An Indoor Passive Positioning Method Using CSI Fingerprint Based on Adaboost

Yong Zhang, Dapeng Li, and Yujie Wang

***Abstract*— Wi-Fi fingerprint-based indoor positioning systems have extensive application prospects. Indoor positioning based on channel state information (CSI) becomes a research hotspot because it can provide fine-grained information. This paper proposes the Adaboost positioning system (ABPS) that uses the phase information in CSI and the ensemble learning (EL) method to train the fingerprint map. In this system, the abnormal phase data is eliminated by density-based clustering and the remainder is linearly transformed to build the fingerprint map. Through continuous iteration with the Adaboost algorithm, the sample weights of the training sets are continuously adjusted to prepare for classification. Finally, it can achieve position coordinates regression by means of confidence level. A series of experiments have been conducted to illustrate the effects of EL parameters, such as the number of iterations, the maximum depth of decision tree, the maximum number of features, and the minimum number of samples required for node subdivision. Meanwhile, the influence of input data sets size is investigated in an open environment and a complex laboratory environment. The experimental results of various classification and regression methods are also discussed, which validates the effectiveness of the proposed method.**

***Index Terms*— Indoor positioning, channel state information (CSI), ensemble learning, Adaboost.**

I. Introduction

POSITIONING methods are promoted by huge commercial demands for location-based services, such as Global Positioning System (GPS) [1], Bluetooth [2], RFID [3], Zigbee [4], Ultra-wideband [5], WiFi [6], etc. Different methods can be used to achieve different goals in different scenarios. The majority of these methods depend on portable devices to obtain current status, which is known as active positioning. On the other hand, passive positioning [7] does not need any portable equipment, it only needs the target in the sensing area, which is more suitable for the practical application scenarios.

An enormous amount of research effort goes into WLAN-Based indoor positioning, including received signal strength (RSS) [8], [9], angle of arrival (AOA) [10], time of arrival (TOA) [11], [12] and the CSI [13], [14]. RSS is widely used because it is easy to access. The Horus system [15] builds a fingerprint map based on RSS, and counts the probability distribution of signal strength at the fingerprint point. However, RSS is a superposition of indoor multipath, Non-line-of-sight (NLOS) paths inevitably appear in practical environments, which will have a serious impact on the positioning accuracy. In NLOS propagation environment, the NLOS error caused by reflections, diffractions, and scatterings of signals is considered as obstacle in signal propagation [16], [17]. The CSI is based on physical layer, describing both the amplitude and phase characteristics, which is more sensitive to changes in the environment [18]. Therefore, using CSI for positioning will be better than RSS [19]–[21]. The FILA positioning method [22] first uses the physical layer CSI information in Orthogonal Frequency Division Multiplexing (OFDM) to establish a signal propagation model, and the FIFS system [23] uses the amplitude information of subcarriers to establish fingerprint map. Since the amplitude information is greatly hindered by other facilities while the phase information is less affected [24], the phase information is also widely used in indoor position and other researches. CSI-MIMO [25] uses the difference matrix of the amplitude and the difference matrix of the phase as fingerprint features. PhaseFi [26] uses the phase information to train a deep network with three hidden layers and stores the weights to build feature-based fingerprint database.

Fingerprint map-based positioning methods are mostly associated with machine learning algorithm. The K nearest neighbor (KNN) method [27] selects K classification labels of the shortest distance between real-time CSI and different fingerprints. The Naive_Bayes method [28],[29] relies on the independence assumption between predictors, which means that it treats each fingerprint point as a separate entity with no relationship. The Support Vector Machine (SVM) method [30], [31] uses kernel functions to address nonlinear classification. EL [32] uses a series of learners to learn and uses a certain rule to integrate individual learning results to achieve a better learning outcome than a single learner. Leo Breiman combines Bagging's EL theory with a random subspace method to propose an EL model based on decision tree, called random forest. The AdaBoost [33] and the Gradient Boost Decision Tree (GBDT) are machine learning algorithms based on the Boosting's EL idea. By adjusting the weights of the samples which are erroneously learned, these samples are processed in the subsequent learning.

In ABPS, the data is processed in three steps. In the pre-processing step, the abnormal phase data of each fingerprint

Manuscript received February 15, 2019; revised March 17, 2019; accepted March 20, 2019. Date of publication March 25, 2019; date of current version June 19, 2019. This work was supported by the National Natural Science Foundation of China under Grant 61801162. The associate editor coordinating the review of this paper and approving it for publication was Prof. Kazuaki Sawada. *(Corresponding author: Dapeng Li.)*

The authors are with the School of Computer and Information, Hefei University of Technology, Hefei 230001, China (e-mail: hfgdwhb@163.com; jackiechen@mail.hfut.edu.cn; wjiejie@hfut.edu.cn).

Digital Object Identifier 10.1109/JSEN.2019.2907109

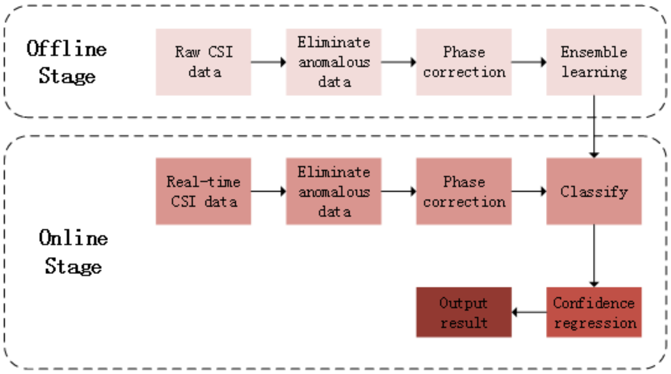


Fig. 1. The architecture of ABPS.

point is eliminated by Density-Based Spatial Clustering of Applications with Noise (DBSCAN), and the acquired phase information is calibrated as the fingerprint feature. And then, the Adaboost algorithm is used to perform the offline training by continuously updating the weights and to classify the test data in the stage of online learning. The position of the test points are estimated by confidence regression in the last step.

