

Article

A Room-Level Indoor Localization Using an Energy-Harvesting BLE Tag

Yutao Chen ^{1,†} , Yun Wang ^{2,3,*} and Yubin Zhao ^{1,*,†}

¹ School of Microelectronics Science and Technology, Sun Yat-Sen University, Zhuhai 519082, China; chenyt328@mail2.sysu.edu.cn

² Industry-Education Integration Development Center, Changchun Technical University of Automobile, No. 1777 Xinhongqi Street, Changchun 130051, China

³ College of Chemistry and Molecular Engineering, Peking University, No. 292, Chengfu Road, Haidian District, Beijing 100871, China

* Correspondence: xiaowangtongxue@pku.org.cn (Y.W.); zhaoyb23@mail.sysu.edu.cn (Y.Z.)

† These authors contributed equally to this work.

Abstract: Energy-efficient and cost-effective localization systems are attractive for large-scale tracking and localization of goods. In this paper, we propose a room-level localization system using energy-harvesting BLE tags to track the targets. We introduce the Dempster–Shafer (D–S) evidence theory combined with fingerprinting technology for location estimation. To reduce the estimation complexity, we divide the indoor environment into clear areas and fuzzy areas. The D–S algorithm is employed to locate the target in the clear areas when the targets are only detected by the anchor nodes within a single room. Conversely, fuzzy areas are characterized by RSSI signals detected by anchor nodes across multiple rooms. Then, the system integrates fingerprint matching to ensure superior positioning accuracy across the deployment. Extensive experiments demonstrate that the proposed system maintains a room-level positioning accuracy above 99% under standard test conditions within an area of approximately 2000 m² with lots of rooms.

Keywords: indoor localization; energy harvesting; BLE; Dempster–Shafer evidence theory



Citation: Chen, Y.; Wang, Y.; Zhao, Y. A Room-Level Indoor Localization Using an Energy-Harvesting BLE Tag. *Electronics* **2024**, *13*, 4493. <https://doi.org/10.3390/electronics13224493>

Academic Editors: Fabio Corti and Francesco Giuseppe Della Corte

Received: 20 September 2024

Revised: 29 October 2024

Accepted: 13 November 2024

Published: 15 November 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

Over the last decade, a wide array of indoor positioning technologies has emerged, and their widespread acceptance and application across society are increasing [1–3]. Currently, the most utilized indoor positioning technologies encompass Wi-Fi [4–6], Bluetooth low energy (BLE) [7–9], and ultra-wide band (UWB) [10–13]. UWB technology, known for its high accuracy and strong anti-interference capabilities, has received considerable attention for indoor positioning, though it is limited by its inability to cover large areas and high costs associated with hardware and deployment [14,15]. Wi-Fi-based indoor positioning benefits from extensive existing infrastructure, which can help reduce costs but suffers from high power consumption, short battery life, and increased susceptibility to interference from other Wi-Fi devices [16,17]. After the introduction of BLE technology, it has drawn wide attention in the field of indoor positioning because of its low power consumption, widespread range, and low cost [18]. Nevertheless, despite the considerable advancements made by these technologies, the domain of indoor positioning continues to encounter a number of pivotal challenges, including the trade-off between accuracy and cost, energy efficiency, environmental adaptability, and scalability. The DSFP algorithm proposed in this paper seeks to address these challenges and achieve a high-precision, low-cost, and energy-efficient indoor positioning solution by integrating D–S evidence theory and fingerprint positioning technology.

Within indoor positioning research, primary observational metrics include time of arrival (TOA) [19], time difference of arrival (TDOA) [20], angle of arrival (AOA) [21],

and received signal strength indicator (RSSI) [18]. Although TOA, TDOA, and AOA can achieve high-precision positioning, they have high requirements for equipment, complex deployment, and high cost [13]. Due to these factors, they may not be the best fit for applications that require tracking personnel or managing logistics, where cost-effectiveness and ease of implementation are crucial. In the management of fixed assets, positioning tags attached to equipment allow for the real-time tracking and management of each item. This capability enables the precise localization of each asset, thereby automating inventory processes [22]. For instance, it is possible to verify whether tool carts are correctly positioned in warehouses or to locate various devices in hospitals that are essential for delivering medical services and conducting clinical care, which supports emergency care management. Additionally, the system can monitor the movements of portable assets; for example, it can trigger alerts when a tool cart exits the premises to ensure that it is used only within authorized areas [23]. In such scenarios, we only need to estimate the location of the positioning target at the room level to meet the requirements at the lowest cost.

The goal of this paper is to achieve room-level accuracy positioning and tracking. RSSI has the advantages of low implementation cost, low equipment requirements, and wide applicability [24]; thus, it is suitable to use RSSI for the observations. However, existing RSSI-based positioning systems encounter a number of challenges, including signal instability resulting from environmental changes, positioning errors caused by multipath effects, and the difficulty of maintaining high accuracy in large-scale deployments. For the room-level indoor localization problem, the target may frequently be estimated in a wrong room, because the coverage area of an anchor is not as same as the actual room. In this case, the estimation accuracy is not improved, even with a higher accurate localization algorithm, in which the MSE is rather small.

In this paper, we propose a Bluetooth-based indoor positioning system that enables room-level localization and real-time tracking of targets. The BLE target consists of a solar panel which supports the energy-harvesting capability and long-term operation. This system requires only a single anchor node device in each room to facilitate room-level localization of BLE tags, eliminating the need for complex and expensive configurations. The primary contributions of this work are as follows:

- (1) We developed a cost-effective real-time indoor positioning and tracking system using Artificial Intelligence of Things (AIoT) devices, capable of synchronously localizing multiple energy harvesting BLE tag devices within complex environments. This system integrates hardware, databases, and algorithms to facilitate a highly customized platform. It features a front-end interface connected to the database, enabling the real-time display of positions and identification of different targets across various rooms.
- (2) We introduce an algorithm named DSFP, which combines the D–S theory of evidence and fingerprint-based positioning to achieve efficient and accurate location estimation. DSFP divides the indoor environment into clear areas and fuzzy areas. The clear areas indicate that each room is equipped with a unique anchor. For fuzzy areas, it is difficult to classify the anchors into the specific rooms. The D–S algorithm is employed for positioning within clear areas. For fuzzy areas, the fingerprinting technique is utilized to ensure reliable localization. The D–S theory has been applied to represent and combine conflicting RSSI information for positioning [25]. The implementation concept of the D–S algorithm is to regard each room as a hypothesis, and the measured value of the target RSSI in each room as evidence. The confidence level is set according to the distance and area size between different rooms. The algorithm calculates the probability of each hypothesis based on the confidence level and the evidence of the target in each room, and the hypothesis with the maximum probability value is used as the position estimation result. For fingerprint-based positioning, we evaluated and compared three machine learning algorithms: k-nearest neighbors (KNN), random forest (RF), and support vector machine (SVM).

