



Article

Improved Particle Filter in Machine Learning-Based BLE Fingerprinting Method to Reduce Indoor Location Estimation Errors

Jingshi Qian ^{1,†}, Jiahe Li ^{1,†}, Nobuyoshi Komuro ^{2,*} , Won-Suk Kim ³ and Younghwan Yoo ³

¹ Graduate School of Science and Engineering, Chiba University, 1-33, Yayoi-cho, Inage-ku, Chiba 263-8522, Japan; 23wd5206@student.gs.chiba-u.jp (J.Q.); lijiahe970212@163.com (J.L.)

² Chiba University Digital Transformation Enhancement Council, Chiba University, 1-33, Yayoi-cho, Inage-ku, Chiba 263-8522, Japan

³ Division of Computer Science and Engineering, Pusan National University, 2 Busandaehak-ro 63 beon-gil, Geumjeong-gu, Busan 43241, Republic of Korea; wonsuccim@pusan.ac.kr (W.-S.K.); ymomo@pusan.ac.kr (Y.Y.)

* Correspondence: kmr@faculty.chiba-u.jp

† These authors contributed equally to this work.

Abstract: Indoor position fingerprint-based location estimation methods have been widely used by applications on smartphones. In these localization estimation methods, it is very popular to use the RSSI (Received Signal Strength Indication) of signals to represent the position fingerprint. This paper proposes the design of a particle filter for reducing the estimation error of the machine learning-based indoor BLE location fingerprinting method. Unlike the general particle filter, taking into account the distance, the proposed system designs improved likelihood functions, considering the coordinates based on fingerprint points using mean and variance of RSSI values, combining the particle filter with the k-NN (k-Nearest Neighbor) algorithm to realize the reduction in indoor positioning error. The initial position is estimated by the position fingerprinting method based on the machine learning method. By comparing the fingerprint method based on k-NN with general particle filter processing, and the fingerprint estimation method based on only k-NN or SVM (Support Vector Machine), experiment results showed that the proposed method has a smaller minimum error and a better average error than the conventional method.

Keywords: indoor positioning; RSSI fingerprinting; k-nearest neighbors; bluetooth low energy; particle filter



Citation: Qian, J.; Li, J.; Komuro, N.; Kim, W.-S.; Yoo, Y. Improved Particle Filter in Machine Learning-Based BLE Fingerprinting Method to Reduce Indoor Location Estimation Errors. *Future Internet* **2024**, *16*, 211. <https://doi.org/10.3390/fi16060211>

Academic Editors: Jiahui Yu, Charlie Yang, Zhenyu Wen, Dalin Zhou, Dongxu Gao and Changting Lin

Received: 7 May 2024

Revised: 12 June 2024

Accepted: 13 June 2024

Published: 15 June 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

Currently, GPS (Global Positioning System) location estimation is used outdoors, and while indoors, signal attenuation and refraction problems make GPS errors even greater [1]. Currently, the use of smartphones, which have a high penetration rate, is often considered for indoor position estimation. Frequently, wireless signals such as Wi-Fi (Wireless Fidelity) [2], Bluetooth, and ZigBee [3] are often used, all of which use a transmitting terminal to measure the RSSI (Received Signal Strength Indication) of the signal. With the launch of Bluetooth 4.0 technology, low-cost power technology is incorporated into Bluetooth 4.0, and location estimation methods based on BLE (Bluetooth Low Energy) [4] have more advantages. There have been some related studies on BLE channel separation [5], or the combination of BLE and the dead reckoning method [6].

Indoor location estimation methods can be divided into two main categories: by distance and by position fingerprints. The distance-based estimation method is based on triangulation [7]. The triangulation method calculates the distance by observing the RSSI values of multiple APs (Access Points). Its accuracy is insufficient due to the propagation loss of RSSI caused by indoor walls and obstructions [8].

The position fingerprinting method [9,10] is a position estimation method that uses the distribution pattern of RSSI in the indoor environment. RSSI measurements are taken in advance at several locations and used as position fingerprints (a collection of RSSI values for each location). Localization is performed by comparing the similarity of the position fingerprint RSSI with the actual measured RSSI. Several methods have been proposed for indoor location estimation using fingerprint positioning method, including fingerprinting using CNN (Convolutional Neural Network) [11–13], k-NN (k-Nearest Neighbor) [14–16], and SVM (Support Vector Machine) [17].

In the CNN-based method, the RSSI value is transformed into an image that becomes a location fingerprint. By learning the pixel values of the image, the location estimation can be calculated. The method using CNN provides high accuracy in simulations but requires a large amount of data for training, which makes it costly in practical experiments. In the method using k-NN, k fingerprints near the observation point are selected based on the measured RSSI values, and the location is estimated by calculating the average value. In the k-NN-based method, the estimation accuracy is greatly affected by the distribution of RSSI noise. In the method using SVM, the RSSI values of the reference points are learned and transformed into a classification problem by a kernel function for position estimation. The choices of the kernel function and parameters greatly affect the accuracy of the estimation.

On the other hand, particle filters [18], which can deal with non-linear cases, have also been widely applied to position estimation for object tracking [19], post-processing, and other purposes using statistical models. The computational complexity of traditional methods using particle filters is also high when the number of particles is large, and therefore methods that can reduce the complexity are considered [20–22]. Related studies have shown that general particle filtering allows for filtering and improving the accuracy of the predicted positions of machine learning algorithms [22]. In [22], the weight function used in the particle filter considers distance. However, there have been few discussions on the suitable weight functions used in the particle filter used in the fingerprinting method.

This study proposes a fingerprint positioning method based on k-NN and improved particle filters, whose weight functions consider the coordinates. The proposed system first estimates the position using k-NN, and then corrects the estimated position using a particle filter with improved likelihood functions to reduce the error of k-NN caused by noise. In the proposed particle filter, improved likelihood functions are designed using the mean and variance of RSSI values of fingerprints around particles, and the results were comparatively analyzed. The proposed system reduces the error of the k-NN and reduces the computational complexity of the method using a particle filter. Experiments demonstrate the effectiveness of the proposed system by comparing the estimation results of the conventional methods.

2. Related Research

2.1. Position Fingerprinting Method

Indoors radio signals reflect and are affected by many things, such as obstacles, temperature, and human movement. Unique RSSI values are generated at different locations.