The main contributions of this work are as follows:

1) Instead of using filtering methods [34], we use density-based clustering method to eliminate abnormal data, which can find abnormal data effectively.
2) We use the Adaboost method to combine multiple weak classifiers to improve the classification accuracy, which has a good effect compared with the single classifier.
3) To improve the accuracy of positioning, we use the method of confidence regression. Compared with Principle Component Regression (PCR) and Ridge, the positioning accuracy can be increased by 1.5 meters.

The rest of this paper is organized as follows: Section II introduces the ABPS structure based on the Adaboost algorithm. Section III introduces the CSI data format and pre-processing. Experimental results with various methods under different conditions are discussed in Section IV. Finally, we draw the conclusion in Section V.

II. ABPS Structure

In this section, we first give an overview of the system architecture, and then introduce the Adaboost algorithm and the confidence regression method.

A. ABPS

The architecture of ABPS is shown in Fig.1. The system is divided into two parts, the offline training stage and the online positioning stage.

Offline stage: At this stage, the main function is to collect the raw CSI information and build a fingerprint map. We collect data through the IWL 5300 NIC and then the raw data is pre-processed. The pre-processing stage includes eliminating the abnormal data and calibrating the phase. The calibrated phase data is used as fingerprint feature to train fingerprint map by Adaboost algorithm.

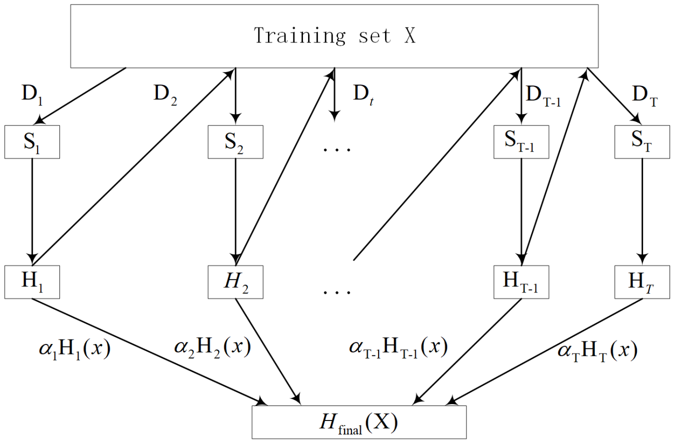


Fig. 2. The flow chart of the Adaboost algorithm.

Online stage: The real-time CSI data of the test points is processed by DBSCAN and then calibrate the phase. The calibrated phase data is also classified by Adaboost algorithm. The results of the classification are regressed in a confidence-regressive manner.

B. Adaboost Algorithm

The goal of Adaboost is to train a series of weak classifiers from the training data and then combine these weak classifiers into a strong classifier. Fig. 2 shows the flow chart of the Adaboost algorithm and the steps are as follows:

1) Initialize the weight distribution of the training data so that each sample is given the same weight:$w_i = 1/M$, $i = 1, \cdots, M$, where M represents the total number of samples. Thus, the initial weights distribution $D_1(i)$ of the training data set satisfies:

$$D_1(i) = (w_1, w_2, \cdots, w_M) = (1/M, 1/M, \cdots, 1/M) \quad (1)$$

2) Perform iterative operations to update the weights of the sample set. S_t represents the data set under the D_t, $t = 1, \cdots, T$, where T represents the number of iterations.

A) Select a weak classifier with the current minimum error rate as the tth basic classifier (H_t), and calculate the error of H_t. The error of H_t on the distribution $D_t(i)$ is:

$$e_t = P(H_t(x_i) \neq y_i) = \sum_{i=1}^{M} w_{ti} I(H_t(x_i) \neq y_i) \quad (2)$$

where x_i and y_i represent the ith training set and label, respectively.

B) Calculate the weight of the basic classifier in the final classifier (basic classifier weight is denoted by α_t):

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1-e_t}{e_t}\right) \quad (3)$$

C) Update the weight distribution of training samples:

$$D_{t+1} = \frac{D_t(i)\exp\left(-\alpha_t y_i H_t(x_i)\right)}{Z_t} \quad (4)$$

where Z_t is the normalization constant,

$$Z_t = 2\sqrt{e_t(1-e_t)} \tag{5}$$

3) Combine the basic classifiers according to the basic classifier weight α_t:

$$f(x) = \sum_{t=1}^{T} \alpha_t H_t(x) \tag{6}$$

Get a strong classifier through the symbolic function $sign$:

$$H_{final} = sign(f(x)) = sign\left(\sum_{t=1}^{T} \alpha_t H_t(x)\right) \tag{7}$$

We choose the Classification And Regression Tree (CART) as the weak classifier. Its parameters mainly include:

(1) The maximum number of features considered in the division (max_features): If the fingerprint points have many features, we can adjust generation time of the decision tree by changing the parameter.

(2) Maximum depth of the decision tree (max_depth): By limiting the depth of the tree, the size of the decision tree can be controlled to prevent over-fitting.

(3) Minimum number of the samples required for internal node subdivision (min_samples_split): This parameter limits the conditions under which the subtree continues to be partitioned. If the number of samples of a node is less than this value, it will not continue to try to select the optimal feature for partitioning.

(4) The minimum number of samples of the leaf node (min_samples_leaf): This parameter limits the minimum number of samples of the leaf node. If the number of leaf nodes is smaller than the number of samples, it will be pruned together with the sibling node(Pruning refers to deleting all the child nodes of a subtree and using the root node as a leaf node).

The output of the strong classifier is the final classification result, and then we use the result to regress the coordinates.

