close to the actual location.

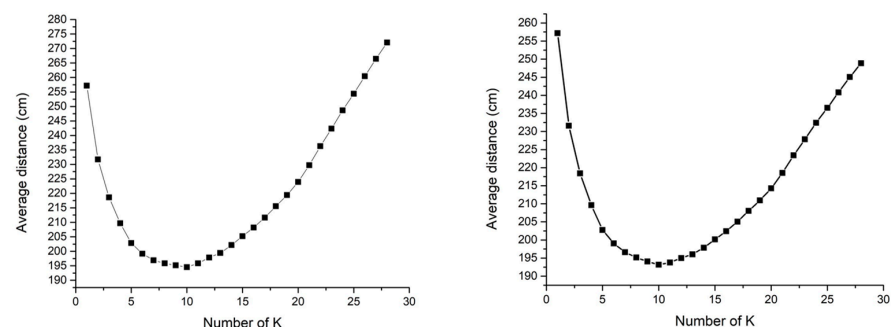


Figure 5. Average Error Distance for Different K in KNN (left) and WKNN (right).

Nonetheless, in the context of Weighted K-Nearest Neighbors (WKNN), a scenario arises wherein a smaller value of K might yield superior outcomes compared to the K-value exhibiting the minimal average distance. This divergence emerges particularly when the entirety of the K-nearest Neighbors fails to converge closely to the actual position. The comparative analysis depicted in Figure 6 showcases the disparity in outcomes across varying distances for K-values of 4 and 10. It is evident that when K equals 4, outcomes within proximity of 1 m tend to exhibit greater fidelity to the actual location. This trend prevails despite a higher count of results being situated at greater distances in contrast to the K-value of 10. Consequently, for the purpose of algorithmic juxtaposition, a value of $K = 10$ was adopted for both K-Nearest Neighbors (KNN) and Weighted K-Nearest Neighbors (WKNN), as it consistently yields the optimal overall performance across diverse K-values.

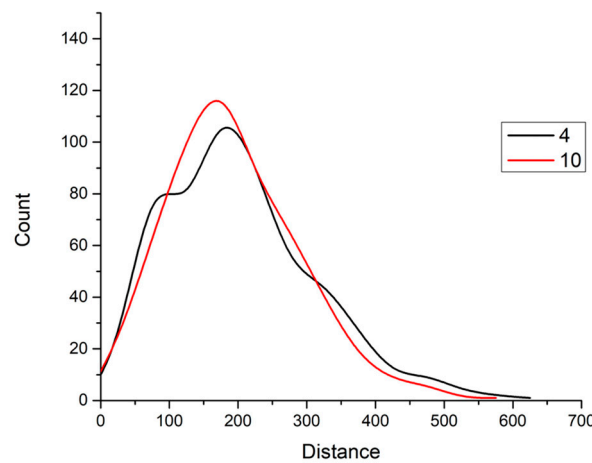


Figure 6. Result error distance performance of WKNN with $K = 4$ and 10.

5.1.2. Optimal Sampling of RSS Data of NB

The concept of averaging, as applied in KNN and WKNN by considering average RSS values within a time window, has demonstrated an optimization effect on the accuracy of NB in our experimental setup. To further evaluate the impact of averaging, Figure 7 presents a comparison of the number of results at different distances for three variants: (i) NB, (ii) averaged NB (ANB), and (iii) weighted averaged NB (WANB).

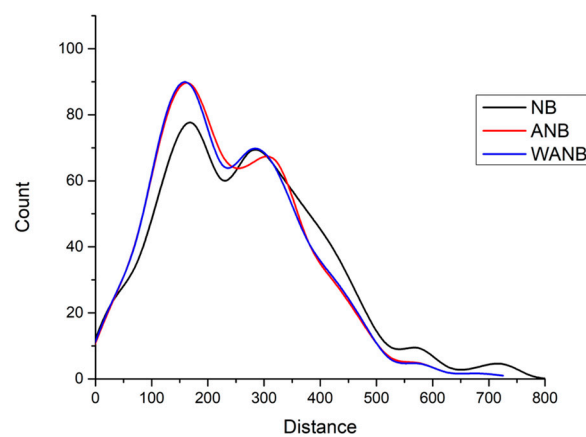


Figure 7. Comparisons error distance performance between NB, ANB, and WANB.