RSSI values are measured and collected at as many points as possible indoors, and when estimating location, the RSSI values are compared to the values of the location points. Since this is similar to the relationship between a person and a fingerprint, it is also known as the position fingerprinting method [9,10]. The position fingerprinting method is divided into two parts: the “online phase” and the “offline phase”. Figure 1 illustrates the procedure of the position fingerprinting method.

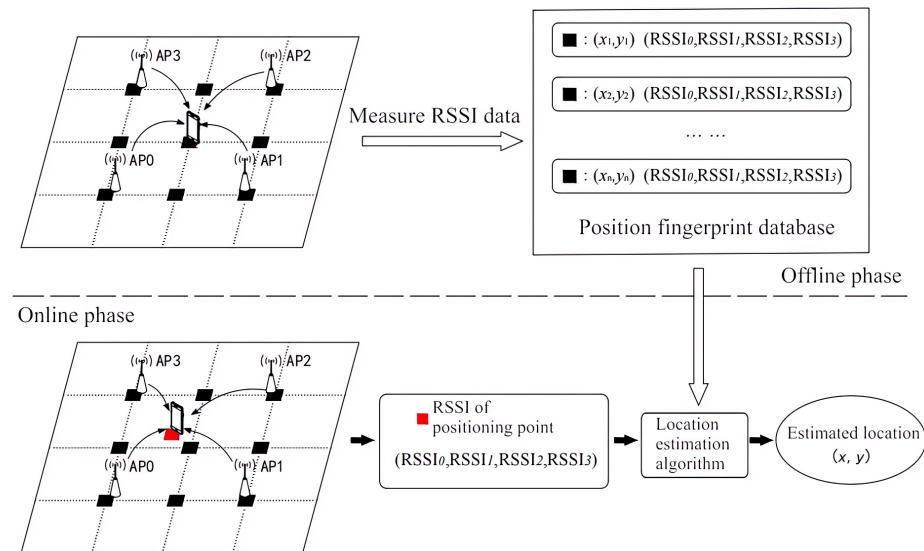


Figure 1. Procedure of fingerprint positioning method.

In the offline phase, first, the experimental environment is determined, classified, and displayed on a map. In the experimental environment, multiple APs are set up in advance, and the RSSI values of each reference point within the entire map are measured sequentially.

In the online phase, when performing location estimation, the RSSI value collected from the terminal AP at the location point is used to compare with the fingerprint points stored in the location fingerprint database, the fingerprint point with the most similar RSSI value characteristics is selected, and the location coordinates of the location point can be estimated by the location estimation algorithm based on the RSSI distribution rules.

2.1.1. CNN-Based Location Estimation Method

CNN (Convolutional Neural Network) simulates the human brain's cognitive process of external information and can extract complex structural features in a large amount of data to establish processing methods [23]. In the literature [24], it was proposed to convert RSSI values into a single vector or matrix to be processed as pixel values in images and to use the features of RSSI values in indoor environments to learn the nonlinear relationship between RSSI and location coordinates.

Although the CNN-based approach can achieve high accuracy in simulation, the number of fingerprint points used will be thousands to tens of thousands when the environment is large. In other words, it is very costly.

2.1.2. k-NN-Based Fingerprinting Method

The k-NN (k-Nearest Neighbors) is a classification algorithm commonly used in machine learning algorithms. k-NN-based fingerprinting [14] is often applied to the multi-classification problem corresponding to location estimation. Fingerprinting with k-NN is achieved by using fingerprint points around a location point by calculating the Euclidean distance between the location point and the fingerprint points. Figure 2 illustrates the concept of the k-NN-based fingerprinting method.

In the fingerprinting method using k-NN, the estimation accuracy may be significantly affected by the distribution of fingerprint points and the noise of observed RSSI.

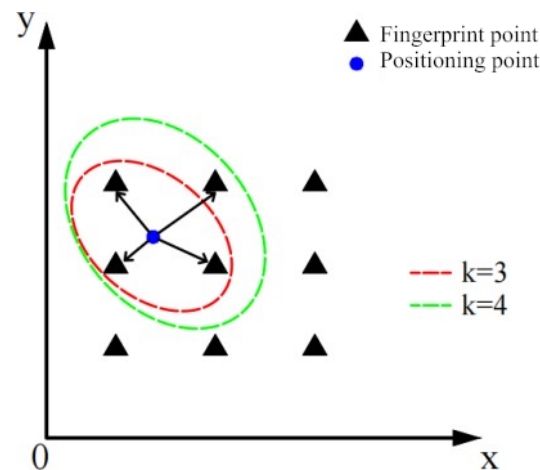


Figure 2. k-NN-based fingerprinting method.

2.1.3. SVM-Based Fingerprinting Method

SVM (Support Vector Machine) [17] is an algorithm capable of handling classification and regression with excellent performance for high-dimensional, nonlinear pattern recognition problems [25]. Data classification is achieved by adding a kernel function $K(x, y)$ to the SVM to construct a partitioned hyperplane that maximizes the geometric interval in the high-dimensional space. The SVM is trained in the offline phase using data from the fingerprint database to determine the nonlinear relationship between RSSI values and location coordinates, and then the localization calculation is performed in the online phase.

The problem is that accuracy is affected by the kernel function and the number of parameters, and as training data increases, the computational complexity becomes enormous.

2.2. Method Using Statistical Models

2.2.1. State-Space Model

By building a state-space model like Figure 3, indoor position estimation can be considered as a state estimation problem. The current position of the measurement point at time t is considered to be related to the position and state at time $t - 1$. The discrete-time state-space model [26] can be used to implement the position estimation problem.

$$x_t = F(x_{t-1}) + \omega_t \quad (1)$$

$$z_t = H(x_{t-1}) + v_t \quad (2)$$

The state Equation (Equation (1)) represents the relationship between the states of the target at different moments. The observation Equation (Equation (2)) is used to represent the relationship between the observed values and the state of the target.

ω_t is the process noise, representing the uncertainty of the movement. v_t is the observation noise, representing the error of the observation. ω_t and v_t are independent of each other.

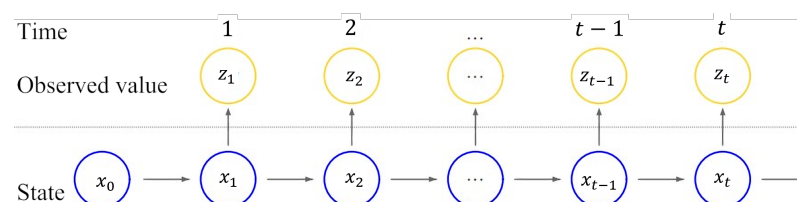


Figure 3. State-space model.