Our proposed system is evaluated via extensive experiments. The results demonstrate that the DSFP can achieve highly accurate room-level estimation, maintaining positioning accuracy above 99% under standard test conditions.

This paper is organized as follows: Section 2 presents an overview of the various methodologies and technologies employed to achieve room-level positioning. Section 3 provides an overview of the system architecture. Section 4 delves into the methods and algorithms employed for location estimation. Section 5 presents the experimental results that validate the proposed scheme. Finally, Section 6 summarizes the conclusions drawn from this work.

2. Related Work

Among the various indoor positioning technologies, RSSI-based methods have gained widespread adoption due to their ease of implementation and cost-effectiveness [18]. These systems typically employ one of following three methods for positioning:

Ranging-based methods: This approach leverages the principle that radio signal strength is degrading with the increased propagation distance. By measuring the received signal strength, an estimate of the distance between the device and the Bluetooth beacon can be obtained, which is then used to calculate the device's position [26]. Rida et al. proposed an indoor positioning system with an average error ranging from 0.5 to 1 m [27]. This system operates by selecting the three anchor nodes with the strongest signal strength when a target enters their coverage area. The target's location is then estimated using a trilateration algorithm developed by the authors. The effectiveness of this method relies on a model that accounts for transmitter power and environmental path loss. However, the complexity of indoor environments, particularly the presence of multipath effects, can significantly reduce positioning accuracy. Furthermore, achieving room-level accuracy using trilateration necessitates at least three anchor nodes, which increases deployment costs. Additionally, the inherent instability of RSSI signals makes real-time target tracking challenging [28].

Fingerprint-based methods: This approach involves two distinct phases: offline training and online estimation. During the offline phase, RSSI measurements are collected at various points within the target area to construct a fingerprint database, often referred to as a "radio map". In the online phase, real-time RSSI measurements are compared against the radio map to estimate the user's location [1]. Fingerprint-based methods offer several advantages over ranging techniques. They eliminate the need for complex infrastructure and, with appropriate fingerprint matching algorithms, exhibit greater resilience to temporal fluctuations in RSSI. However, deploying these systems across large areas necessitates extensive upfront effort for fingerprint data collection. Additionally, any environmental changes may alter the radio map, requiring recalibration or retraining [29]. Akram et al. presented a framework for room-level positioning using Wi-Fi fingerprinting and a random forest ensemble classifier [6]. Their approach achieved an accuracy of 89% when validated with a Wi-Fi fingerprint dataset. Lazaro et al. developed a room-level positioning system based on LoRa signals [30]. They employed the Linear Discriminant Analysis (LDA) algorithm and achieved a notable accuracy of 89.7%.

Proximity positioning methods: A target device is simply identified as being within the vicinity of an anchor node if its RSSI value exceeds a predefined threshold when it enters the anchor node's range. Kyritsis et al. developed a room-level positioning algorithm based on thresholding, where the threshold calculation takes into account both the RSSI measurements and the geometric shape of the room containing the beacon [31]. Their algorithm specifically focused on analyzing and refining the boundary points between rooms, resulting in an impressive overall accuracy of 90.1%. However, the algorithm prioritizes larger rooms, leading to improved accuracy for most locations but at the expense of reduced accuracy in smaller rooms. Bianchi et al. introduced a method based on the receiver operating characteristics (ROCs) to assess the optimal placement of anchor nodes, the number of RSSI measurements required, and the appropriate threshold values [32]. Al-

though the system implementation is relatively complex, it achieves a remarkable accuracy rate of up to 98%.

After reviewing the above positioning systems and algorithms, we found that there are some challenges with the existing positioning systems (as summarized in Table 1). In systems using fingerprint recognition (scene analysis), it requires the creation of a comprehensive fingerprint map (radio map) of the target environment, which is a labor-intensive and time-consuming process. Systems using multilateral positioning algorithms require higher deployment costs and cannot achieve real-time positioning. Some systems use proximity algorithms in conjunction with fingerprint recognition algorithms but cannot achieve high-accuracy position estimation.

Table 1. Comparison of indoor positioning methods.

Method	Advantages	Disadvantages	Accuracy
Range-based	Simple implementation	Highly affected by multipath effects	0.5–1 m error
Fingerprint-based	Strong environmental adaptability	Requires extensive preliminary work	89%
Proximity-based	Low computational complexity	Lower precision	90.1–98%
DSFP	High precision, low cost, energy efficient	Requires fingerprint database initialization	Close to 100%

The DSFP method proposed in this paper combines the advantages of D–S theory and fingerprint positioning. While maintaining high accuracy, it reduces system complexity and energy consumption, providing a new and effective solution for indoor positioning. Our proposed system boasts several advantages, including room-level accuracy, cost-effectiveness, ease of maintenance, and real-time positioning capabilities. By leveraging existing infrastructure and Bluetooth beacons, we achieve a positioning accuracy above 99% within each room. Furthermore, we evaluate the impact of sample size and computational complexity on the training of three machine learning algorithms: KNN, RF, and SVM. We also compare the accuracy of the D–S algorithm alone against the combined D–S and fingerprint approach employed in DSFPs. Lastly, we conduct a moving trajectory tracking test to assess the system’s performance in dynamic scenarios.

3. System Architecture

The system comprises four primary components: hardware, database, algorithmic processing, and frontend interface, which are illustrated in Figure 1. The hardware configuration consists of target nodes and anchor nodes. The BLE target nodes, which are also called tags, are equipped with solar panels for energy harvesting and continuously providing the signals to the anchors. The anchors are typically the AIoT nodes which can detect the signals from the BLE targets and process the data locally. The database component stores not only the transmitted data from the anchor nodes, including each node’s ID and timestamps, but also the calculated RSSI values from target nodes. This process involves decoding the beacon data and applying the DSFP algorithm to ascertain precise locations in the following process. Finally, the frontend interface retrieves and displays these location data from a MySQL database, offering real-time and historical tracking of target positions as well as compiling information about targets encountered in each room.

The experimental deployment overview illustrates the DSFP positioning system, as depicted in Figure 2. In each room, an anchor node is installed to facilitate the localization of several undetermined targets within the vicinity. These nodes capture and transmit signal data to a centralized database via Wi-Fi APs. Subsequently, the DSFP algorithm is executed on a computer, processing these data to derive accurate positional estimates for various targets. These results are then presented on the user interface.

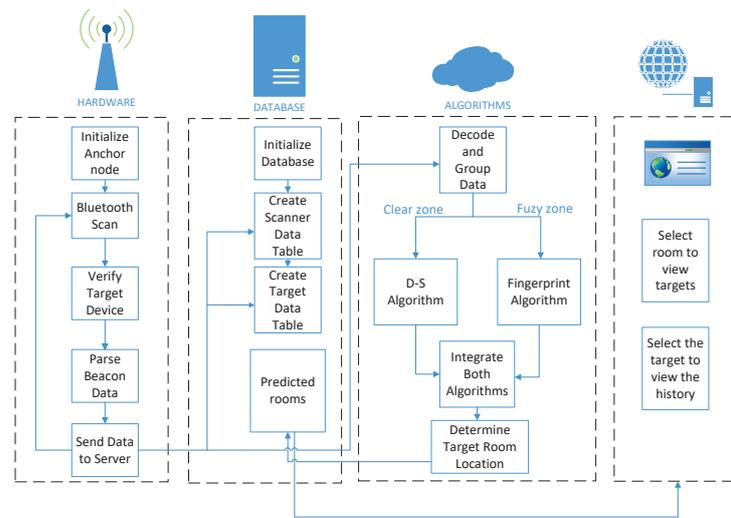


Figure 1. The overall system processing flow.

