

Research Article

Mobile Robot Indoor Positioning System Based on K-ELM

Haixia Wang, Junliang Li, Wei Cui , Xiao Lu , Zhiguo Zhang, Chunyang Sheng, and Qingde Liu

Key Laboratory for Robot & Intelligent Technology of Shandong Province, Shandong University of Science and Technology, Qingdao 266590, China

Correspondence should be addressed to Wei Cui; cuiwei@sdust.edu.cn

Received 13 September 2018; Revised 15 November 2018; Accepted 4 December 2018; Published 14 February 2019

Academic Editor: Fanli Meng

Copyright © 2019 Haixia Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Mobile Robot Indoor Positioning System has wide application in the industry and home automation field. Unfortunately, existing mobile robot indoor positioning methods often suffer from poor positioning accuracy, system instability, and need for extra installation efforts. In this paper, we propose a novel positioning system which applies the centralized positioning method into the mobile robot, in which real-time positioning is achieved via interactions between ARM and computer. We apply the Kernel extreme learning machine (K-ELM) algorithm as our positioning algorithm after comparing four different algorithms in simulation experiments. Real-world indoor localization experiments are conducted, and the results demonstrate that the proposed system can not only improve positioning accuracy but also greatly reduce the installation efforts since our system solely relies on Wi-Fi devices.

1. Introduction

With the development of the global positioning system (GPS), extremely high accuracy and reliable positioning have been achieved in most outdoor environments. And it has been widely used in various fields of daily life, such as transportation, surveying, military, and other fields. Nevertheless, most of human activities are carried out indoors; the GPS devices have poor performance in indoor environments owing to its weak reception of satellites' signals. It is necessary to study the corresponding indoor positioning method because GPS cannot meet the needs of indoor positioning. Compared with the matured satellite positioning navigation system, indoor positioning technology is still in its infancy and has great application prospects. Indoor position-based service includes indoor location, navigation, surveillance, and home automation [1–3].

Over the past decade, numerous universities and research institutes have conducted in-depth research on indoor positioning technology and proposed many indoor positioning technologies such as UWB [4], RFID [5], Zigbee [6], and Bluetooth [7]. Nevertheless, all of the indoor positioning technologies require the installation of additional

infrastructure. Wi-Fi-based indoor positioning has received widespread attention because commercial Wi-Fi devices widely exist in indoor environments and a lot of mobile devices are equipped with Wi-Fi modules. Furthermore, Wi-Fi-based indoor positioning has higher accuracy than other indoor positioning technologies.

Wi-Fi indoor positioning technology can be divided into two categories: range-based and range-free. According to the way of obtaining characteristic parameters, the range-based positioning technology includes positioning based on time of arrival (TOA) [8], positioning based on time difference of arrival (TDOA) [9], positioning based on angle of arrival (AOA) [10], and positioning based on received signal strength indicator (RSSI) [11].

Wi-Fi indoor positioning techniques which do not require ranging are divided into two ways: communication hop-based and matching-based. When we utilize matching-based positioning techniques, firstly we should collect a number of reference positions in the positioning area and establish a database of position and feature parameters, then the measured characteristic parameters are compared with the database to obtain the current target position information [12]. The location based on the number of communication

hops is estimated by calculating the minimum hop count and the hop distance between the target node and the beacon node [13]; the DV-Hop algorithm is recognized as a classic algorithm to solve it. Most of the research is focused on how to improve the accuracy of the DV-Hop algorithm, and those improvements mainly include three aspects: node topology [14], interhop distance calculation method [15], and some hybrid improvement methods [16].

Mobile robots can replace people in many dangerous tasks. Therefore, mobile robots are of important significance to military reconnaissance, anti-terrorism, explosion proof, and anti-nuclearization. Of course, mobile robots also have a wide range of applications in daily life. Entering a large art or technology museum, we often get lost and hard to find exhibits we are interested in. It will be convenient to have mobile robot positioning and explanation. If a mobile robot integrated an indoor positioning system, we need a high-frequency and outstanding portability device. Additionally, simultaneous localization and mapping (SLAM) is applied to the autonomous navigation and positioning of the robot. It employs a large number of sensors such as vision, laser, and odometer to assist in positioning and constructing maps in unfamiliar environments [17–19]. Its advantage is that it can be positioned in unknown environments, and the positioning accuracy is relatively high through the cooperative operation of various sensors. Although the monocular visual SLAM has simple structure and is inexpensive, its reliability is poor. The binocular visual SLAM solved the shortcoming of monocular vision, but the cost of the whole system is high and the design is complicated, and it is limited by a certain distance. Due to a large amount of computation requirements, it is difficult to achieve real-time positioning and flexibility is worse.

Based on the above opinions, this paper designs a Wi-Fi-based indoor positioning mobile robot based on the ARM. The gec210 high-performance Cortex-A8 development board is employed as the hardware part of the mobile robot and performs data interaction during offline and real-time positioning. The high-performance development board collects signal strength data up to four times a second through the wireless network card, which can improve the real-time performance of the overall positioning system. The STM32F103ZET6 development board and the L298N motor drive board are utilized to control the robot with the high-performance development board. The computer is employed for remote monitoring; we can observe real-time positioning and the actual position of the mobile robot and control the moving route of the mobile robot.

Usage scenarios: there are many real-world applications that rely on indoor positioning, and the positioning accuracy of the requirements is diverse. Our objective in this work is to design a low-cost, high-accuracy solution for robot localization using Wi-Fi. It is necessary to point out that our system should not be used in many scenarios such as locate a robot for military reconnaissance or emergency rescue where localization error should be at centimetre or even millimeter level. Instead, our system is useful for purposes of convenience and entertainment in homes and offices where localization error of robots is not catastrophic: for instance, robots for location

finding and explanation in the museum, position detection of medical staff or equipment in hospitals, and finding tagged maintenance tools and equipment scattered all over the smart plant [20]. Both theoretical analysis and experiments are carried out to verify the capability and demonstrate the superiority of the proposed system.

The rest of the paper is organized as follows. K-ELM and fingerprint indoor location algorithms are briefly reviewed in Section 2. Section 3 introduces the detailed system design of the indoor positioning system based on the ARM. In Section 4, the experiment results of the proposed indoor location system are presented, and the performance is then analyzed. The conclusions are presented in Section 5.