2.2.2. Bayesian Filter

Unlike fingerprinting methods using CNN, k-NN, and SVM, Bayesian filters [27] view the state estimation problem in the state-space model as probabilistic inference and transform the location estimation of a geodetic point into a calculation of the posterior probability density. Position estimation is performed by calculating the probability of positioning points and reference fingerprint points.

It is difficult to apply the Bayesian filter directly. For different cases, the Bayesian filter is divided into Kalman filter [28] (KF) and particle filter [29] (PF). The Kalman filter is applied to linear and Gaussian state spaces. However, for indoor positioning, particle filters are often used for nonlinear and non-Gaussian state spaces, where it is difficult to satisfy linear and Gaussian [30].

2.2.3. Particle Filter

The particle filter combines Bayesian estimation and the method of Monte Carlo approximation [31], and considers N particle samples to represent the prediction and state distributions. The state distribution is simulated by the particle samples, and the particle weights and position distribution at t are adjusted using samples rather than integration operations, and the state distribution is updated according to the adjusted particles. Particle filtering consists of four parts: initialization, sequential inertial importance sampling, attaching new weights, and sampling. The two most important parts are sequential importance sampling [29] and resampling [32,33].

3. Proposed System

3.1. Configuration of the Proposed System

The procedure of the system proposed in this study is shown in Figure 4.

1. Measure the RSSI values and create fingerprint points using the position fingerprinting method.
2. Determine the initial state using the k-NN-based position estimation method based on the RSSI values of the location point.
3. Correct the estimated coordinates by k-NN using a particle filter.

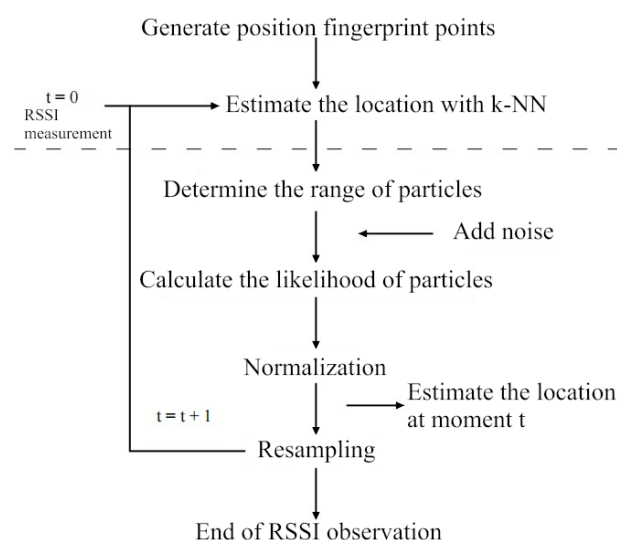


Figure 4. Configuration of the proposed system.

3.2. Position Correction with Particle Filter

Define a set of N particles $P = \{p_1, p_2, \dots, p_N\}$. The coordinates of particle p_i at time t are $(x p_t^{(i)}, y p_t^{(i)})$ and the weight is $\omega_t^{(i)}$. The process of k-NN-based particle filtering algorithm is as follows.

1. Initial state At $t = 0$, determine the initial estimated location using the fingerprinting method with k-NN and generate N particles near the initial location, each randomly scattered and uniformly distributed. Let $\omega_0^{(i)} = 1/N$ be the initial weight.
2. Movement of particles Move N particles. Based on an average stride length, add noise at a distance of $d = 0.65 \text{ m} \pm 0.1 \text{ m}$ and completely randomize the direction of movement. Delete particles whose coordinates are outside the wall by comparing the coordinates of the experimental environment with the coordinates of the particles after moving. The particle movement equation, with the movement angle θ , can be expressed as

$$xp_t^{(i)} = xp_t^{(i)} + d \cos \theta \quad (3)$$

$$yp_t^{(i)} = yp_t^{(i)} + d \sin \theta \quad (4)$$

where $\theta \in (0, 2\pi)$.

3. Weighted by likelihood function

In this study, the designed likelihood function is $fp = (xp_t^{(i)}, yp_t^{(i)})$, where the new weights are obtained by the distance between the particle and the surrounding fingerprint points. The particle weights are $\omega_t^{(i)} = fp(xp_t^{(i)}, yp_t^{(i)})$. At this time, the weight of the particle is defined as the magnitude of the likelihood.

4. Normalization

For each particle, normalize as follows.

$$\tilde{\omega}_t^{(i)} = \frac{\omega_t^{(i)}}{\sum_{i=1}^N \omega_t^{(i)}} \quad (5)$$

5. Resampling

The weight $\omega_t^{(i)}$ of the resampled particles is set to $1/N$. The state can be predicted and the result of the position estimation can be output.

3.3. Design of Likelihood Functions

In this study, two likelihood functions are designed to calculate the likelihood of particles to investigate the effect of weight functions considering position coordinates. The likelihood functions can be designed by calculating the weights of particles using a weighting function by distance based on a mixture normal distribution.

3.3.1. Generating a Mixed Normal Distribution

The RSSI distribution approximates a normal distribution, which allows the Gaussian Mixture Model [34] to be used to generate a mixture normal distribution. The mixture of normal distribution is

$$f_{mix}(RSSI) = \sum_{id} \frac{\omega_{mix}^{id}}{\sqrt{2\pi\sigma_{mix}^{id}}} \exp \left\{ -\frac{(RSSI - \mu_{mix}^{id})^2}{\sigma_{mix}^{id}} \right\} \quad (6)$$

where ω_{mix}^{id} , μ_{mix}^{id} , and σ_{mix}^{id} are the weight, mean, and variance of the mixture normal distribution generated, respectively. Furthermore, id indicates the ID of the BLE.

3.3.2. Weighting Function by Distance

When locating at a given position, the distribution of fingerprint points indicates that four fingerprint points around the particle can be used. Let $c_{(x,y)}^i$ be the four surrounding fingerprint points and $d^i(x, y)$ be the distance from the particle to the fingerprint points. The weight value of the particle varies depending on the distance of the fingerprints.