The results clearly indicate that the performance of ANB and WANB is significantly superior to that of the original NB algorithm without averaging. Moreover, WANB exhibits a slight advantage over ANB in terms of performance. Consequently, WANB was selected as the variant to be compared with other algorithms in our analysis.

5.1.3. Optimal the Hidden Neurons for RSS-Based NN

The provided histogram in Figure 8 compares the performance of two neural network variants—averaged Neural Network (avgNN) and non-averaged Neural Network (NotAvgNN)—by illustrating the frequency of estimated positions at various distance intervals from the true location. The histogram bars represent the count of estimated positions falling within specified distance ranges (measured in centimeters). Black bars denote avgNN, while red bars represent NotAvgNN.

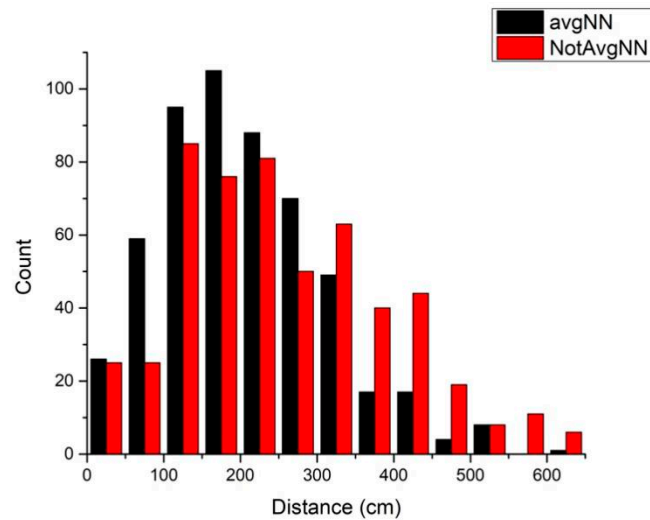


Figure 8. Comparison error distance performance between averaged and not averaged NN.

From the histogram, we observed a higher concentration of black bars at lower distance values, particularly within the 300 cm range. This pattern indicates that avgNN generally predicts positions closer to the actual location more frequently than NotAvgNN. The reduced height of the red bars within the same distance interval suggests that NotAvgNN has a lower frequency of close approximations to the true position. Therefore, the visualization demonstrates that incorporating an averaging technique in the neural network (avgNN) significantly improves the positioning accuracy by yielding a greater number of predictions nearer to the true location, compared to the variant without averaging (NotAvgNN). The data emphasize the benefit of the averaging approach in reducing the error in the position estimates produced by the neural network algorithm.

The determination of the number of neurons in the hidden layer of the neural network is an important consideration. In order to investigate this, Figure 9 presents a comparison of the average distance for different numbers of hidden neurons.

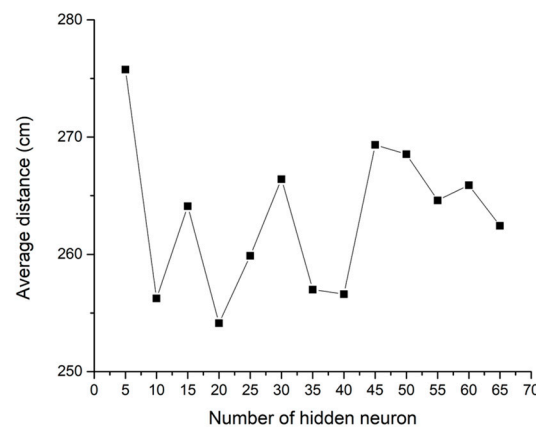


Figure 9. Average error distance for different hidden neurons in DNN.

Upon analysis, it is observed that the closest average distance is achieved when the number of hidden neurons is set to 20. Based on this finding, we set the number of hidden neurons in the hidden layer to 20 for the purpose of comparison with other algorithms.