2. KELM-Based Indoor Location Algorithm

Nowadays, there are many indoor positioning algorithms based on Wi-Fi. The geometric positioning algorithm and position fingerprint algorithm employed in WLAN (wireless local area network) positioning are also applicable for Wi-Fi indoor positioning. In our indoor positioning system, we do not adopt TOA, TDOA, and AOA methods due to these required additional specialized equipment to achieve indoor Wi-Fi positioning, generally applied to outdoor satellite positioning systems. The positioning method based on the propagation model is simple to implement; unfortunately, the multipath non-line-of-sight effect cannot make the model fully describe the indoor environment, leading to poor positioning accuracy [21, 22]. Compared with other positioning methods, the fingerprinting technique does not rely on precision instruments and is less affected by the indoor environment and has better positioning accuracy, which is a widely used method. Therefore, the fingerprinting technique is utilized as an indoor positioning method in this paper.

2.1. The Implementation of Fingerprinting Technique. It is acknowledged that wireless signals are easily affected by the indoor environment and gradually weaken as the distance increases. Furthermore, the signal has distinct multipath characteristics at different locations, after refraction, reflection, and scattering; the only signal associated with its transmission channel is generated. We call this multipath feature the “fingerprint” of the location of the signal. The multipath characteristics of the signal include various signal characteristics such as signal-to-noise ratio, signal strength, and signal impulse response time.

In this paper, signal strength is employed as the fingerprint feature of the location, as shown in Table 1. We combine coordinates of a sampling position and signal strength of eight access point (AP) collected at a sampling position to a matrix; the matrix is one fingerprint of a two-dimensional sampling position.

The fingerprint-based method includes two phases in the positioning process: “offline/training” and “online/positioning.” The positioning flowchart is shown in Figure 1, L represents position coordinates, and RSS is the received signal strength received from APs.

TABLE 1: Fingerprint feature.

Location	AP1	AP2	AP3	AP4	AP5	AP6	AP7	AP8
(x1, y1)	-62	-36	-53	-59	-53	-67	-68	-60
(x2, y2)	-54	-37	-47	-61	-52	-63	-67	-58
(x3, y3)	-68	-53	-45	-51	-41	-47	-68	-54
...	RSS							
(xm, ym)	RSS _{m1}	RSS _{m2}	RSS _{m3}	RSS _{m4}	RSS _{m5}	RSS _{m6}	RSS _{m7}	RSS _{m8}

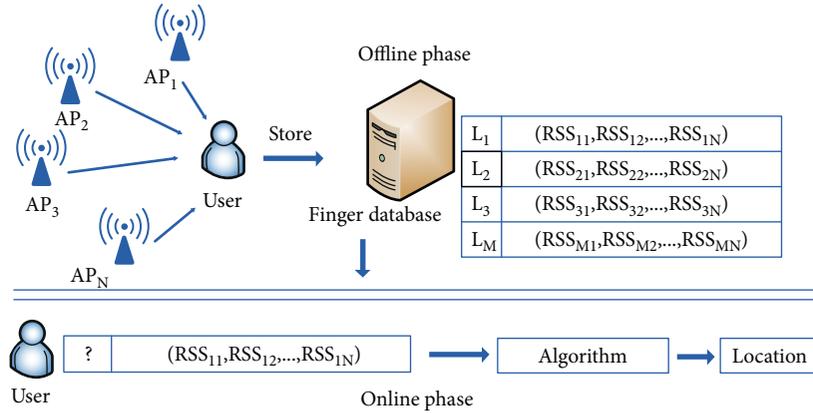


FIGURE 1: The flowchart of fingerprint localization.

(1) *Offline Training Phase.* In the offline training phase, the user mainly collects RSS and coordinates of sampling positions and constructs fingerprint datasets.

(2) *Online Positioning Phase.* The user receives the real-time signal feature vector of the surrounding AP access point at a location to be located. The matching algorithm can be used to match the RSS vector with the fingerprint in the location fingerprint database. After the fingerprinting technique finds the matching fingerprint, the location corresponding to the fingerprint in the location fingerprint database is considered to be the location of the current user. Most machine learning algorithms such as ELM, KNN, and BYS can be used for positioning, and during the offline phase, these algorithms learn the relationship functions between the location and the fingerprint. In the online phase, we employ fingerprint matching technology to match the received signal fingerprints with the fingerprint database to predict the user's location.

2.2. ELM Algorithm. The extreme learning machine (ELM) is an algorithm for solving single hidden layer feedforward networks (SLFNs) [23, 24]. We can randomly initialize the input weight and bias of the ELM, thus simply manually setting the number of hidden layer nodes to find the unique optimal solution. But traditional neural network algorithms (such as BP neural network) need to manually set a large number of training parameters in the training process and are easy to fall into the local optimum. Therefore, compared with the traditional neural network, ELM is faster than traditional neural network algorithms and has higher accuracy, so it is more suitable for indoor positioning in practical applications.

The single hidden layer feedforward neural network is a common model of feedforward neural network in which

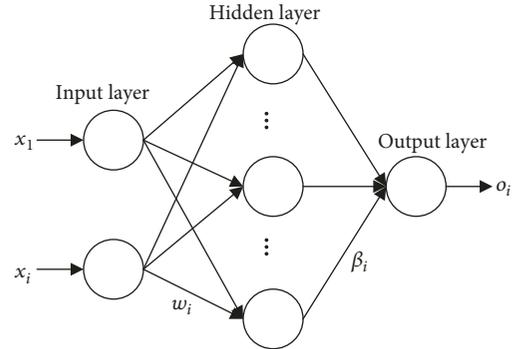


FIGURE 2: The structure of SLFNs.

the training process is the process of continuous approximation of the training samples. ELM is an extremely fast algorithm for solving single hidden layer neural networks. Its network structure is shown in Figure 2. Suppose there is M arbitrary independent sample $(\mathbf{x}_i, \mathbf{t}_i)$, where $\mathbf{x}_i = [x_{i,1}, x_{i,2}, \dots, x_{i,n}]^T \in R^n$, $\mathbf{t}_i = [t_{i,1}, t_{i,2}, \dots, t_{i,m}]^T \in R^m$. For a single hidden layer neural network with \tilde{N} hidden layer nodes, it can be described as

$$\sum_{i=1}^{\tilde{N}} \beta_i h(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) = o_j, \quad j = 1, 2, \dots, M, \quad (1)$$

where $h(x)$ is the activation function, $\mathbf{w}_i = [w_{i,1}, w_{i,2}, \dots, w_{i,m}]^T$ is the input weight matrix between the input neurons and the i -th hidden layer neurons, and $\beta_i = [\beta_{i,1}, \beta_{i,2}, \dots, \beta_{i,m}]^T$ is the output weight matrix between the i -th hidden layer neurons and the output neurons. b_i is the bias of the

i -th hidden layer neuron, \mathbf{o}_j is the corresponding output vector. $\mathbf{w}_i \cdot \mathbf{x}_j$ represents the inner product of \mathbf{w}_i and \mathbf{x}_j .