The distance from the particle to each of the four surrounding fingerprint points is $d_i (i = 1, 2, 3, 4)$, and the following function is used to find the distance and weighting function $\omega^i(x, y)$. The function is altered from the classical sigmoid distance weighting function by adding coefficients and a logarithmic function to make the curve fall more gently and the weights are close to 1 at a distance of zero.

$$l(a, d) = \frac{1}{(0.7)^{-1} \log\{1 + \exp(-ad)^{-1}\}}, a \in \mathbb{N} \quad (7)$$

The determination of a is shown in Figure 5. Here, the distance weighting function $\omega^i(x, y)$ is expressed as follows:

$$\omega^i(x, y) = \frac{l(a, d^i(x, y))}{\sum l(a, d^i(x, y))} \quad (8)$$

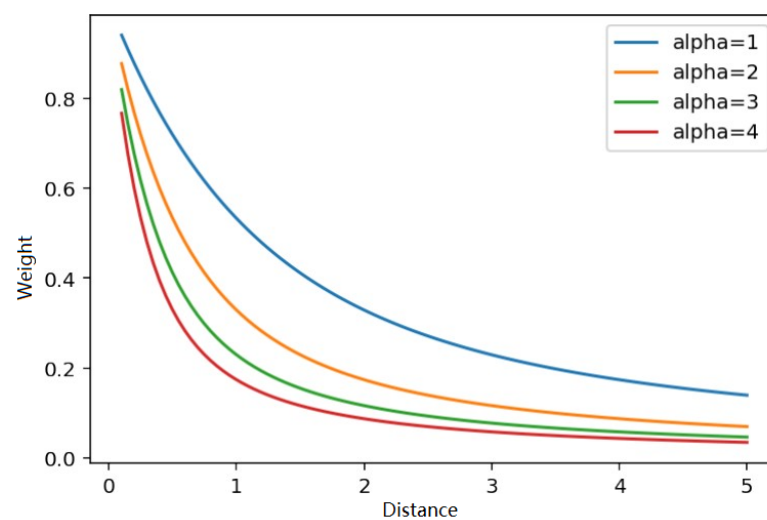


Figure 5. Graph of Equation (7).

3.3.3. Likelihood Function

Let id be the BLE ID, $R_{(x,y)}^{id}$ be the RSSI value measured by BLE^{id} at point (x, y) , $\mu_{(x,y)}^{id}$ be the mean, and $\sigma_{(x,y)}^{id}$ be the variance; the distribution of $R_{(x,y)}^{id}$ is

$$f_{(x,y)}^{id}(R_{(x,y)}^{id}) = \frac{1}{\sqrt{2\pi\sigma_{(x,y)}^{id}}} \exp\left\{-\frac{(R_{(x,y)}^{id} - \mu_{(x,y)}^{id})^2}{\sigma_{(x,y)}^{id}}\right\}. \quad (9)$$

Let R_t^{id} be the RSSI value measured from the BLE beacon at time t . The following equation is used to determine the upper probability of each R_t^{id} .

$$F_{mix}^{id}(R_{(x,y)}^{id}) = \int_{R_t^{id}} f_{mix}^{id}(R_{(x,y)}^{id}) dR_{(x,y)}^{id}. \quad (10)$$

Let D be the set of id 's ordered by decreasing the value of $F_{mix}^{id}(R)$. Furthermore, let $(xp_t^n, yp_t^n) = p(n)$. The distribution of $R_{(x,y)}^{id}$ is used to calculate the likelihood of the particles in $p(n)$. When the particle is within a radius r of the BLE beacon

$$fc(c_{p(n)}^j) = \sum_D f_{(c_{p(n)}^j)}^D R_t^D \{F_{mix}^D(R_{(x,y)}^{id})\}^{-1}, \quad (11)$$

otherwise

$$fc(c_{p(n)}^j) = \sum_D fn^D(c_{p(n)}^j)(R_t^D), \quad (12)$$

Thus, the likelihood function $fp(xp_t^n, yp_t^n)$ of the particle corresponding to the coordinate (xp_t^n, yp_t^n) is expressed as follows:

$$fp(xp_t^n, yp_t^n) = \sum_j \omega^j(xp_t^n, yp_t^n) fc(c_{(xp_t^n, yp_t^n)}^j) \quad (13)$$

Likelihood Function A

The mean $\mu_{(x,y)}^{id}$ and variance $\sigma_{(x,y)}^{id}$ are used to calculate the likelihood of the fingerprint points around the particle, and then the distance weighting function is used to process the estimates to calculate the likelihood of the particle, resulting in a location estimate. Figure 6 shows the flow of calculating the likelihood function A.

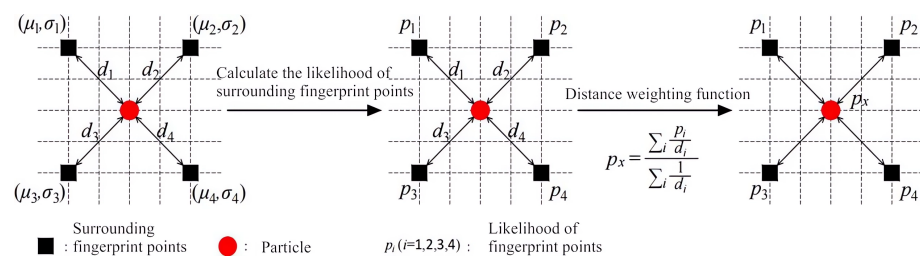


Figure 6. Likelihood function A.

Likelihood Function B

The likelihood function B is more accurate, as it uses more reference points around the particle. An interpolated positional fingerprint with a weighting function is generated based on distance. By using the mean $\mu_{(x,y)}^{id}$ and variance $\sigma_{(x,y)}^{id}$ in the interpolated data, the likelihood of the mixed normal distribution of the location points can be calculated. Figure 7 shows the flow of calculating the likelihood function B.