C. Confidence Regression

Setting the coordinates as labels, the classification results of each test point are counted and the top four classification results are selected. Position coordinates are expressed as:

$$L_i(x, y), \quad i = 1, 2, 3, 4 \tag{8}$$

The numbers of corresponding classification results are n_1, n_2, n_3, n_4. The total number of these four classification results is recorded as N:

$$N = n_1 + n_2 + n_3 + n_4 \tag{9}$$

Then calculate the probability of each coordinate L_i in these results:

$$P_i = \frac{n_i}{N} \tag{10}$$

Finally, the obtained four coordinates are weighted to obtain the final output coordinates according to the probability distribution result:

$$position_x = \sum_{i=1}^{4} L_i(x) P_i \tag{11}$$

$$position_y = \sum_{i=1}^{4} L_i(y) P_i \tag{12}$$

III. CSI Data Format and Pre-Processing

A. CSI Data Format

With the development of digital wireless communication technology, OFDM technology has been adopted as a physical layer standard. The OFDM technique can be used to divide the total spectrum into multiple orthogonal subcarriers and transmit data on the subcarriers using the same modulation scheme to combat frequency selective fading in a multipath propagation environment.CSI is physical layer information which can be obtained by a device conforming to the IEEE 802.11a / g / n WiFi standard, such as the IWL 5300 NIC. The raw CSI data format is $m \times n \times 30$, where m and n represent the number of transmitted and received antennas, respectively. There are 30 subcarriers frequency, and each subcarrier corresponds to a complex matrix of $m \times n$. The signal propagation model can be expressed as:

$$Y = HX + N \tag{13}$$

where Y, X, H and N represent the received signal vector, the transmitted signal vector, the channel gain matrix, and the channel noise matrix, respectively. H can be expressed as:

$$H = \begin{bmatrix} H_{1.1} & H_{1.2} & \cdots & H_{1.M} \\ H_{2.1} & H_{2.2} & \cdots & H_{2.M} \\ \vdots & \vdots & \ddots & \vdots \\ H_{m\times n.1} & H_{m\times n.2} & \cdots & H_{m\times n.M} \end{bmatrix} \tag{14}$$

Each item in the formula is a complex number representing the magnitude and phase of the channel state, where M represents different subcarrier numbers. Therefore, Y can also be expressed as:

$$Y = \|Y\| \bullet e^{j\times\angle Y} \tag{15}$$

where $\|Y\|$ and $\angle Y$ represent the amplitude and the phase of the received signal, respectively. Because the phase of the signal periodically changes over the propagation distance, which is more robust than amplitude, especially in cluttered propagation environments, we select the phase information as the fingerprint feature in this paper.

B. Eliminate Anomalous Data

For wireless communications, the received signal is typically affected by channel state, such as path loss and multipath fading, which causes the received signal to inevitably fluctuate or even vary greatly. This paper uses DBSCAN method to find out the abnormal data and eliminate them.

The DBSCAN algorithm utilizes the concept of density-based clustering, which requires that the number of objects contained in a certain region of the cluster space is not less than a given threshold. It is also possible to find clusters of arbitrary shape in a spatial database with noise, and connect adjacent regions of sufficient density so as to process abnormal data effectively.

The processing of the DBSCAN algorithm in the proposed system can be expressed as follows:

1) DBSCAN searches for clusters by examining the neighborhood of each point in the data set. If the neighborhood of point p contains more points than the minimum, then create a cluster with p as the core object;
2) DBSCAN iteratively aggregates objects that are directly reachable from these core objects;
3) The process ends when there is no new point added to any clusters.

The DBSCAN method is used to process the data as shown in Figure 3a, 3b. Figure 3a shows fingerprint point data with one abnormal data marked as noise. The DBSCAN method is used to find out the abnormal data and remove it. Figure 3b shows the clustering results after eliminating the abnormal data by the DBSCAN method.

C. Calibrating the Phase

Due to the synchronization error between the transmitting port and the receiving port, the CSI will contain the clock synchronization error and the residual error of the carrier frequency synchronization. For residual errors in the phase, it can be eliminated by linear transformation. The processed phase can be obtained by the following method:

$$\angle Y_{\text{mod}\,i} = \angle Y_{raw_i} - A wave_i - b \tag{16}$$

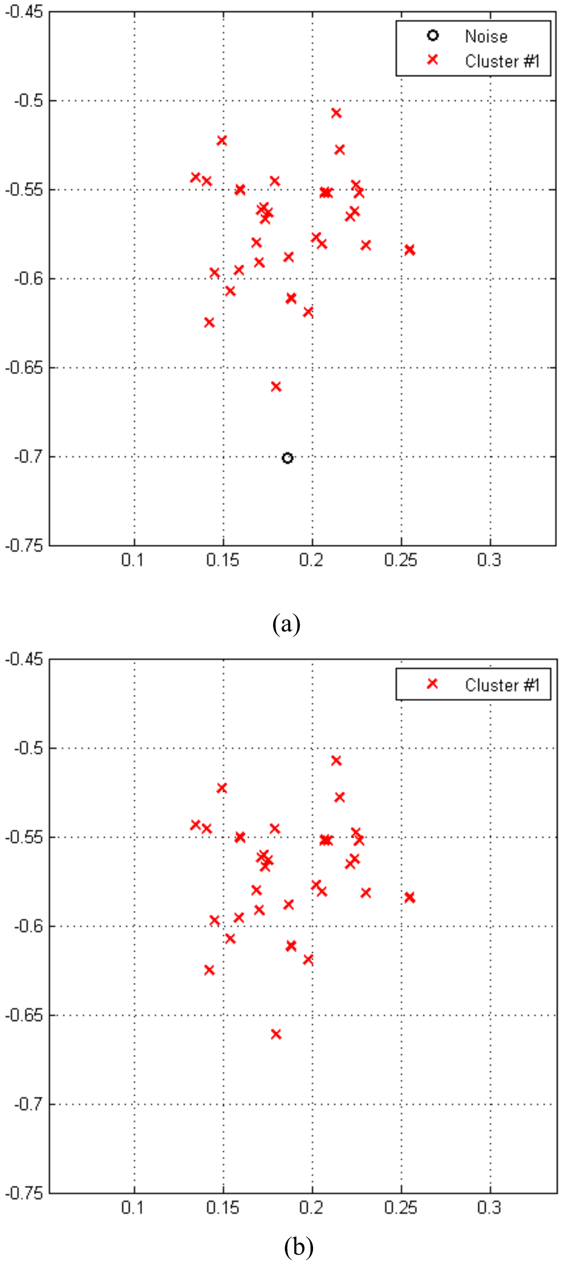
where $\angle Y_{\text{mod}\,i}$, $\angle Y_{raw_i}$, A and b represent respectively the calibrated phase of the ith subcarrier, the uncalibrated phase of the ith subcarrier, and the two parameters of the linear transformation. And $wave_i$ represents the index value of the ith subcarrier. This index conforms to the IEEE 802.11n standard. $i \in (1-30)$, $wave_i \in [(-28)-(+28)]$. Let:

$$A = \frac{\angle Y_{raw_{30}} - \angle Y_{raw_1}}{wave_{30} - wave_1}, \quad b = \frac{\sum_{i=1}^{30} \angle Y_{raw_i}}{30} \tag{17}$$

so the calibrated phase can be obtained by formula (16). The raw and calibrated phase data from the same subcarrier are shown in Fig. 4. The dotted circle indicates the amplitude of the signal, and 0° to 360° indicates the phase of the signal. The raw and calibrated phase data are indicated in red and blue, respectively. It can be seen that the raw phases distribute in 0° to 360°, and the distribution is irregular while the calibrated phase is concentrated in the range of 60° to 90°. It can be seen that the calibrated phase eliminates residual errors, which is suited to be used as the features for indoor positioning.

IV. Experiments and Discussion

In this chapter, we will discuss the experimental conditions and the experimental results, including the establishment of fingerprint map, the influences of Adaboost parameters, the number of antennas, different classification and regression methods.



(a)

(b)

Fig. 3. Clustering results at the same fingerprint point: (a) before eliminating abnormal data points, (b) after eliminating abnormal data points.

A. Experimental Condition

In this paper, the TL-WDR6500 is used as the Access Port (AP). The desktop computer equipped with the Ubuntu 12.04 system and the IWL 5300 NIC is used as the Receive Port (RP). The AP has two 5GHz transmit antennas and the RP has three receive antennas. The received data format is 2 × 3 × 30. The experimental scenes are shown in Figures 5a, 5b. Figure 5a shows a relatively open environment. Figure 5b shows the laboratory environment where multipath effects are aggravated by the influence of facilities such

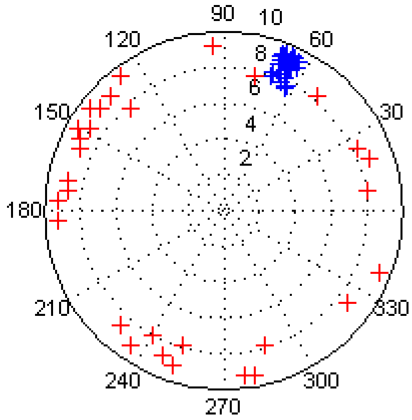
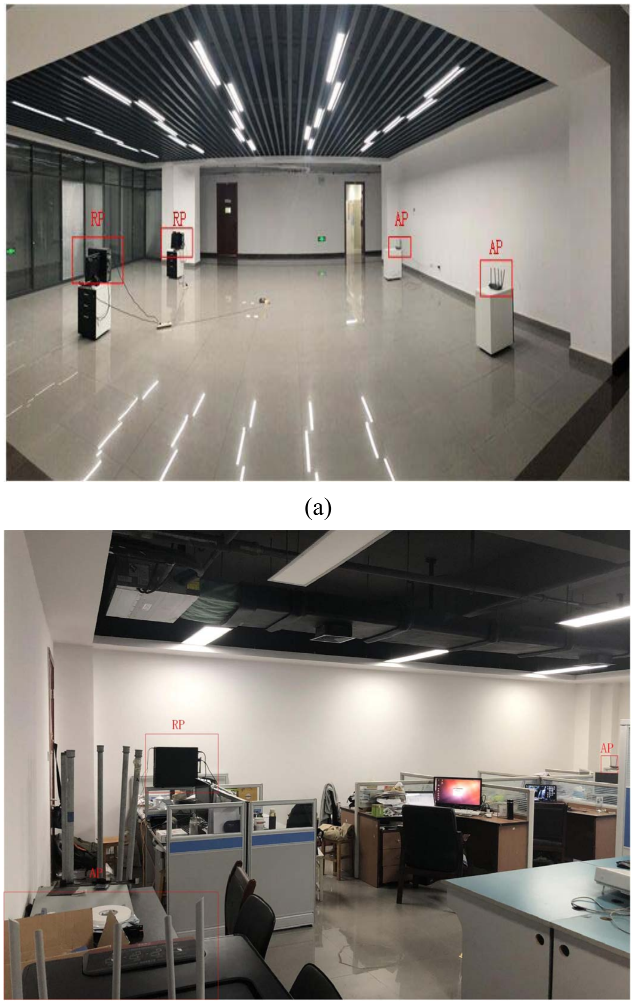


Fig. 4. Raw and processed phase of the same subcarrier from different data packets.



(a)

(b)

Fig. 5. Experimental scene: (a) open environment, (b) laboratory environment.

as tables and chairs. The two scenarios will be discussed separately in this paper.

B. Establishment of Fingerprint Map

For the open environment shown in Figure 5a, we placed the AP and RP in four corners and divided them into two groups. The fingerprint point spacing is divided into $1m \times 1m$, $1.5m \times 1.5m$ and $2m \times 2m$, respectively. The experimental results are given in the form of a cumulative distribution function (CDF), as shown in Fig. 6.

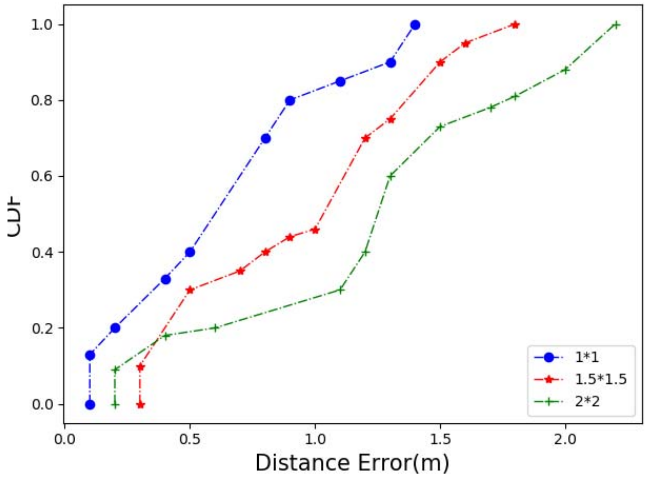


Fig. 6. CDF of different fingerprint spacing.