The goal of SLFN learning is to allow zero-error approximation of the sample output, namely,

$$\sum_{j=1}^M \|\mathbf{o}_j - \mathbf{t}_j\| = 0. \quad (2)$$

The existence of β_i , \mathbf{w}_i , and b_i satisfies

$$\sum_{i=1}^{\tilde{N}} \beta_i h(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) = \mathbf{t}_j, \quad j = 1, \dots, M. \quad (3)$$

Equation (3) is abbreviated as the following matrix equation:

$$H\beta = T, \quad (4)$$

where H is the hidden layer output:

$$H(\mathbf{w}_1, \dots, \mathbf{w}_{\tilde{N}}; b_1, \dots, b_{\tilde{N}}; \mathbf{x}_1, \dots, \mathbf{x}_M) = \begin{bmatrix} h(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \dots & h(\mathbf{w}_{\tilde{N}} \cdot \mathbf{x}_1 + b_{\tilde{N}}) \\ \vdots & \dots & \vdots \\ h(\mathbf{w}_1 \cdot \mathbf{x}_M + b_1) & \dots & h(\mathbf{w}_{\tilde{N}} \cdot \mathbf{x}_M + b_{\tilde{N}}) \end{bmatrix}_{M \times \tilde{N}}, \quad (5)$$

β is the output weight:

$$\beta = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\tilde{N}}^T \end{bmatrix}_{\tilde{N} \times m}, \quad (6)$$

and T is the expected output:

$$T = \begin{bmatrix} \mathbf{t}_1^T \\ \vdots \\ \mathbf{t}_M^T \end{bmatrix}_{M \times m}. \quad (7)$$

In order to train the single-layer neural network approaching the sample output with zero error, we need to make $\hat{\beta}_i$, $\hat{\mathbf{w}}_i$, and \hat{b}_i satisfy

$$\|H(\hat{\mathbf{w}}_i, \hat{b}_i) \hat{\beta}_i - T\| = \min_{\mathbf{w}_i, b_i, \beta_i} \|H(\mathbf{w}_i, b_i) \beta_i - T\|, \quad i = 1, \dots, \tilde{N}. \quad (8)$$

Equation (8) is equivalent to minimizing the loss function

$$E = \sum_{j=1}^M \left(\sum_{i=1}^{\tilde{N}} \beta_i h(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) - \mathbf{t}_j \right)^2. \quad (9)$$

The gradient descent method can be leveraged to obtain the optimal solution. The gradient descent method is achieved through continuous iteration. The iteration process of the parameter \mathbf{w} is as follows:

$$\mathbf{w}_k = \mathbf{w}_{k-1} - \eta \frac{\partial E(\mathbf{w})}{\partial \mathbf{w}}, \quad (10)$$

where η is the learning rate. The typical representative based on the gradient descent method is the BP neural network learning algorithm, which can be widely used in the feedforward neural network learning because it can efficiently carry out the gradient learning through the backpropagation of output and input. Nevertheless, the BP neural network learning algorithm has many defects: its learning rate is difficult to choose, the convergence rate of the iterative process is slow and the learning rate is small, and a larger rate may result in the algorithm being unstable and unable to converge. The algorithm is easy to fall into the local optimum due to overtraining and takes a long time.

The ELM algorithm can effectively solve the defects of the above single hidden layer neural network. In the ELM algorithm, it is not necessary to adjust the network input weight \mathbf{w}_i and the hidden layer bias b_i , and the hidden layer output matrix H is also determined after the input parameters are randomly determined.

When the algorithm starts the operation, randomly assign the weight \mathbf{w}_i between the input layer and the hidden layer and the bias b_i of the hidden layer randomly. Then, the least squares solution of Equation (8) for $\hat{\beta}_i$ is obtained as

$$\|H(\mathbf{w}_i, b_i) \hat{\beta}_i - T\| = \min_{\beta} \|H(\mathbf{w}_i, b_i) \beta - T\|, \quad i = 1, \dots, \tilde{N}, \quad (11)$$

which can be deduced by the least squares

$$\hat{\beta}_i = H^\dagger T, \quad (12)$$

where H^\dagger is the Moore-Penrose generalized inverse matrix of H .

2.3. K-ELM Algorithm. Considering that the classical ELM algorithm randomly gives the input weight and hidden layer bias may lead to instability of the output result; the literature [25] based on the traditional ELM, following the SVM learning principle, adds the kernel function to constitute the K-ELM algorithm. Numerous studies have shown that K-ELM has better generalization ability than SVM [26].

According to the optimization theory, the classical ELM algorithm (11) is rewritten as

$$\text{Minimize : } L_p = \frac{1}{2} \|\beta\|^2 + C \frac{1}{2} \sum_{i=1}^M \xi_i^2, \quad (13)$$

$$\text{Subject to : } f(x_i) = h(x_i) \beta = t_i - \xi_i,$$

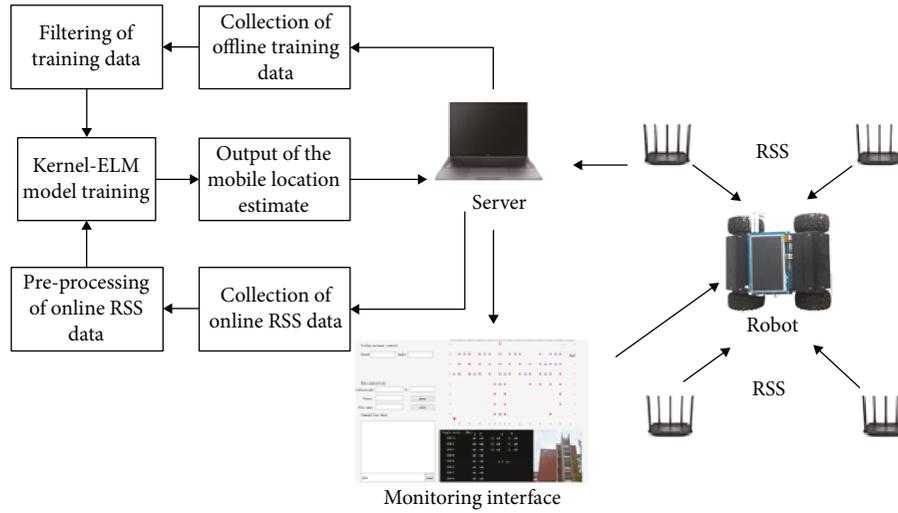


FIGURE 3: The architecture of the system.

where C is the regularization coefficient and ξ_i is the training error of the theoretical output relative to the training output.