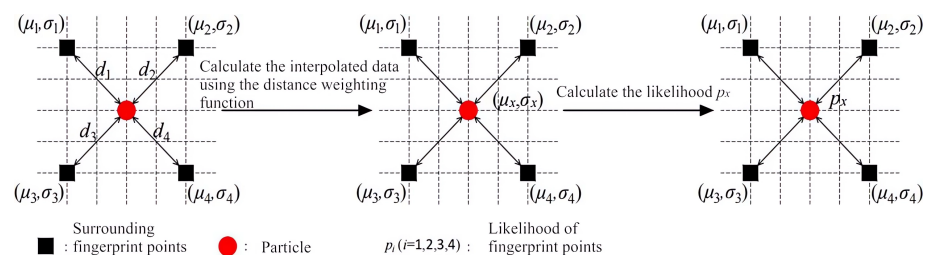


Figure 7. Likelihood function B.

The following equation can be used to generate an interpolated fingerprint with a distance weighting function.

$$\mu_{(x,y)}^{id} = \sum_i \left\{ \omega_{(x,y)}^i \right\} \mu_{(c_{(x,y)}^i)}^{id} \quad (14)$$

$$\sigma_{(x,y)}^{id} = \sum_i \left\{ \omega_{(x,y)}^i \right\} \sigma_{(c_{(x,y)}^i)}^{id} \quad (15)$$

4. Experiment

Indoor location estimation was performed using the proposed system based on RSSI values actually measured from each BLE beacon. The estimation accuracy of the proposed system is evaluated by comparing it with the existing methods using k-NN and SVM.

4.1. Generation of Position Fingerprints

4.1.1. Environment

In this study, the experiment was conducted in a university research room (about $7.8 \text{ m} \times 7.9 \text{ m} \times 2.8 \text{ m}$). Obstacles included desks, lockers, etc. No one other than the experimenter was present during the measurement. Four BLE beacons $\{\text{BLE}^{id} | id = 0, 1, 2, 3\}$ were placed at different locations on the ceiling. The height of the measurement terminal was fixed at 0.65 m when measuring RSSI values. The experiment environment is shown in Figure 8a. Specifications about the experiment devices are in Table 1.

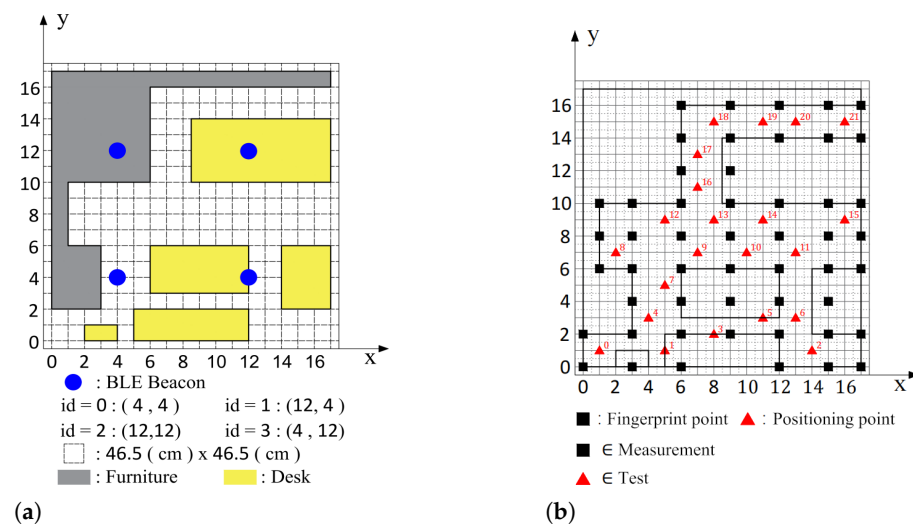


Figure 8. (a) Layout of furniture and location of beacons. (b) Layout of fingerprint points. The grids on the floor represent floor tiles with a side length of approximately 46.5 cm.

Table 1. Specifications of experiment devices.

BLE Products	BLEAD-B Ver.2 [35]
Standard	Bluetooth Ver.4.0
Size	Dia. $50 \times 16.5 \text{ mm}$
Weight	28 g
Range	Approx. 5–100 m
Locating Device	Honor 8 [36]
OS	Android ver.7

4.1.2. Measurement of Fingerprints

In this study, RSSI values were measured evenly across 73 locations for 3 min each. 51 of these locations were used as training and reference, and the remaining 22 locations were used as locating points. The distribution of reference and test fingerprints is shown in Figure 8b.

4.2. Location Estimation by Existing Methods

4.2.1. Location Estimation by SVM

In SVM-based location estimation methods (described in Section 2.1.3), accuracy is affected by the kernel function [37] and parameters. When different kernel functions are used, the SVM algorithm is also different.

There are four general types of kernel functions: linear, polynomial, radial basis function (RBF), and sigmoid. The polynomial kernel function and RBF are applied to the RSSI value distribution for nonlinear and non-Gaussian states [38]. Their equations are expressed as follows. The angle brackets represent the dot product.

$$K_{\text{polynomial}}(x, y) = (\langle x, y \rangle + p)^q \quad p \geq 0, q \in \mathbb{N}^+ \quad (16)$$

$$K_{\text{RBF}}(x, y) = \exp\left(-\frac{\|x - y\|^2}{2\sigma^2}\right) \quad \sigma \geq 0 \quad (17)$$

In this study, grid search is used to determine the parameters of the optimal solutions for two kernel functions. The following table counts the location estimation errors of the two kernel functions, and it can be seen that the radial basis kernel function is more accurate (Table 2).

Table 2. Estimation errors of SVM-based method.

	SVM (RBF)	SVM (Polynomial)
Minimum error (m)	0.77	0.32
Maximum error (m)	3.38	4.56
Average error (m)	2.18	2.56
Std Dev (m)	0.31	0.38

4.2.2. Location Estimation by k-NN

The proposed method first determines the initial location of location estimation based on k-NN (Section 2.1.2), then uses a particle filter in order to improve the estimation accuracy of the k-NN-based method. The k-NN was first experimented separately, Figure 9 shows the value of k influences the accuracy. This study sets $k = 4$, which offers the best estimation accuracy.