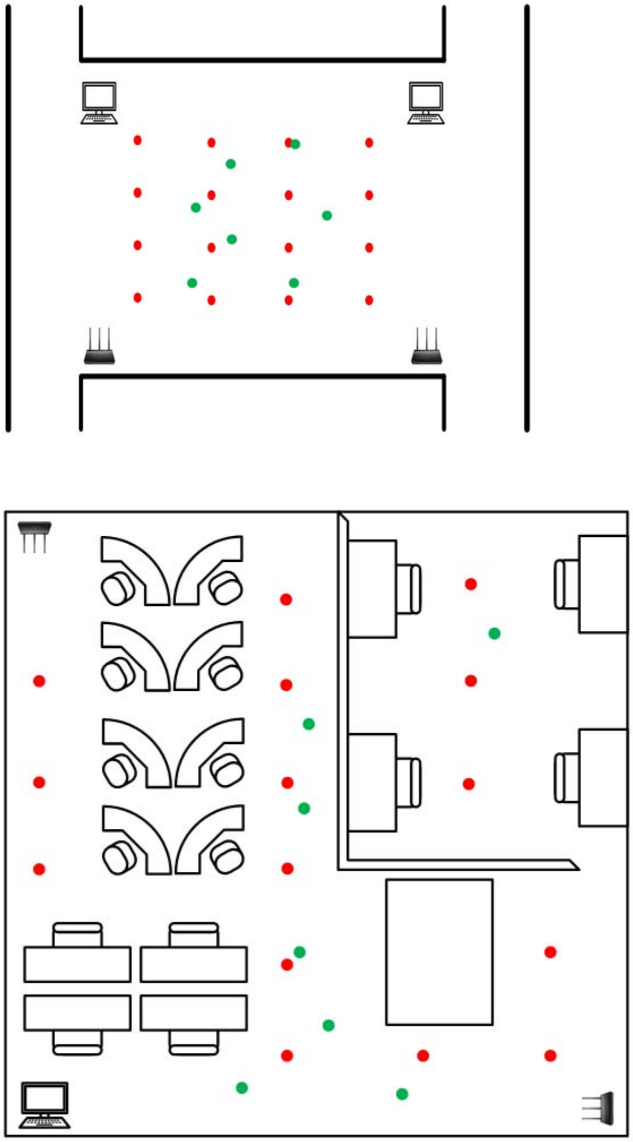
Fig. 6 shows that when the fingerprint point spacing is 1 m, 40% of the test points positioning errors are concentrated between 0.5 m and 1 m, and 80% of these errors are within 1 m. When the fingerprint point spacing is 1.5m, 50% of the test points positioning errors are concentrated between 1m and 1.5m, and 90% of these errors are within 1.5m. When the fingerprint point spacing is 2m, 40% of the test points positioning errors are also concentrated between 1m and 1.5m, and 90% of these errors are within 2m.

It can be seen that the error increases slightly as the spacing between the fingerprint points becomes larger. At the same time, when the interval between fingerprint points is 1m, we need to collect a large number of fingerprint points. When the interval is 2m, the number of fingerprint points collected can be effectively reduced, but the experimental error is relatively large. As a result, we set the interval to be 1.5m, and further discussion will be based on the 1.5m interval. The experimental environments are shown in Figure 7a, 7b.

We put the AP and DP in four corners, divided into two groups. The fingerprint point spacing is set to $1.5m \times 1.5m$, wherein red indicates the fingerprint points collected, and green indicates the test points. In the open area of $7m \times 7m$ shown in Figure 7a, we collected 16 regularly distributed fingerprint points and 7 test points. In the laboratory environment of $8m \times 9m$ shown in Figure 7b, we collected 15 fingerprint points and 7 test points. The acquisition frequency of each point is 100 packets/s, and the acquisition time is 2 seconds.

C. Influence of Eliminating Abnormal Data

In this section, we will discuss the positioning accuracy influenced by the number of abnormal data. By adjusting the number of thresholds and the distance of neighborhood, we can treat different quantity of phase data as abnormal data. We select a test point as an experimental point and analyze the effect of the number of eliminating data on the



(b)

Fig. 7. Experimental environment: (a) open environment, (b) laboratory environment.

experimental results. In the experiment, by adjusting the parameters, the number of data we eliminated was 8,12,15,17,23 and 29. The experimental results are given in the form of average error distance, as shown in Fig. 8.

Fig. 8 shows that when the number of data to be eliminated is 15, the average error is 1.05m and reaches the minimum. If the number of eliminated data is less than 15, the average error distance decreases as the number of eliminated data increases. This phenomenon indicates that the existence of abnormal data will affect the accuracy of the experiment. When the number of data to be eliminated is more than 15, the average error is steady. This shows that when the number of data to be removed reaches a certain value, the average error distance remains stable.

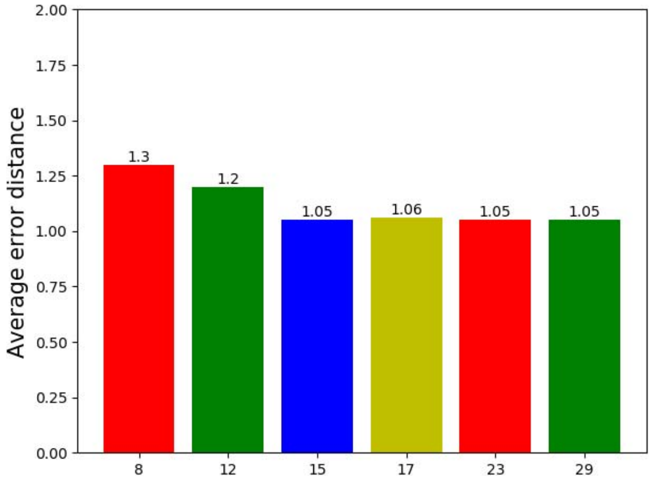


Fig. 8. Effect of eliminating different amounts of data.