According to KKT optimal conditions, the following solution can be obtained:

$$\beta = H^T \left(HH^T + \frac{1}{C} \right)^{-1} T. \quad (14)$$

Applying the Mercer condition to define Ω_{ELM} as the kernel matrix,

$$\begin{aligned} \Omega_{\text{ELM}} &= HH^T, \\ \Omega_{\text{ELM}(i,j)} &= h(x_i) \cdot h(x_j) = K(x_i, x_j), \end{aligned} \quad (15)$$

where $K(x_i, x_j)$ is a kernel function, which is the element of the j -th column of the i -th row of Ω_{ELM} , then the output of the hidden layer of K-ELM can be expressed as

$$f(x) = h(x) \hat{\beta} = h(x) H^T \left(HH^T + \frac{1}{C} \right)^{-1} T. \quad (16)$$

2.4. Locating Process of the Indoor Positioning System Based on K-ELM. First, we develop a software on the computer to enable the mobile robot to communicate with the server; simultaneously, we utilize software and mobile devices to collect and send RSS data from different AP in the positioning region. The whole positioning process includes two stages: offline training and online positioning.

During offline training, a mobile robot installed synchronous reception signal software for collecting RSS data at a known reference location. Using the reference point position fingerprint database $\{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_M\}$ as input and the corresponding spatial position coordinate set $\{\mathbf{l}_1, \mathbf{l}_2, \dots, \mathbf{l}_M\}$ as output, we also need to set the kernel function and the relevant parameters and determine the regularization coefficient C . Then, the hidden layer output matrix H of the network was calculated. Finally, the output weight matrix

$\hat{\beta} = H^T (HH^T + (1/C))^{-1} T$ will be calculated; at the same time, we will obtain the K-ELM model.

Online phase: enter the fingerprint database $\mathbf{r}_o = (r_{o,1}, r_{o,2}, \dots, r_{o,N})$ of the test subject received in real time, then calculate $f(\mathbf{r}_o) = h(\mathbf{r}_o) \hat{\beta}$ and output point coordinates.

The collected RSS signal contains noise errors due to its being easily disturbed by the indoor environment and other wireless signals. We utilize the smooth filter to preprocess the collected training data to eliminate the unstable and abnormal data caused by non-line-of-sight or other factors to reduce the impact of noise.

3. Indoor Positioning System Based on ARM

The existing indoor positioning system suffers from low positioning accuracy, poor stability, energy consumption, and poor expansibility. So we propose an indoor positioning system based on ARM with low energy consumption, high stability, and better accuracy.

3.1. System Architecture. The entire indoor positioning system consists of a server and a mobile robot, in which the server employs the Acer E1-470G notebook computer, and the mobile robot is controlled by a “high-performance Cortex-A8 development board GEC 210” (A8) and the “STM32 microcontroller” (S32). The server and the mobile robot communicate via the wireless network (Wi-Fi) and support the IEEE802.11b standard; its transmission rate is fast and can reach 11 Mbps. During data interaction, we leverage the TCP/IP protocol as the rule for transmitting data; the underlying Wi-Fi physical layer is called by TCP/IP when sending and receiving data. We utilize the K-ELM algorithm to train and predict on the server. The system architecture is shown in Figure 3.

In Figure 3, the mobile robot collects the Wi-Fi signal strength data which is then sent to the server via a wireless network card. The server receives and processes the data in real time and displays the positioning results on the software



FIGURE 4: The hardware architecture of the system.