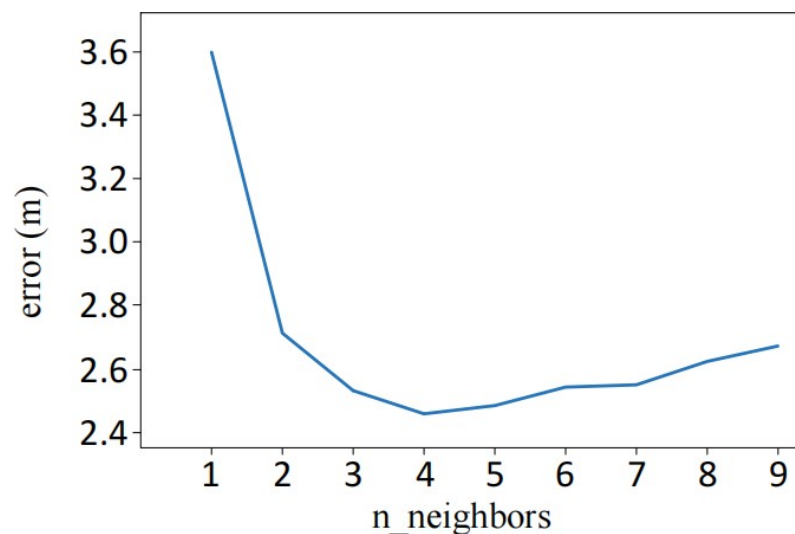


Figure 9. Errors vary with the value of k .

4.2.3. Location Estimation by Proposed Method

The proposed system first estimates the position using a k-NN-based position fingerprinting method and then corrects the estimated position using a particle filter. Figure 10 shows the particle distribution and the variation in the coordinate positions estimated using k-NN for each minute.

The proposed A and B use the likelihood functions A and B, respectively. Figure 11 shows the error distribution in the room. It shows that errors are higher near the walls and obstacles because the RSSI values are affected.

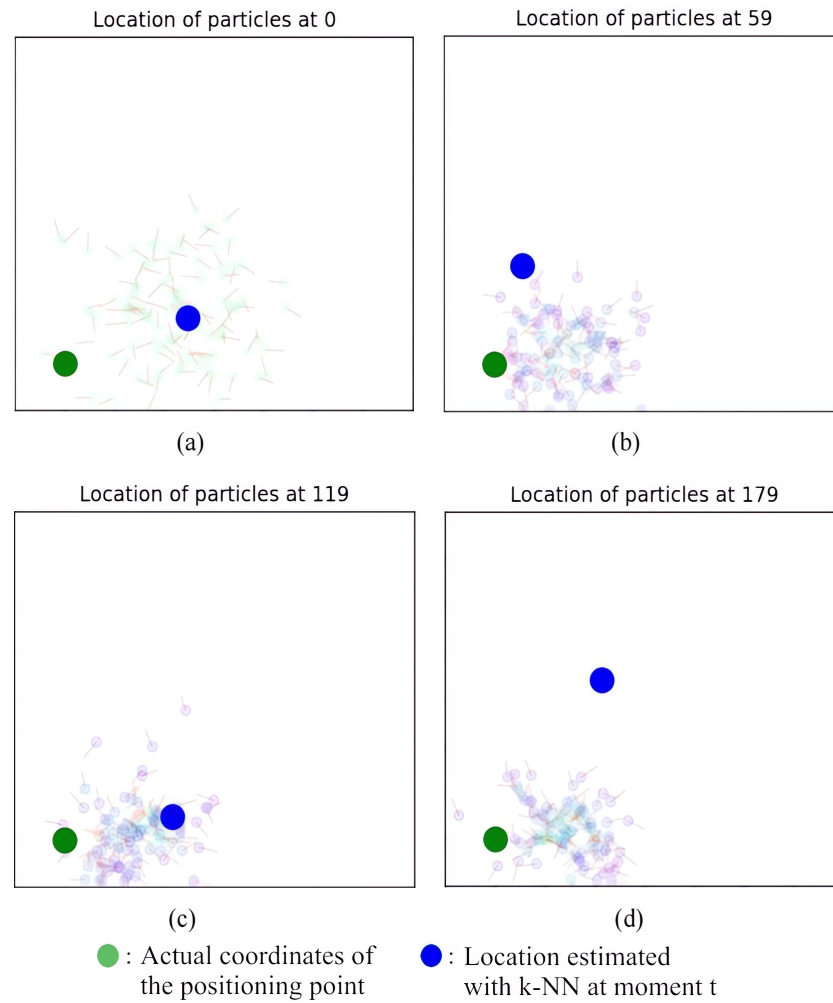


Figure 10. Distribution of particles and estimation of k-NN at different moments t (s).

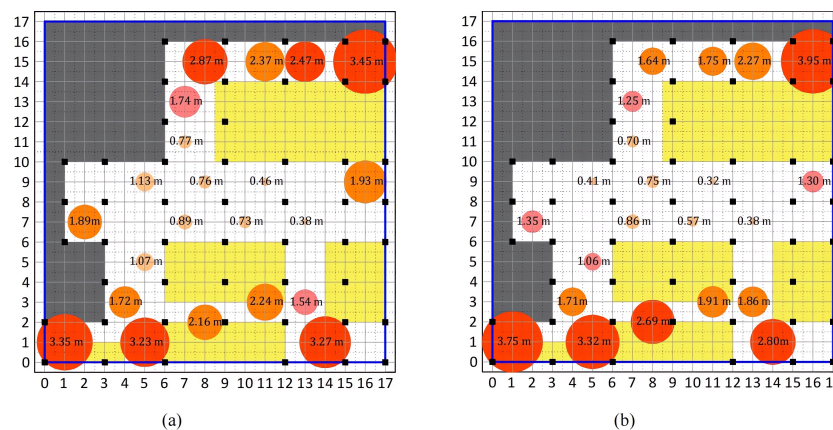


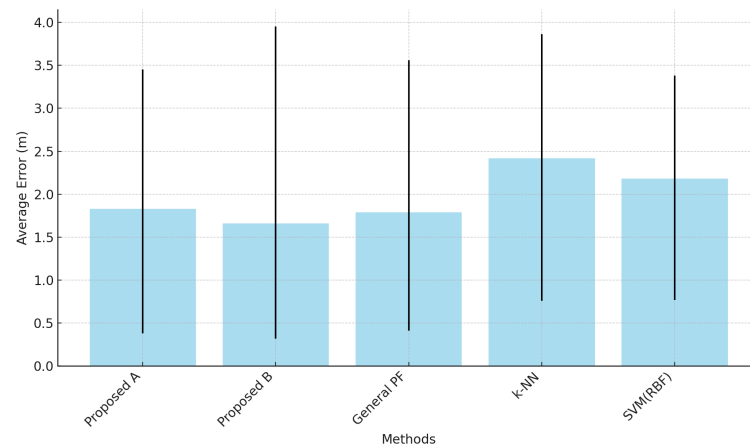
Figure 11. Error distribution in the room. The gray areas represent lockers and the yellow areas represent desks. (a) Proposed A (b) Proposed B.