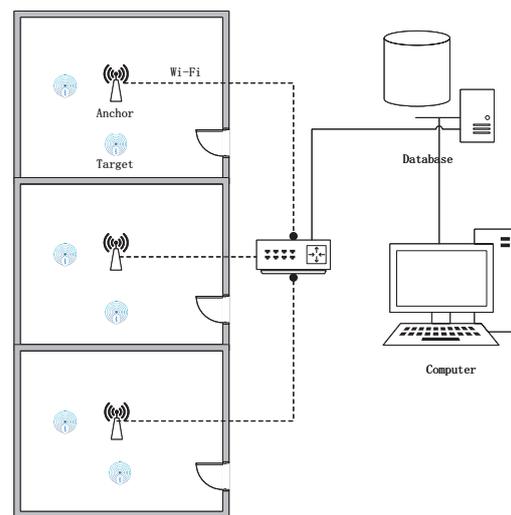


Figure 2. System deployment architecture.

Given that signals can be detected via various anchors in the complicated indoor environment, we classify the environment into clear areas and fuzzy areas. Fuzzy areas indicate the scenarios where a target is located at the boundary between Room A and Room B as depicted in Figure 3. In this case, the RSSI values received from the target in both rooms are similar. Relying solely on the anchor node that captures the strongest signal for determining the target’s location can result in significantly inaccurate estimation. This exemplifies the need for more sophisticated approaches in areas where signal overlap complicates straightforward location estimation. On the contrary, clear areas indicate that each anchor can only detect the signals from solo specific room. In this case, the D–S theory can be directly used to locate the target to the related room. To more clearly distinguish between clear areas and fuzzy areas, we define them as shown in Table 2:

Table 2. Definition of clear and fuzzy areas.

Area Type	Definition	Characteristics	Positioning Method
Clear Area	Area where the target is mainly detected by anchor nodes within a single room	RSSI values from a single anchor are significantly stronger than others	Direct use of D-S algorithm for position estimation
Fuzzy Area	Area where the target is detected by anchor nodes from multiple rooms with similar signal strengths	RSSI values from multiple anchors are similar, making it difficult to directly determine the target's room	Combination of D-S algorithm and fingerprint positioning technique for more accurate position estimation

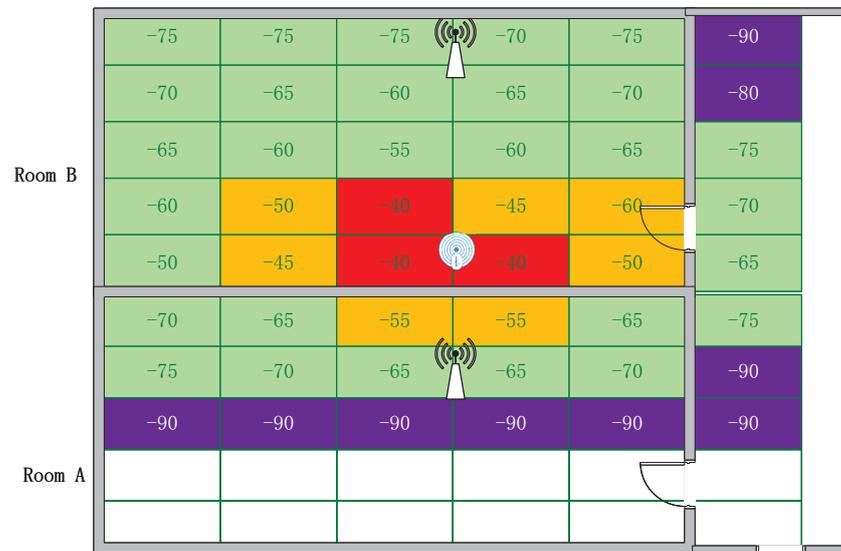


Figure 3. Illustrations of clear areas and fuzzy areas.

3.1. Energy Harvesting BLE Target Node

The target node consists of a solar panel as an energy harvesting module, which is shown in Figure 4. The signals are periodically transmitted via the Bluetooth module, which is depicted in Figure 4c. The principle of the energy harvesting module circuit is shown in Figure 5, and the real component is depicted in Figure 4a. The main working procedure of the target is as follows: when the toggle switch is set to the output gear, the super capacitor is used as the input voltage of the circuit, and when the voltage is greater than 1.8 V, the buck/boost converter TPS63900 starts to work. By adjusting the resistance value of RCFG1 and RCFG2, the output voltage is set to 3.3 V. When the toggle switch is set to the charging gear, the device focuses on charging the super capacitor by collecting solar energy. The solar panel (Figure 4b) measures 37 mm × 22 mm × 1 mm for capacitor charging. The Bluetooth module ATK-MW579D is debugged through the serial port, and its working mode is set to iBeacon mode, while the identification markers Major and Minor are set to broadcast Bluetooth signals at a frequency of 1 Hz.

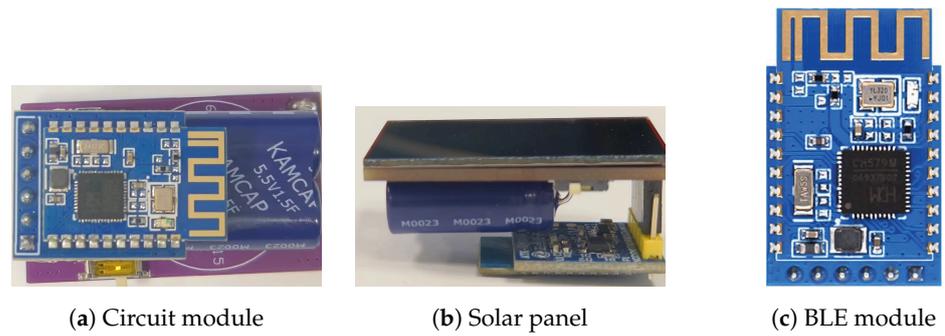


Figure 4. Target node module.