5.1.4. Optimal Parameters of CNN

A. RSS measurement pre-processing

The RSS measurements were conducted at 28 training points. Each point was usually detected with multiple beacons, and RSS measurements were captured. The entire labelled data set contains 2699 labelled data points, in which 2159 data points were used for training and 540 data points were used for testing.

The application of (PCA) methodology serves to mitigate the inherent redundancy within the RSS measurements, while the Kalman filter technique was employed to temporally enhance the coherence of the RSS measurements. This pre-processing regimen, characterized by its composite implementation of PCA and Kalman filtering, is instrumental in preserving pertinent features while concurrently ameliorating the presence of extraneous noise and multifaceted interference, thereby engendering a discernible enhancement in the overall efficacy of the localization performance. Due to the significance of RSS information, it is imperative to uphold the information dimensionality inherent in the RSSI data during subsequent stages of network learning. Moreover, normalization procedures were applied to the RSS measurements to ensure their coherent treatment.

B. Feature extraction network parameters

The proposed CNN network with a multi-channel input configuration was harnessed to effectuate the extraction of pertinent features from the RSS measurements. The devised network architecture comprised a duo of convolutional layers, succeeded by a pair of max-pooling layers, culminating in a single flatten layer. The underlying convolutional operations served to extract intricate features intrinsic to the RSS measurements. The ensuing max-pooling stratum functions to selectively preserve salient attributes, concomitantly effecting a reduction in data dimensionality. This process, in turn, engendered a reduction in parameters and computational overhead, thereby serving as a constraint against overfitting. The configuration parameters governing the feature extraction module are detailed in Table 2. The input data, characterized by their ten-dimensional scale, undergoes transformation, yielding an RSS feature output of two-dimensional dimensionality.

Table 2. Feature extraction network parameters.

Parameter	Value
Input size	[10, 10]
Convolutional layers	[3, 3]
Max pooling layers	[2, 2]
Kernel size	3
Stride	1
Pooling	Max pooling
Pooling size	2
Output size	[1, 2]

C. Location classification learning network parameters

The network architecture parameters for the purpose of classification learning are comprehensively documented in Table 3. These parameters encompass a pair of fully connected layers, followed by a Softmax layer that serves as the classifier. The RSS feature vector was propagated through the fully connected layers, ultimately leading to the Softmax layer for the attainment of the classification outcome. The Rectified Linear Unit (ReLU) was designated as the activation function, selected for its efficacy in augmenting the nonlinear attributes inherent within the network. The Mean Squared Error (MSE) was adopted as the loss function, imparting a quantifiable measure of the dissimilarity between predicted and

actual values. In the context of network optimization, the Adam algorithm was enlisted. Furthermore, the network’s training regime incorporates a polynomial learning strategy, underpinned by a cyclically modulated adjustment of the learning rate, thereby facilitating dynamic convergence towards the global minimum loss.

Table 3. Classification learning network parameters.

Parameter	Value
Epochs	900
Batch Size	80
Loss function	MSE
Optimizer	SGD
Activation	ReLU
Learning schedule	Polynomial Decay
Learning rate	0.001

Set the optimal configuration of the number of epochs to 900 and the batch size to 80. The training process reveals the progression of the loss values, as depicted in Figure 10, wherein the lowest records at 0.58, and the mean loss stabilizes at 1.02. We take this mean loss as the average distance. Similarly, the mean squared error (MSE) is around 0.55, the lowest MSE observed at 0.54.

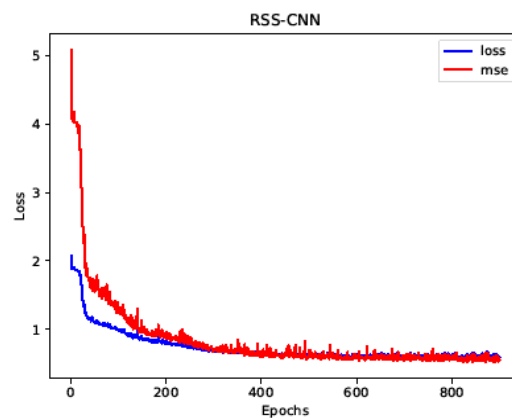


Figure 10. Loss and MSE.