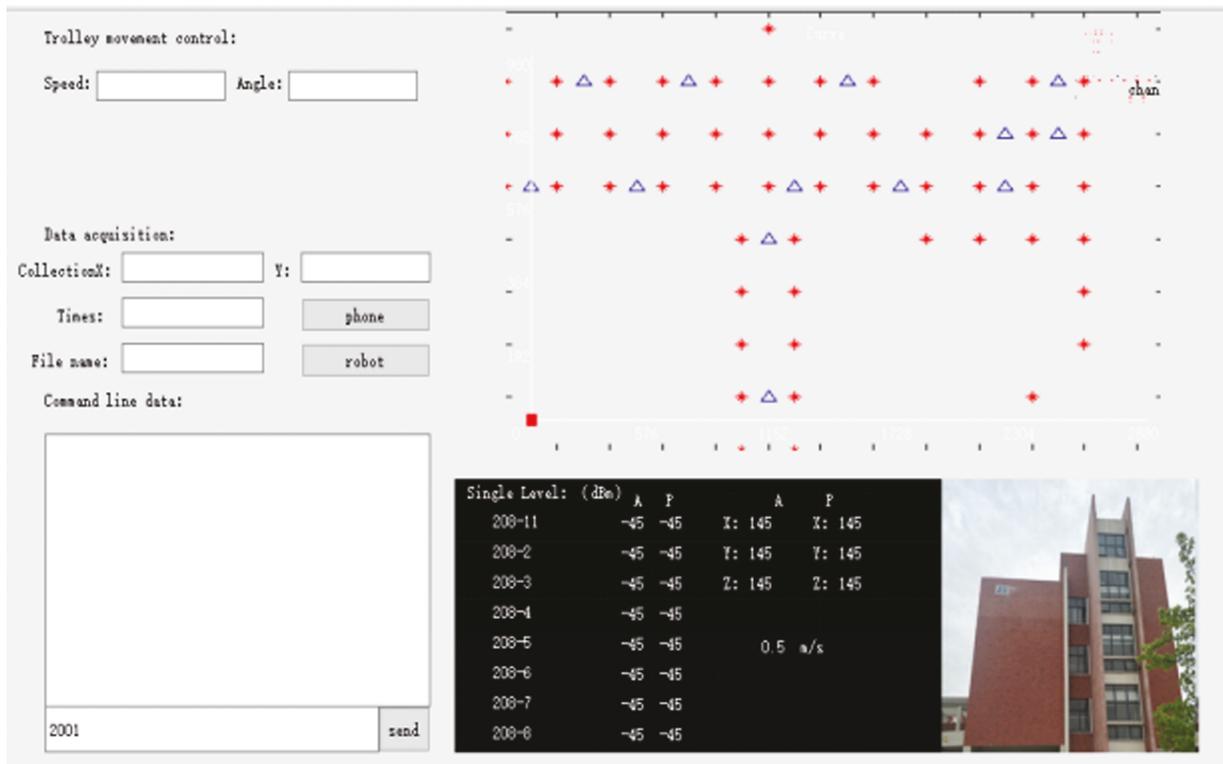


FIGURE 5: Software interface.

interface. At the same time, the movement of the robot can also be controlled by the server. The mobile robot easily adapts to a complex indoor environment through the rapid and accurate coordination of the server and the mobile robot, achieves better accuracy indoor positioning, and quickly reaches the designated target point.

3.2. Hardware Architecture of the Mobile Robot. This section describes the hardware of the mobile robot of the indoor positioning system-based Wi-Fi. The hardware architecture includes six parts, as shown in Figure 4.

The A8 in Figure 4 is the master module which receives data sent by the server and sends real-time data collecting

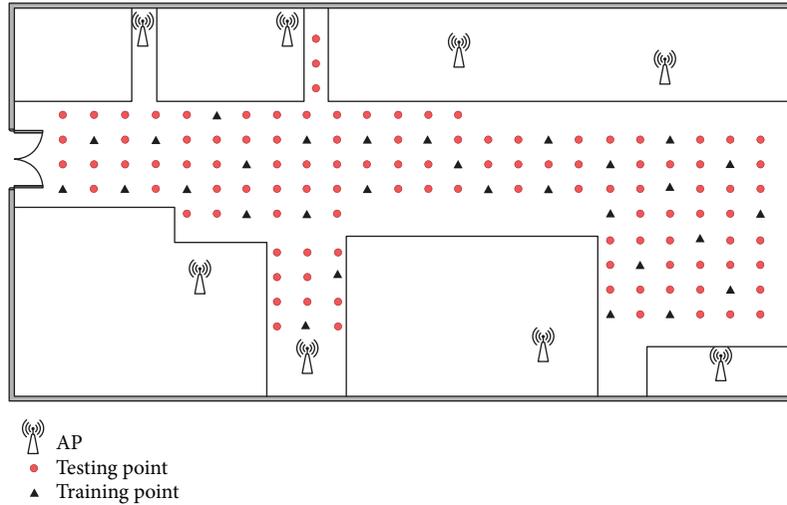


FIGURE 6: Positions of the APs, training points, and testing points in the lab.

Wi-Fi signal strength through wireless network cards to the server. Meanwhile, the S32 receives signals sent by the A8; after unpacking, the A8 cooperates with the motor drive board communicated with S32 through the universal asynchronous receiver/transmitter (UART) protocol to control the movement of the robot. The battery module is the energy supply module of the entire mobile terminal, and we select a 12 V lithium battery. Because the rated voltage of the A8 and S32 is distinct from the rated voltage of the motor drive plate, we utilize a buck module to reduce the voltage to 5 V to supply the A8 and S32, respectively, while the motor drive module was directly connected to the 12 V lithium battery.

3.3. Software. We developed the software using the Qt Creator, as shown in Figure 5. The software has outstanding portability and is fast to operate. Meanwhile, MATLAB's program can be called to process Wi-Fi data in our software.

The software interface based on the indoor Wi-Fi positioning system can monitor the movement of the mobile robot without the assistance of the camera. Meanwhile, it can directly communicate with the A8 and receive the Wi-Fi data collected by the A8. In Figure 5, the interface contains a map of the laboratory, a real-time coordinate display of the position of the robot, a console window, and a box for setting the speed of the car and the angle of turning. When collecting Wi-Fi data, we can set the collected coordinate points and the number of data collections in the interface and also display the signal strength of eight routers in real time.

4. Experimental Study

Extensive experiments were conducted to evaluate the performance of the proposed system. We first describe the setup of our testbed, and then detail the experimental results and performance evaluation.

4.1. Deployment of the Experiment. The testbed for our experiments was installed at our Industrial Robot Laboratory.

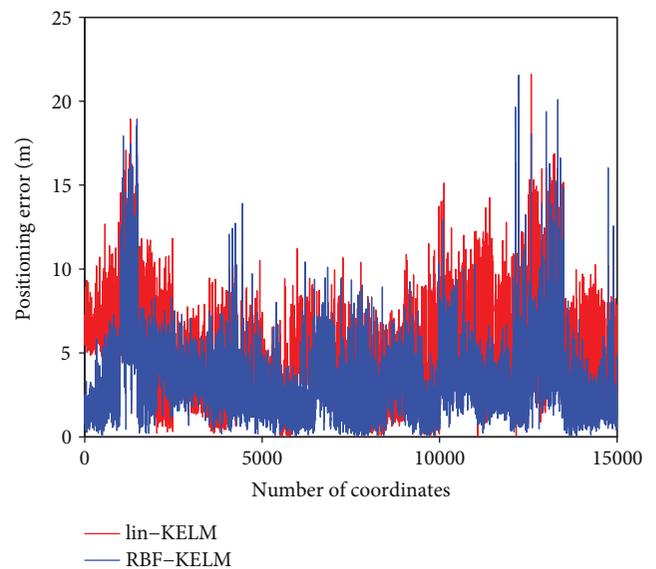


FIGURE 7: Comparison of position error distribution for different activation functions lin-kernel and RBF-kernel.

Figure 6 illustrates the layout of the lab, with a total area of around 512 m^2 ($32 \text{ m} \times 16 \text{ m}$). The circle indicates the position of the offline training reference point, and the location of the test point is marked by the triangle. Eight TP-Link WDR6500 Wi-Fi cloud routers were installed as APs for our experiments, and all the APs were fixed on a 1.2-meter-high table to keep them at the same height level.