4.3. Results and Error Comparison

Based on the above results, the specific errors of each method (Table 3) and the comparison of errors (Figure 12) are summarized.

Table 3. Estimation errors by proposed and existing methods.

	Proposed A	Proposed B	k-NN + General PF	k-NN	SVM (RBF)
Minimum error (m)	0.38	0.32	0.41	0.76	0.77
Maximum error (m)	3.45	3.95	3.56	3.86	3.38
Average error (m)	1.83	1.66	1.79	2.42	2.18
Std Dev (m)	0.18	0.25	0.19	0.30	0.31

**Figure 12.** Error comparison.

The proposed methods obtain higher average accuracy than k-NN and SVM. Furthermore, it can be seen that proposed B has a higher average accuracy than the general particle filtering process [39]. However, proposed A has a relatively more stable standard deviation. Proposed B has several more large errors at the wall than proposed A, but elsewhere, proposed B has relatively small errors. This makes the maximum error and variance of proposed B larger than that of proposed A. However, the average error of proposed B is the smallest. The choice of which method to use can be combined with the needs of the location estimation environment. For example, proposed B can be used in a general environment, and proposed A in case of signal instability.

To additionally validate the usability in larger environments, we used a BLE RSSI dataset [40]. This dataset contains RSSI from multiple points, and we selected a number of continuous points to simulate continuous RSSI measurements. The results of different methods are shown in Table 4. It is worth noting that because of the distribution pattern of fingerprints, k-NN sometimes has points with 0 errors. However, the proposed method has improved the average error and stability.

Table 4. Estimation errors on dataset.

	Proposed A	Proposed B	k-NN + General PF	k-NN	SVM (RBF)
Minimum error (m)	0.30	0.18	2.83	0.00	0.03
Maximum error (m)	18.59	24.08	18.26	25.88	19.84
Average error (m)	5.97	5.52	6.22	6.13	6.22
Variance (m ²)	3.75	3.38	3.87	5.24	4.66

5. Conclusions

This paper proposed a location estimation system using an improved particle filter regarding the improvement of indoor location accuracy by the BLE position fingerprinting method. Improved likelihood functions considering position coordinates in particle filters were designed based on the position fingerprinting method. Especially, by using the

likelihood function B , this method can extend position fingerprints with interpolated data and improve estimation accuracy. Meanwhile, the error of location estimation becomes large when the location is close to a wall or obstacle, which is probably because of the extended error of the position fingerprint.

Anyway, comparison and evaluation between position fingerprint methods based on only k-NN and SVM were conducted by experiments. It was confirmed that the proposed method has improved estimation accuracy and stability.

In future studies, it is considered possible to demonstrate the accuracy of location estimation drawn by the proposed method in different environments, e.g., location estimation in rooms of different shapes or sizes, in multiple rooms, for multiple people, etc. In addition, to further improve the estimation accuracies, this study will design a method for outliers in the RSSI values of reference fingerprints.

Author Contributions: J.Q., J.L., and N.K. suggested the basic concept and the algorithm demonstrated in this study. J.Q. and J.L. designed and performed the experiments. J.Q., J.L., and N.K. analyzed the data. W.-S.K. and Y.Y. provided help with algorithms and data analysis. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The RSSI dataset collected for this experiment has been uploaded to Zenodo. <https://doi.org/10.5281/zenodo.11604290> (accessed on 12 June 2024).

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

References

1. Seo, W.; Hwang, S.; Park, J.; Lee, J.M. Precise outdoor localization with a GPS-INS integration system. *Robotica* **2013**, *31*, 371–379. [CrossRef]
2. 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]
3. Niu, J.; Wang, B.; Shu, L.; Duong, T.Q.; Chen, Y. ZIL: An energy-efficient indoor localization system using ZigBee radio to detect WiFi fingerprints. *IEEE J. Sel. Areas Commun.* **2015**, *33*, 1431–1442. [CrossRef]
4. Ji, M.; Kim, J.; Jeon, J.; Cho, Y. Analysis of positioning accuracy corresponding to the number of BLE beacons in indoor positioning system. In Proceedings of the 2015 17th International Conference on Advanced Communication Technology (ICACT), Pyeong Chang, Republic of Korea, 1–3 July 2015; pp. 92–95.
5. Zhuang, Y.; Yang, J.; Li, Y.; Qi, L.; El-Sheimy, N. Smartphone-Based Indoor Localization with Bluetooth Low Energy Beacons. *Sensors* **2016**, *16*, 596. [CrossRef] [PubMed]
6. R besaat, J.; Zhang, P.; Abdelaal, M.; Theel, O. An Improved BLE Indoor Localization with Kalman-Based Fusion: An Experimental Study. *Sensors* **2017**, *17*, 951. [CrossRef] [PubMed]
7. Goldoni, E.; Savioli, A.; Risi, M.; Gamba, P. Experimental analysis of RSSI-based indoor localization with IEEE 802.15. 4. In Proceedings of the 2010 European Wireless Conference (EW), Lucca, Italy, 12–15 April 2010; pp. 71–77.
8. Bose, A.; Foh, C.H. A practical path loss model for indoor WiFi positioning enhancement. In Proceedings of the 2007 6th International Conference on Information, Communications & Signal Processing, Singapore, 10–13 December 2007; pp. 1–5.
9. Patwari, N.; Hero, A.O.; Perkins, M.; Correal, N.S.; O’dea, R.J. Relative location estimation in wireless sensor networks. *IEEE Trans. Signal Process.* **2003**, *51*, 2137–2148. [CrossRef]
10. Shu, Y.; Huang, Y.; Zhang, J.; Cou  , P.; Cheng, P.; Chen, J.; Shin, K.G. Gradient-based fingerprinting for indoor localization and tracking. *IEEE Trans. Ind. Electron.* **2015**, *63*, 2424–2433. [CrossRef]
11. Sinha, R.S.; Hwang, S.H. Comparison of CNN applications for RSSI-based fingerprint indoor localization. *Electronics* **2019**, *8*, 989. [CrossRef]
12. Zhang, W.; Liu, K.; Zhang, W.; Zhang, Y.; Gu, J. Deep neural networks for wireless localization in indoor and outdoor environments. *Neurocomputing* **2016**, *194*, 279–287. [CrossRef]
13. Shao, W.; Luo, H.; Zhao, F.; Ma, Y.; Zhao, Z.; Crivello, A. Indoor positioning based on fingerprint-image and deep learning. *IEEE Access* **2018**, *6*, 74699–74712. [CrossRef]
14. Pu, Y.C.; You, P.C. Indoor positioning system based on BLE location fingerprinting with classification approach. *Appl. Math. Model.* **2018**, *62*, 654–663. [CrossRef]
15. Yadav, R.K.; Bhattarai, B.; Gang, H.S.; Pyun, J.Y. Trusted k nearest bayesian estimation for indoor positioning system. *IEEE Access* **2019**, *7*, 51484–51498. [CrossRef]