5.2. Evaluation of Algorithms

The present study compares the performance of five algorithms, namely KNN, WKNN, NB, RSS-NN, and CNN. As depicted in Figure 11a, the average error distances of all five algorithms are observed to be below 2 m. The minimum error distance is observed for CNN at 1.22, which calculated from 1.02 by 1.2, while the maximum error distance is recorded for NB at 2.37 m. Notably, CNN exhibits the minimum error distance of 1.28 m, while KNN displays the largest average error distance, which is marginally greater than that of WKNN.

In Figure 11b, we illustrate a comparison of the distribution of results in different distance ranges for the five algorithms, namely KNN, WKNN, NB, RSS-NN, and CNN. The graph indicates that CNN performs the best, with approximately 20% of the results having a distance error of 0.5 m or less, and the vast majority of results having errors of less than 3 m. The histogram confirms that WKNN outperforms KNN, although each algorithm has its own set of strengths and weaknesses. For example, KNN and WKNN may be better suited for small areas with fewer reference points, while CNN may be better suited for larger areas with more reference points. However, each algorithm has its advantages and disadvantages. KNN and WKNN may be more suitable for small places with fewer reference points, whereas CNN may be more suitable for large places with more reference points.

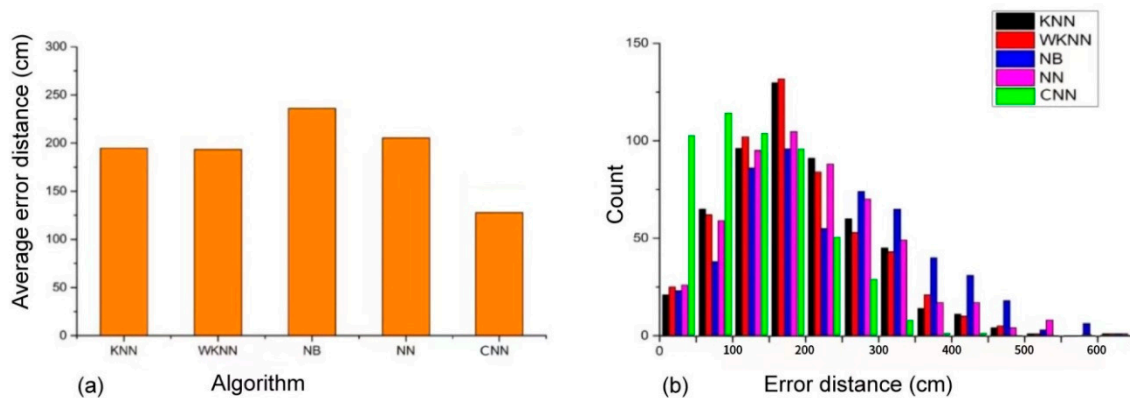


Figure 11. A comparison between the algorithms: (a) shows the average error distance, and (b) shows a histogram of error values. Generally, the average error performance with CNN is the best one in the five algorithms, and KNN and WKNN are better than the rest of the two algorithms, and this is confirmed by the histogram in (b).

However, a limitation of the KNN, WKNN, and NB algorithms lies in the size of the training set. As these algorithms require calculating the distance between the current RSS vector and the entire training set, their performance can be impacted when dealing with large training sets. This can lead to a significant increase in the time required to estimate the position of each test point. On the other hand, the performance of the NN algorithm is comparable to that of KNN and WKNN. One advantage of NN and CNN is its fast prediction time after the model is trained. However, it should be noted that determining the optimal number of hidden neurons in NN can be time-consuming. The more hidden neurons there are, the longer it takes to train the neural network model. For the CNN approach, the design of the network structure and setting the hyper parameters are crucial. Consequently, the KNN, WKNN, and NB algorithms may be more suitable for smaller areas with a limited number of reference points, while NN and CNN may be more suitable for larger areas with a higher density of reference points.