A large number of documents demonstrate that small data sets do not clearly describe the environment and the signal characteristics, resulting in low positioning accuracy. However, if the amount of data is too large, the data acquisition will lead to high labor costs and high computation costs. In order to do a trade-off between the cost and accuracy, when we collect training data and testing data, 500 sets of RSS vector data are collected at each location, and utilizing the advantage of A8, each collection interval is set to 250 ms. Then, the scanned data is sent to the server. We

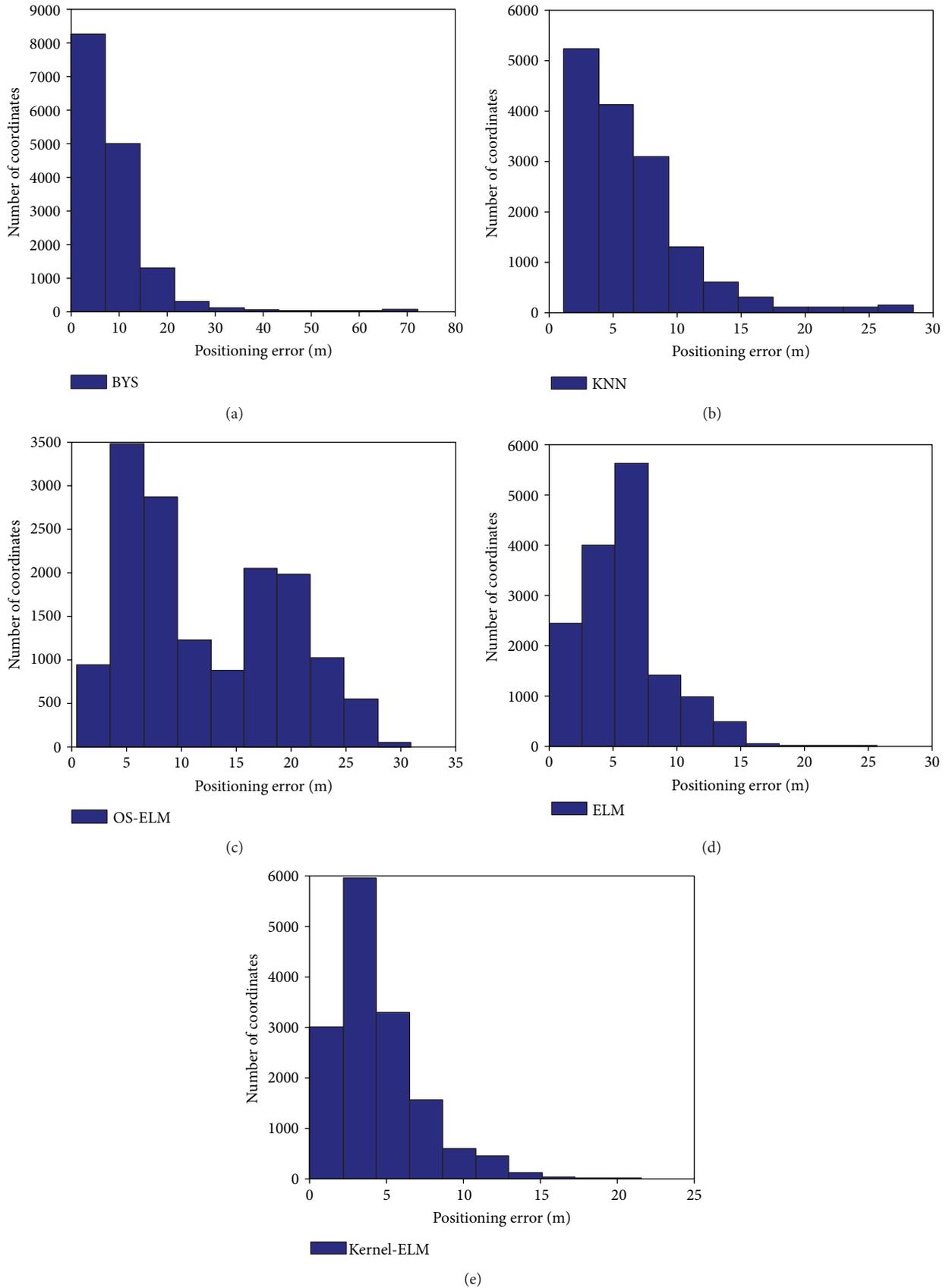


FIGURE 8: The localization error's statistical histogram of different algorithms. (a) BYS, (b) KNN, (c) ELM, (d) OS-ELM, and (e) K-ELM.

collected 53,500 sets of position fingerprint data at 107 points as a training database. A total of 15,000 sets of position fingerprint data as a test database were collected at 30 test data points. Because the dynamic environment has a great influence on RSS, the database collection is carried out in five time periods. It is noteworthy that when we use fingerprint location methods, we can locate in the same area only because we only have a fingerprint database in this area. If we collect training data sets in one area and collect test data sets in another area, the positioning accuracy will not be guaranteed and cannot meet the actual needs. After the database is pre-processed, we trained K-ELM in the server. At the stage of testing the positioning accuracy, the server preprocesses the test database and enters the trained K-ELM model, then the model outputs the position coordinates, then we compare these coordinates with the original coordinates of the test database and finally get the positioning accuracy.

4.2. Selection of Model Parameters. Based on the description of the ELM algorithm in Section 2.2, it can be seen that the activation function and the number of hidden layer neurons are important parameters in the ELM model, which have a great influence on the accuracy of constructing the ELM prediction model during offline training. In the K-ELM algorithm, the kernel function type also has a great influence on the accuracy of the algorithm.

The algorithm positioning error statistics for the K-ELM of lin-kernel and RBF-kernel are shown in Figure 7. As we can see from the figure, the RBF-kernel has better average positioning error. Therefore, we apply the RBF-kernel as the kernel function of the K-ELM algorithm in our indoor positioning system.

4.3. Experimental Results and Evaluation. To evaluate the positioning performance of the K-ELM-based positioning system, KNN (k-nearest neighbor) [27], BYS (Bayesian) [28, 29], classic ELM [30], and OS-ELM (online sequential ELM) [31] algorithms are utilized for comparison; we compared the above algorithms' experimental result with the results of K-ELM in the positioning system.