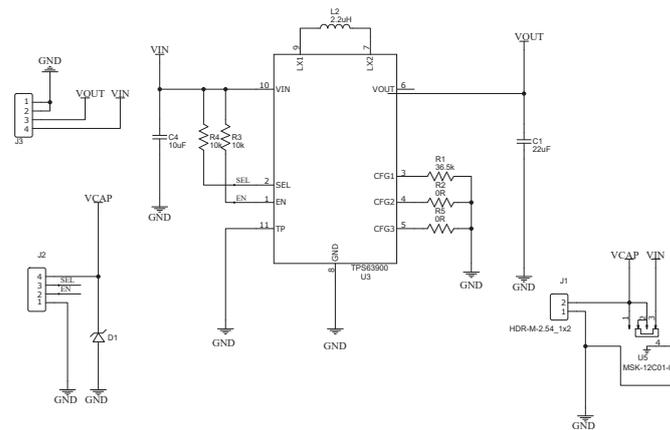


Figure 5. Energy-harvesting circuit schematic.

3.2. Anchor Node

Here, we use the ESP32-C3 development board, which is a typical AIoT programmable node platform, as the anchor node with the objective of scanning and recognizing the Bluetooth data broadcast by the target node. The anchor node contains the signal detection, processing, and communication capabilities. The deployment of an anchor node in each room allows for the acquisition of information pertaining to the target node currently situated therein. The anchor nodes situated in all rooms collectively form a sensor network, the purpose of which is to obtain information regarding the target’s movements. The workflow of the anchor node is illustrated in Figure 6. Initially, the parameters for connecting to Wi-Fi are configured, and then the scanning process for Bluetooth devices is initiated. Upon detecting a Bluetooth signal that matches the iBeacon data format, the Major and Minor values are examined to ascertain whether the signal originates from the target node that has been previously deployed. The data format is illustrated in Table 3.

Table 3. BLE target data format.

Field Name	Size	Description
UUID	16 bytes	A unique identifier for the device or application.
Major	2 bytes	Further specifies a specific iBeacon and use case. For example, this could define a sub-region within a larger region defined by the UUID.
Minor	2 bytes	Allows further subdivision of region or use case.

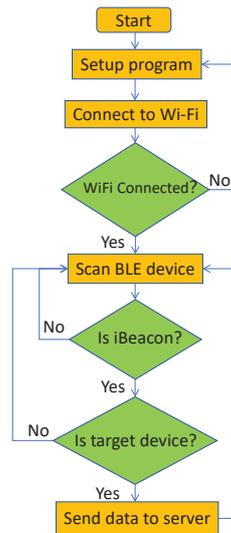


Figure 6. Flow chart of anchor node.

3.3. Database Component

We design a distributed data collection and processing system based on the MQTT protocol, which consists of an anchor node, an MQTT Broker, and a back-end server for collecting, transmitting, and processing data from target nodes in real-time.

The anchor node, as an MQTT client, encapsulates the collected target node data into MQTT messages and publishes them to predefined custom topics through the MQTT Broker. The MQTT Broker receives these messages and routes them to corresponding subscribers based on subscription relationships. The backend server subscribes to the relevant topics through the MQTT client to receive real-time anchor node data and processes them. The processed data are stored by the server in the `iBeacon_xx` database table, which is used by the front-end to retrieve the historical trajectory of the target. Subsequently, for each anchor identified by its unique ID, a dedicated table named `scanner_xxx` is created for storing and retrieving the MAC address and RSSI value of the detected targets in each room, and the `iBeacon_time` table serves as a dynamic table recording the last time when each target is scanned by the anchor node. This timestamp information is used by the front-end to display the target's path timing in different rooms. The `predicted_rooms` table stores predicted room information including timestamps, addresses, anchor nodes, and algorithm sources, which are used for display and discrimination acquired by the front-end. The relationship of the data tables is as shown in in Table 4.

Table 4. Database table relationships.

<code>ibeacon_xx</code>	Stores data of a specific targets device (xx represents the device ID)
<code>ibeacon_scannerxxx</code>	Stores data of a specific anchor (xxx represents the anchor ID)
<code>ibeacon_time</code>	Stores the time when the target was last scanned
<code>predicted_rooms</code>	Stores predicted room information, including timestamp, address, scanner, and source

4. DSFP

After collecting the RSSI data from the anchors, we employ the DSFP algorithm which combines the D–S algorithm and fingerprinting localization for the entire area. The DSFP algorithm uses the D–S algorithm for clear areas while employing fingerprinting localization for targets in fuzzy areas.

The DSFP algorithm possesses environmental adaptability. It divides the indoor space into distinct clear areas with obvious signal characteristics and complex fuzzy areas,

and then adopts the optimal positioning strategy accordingly. This dynamic area division method enables the algorithm to flexibly respond to different environmental features. DSFP achieves a balance between accuracy and efficiency. The algorithm uses the computationally efficient D–S algorithm in clear areas, while employing more precise fingerprinting technology in signal-complex fuzzy areas. This ensures overall positioning accuracy while avoiding unnecessary computational overhead. The algorithm is adept at handling uncertainty. By introducing D–S theory, DSFP can effectively manage uncertainty information in RSSI measurements, thereby enhancing the reliability of positioning results. DSFP demonstrates good scalability, making this method easy to extend to large, multi-room complex environments.

4.1. Dempster–Shafer Theory in Indoor Positioning

D–S evidence theory is employed to aggregate multiple sources of evidence to derive more reliable conclusions, especially useful in environments where individual evidence sources may not be entirely dependable. The D–S algorithm is based on the D–S theory reasoning, where we consider the RSSI values of the targets captured by the anchor nodes in each room as independent evidence, and using the evidence aggregation rules of the theory, the algorithm combines the signal strengths of the individual anchor nodes and the room area characteristics. By deploying a decision rule for the trust function, it constructs a probability profile for each room. The D–S algorithm then determines the estimated location of the beacon by selecting the room with the highest probability of existence. The steps are as follows:

4.1.1. Define the Recognition Framework

Different rooms are considered as different hypotheses and the recognition framework is the set of all rooms $\mathcal{L} = \{L_1, L_2, \dots, L_n\}$.

4.1.2. Assign Quality Function

For each anchor point S_j , the RSSI values within a time window are accumulated and denoted as E_j^{sum} . The accumulated RSSI value E_j^{sum} is then multiplied by the room area A'_j to obtain the un-normalized quality value for each room as illustrated in Equation (1).

$$m_j(L_j) = \frac{E_j^{\text{sum}} \cdot A'_j}{\sum_{i=1}^{|\mathcal{L}|} E_i^{\text{sum}} \cdot A'_i}, \quad (1)$$

where

$m_j(L_i)$: Belief mass for location L_i based on evidence from anchor S_j ,

A'_i : Normalized room area for location L_i ,

E_{ij} : RSSI value received at location L_i from anchor S_j .

4.1.3. Evidence Fusion and Decision Making

The mass functions of all anchor points are merged to obtain the final mass function m :

$$m(L_i) = \frac{1}{1 - K} \sum_{\substack{A, B \subseteq \mathcal{L} \\ A \cap B = \{L_i\}}} m_1(A) \cdot m_2(B), \quad (2)$$

where

$m(L_i)$: Combined belief for location L_i ,

m_1 and m_2 : Two mass functions to be combined,

$K = \sum_{\substack{A, B \subseteq \mathcal{L} \\ A \cap B = \emptyset}} m_1(A) \cdot m_2(B)$ represents the conflict factor.