RSS fingerprint-based BLE positioning algorithms like KNN, WKNN, NB, NN, and CNN all face several drawbacks, including sensitivity to environmental changes which can significantly affect positioning accuracy, high initial setup and maintenance costs due to the need for extensive calibration, and susceptibility to signal multipath effects which distort signal strength readings. These methods also have scalability issues in large areas and can be computationally demanding. Moreover, sophisticated models like Neural Networks and CNN require careful tuning to avoid overfitting.

5.3. Comparison with Commercial Equipment

The aim of these tests was to assess the order of magnitude of results obtained previously, with some taken using commercial equipment. It would not be fair to call this a direct quantitative comparison as (i) the commercial equipment is expensive, so fewer beacons were used; (ii) a room adjacent to that tested previously had to be used (but was the same construction); and (iii) no detailed investigation of optimum locations for the commercial beacons or their optimum settings was carried out. However, it does illustrate well the difficulty of getting accurate results in the environment, although it is acknowledged that a more painstaking investigation of these settings could well have increased the accuracy significantly, but that was not the purpose of this work. Clearly, from the photograph in Figure 12, there is the potential for reflections and absorption of the radio signals, and the error values are a similar order of magnitude to the values we obtained in the experiments described in this section.

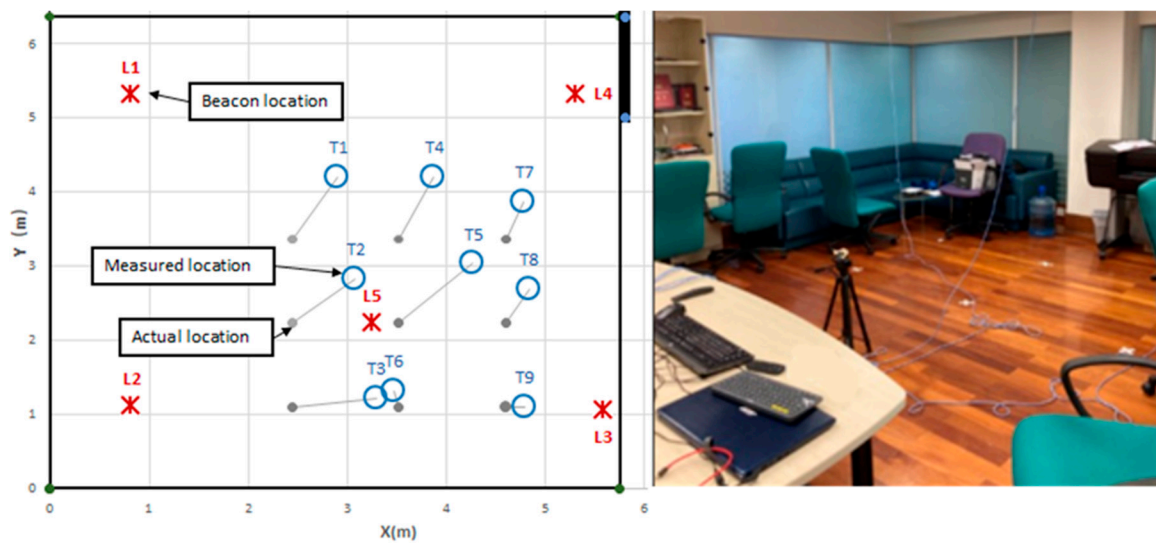


Figure 12. Results and environment for tests with commercial equipment.

The greatest disparity in distance error amounts to 1.1 m, manifested at position T5. Conversely, the most minimal discrepancy of 0.19 m transpires at position T9. The mean value of the distance errors stands at 0.68 m, as observed in Table 4.

Table 4. Comparison of the actual and measured positions.

	T1	T2	T3	T4	T5	T6	T7	T8	T9
Actual	(2.45, 3.36)	(2.45, 2.23)	(2.45, 1.1)	(3.52, 3.36)	(3.52, 2.23)	(3.52, 1.1)	(4.6, 3.36)	(4.6, 2.23)	(4.6, 1.1)
Measured	(2.9, 4.19)	(3.08, 2.83)	(3.29, 1.21)	(3.87, 4.19)	(4.27, 3.03)	(3.47, 1.32)	(4.78, 3.86)	(4.83, 2.69)	(4.79, 1.09)

The algorithms used in this commercial equipment are proprietary, but the understanding of the authors is that both RSS and AoA are included. This led us to investigate the performance of an AoA-based system and how to combine the results from RSS and AoA.