TABLE I
CLASSIFICATION RESULTS OF DIFFERENT PARAMETERS IN THE OPEN ENVIRONMENT

iterations	max_depth	max_features	min_samples_split	accuracy
50	5	10	5	70%
	6			72%
	7			74%
100	5	10	5	74%
	6			76%
	7			77%
150	5	10	5	76%
	6			80%
	7			78%

D. Influence of Adaboost Parameters

In this system, the most important parameter is the number of iterations. By changing the number of iterations, the weight of each sample will change accordingly. In addition, other parameters of the basic classifier affect the accuracy of the classification as well. TABLE I shows the classification accuracy of different parameters in the open environment.

While keeping other parameters unchanged, by adjusting the parameters max_features and min_sample_split constantly, we found that the best precision could be achieved when max_features was set to 10 and min_sample_split was set to 5. To verify the impact of the number of iterations, we set the number of iterations to 50, 100, and 150, respectively. The experimental results show that as the number of iterations increases, the overall accuracy will increase. When the number of iterations is 50, the average accuracy is 72%, while the average accuracy is 78% when the number of iterations is 150. Therefore, we set the number of iterations, max_depth, max_features and min_sample_split to be 150, 6, 10 and 5, respectively.

E. Influence of the Number of Antennas

We selected one antenna pairs, two antenna pairs, and three antenna pairs respectively for test. Fig. 9 shows the effect of selecting different antenna pairs.

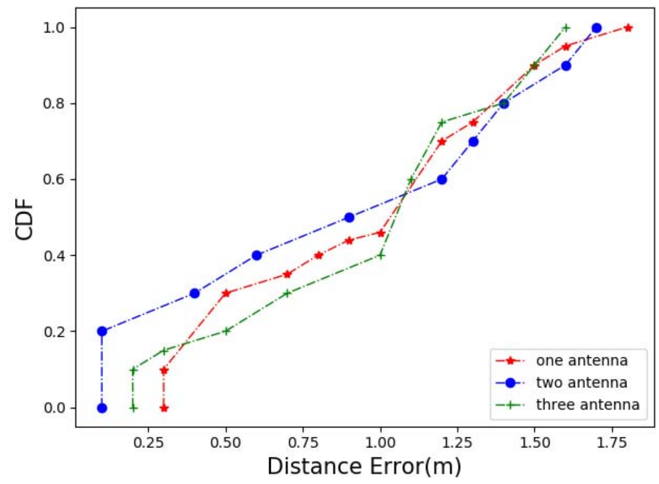


Fig. 9. CDF of different antennas.

TABLE II
CLASSIFICATION RESULTS OF DIFFERENT CLASSIFICATION METHODS

Environment	KNN	Naive_Bayes	SVM	Random Forest	Adaboost
Open environment	68%	70%	75%	78%	81%
Laboratory	60%	65%	72%	75%	80%

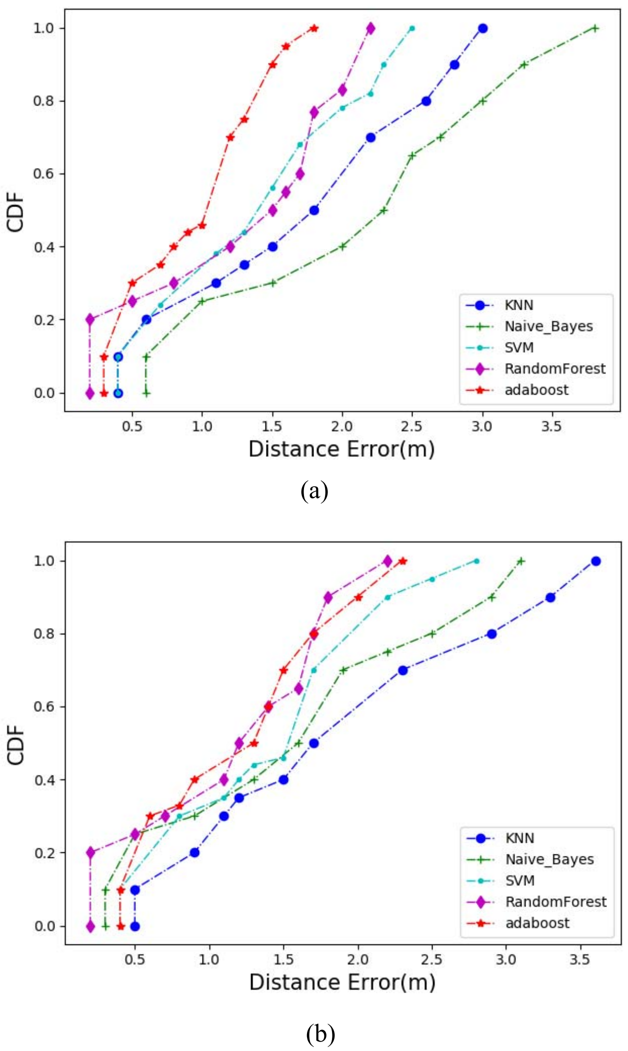
It can be seen that the error of two antennas within 1m is 50% while the error of one antenna pairs within 1m is 40%, but the errors are basically equal when the error range is 1.5m. The results show that the antenna pairs of different groups have slight effect on the experimental results, and the experimental precision does not increase significantly with the increase of the number of antennas. Therefore, we only use one antenna pairs in the following experiments.

F. Comparison of Different Classification Methods

We compared the Adaboost method with KNN, Naive_Bayes, SVM and Random Forest. TABLE II shows a comparison of the different classification methods in the two environments.

In the open environment, KNN and Naive_Bayes have a low classification accuracy, about 70%. Once the features are mapped to high-dimensional space with SVM, the accuracy reaches 75%. Random Forest and Adaboost are as important as EL methods, the classification accuracy are better than other classifiers, and the Adaboost algorithm has the highest accuracy, which can reach 81%. In comparison, due to the effect of path loss and multipath fading, classification accuracy has declined overall in laboratory. Figure 10a, 10b show the results after regression in the two environments.