To evaluate the experimental test results, we apply evaluation indicators: RMSE (root mean square error) and STD (standard deviation):

$$\begin{aligned} \text{RMSE} &= \frac{1}{s} \sum_{i=1}^s \left\| \mathbf{l}_i - h_i \hat{\boldsymbol{\beta}} \right\|, \\ \text{STD} &= \sqrt{\sum_{i=1}^s \left(\left\| \mathbf{l}_i - h_i \hat{\boldsymbol{\beta}} \right\| - \frac{1}{s} \sum_{i=1}^s \left\| \mathbf{l}_i - h_i \hat{\boldsymbol{\beta}} \right\| \right)^2}, \end{aligned} \quad (17)$$

where s is the total number of test samples. In this experiment, there are 30 test points, and we collect 500 data at each test point, so s is 15,000.

The statistical histogram of the positioning error of each algorithm is shown in Figure 8. After analyzing the error statistics histogram, we know that the K-ELM algorithm has a smaller positioning error than others, and its distribution is more concentrated and its stability is better.

TABLE 2: Experimental results.

Algorithm	RMSE (m)	STD (m)	Training time (s)	Testing time (s)
ELM	5.8424	3.2616	0.9656	6.2×10^{-3}
OSELM	12.3160	6.9533	0.4688	3.2×10^{-3}
KNN	6.3185	4.4342	349	≈ 0
BYS	6.9662	5.2584	10.515	≈ 0
K-ELM	2.9322	2.4178	60.847	1.8×10^{-2}

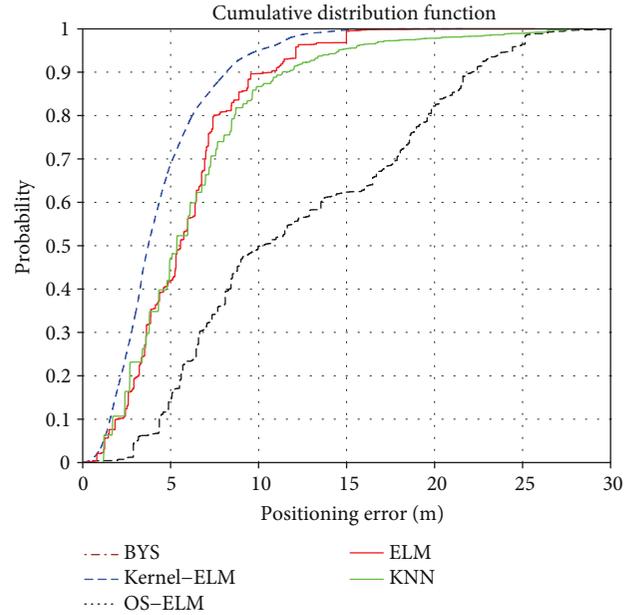


FIGURE 9: Cumulative percentile of localization error for different methods.

Table 2 shows the positioning results of ELM, OS-ELM, KNN, BYS, and K-ELM. From Table 2, it can be seen that ELM, OS-ELM, and BYS have extremely fast training speeds. Although K-ELM positioning is not as good as the above algorithm, its training time takes only 60.847s, which can meet the requirements of actual application. The RMSE of K-ELM is the highest in Table 2; up to 2.4178 m, ELM is followed by 3.2616 m, and the RMSE of other algorithms is above 4 m.

Figure 9 shows the cumulative error diagram of all algorithms. Obviously, the positioning system based on the K-ELM algorithm has better positioning accuracy and stability than other algorithms. And from the above analysis, it can be seen that the operating speed of the K-ELM algorithm is sufficient to meet the requirements of the application. To sum up, the K-ELM algorithm performs better than other algorithms in indoor test scenarios do.

In the online positioning phase, the server preprocesses the test data and enters the trained K-ELM model, after calculation using MATLAB, and outputs the position coordinates, and then the server sends the position coordinates to the mobile robot; finally, the user can obtain the position of the mobile robot. The real-time display of the server interface

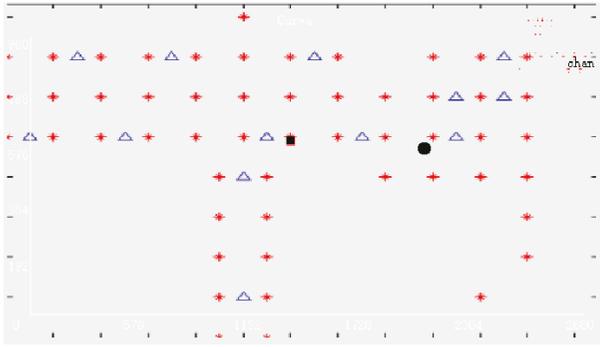


FIGURE 10: Map display of anchor points.

at the actual test is shown in Figure 10, where the square is marked as a predicted position.

5. Conclusion

In this paper, a new real-time positioning system for mobile robot is proposed based on the centralized positioning method. We clearly describe the design idea and function realization based on the fingerprint matching method. Meanwhile, the K-ELM algorithm is compared with the KNN, OS-ELM, BYS, and classical ELM algorithms, and the actual test results demonstrate that the K-ELM algorithm applied in our indoor positioning system can achieve better positioning accuracy and higher efficiency than other algorithms can. In general, the real-time mobile robot system designed and developed in this paper has lower cost, simple installation and maintenance, and high positioning accuracy. It is notable that our system has the drawbacks of the Wi-Fi based positioning method; the approach suffers from relatively low accuracy, time-consuming and laborious data collection, and device heterogeneity.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61503224, 61773245, 61473177, 61603068, 61806113), Natural Science Foundation of Shandong Province (ZR2018ZC0436, ZR2018PF011, ZR2018BF020), Key Research and Development Program of Shandong Province (2018GGX101053), China Postdoctoral Science Foundation (2016T90640), Scientific Research Foundation of Shandong University of Science and Technology for Recruited Talents (2017RCJJ061, 2017RCJJ062, 2017RCJJ063), and Taishan Scholarship Construction Engineering.