6. Conclusions

In this paper, we investigated the positioning capability of BLE using radio signal strength and compared the performance of five RSS-based positioning algorithms in a typical indoor environment. Five popular RSS based positioning algorithms were evaluated on their performance in a typical indoor environment in addition to a commercial indoor positioning system (the Quuppa Intelligent Locating System). The effectiveness of the five algorithms was evaluated for improving the localization accuracy. The results demonstrate that our CNN method outperforms KNN, WKNN, NB, and RSS-NN.

However, there are still many issues that should be considered and addressed in future work. In a real environment there will be objects absorbing the radio signal, objects reflecting it and dynamic changes in these impairments. Since the perceived RSS may be affected differently by these impairments, we considered how they may be combined to provide more accurate positioning, especially when using AoA over larger distances where small errors in the angle can lead to fairly large errors in position. This is a new approach and the work will be the first step in evaluating the concept, although it does seem clear from the results of the training set that it has potential. To improve the accuracy of the positioning, we also consider combining RSS with AoA in the presence of these impairments.

Author Contributions: Investigation, X.Y.; Data curation, I.-C.W.; Writing—original draft, C.W.; Writing—review & editing, Y.W.; Supervision, W.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work is funded by Macao Polytechnic University under grant No. RP/ESCA-08/2021.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bai, L.; Ciravegna, F.; Bond, R.; Mulvenna, M. A low cost indoor positioning system using bluetooth low energy. *IEEE Access* **2020**, *8*, 136858–136871. [CrossRef]
2. Li, H.; Trocan, M.; Galayko, D. Virtual fingerprint and two-way ranging-based Bluetooth 3D indoor positioning with RSSI difference and distance ratio. *J. Electromagn. Waves Appl.* **2019**, *33*, 2155–2174. [CrossRef]
3. Subhan, F.; Ahmed, S.; Haider, S.; Saleem, S.; Khan, A.; Ahmed, S.; Numan, M. Hybrid Indoor Position Estimation using K-NN and MinMax. *KSII Trans. Internet Inf. Syst.* **2019**, *13*, 9. Available online: <http://itiis.org/journals/tiis/digital-library/manuscript/file/22203/TIISVol13No9-5.pdf> (accessed on 31 October 2023).
4. Urano, K.; Hiroi, K.; Yonezawa, T.; Kawaguchi, N. An end-to-end BLE indoor location estimation method using LSTM. In Proceedings of the 2019 Twelfth International Conference on Mobile Computing and Ubiquitous Network (ICMU), Kathmandu, Nepal, 4–6 November 2019; pp. 1–7. Available online: <https://ieeexplore.ieee.org/abstract/document/9006638/> (accessed on 31 October 2023).
5. Weerasinghe, Y.S.P.; Maduranga, M.W.P.; Dissanayake, M.B. RSSI and feed forward neural network (FFNN) based indoor localization in WSN. In Proceedings of the 2019 National Information Technology Conference (NITC), Colombo, Sri Lanka, 8–10 October 2019; pp. 35–40. Available online: <https://ieeexplore.ieee.org/abstract/document/9114515/> (accessed on 6 November 2023).
6. Choi, J.; Choi, Y.S.; Talwar, S. Unsupervised learning technique to obtain the coordinates of Wi-Fi access points. In Proceedings of the 2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN), Pisa, Italy, 30 September–3 October 2019; pp. 1–6.