Then, the room with the largest mass value is selected as the final localization result:

$$\hat{L} = \arg \max_{L_i \in \mathcal{L}} m(L_i). \quad (3)$$

The detailed algorithm of D–S approach is illustrated in Algorithm 1.

Algorithm 1 Indoor localization using a simplified D–S approach with room size

Require: \mathcal{L} : Set of possible locations

Require: \mathcal{S} : Set of anchors

Require: E : RSSI values received at each location from the target

Require: A' : Normalized room area for each location

Ensure: \hat{L} : Estimated location

for each anchor S_j in \mathcal{S} **do**

$m_j \leftarrow 0$

for each location L_i in \mathcal{L} **do**

$$m_j(L_i) \leftarrow \frac{E_j^{\text{sum}} \cdot A'_i}{\sum_{k=1}^{|\mathcal{L}|} E_k^{\text{sum}} \cdot A'_k} \quad \triangleright \text{Calculate mass by (1)}$$

end for

 Normalize m_j such that $\sum_{L_i \in \mathcal{L}} m_j(L_i) = 1$

$m \leftarrow$ Combine m and m_j by (2)

end for

$\hat{L} \leftarrow \arg \max_{L_i \in \mathcal{L}} m(L_i) \quad \triangleright$ Select location with highest mass

4.2. Fingerprinting in Fuzzy Zones

In fuzzy zones, where targets' signals overlap across multiple rooms, it is essential to employ robust location estimation methods to accurately determine a target's position. To address this challenge, a specialized fingerprinting approach is used. This method leverages a fingerprinting database alongside sophisticated machine learning algorithms, e.g., random forest (RF), k-nearest neighbors (KNN), and support vector machines (SVMs), to analyze RSSI distribution patterns and predict the target's location. The fingerprint database captures the unique characteristics of RSSI data within these zones, providing a critical foundation for the effective application of these algorithms.

The steps of the fingerprinting localization algorithm are divided into three steps:

4.2.1. Data Preprocessing

The raw RSSI data (denoted as X) are subjected to sliding average filtering and normalization. These steps are essential to reduce noise and standardize the data format, making it suitable for further analysis by machine learning models.

4.2.2. Model Training

A fingerprint localization model is trained on a subset of the preprocessed data X' . The model learns to map RSSI features to latent locations $\mathcal{L} = \{L_1, L_2, \dots, L_n\}$, which allows it to predict the most likely location of new RSSI readings.

4.2.3. Location Prediction

Given new RSSI data $x \in X'$, the fingerprint model predicts the most likely location by matching with the data in the fingerprint database to find the most similar fingerprint (\hat{L}).

Algorithm 2 details the operational steps of our room-level localization system, DSFPS, for processing RSSI data to efficiently estimate indoor locations.

Algorithm 2 Indoor Location Estimation

```

1:  $W \leftarrow 10$                                 ▷ Window size for data segmentation
2:  $D \leftarrow \{\}$                                 ▷ Data windows for each address
3:  $C \leftarrow \{\}$                                 ▷ Count of data points for each address
4:  $S \leftarrow \{\dots\}$                             ▷ Mapping of device addresses to scanners
5:  $T \leftarrow \{\dots\}$                             ▷ Trust values associated with each scanner
6:  $H \leftarrow T.keys()$                             ▷ Set of hypotheses based on scanner IDs
7: Input:  $data, model$ 
8: for row in  $Data(data)$  do
9:    $a, s, r \leftarrow ExtractData(row)$             ▷ Extract address, scanner, and RSSI
10:   $AddData(a, s, r, D, W)$ 
11:   $Increment(a, C)$ 
12: end for
13: if  $Sufficientdata(C, W)$  then
14:   $L \leftarrow \{\}$                                 ▷ Dictionary to hold estimated locations
15:  for  $a, d$  in  $D.items()$  do
16:     $g \leftarrow GroupByScanner(d)$ 
17:     $c \leftarrow CountScanners(g)$ 
18:    if  $c == 1$  then
19:       $L[a] \leftarrow EstimateDS(g, H, T)$ 
20:    else
21:       $L[a] \leftarrow EstimateRF(g, model)$ 
22:    end if
23:  end for
24:  return  $L$ 
25: end if

```

5. Experiments

To assess the accuracy of our proposed system, we conduct several experiments, and the specific experimental equipment is shown in Figure 7. The experiment takes place in seven designated areas on the second floor of our college, covering an area of approximately 2000 m². Each area is equipped with an anchor node, the location is shown in Figure 8. The environmental description of the test area is as shown in the Table 5. Ten targets are deployed in each area, and the targets are launched once per second. The formula for calculating accuracy is as follows:

$$\text{Accuracy} = \frac{\text{correct estimations}}{\text{total estimations}} \quad (4)$$

This process requires collecting RSSI readings from each target, and then applying the sliding window method with a window size of 10 to evaluate the position in turn according to the RSSI data of each window, as shown in Table 6.

Based on the current deployment locations of targets, the regions associated with targets b, c, e, and i are identified as regions of uncertainty. We compare the accuracy of the D–S algorithm when used alone versus its combination with a fingerprinting positioning algorithm. Prior to this comparison, it is noteworthy that the fingerprinting algorithm implemented in our system requires only a minimal set of location data to achieve highly accurate determinations in these uncertain regions, as illustrated in Figure 8. In Figure 8, targets b, c, e, and i are situated in the fuzzy areas. Subsequently, we conduct an accuracy comparison between using solely the D–S algorithm and employing DSFP location algorithms.

Prior to the comparative analysis, to validate the robustness of the DSFP algorithm, we conducted training using 5, 10, 20, 30, 40, 50, 100, and 300 data points from each room and evaluated the algorithm's performance in situations of data scarcity. As depicted in Figure 9, the accuracy of location estimates using different training set sizes shows that each of the three algorithms reaches optimal accuracy when trained with 50 data points.

Figure 10 illustrates the computational time required by various algorithms and models trained with different data volumes, using approximately 2400 test data points. It is evident that the SVM algorithm requires the least time for execution; however, the KNN algorithm is able to produce a highly accurate model with just 10 training data points. Considering running time, accuracy, and data requirements collectively, the KNN algorithm exhibits superior performance.



Figure 7. Experimental equipment deployment diagram.

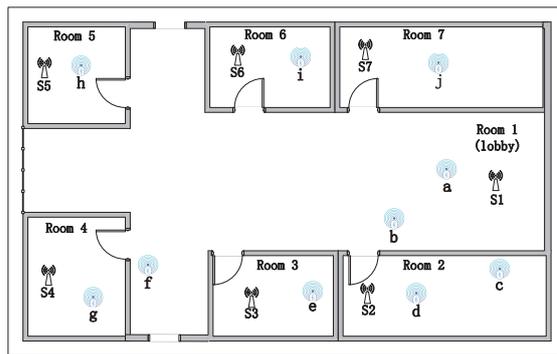


Figure 8. Test site environment.