16. Fang, Y.; Deng, Z.; Xue, C.; Jiao, J.; Zeng, H.; Zheng, R.; Lu, S. Application of an improved K nearest neighbor algorithm in WiFi indoor positioning. In Proceedings of the China Satellite Navigation Conference (CSNC) 2015 Proceedings: Volume III, Xi'an, China, 13–15 May 2015; Springer: Berlin/Heidelberg, Germany, 2015; pp. 517–524.
17. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
18. Yi, Y.; Grejner-Brzezinska, D.A. Tightly-coupled GPS/INS integration using unscented Kalman filter and particle filter. In Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2006), Fort Worth, TX, USA, 26–29 September 2006; pp. 2182–2191.
19. Fan, G.; Sheng, C.; Yu, B.; Huang, L.; Rong, Q. An Indoor and Outdoor Multi-Source Elastic Fusion Navigation and Positioning Algorithm Based on Particle Filters. *Future Internet* **2022**, *14*, 169. [[CrossRef](#)]
20. Gustafsson, F.; Gunnarsson, F.; Bergman, N.; Forssell, U.; Jansson, J.; Karlsson, R.; Nordlund, P.J. Particle filters for positioning, navigation, and tracking. *IEEE Trans. Signal Process.* **2002**, *50*, 425–437. [[CrossRef](#)]
21. Shin, S.; Park, C.; Kim, J.; Hong, H.; Lee, J. Adaptive step length estimation algorithm using low-cost MEMS inertial sensors. In Proceedings of the 2007 IEEE Sensors Applications Symposium, San Diego, CA, USA, 6–8 February 2007; pp. 1–5.
22. Ascher, C.; Kessler, C.; Wankerl, M.; Trommer, G.F. Dual IMU indoor navigation with particle filter based map-matching on a smartphone. In Proceedings of the 2010 International Conference on Indoor Positioning and Indoor Navigation, Zurich, Switzerland, 15–17 September 2010; pp. 1–5.
23. Zhang, S.; Yao, L.; Sun, A.; Tay, Y. Deep learning based recommender system: A survey and new perspectives. *ACM Comput. Surv. (CSUR)* **2019**, *52*, 1–38. [[CrossRef](#)]
24. Xie, L.; Yuille, A. Genetic cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1379–1388.
25. Smola, A.J.; Schölkopf, B. A tutorial on support vector regression. *Stat. Comput.* **2004**, *14*, 199–222. [[CrossRef](#)]
26. Altendorfer, R.; Matzka, S. A new confidence estimator for vehicle tracking based on a generalization of Bayes filtering. *IEEE Intell. Transp. Syst. Mag.* **2012**, *4*, 30–41. [[CrossRef](#)]
27. Kruschke, J.K. Bayesian estimation supersedes the t test. *J. Exp. Psychol. Gen.* **2013**, *142*, 573. [[CrossRef](#)]
28. Kalman, R.E. A new approach to linear filtering and prediction problems. *J. Basic Eng.* **1960**, *82*, 35–45. [[CrossRef](#)]
29. Doucet, A.; Godsill, S.; Andrieu, C. On sequential Monte Carlo sampling methods for Bayesian filtering. *Stat. Comput.* **2000**, *10*, 197–208. [[CrossRef](#)]
30. Zhou, L.; Li, G.; Zheng, Z.; Xiao, T. A trust region-based particle filter algorithm for indoor tracking. *Wirel. Pers. Commun.* **2015**, *80*, 739–750. [[CrossRef](#)]
31. Hammersley, J.M.; Morton, K.W. Poor man's monte carlo. *J. R. Stat. Soc. Ser. B (Methodol.)* **1954**, *16*, 23–38. [[CrossRef](#)]
32. Fu, X.; Jia, Y. An improvement on resampling algorithm of particle filters. *IEEE Trans. Signal Process.* **2010**, *58*, 5414–5420. [[CrossRef](#)]
33. Murphy, K.; Russell, S. Rao-Blackwellised particle filtering for dynamic Bayesian networks. In *Sequential Monte Carlo Methods in Practice*; Springer: Berlin/Heidelberg, Germany, 2001; pp. 499–515.
34. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
35. Houwa System Design Company. Bluetooth Low Energy Beacon Module BLEAD-B Version 2. 2017. Available online: <https://houwa-js.co.jp/service/product/blead-b/> (accessed on 19 December 2022).
36. Huawei Technologies Co., Ltd. Honor 8. 2016. Available online: <https://consumer.huawei.com/jp/support/phones/honor-8/> (accessed on 19 December 2022).
37. Zhao, Y. Study of Gaussian Kernel Functions in Support Vector Machines. Ph.D. Thesis, East China Normal University, Shanghai, China, 2007.
38. Wang, X.L.; Li, Z.B. Identifying the Parameters of the Kernel Function in Support Vector Machines Based on the Grid-Search Method. *Period. Ocean. Univ. China* **2005**, *5*, 859–862.
39. Yuan, Z.; Zhang, X.; Zhou, P.; Wang, S. Research on Indoor Position Fingerprint Location Based on Machine Learning combined Particle Filter. In Proceedings of the 2019 2nd International Conference on Safety Produce Informatization (IICSPI), Chongqing, China, 28–30 November 2019; pp. 456–459. [[CrossRef](#)]
40. Mohammadi, M. BLE RSSI Dataset for Indoor Localization. 2018. Available online: <https://www.kaggle.com/datasets/mehdimka/ble-rssi-dataset/data> (accessed on 29 May 2024).

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.