In the open environment, 80% of the test points have an error within 2.5m using KNN while 80% of the test points have an error within 1.5m using Adaboost. In the laboratory, the error is larger than in open environment. However, the EL method still maintains its own superiority, and is still lower than other methods in terms of error. In general, 90% of test points have an error within 2m using Adaboost.



(a)

(b)

Fig. 10. CDF after regression of classification results with different classification methods. (a) Open environment. (b) Laboratory environment.

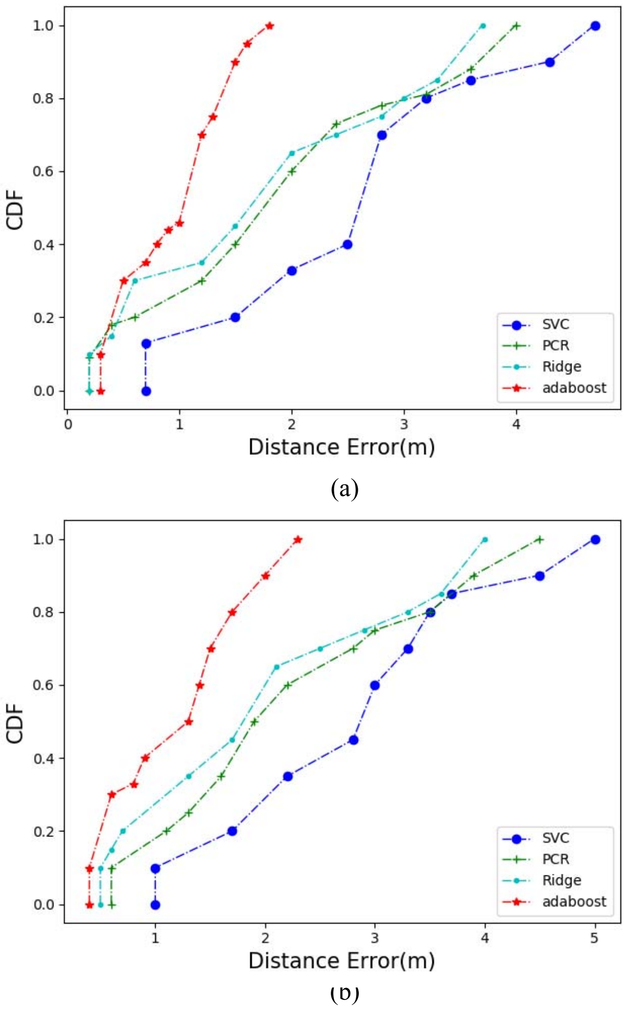
G. Comparison of Different Regression Algorithms

We compare this method with some other regression algorithms, such as SVC, PCR, Ridge. The experimental results are shown in Figure 11a, 11b.

It can be seen that, compared with the traditional regression algorithm, the regression accuracy can be improved by the proposed method. In the open environment, 80% of the test point errors can be concentrated in the range of 2.5m to 3m with traditional regression method. By means of confidence regression, 80% of test points have an error within 1.5m. The accuracy range is improved by about 1m using confidence regression. In the laboratory environment, the error is increased by 0.5m compared to the open environment.

H. Comparison of Different Systems

We compared ABPS with PLS-Lssvm [35] and DeepFi [36] in the two environments. Figure 12 shows the average error distance in different environments.



(a)

(b)

Fig. 11. CDF after regression compared with different regression algorithms. (a) Open environment. (b) Laboratory environment.

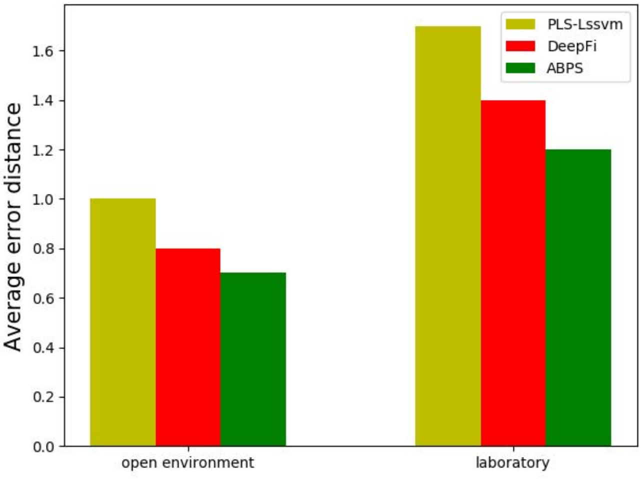


Fig. 12. The average error distance of different systems.

PLS-Lssvm adopts least squares support vector machine while DeepFi adopts deep learning. Compared with PLS-Lssvm, the proposed method can reduce the error about 0.3m in open environment. Compared with DeepFi, the proposed method also can increase positioning accuracy about 0.2m in laboratory.

V. Conclusion and Future Work

In this paper, we proposed ABPS, a fingerprinting system for indoor localization. In this system, the phase of the CSI is used as fingerprint feature, and the test points are classified by means of Adaboost algorithm. In the regression stage, the classification results of the test point are regressed by confidence regression to obtain the final coordinates. The system is validated in two different indoor environments, and the influences of several factors such as the number of abnormal data, ABPS parameters and the number of antennas are discussed. Experimental results show that the system has an excellent performance. Compared with some other classification methods, the classification accuracy of the Adaboost algorithm is higher, and the confidence regression outperform the regression algorithm such as SVC, PCR and Ridge.

In this work, we only deal with the phase information simply. In the future work, we can further explore the phase information and find more valuable parts. In addition, the problem of multi-person positioning in the target area is also our next work.

References

[1] H. Xiong, J. Tang, H. Xu, W. Zhang, and Z. Du, “A robust single GPS navigation and positioning algorithm based on strong tracking filtering,” *IEEE Sensors J.*, vol. 18, no. 1, pp. 290–298, Jan. 2018.

[2] S. Thongthammachart and H. Olesen, “Bluetooth enables in-door mobile location services,” in *Proc. 57th IEEE Semiannual Veh. Technol. Conf.*, Jeju, South Korea, Apr. 2003, pp. 2023–2027.