Table 5. Characteristics of different rooms in the test environment.

Room No.	Room Type	Area Feature	Main Characteristics
Room 1	Hall	-	Some concrete pillars
Room 2	Office A	-	Wooden partitions dividing spaces
Room 3	Laboratory A	-	Densely populated, high interference
Room 4	Laboratory B	Spacious	Less populated
Room 5	Laboratory C	Small area	Densely populated
Room 6	Office B	Small area	Wooden partitions dividing spaces
Room 7	Office C	Small area	Wooden partitions dividing spaces

Using a model trained with 50 data points, we assessed the accuracy of various algorithms at different data window sizes. Figure 11 illustrates that setting the window size to 10 yields the highest accuracy.

Table 6. Scanner data example.

Timestamp	S001	S002	S003	S004	S005	S006	S007
24 November 2023 21:24:38	-100	-100	-71	-100	-100	-100	-100
24 November 2023 21:24:43	-100	-100	-80	-100	-100	-100	-100
24 November 2023 21:24:45	-100	-87	-100	-100	-100	-100	-100
24 November 2023 21:24:48	-100	-86	-100	-100	-100	-100	-100
24 November 2023 21:24:49	-100	-100	-71	-100	-100	-100	-100
24 November 2023 21:24:50	-100	-100	-72	-100	-100	-100	-100
24 November 2023 21:24:51	-100	-100	-85	-100	-100	-100	-100
24 November 2023 21:24:59	-100	-85	-100	-100	-100	-100	-100
24 November 2023 21:25:00	-100	-100	-85	-100	-100	-100	-100
24 November 2023 21:25:01	-100	-100	-85	-100	-100	-100	-100
24 November 2023 21:25:02	-100	-100	-73	-100	-100	-100	-100

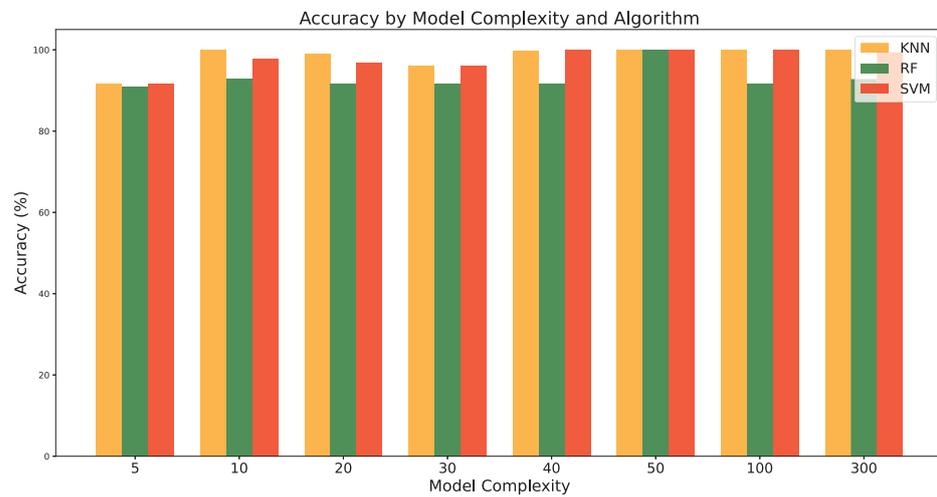


Figure 9. Outcomes of employing varying amounts of training data for fingerprinting positioning.

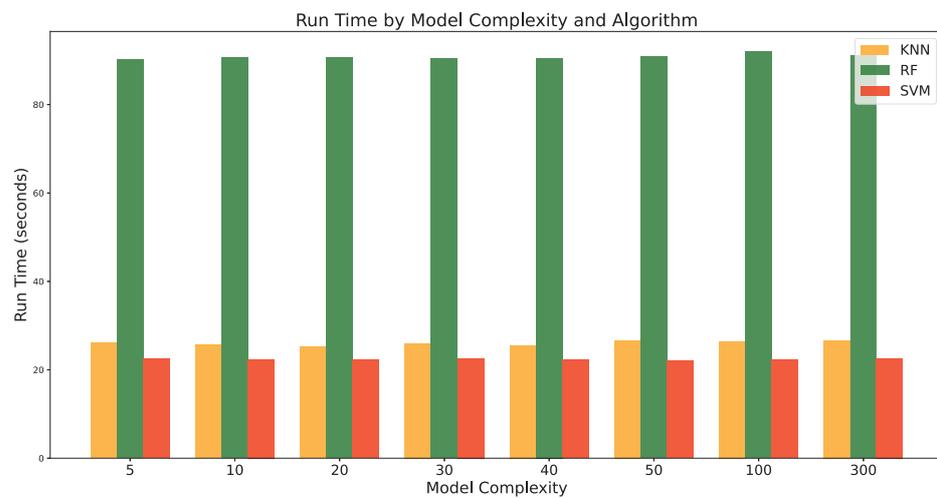


Figure 10. Computational time required for different algorithms and training sizes.

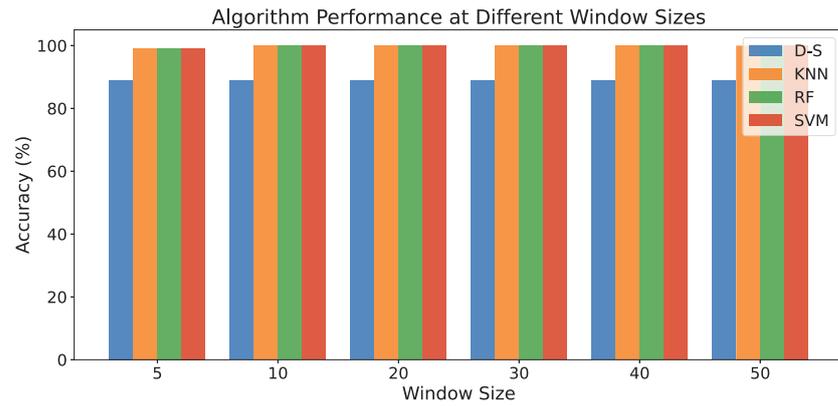


Figure 11. Comparison of Algorithm Performance at Different Window Sizes.

Following the static accuracy evaluation, we proceed with mobile testing. The trajectory for this testing is depicted in Figure 12. Figure 13 presents both the location estimates derived solely from the D–S algorithm and those obtained with the DSFP algorithm. The D–S algorithm results reveal that, during the transition from Room 3 to Room 2, the anchor node in Room 1 (hall) also detected the target signal, which compromises the accuracy of the pure D–S algorithm. In contrast, the DSFP algorithm demonstrates its ability to accurately determine the correct location of the target throughout the transition.