References

- [1] W. Cui, C. Wu, Y. Zhang, B. Li, and W. Fu, "Indoor robot localization based on multidimensional scaling," *International Journal of Distributed Sensor Networks*, vol. 11, no. 8, Article ID 719658, 2015.
- [2] R. F. Brena, J. P. García-Vázquez, C. E. Galván-Tejada, D. Muñoz-Rodríguez, C. Vargas-Rosales, and J. Fangmeyer, "Evolution of indoor positioning technologies: a survey," *Journal of Sensors*, vol. 2017, Article ID 2630413, 21 pages, 2017.
- [3] W. Shao, F. Zhao, C. Wang et al., "Location fingerprint extraction for magnetic field magnitude based indoor positioning," *Journal of Sensors*, vol. 2016, Article ID 1945695, 16 pages, 2016.
- [4] W. Cui, C. Wu, W. Meng, B. Li, Y. Zhang, and L. Xie, "Dynamic multidimensional scaling algorithm for 3-d mobile localization," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 12, pp. 2853–2865, 2016.
- [5] H. Zou, L. Xie, Q.-S. Jia, and H. Wang, "Platform and algorithm development for a rfid-based indoor positioning system," *Unmanned Systems*, vol. 2, no. 3, pp. 279–291, 2014.
- [6] S.-H. Fang, C.-H. Wang, T.-Y. Huang, C.-H. Yang, and Y.-S. Chen, "An enhanced zigbee indoor positioning system with an ensemble approach," *IEEE Communications Letters*, vol. 16, no. 4, pp. 564–567, 2012.
- [7] S. Feldmann, K. Kyamakya, A. Zapater, and Z. Lue, "An indoor bluetooth-based positioning system: concept, implementation and experimental evaluation," in *Proceedings of the International Conference on Wireless Networks*, vol. 272, Las Vegas, NV, USA, June 2003.
- [8] I. Guvenc and Z. Sahinoglu, "Threshold-based TOA estimation for impulse radio UWB systems," in *2005 IEEE International Conference on Ultra-Wideband*, Zurich, Switzerland, September 2005.
- [9] W. Meng, L. Xie, and W. Xiao, "TDOA sensor pairing in multi-hop sensor networks," in *Proceedings of the 11th international conference on Information Processing in Sensor Networks*, Beijing, China, April 2012.
- [10] Y. S. Lee, J. W. Park, and L. Barolli, "A localization algorithm based on AOA for ad-hoc sensor networks," *Mobile Information Systems*, vol. 8, no. 1, p. 72, 2012.
- [11] W. Cui, L. Zhang, B. Li et al., "Received signal strength based indoor positioning using a random vector functional link network," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 5, pp. 1846–1855, 2018.
- [12] X. Luo, W. J. O'Brien, and C. L. Julien, "Comparative evaluation of Received Signal-Strength Index (RSSI) based indoor localization techniques for construction jobsites," *Advanced Engineering Informatics*, vol. 25, no. 2, pp. 355–363, 2011.
- [13] H. Xu, Y. Tu, W. Xiao, Y. Mao, and T. Shen, "An archimedes curve-based mobile anchor node localization algorithm in wireless sensor networks," in *2010 8th World Congress on Intelligent Control and Automation*, Jinan, China, July 2010.
- [14] Z. Zhong and T. He, "RSD: a metric for achieving range-free localization beyond connectivity," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 11, pp. 1943–1951, 2011.
- [15] J. Lee, W. Chung, and E. Kim, "A new range-free localization method using quadratic programming," *Computer Communications*, vol. 34, no. 8, pp. 998–1010, 2011.
- [16] P. Brída, J. Machaj, and J. Duha, "A novel optimizing algorithm for DV based positioning methods in ad hoc networks," *Elektronika ir Elektrotechnika*, vol. 1, no. 97, pp. 33–38, 2010.

- [17] J. Fuentes-Pacheco, J. Ruiz-Ascencio, and J. M. Rendón-Mancha, "Visual simultaneous localization and mapping: a survey," *Artificial Intelligence Review*, vol. 43, no. 1, pp. 55–81, 2015.
- [18] B. Olszewski, S. Fenton, B. Tworek, J. Liang, and K. Yelamarthi, "RFID positioning robot: an indoor navigation system," in *IEEE International Conference on Electro-Information Technology, EIT 2013*, Rapid City, SD, USA, May 2013.
- [19] C. Cadena, L. Carlone, H. Carrillo et al., "Past, present, and future of simultaneous localization and mapping: Toward the robust-perception age," *IEEE Transactions on Robotics*, vol. 32, no. 6, pp. 1309–1332, 2016.
- [20] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1067–1080, 2007.
- [21] W. Cui, B. Li, L. Zhang, and W. Meng, "Robust mobile location estimation in NLOS environment using GMM, IMM, and EKF," *IEEE Systems Journal*, vol. 99, pp. 1–11, 2018.
- [22] B. Li, W. Cui, and B. Wang, "A robust wireless sensor network localization algorithm in mixed LOS/NLOS scenario," *Sensors*, vol. 15, no. 9, pp. 23536–23553, 2015.
- [23] G.-B. Huang, Q.-Y. Zhu, and C.-K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No.04CH37541)*, Budapest, Hungary, July 2004.
- [24] G. Huang, S. Song, J. N. D. Gupta, and C. Wu, "Semi-supervised and unsupervised extreme learning machines," *IEEE Transactions on Cybernetics*, vol. 44, no. 12, pp. 2405–2417, 2014.
- [25] G.-B. Huang and C. K. Siew, "Extreme learning machine: RBF network case," in *ICARCV 2004 8th Control, Automation, Robotics and Vision Conference, 2004*, p. 2, Kunming, China, December 2004.
- [26] G.-B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 2, pp. 513–529, 2012.
- [27] D. Wu, Y. Xu, and L. Ma, "Research on RSS based indoor location method," in *2009 Pacific-Asia Conference on Knowledge Engineering and Software Engineering*, pp. 205–208, Shenzhen, China, December 2009.
- [28] Y. Liu, Z. Yang, X. Wang, and L. Jian, "Location, localization, and localizability," *Journal of Computer Science and Technology*, vol. 25, no. 2, pp. 274–297, 2010.
- [29] L. Chen, L. Pei, H. Kuusniemi, Y. Chen, T. Kröger, and R. Chen, "Bayesian fusion for indoor positioning using Bluetooth fingerprints," *Wireless Personal Communications*, vol. 70, no. 4, pp. 1735–1745, 2013.
- [30] X. Lu, H. Zou, H. Zhou, L. Xie, and G. B. Huang, "Robust extreme learning machine with its application to indoor positioning," *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 194–205, 2016.
- [31] H. Zou, H. Jiang, X. Lu, and L. Xie, "An online sequential extreme learning machine approach to WiFi based indoor positioning," in *2014 IEEE World Forum on Internet of Things (WF-IoT)*, pp. 111–116, Seoul, South Korea, March 2014.