7. Ding, H.L.; Ng, W.W.; Chan, P.P.; Wu, D.L.; Chen, X.L.; Yeung, D.S. RFID indoor positioning using RBFNN with L-GEM. In Proceedings of the 2010 International Conference on Machine Learning and Cybernetics, Qingdao, China, 11–14 July 2010; Volume 3, pp. 1147–1152. [CrossRef]
8. Cheng, C.H.; Syu, S.J. Improving area positioning in ZigBee sensor networks using neural network algorithm. *Microsyst. Technol.* **2021**, *27*, 1419–1428. [CrossRef]
9. Le, A.T.; Tran, L.C.; Huang, X.; Ritz, C.; Dutkiewicz, E.; Phung, S.L.; Bouzerdoun, A.; Franklin, D. Unbalanced hybrid AOA/RSSI localization for simplified wireless sensor networks. *Sensors* **2020**, *20*, 3838. [CrossRef] [PubMed]
10. Yang, C.; Shao, H. Wi-Fi-based indoor positioning. *IEEE Commun. Mag.* **2015**, *53*, 150–157. [CrossRef]
11. Duong, N.; Trinh, V.; Dinh, T. Bluetooth Low Energy Based Indoor Positioning on iOS Platform. In Proceedings of the 2018 IEEE 12th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoc), Hanoi, Vietnam, 12–14 September 2018; pp. 57–63. [CrossRef]
12. Xu, H.; Ding, Y.; Li, P.; Wang, R.; Li, Y. An RFID indoor positioning algorithm based on Bayesian probability and K-nearest neighbor. *Sensors* **2017**, *17*, 1806. [CrossRef] [PubMed]
13. Poulouse, A.; Eyobu, O.S.; Kim, M.; Han, D.S. Localization error analysis of indoor positioning system based on UWB measurements. In Proceedings of the 2019 Eleventh International Conference on Ubiquitous and Future Networks (ICUFN), Zagreb, Croatia, 2–5 July 2019; pp. 84–88.
14. *IEEE 802.15.4-2015; Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)*. IEEE: New York, NY, USA, 2015.
15. Chen, Y.; Ren, Z.; Han, Z.; Liu, H.; Shen, Q.X.; Wu, Z. LED based high accuracy indoor visible light positioning algorithm. *Optik* **2021**, *243*, 166853. [CrossRef]
16. Lee, S.; Chae, S.; Han, D. ILoA: Indoor localization using augmented vector of geomagnetic field. *IEEE Access* **2020**, *8*, 184242–184255. [CrossRef]
17. Bai, L.; Yang, Y.; Chen, M.; Feng, C.; Guo, C.; Saad, W.; Cui, S. Computer vision-based localization with visible light communications. *IEEE Trans. Wirel. Commun.* **2021**, *21*, 2051–2065. [CrossRef]
18. Shang, S.; Wang, L. Overview of Wi-Fi fingerprinting-based indoor positioning. *IET Commun.* **2022**, *16*, 725–733. [CrossRef]
19. Alhulayil, M.; Lopez-Benitez, M. LTE/Wi-Fi coexistence in unlicensed bands based on dynamic transmission opportunity. In Proceedings of the 2020 IEEE Wireless Communications and Networking Conference Workshops (WCNCW), Seoul, Korea, 6–9 April 2020; pp. 1–6.
20. Gomez, C.; Oller, J.; Paradells, J. Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology. *Sensors* **2012**, *12*, 11734–11753. [CrossRef]
21. Suryavanshi, N.B.; Viswavidharan Reddy, K.; Chandrika, V.R. Direction finding capability in bluetooth 5.1 standard. In Proceedings of the Ubiquitous Communications and Network Computing: Second EAI International Conference, Bangalore, India, 8–10 February 2019; pp. 53–65. [CrossRef]
22. Kriz, P.; Maly, F.; Kozel, T. Improving indoor localization using bluetooth low energy beacons. *Mob. Inf. Syst.* **2016**, *2016*, 2083094. [CrossRef]

23. Jeon, K.E.; She, J.; Soonsawad, P.; Ng, P.C. BLE Beacons for Internet of Things Applications: Survey, Challenges, and Opportunities. *IEEE Internet Things J.* **2018**, *5*, 811–828. [[CrossRef](#)]
24. Pydipaty, R.; George, J.; Selvaraju, K.; Saha, A. Improving the performance of deep learning for wireless localization. *arXiv* **2020**, arXiv:2006.08925. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.