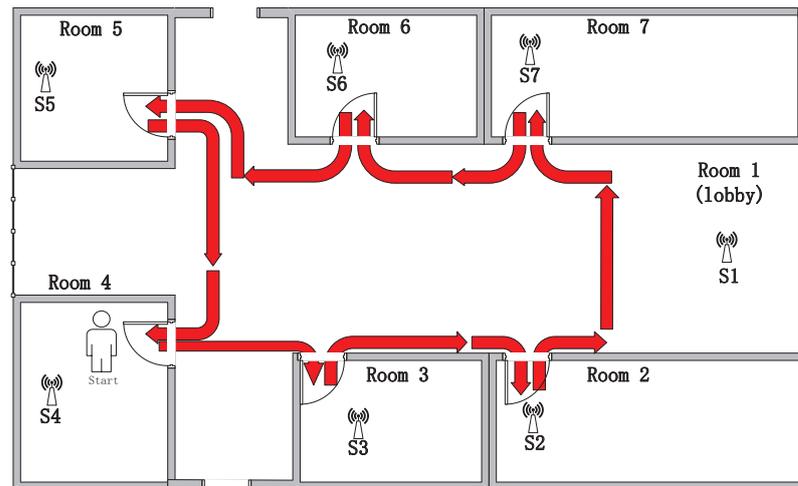


Figure 12. Testing trajectory.

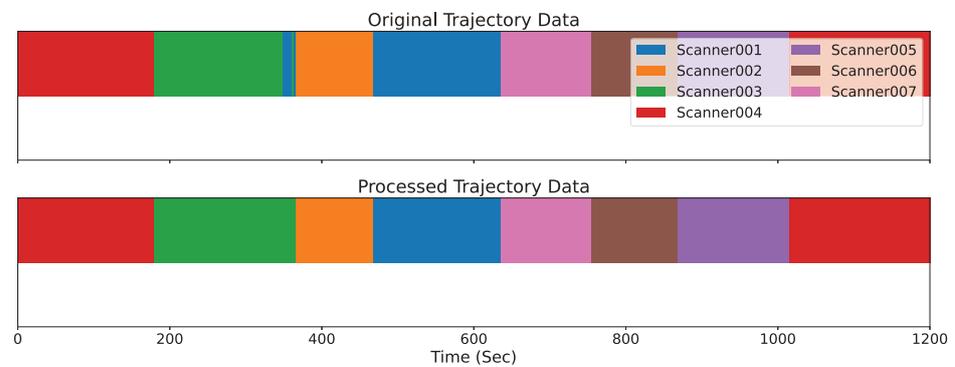


Figure 13. Comparison of raw data trajectory and the correct trajectory after processing.

Figure 14 shows the interface of the DSFP system, which allows users to view the scanned target records in a selected room. After selecting a target, the system displays the target's current room on the floor plan and lists the most recent 10 location records along with the algorithm used to derive those results.

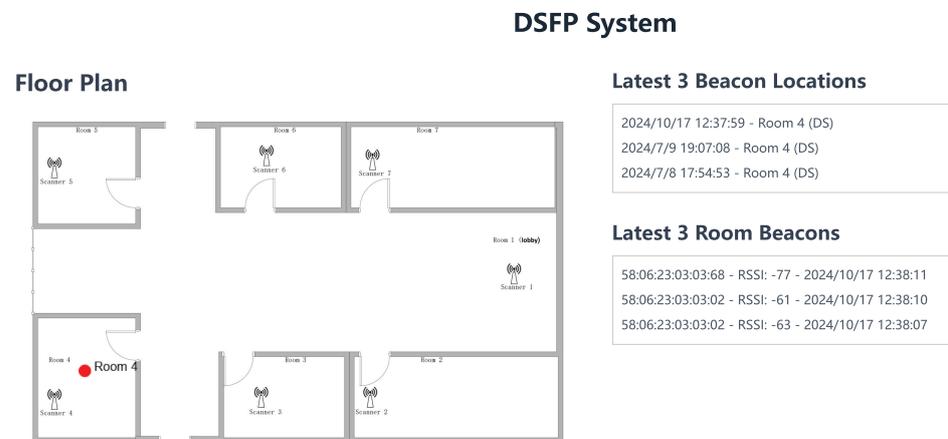


Figure 14. Frontend of the DSFP system.

6. Conclusions

This paper proposes a room-level localization system, which employs the energy harvesting Bluetooth tags as targets, the AIoT nodes as anchors, and the DSFP algorithm to locate the targets. We introduce the D–S theory in conjunction with fingerprinting techniques for an integrated solution to Bluetooth target positioning. Specifically, the D–S theory facilitates the computation of the most probable location by analyzing beacon RSSI values across different rooms. Meanwhile, the fingerprinting technique addresses the issue of RSSI signal overlap in areas where room boundaries are indistinct. Extensive experimental validation demonstrates that our system performs robustly in the test environment, maintaining room-level positioning accuracy above 99% under standard test conditions. This performance metric may fluctuate due to environmental changes, such as personnel movement and equipment layout modifications, particularly in fuzzy areas. Nevertheless, the system consistently provides reliable room-level positioning, meeting the requirements for practical applications in industrial warehousing and medical equipment management. Lastly, DSFP implements a simple and easy-to-use front page.

Author Contributions: Conceptualization of room level localization using BLE, Y.Z. and Y.C.; system design and analyze, Y.W.; experiments, Y.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by China University Industry-University-Research Innovation Fund (No. 2023BD027).

Data Availability Statement: Data are contained within the article.

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

References

1. Zafari, F.; Gkelias, A.; Leung, K.K. A survey of indoor localization systems and technologies. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 2568–2599. [[CrossRef](#)]
2. Yassin, A.; Nasser, Y.; Awad, M.; Al-Dubai, A.; Liu, R.; Yuen, C.; Raulefs, R.; Aboutanios, E. Recent advances in indoor localization: A survey on theoretical approaches and applications. *IEEE Commun. Surv. Tutor.* **2016**, *19*, 1327–1346. [[CrossRef](#)]
3. Brena, R.F.; García-Vázquez, J.P.; Galván-Tejada, C.E.; Muñoz-Rodríguez, D.; Vargas-Rosales, C.; Fangmeyer, J. Evolution of indoor positioning technologies: A survey. *J. Sens.* **2017**, *2017*, 2630413. [[CrossRef](#)]

4. Li, S.; Hedley, M.; Bengston, K.; Humphrey, D.; Johnson, M.; Ni, W. Passive localization of standard WiFi devices. *IEEE Syst. J.* **2019**, *13*, 3929–3932. [[CrossRef](#)]
5. Salamah, A.H.; Tamazin, M.; Sharkas, M.A.; Khedr, M. An enhanced WiFi indoor localization system based on machine learning. In Proceedings of the 2016 IEEE International Conference on Indoor Positioning and Indoor Navigation (IPIN), Madrid, Spain, 4–7 October 2016; pp. 1–8.
6. Akram, B.A.; Akbar, A.H. Wi-Fi Fingerprinting Based Room Level Indoor Localization Framework Using Ensemble Classifiers. *Mehran Univ. Res. J. Eng. Technol.* **2019**, *38*, 151–174. [[CrossRef](#)]
7. Thaljaoui, A.; Val, T.; Nasri, N.; Brulin, D. BLE localization using RSSI measurements and iRingLA. In Proceedings of the 2015 IEEE International Conference on Industrial Technology (ICIT), Seville, Spain, 17–19 March 2015; pp. 2178–2183.
8. Röbesaat, J.; Zhang, P.; Abdelaal, M.; Theel, O. An improved BLE indoor localization with Kalman-based fusion: An experimental study. *Sensors* **2017**, *17*, 951. [[CrossRef](#)]
9. Mackey, A.; Spachos, P.; Song, L.; Plataniotis, K.N. Improving BLE Beacon Proximity Estimation Accuracy Through Bayesian Filtering. *IEEE Internet Things J.* **2020**, *7*, 3160–3169. [[CrossRef](#)]
10. Poulouse, A.; Han, D.S. UWB indoor localization using deep learning LSTM networks. *Appl. Sci.* **2020**, *10*, 6290. [[CrossRef](#)]
11. Ruiz, A.R.J.; Granja, F.S. Comparing ubisense, bespoon, and decawave uwb location systems: Indoor performance analysis. *IEEE Trans. Instrum. Meas.* **2017**, *66*, 2106–2117. [[CrossRef](#)]
12. Barbieri, L.; Brambilla, M.; Trabattoni, A.; Mervic, S.; Nicoli, M. UWB localization in a smart factory: Augmentation methods and experimental assessment. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–18. [[CrossRef](#)]
13. Alarifi, A.; Al-Salman, A.; Alsaleh, M.; Alnafessah, A.; Al-Hadhrani, S.; Al-Ammar, M.A.; Al-Khalifa, H.S. Ultra wideband indoor positioning technologies: Analysis and recent advances. *Sensors* **2016**, *16*, 707. [[CrossRef](#)] [[PubMed](#)]
14. Shit, R.C.; Sharma, S.; Puthal, D.; James, P.; Pradhan, B.; Van Moorsel, A.; Zomaya, A.Y.; Ranjan, R. Ubiquitous localization (UbiLoc): A survey and taxonomy on device free localization for smart world. *IEEE Commun. Surv. Tutor.* **2019**, *21*, 3532–3564. [[CrossRef](#)]
15. Yang, B.; Li, J.; Shao, Z.; Zhang, H. Robust UWB indoor localization for NLOS scenes via learning spatial-temporal features. *IEEE Sens. J.* **2022**, *22*, 7990–8000. [[CrossRef](#)]
16. Roy, P.; Chowdhury, C. A survey on ubiquitous WiFi-based indoor localization system for smartphone users from implementation perspectives. *CCF Trans. Pervasive Comput. Interact.* **2022**, *4*, 298–318. [[CrossRef](#)]
17. He, S.; Chan, S.H.G. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Commun. Surv. Tutor.* **2015**, *18*, 466–490. [[CrossRef](#)]
18. Sadowski, S.; Spachos, P. Rssi-based indoor localization with the internet of things. *IEEE Access* **2018**, *6*, 30149–30161. [[CrossRef](#)]
19. Guvenc, I.; Chong, C.C. A survey on TOA based wireless localization and NLOS mitigation techniques. *IEEE Commun. Surv. Tutor.* **2009**, *11*, 107–124. [[CrossRef](#)]
20. Li, X.; Deng, Z.D.; Rauchenstein, L.T.; Carlson, T.J. Contributed Review: Source-localization algorithms and applications using time of arrival and time difference of arrival measurements. *Rev. Sci. Instrum.* **2016**, *87*, 41502. [[CrossRef](#)]
21. Zheng, Y.; Sheng, M.; Liu, J.; Li, J. Exploiting AoA estimation accuracy for indoor localization: A weighted AoA-based approach. *IEEE Wirel. Commun. Lett.* **2018**, *8*, 65–68. [[CrossRef](#)]
22. Paul, S.; Chatterjee, A.; Guha, D. Study of smart inventory management system based on the internet of things (IOT). *Int. J. Recent Trends Bus. Tour.* **2019**, *3*, 27–34.
23. Wikipedia Contributors. Real-Time Locating System—Wikipedia, The Free Encyclopedia. 2024. Available online: https://en.wikipedia.org/w/index.php?title=Real-time_locating_system&oldid=1222348337 (accessed on 7 May 2024).
24. Singh, N.; Choe, S.; Punmiya, R. Machine learning based indoor localization using Wi-Fi RSSI fingerprints: An overview. *IEEE Access* **2021**, *9*, 127150–127174. [[CrossRef](#)]
25. Achroufene, A.; Chibani, A.; Amirat, Y. Using Dempster–Shafer Theory for RSS-based Indoor Localization. In Proceedings of the 2020 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Glasgow, UK, 19–24 July 2020; pp. 1–8. [[CrossRef](#)]
26. Cantón Paterna, V.; Calveras Augé, A.; Paradells Aspas, J.; Pérez Bullones, M.A. A Bluetooth Low Energy Indoor Positioning System with Channel Diversity, Weighted Trilateration and Kalman Filtering. *Sensors* **2017**, *17*, 2927. [[CrossRef](#)] [[PubMed](#)]
27. Rida, M.E.; Liu, F.; Jadi, Y.; Algawhari, A.A.A.; Askourih, A. Indoor Location Position Based on Bluetooth Signal Strength. In Proceedings of the 2015 2nd International Conference on Information Science and Control Engineering, Shanghai, China, 24–26 April 2015; pp. 769–773. [[CrossRef](#)]
28. Gu, Y.; Quan, L.; Ren, F.; Li, J. Fast Indoor Localization of Smart Hand-Held Devices Using Bluetooth. In Proceedings of the 2014 10th International Conference on Mobile Ad-hoc and Sensor Networks, Maui, HI, USA, 19–21 December 2014; pp. 186–194. [[CrossRef](#)]
29. Tariq, Z.B.; Cheema, D.M.; Kamran, M.Z.; Naqvi, I.H. Non-GPS Positioning Systems: A Survey. *ACM Comput. Surv.* **2017**, *50*, 57. [[CrossRef](#)]
30. Lazaro, A.; Lazaro, M.; Villarino, R. Room-Level Localization System Based on LoRa Backscatters. *IEEE Access* **2021**, *9*, 16004–16018. [[CrossRef](#)]

31. Kyritsis, A.I.; Kostopoulos, P.; Deriaz, M.; Konstantas, D. A BLE-based probabilistic room-level localization method. In Proceedings of the 2016 International Conference on Localization and GNSS (ICL-GNSS), Barcelona, Spain, 28–30 June 2016; pp. 1–6. [[CrossRef](#)]
32. Bianchi, V.; Ciampolini, P.; De Munari, I. RSSI-Based Indoor Localization and Identification for ZigBee Wireless Sensor Networks in Smart Homes. *IEEE Trans. Instrum. Meas.* **2019**, *68*, 566–575. [[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.