[3] F. Seco, C. Plagemann, A. R. Jiménez, and W. Burgard, “Improving RFID-based indoor positioning accuracy using Gaussian processes,” in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Zurich, Switzerland, Sep. 2010, pp. 1–8.

[4] Y. Zhao, L. Dong, J. Wang, B. Hu, and Y. Fu, “Implementing indoor positioning system via ZigBee devices,” in *Proc. Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, Oct. 2008, pp. 1867–1871.

[5] M. Erić, R. Zetik, and D. Vučić, “An approach for determination of antenna positions in distributed antenna system used for UWB indoor self-localization: Experimental results,” in *Proc. Telecommun. Forum Telfor (TELFOR)*, Belgrade, Serbia, Nov. 2013, pp. 204–207.

[6] J. Xiong and K. Jamieson, “ArrayTrack: A fine-grained indoor location system,” in *Proc. Netw. Syst. Design Implement. (NSDI)*, 2013, pp. 71–84.

[7] M. Youssef, M. Mah, and A. Agrawala, “Challenges: Device-free passive localization for wireless environments,” in *Proc. 13th Annu. Int. Conf. Mobile Comput. Netw.*, Montréal, QC, Canada, 2007, pp. 222–229.

[8] Y. Tao and L. Zhao, “A novel system for WiFi radio map automatic adaptation and indoor positioning,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10683–10692, Nov. 2018.

[9] W. Sun, M. Xue, H. Yu, H. Tang, and A. Lin, “Augmentation of fingerprints for indoor WiFi localization based on Gaussian process regression,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 11, pp. 10896–10905, Nov. 2018.

[10] Y. Xie, Y. Wang, P. Zhu, and X. You, “Grid-search-based hybrid TOA/AOA location techniques for NLOS environments,” *IEEE Commun. Lett.*, vol. 13, no. 4, pp. 254–256, Apr. 2009.

[11] S. Venkatraman, J. Caffery, and H.-R. You, “A novel ToA location algorithm using LoS range estimation for NLoS environments,” *IEEE Trans. Veh. Technol.*, vol. 53, no. 5, pp. 1515–1524, Sep. 2004.

[12] Z. Wang and S. A. Zekavat, “Omnidirectional mobile NLOS identification and localization via multiple cooperative nodes,” *IEEE Trans. Mobile Comput.*, vol. 11, no. 12, pp. 2047–2059, Dec. 2012.

[13] J. Wang *et al.*, “LiFS: Low human-effort, device-free localization with fine-grained subcarrier information,” in *Proc. 22nd Annu. Int. Conf. Mobile Comput. Netw.*, New York, NY, USA, 2016, pp. 243–256.

[14] X. Li, S. Li, D. Zhang, J. Xiong, Y. Wang, and H. Mei, "Dynamic-music: Accurate device-free indoor localization," in *Proc. ACM Int. Joint Conf. Pervasive Ubiquitous Comput.*, Berlin, Germany, 2016, pp. 196–207.

[15] M. Youssef and A. Agrawala, "The horus WLAN location determination system," in *Proc. 3rd Int. Conf. Mobile Syst., Appl., Serv.*, Seattle, WA, USA, 2005, pp. 205–218.

[16] X. Li, X. Cai, Y. Hei, and R. Yuan, "NLOS identification and mitigation based on channel state information for indoor WiFi localisation," *IET Commun.*, vol. 11, no. 4, pp. 531–537, Mar. 2017.

[17] H. Xiong *et al.*, "Efficient bias reduction approach of time-of-flight-based wireless localisation networks in NLOS states," *IET Radar. Sonar, Navigat.*, vol. 12, no. 11, pp. 1353–1360, Nov. 2018.

[18] Z. Tao-Yun, L. Bao-Wang, Z. Yi, and L. Sen, "Amp-Phi: A CSI-based indoor positioning system," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 32, no. 9, 2018.

[19] J. Xiao, K. Wu, Y. Yi, L. Wang, and L. M. Ni, "Pilot: Passive device-free indoor localization using channel state information," in *Proc. Int. Conf. Distrib. Comput. Syst.*, Philadelphia, PA, USA, Jul. 2013, pp. 236–245.

[20] K. Wu, J. Xiao, Y. Yi, D. Chen, X. Luo, and L. M. Ni, "CSI-based indoor localization," *IEEE Trans. Parallel Distrib. Syst.*, vol. 24, no. 7, pp. 1300–1309, Jul. 2013.

[21] R. Niu, A. Vempaty, and P. K. Varshney, "Received-signal-strength-based localization in wireless sensor networks," *Proc. IEEE*, vol. 106, no. 7, pp. 1166–1182, Jul. 2018.

[22] K. Wu, J. Xiao, Y. Yi, M. Gao, and L. M. Ni, "FILA: Fine-grained indoor localization," in *Proc. INFOCOM*, Orlando, FL, USA, Mar. 2012, pp. 2210–2218.

[23] J. Xiao, K. Wu, Y. Yi, and L. M. Ni, "FIFS: Fine-grained indoor fingerprinting system," in *Proc. Int. Conf. Comput. Commun. Netw.*, Munich, Germany, Jul./Aug. 2012, pp. 1–7.

[24] C. Wu, Z. Yang, Z. Zhou, K. Qian, Y. Liu, and M. Liu, "PhaseU: Real-time LOS identification with WiFi," in *Proc. IEEE Conf. Comput. Telecommun. (INFOCOM)*, Kowloon, Hong Kong, Apr./May 2015, pp. 2038–2046.

[25] Y. Chapre, A. Ignjatovic, A. Seneviratne, and S. Jha, "CSI-MIMO: Indoor Wi-Fi fingerprinting system," in *Proc. 39th Annu. IEEE Conf. Local Comput. Netw.*, Edmonton, AB, Canada, Sep. 2014, pp. 202–209.

[26] X. Wang, L. Gao, and S. Mao, "PhaseFi: Phase fingerprinting for indoor localization with a deep learning approach," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, San Diego, CA, USA, Dec. 2015, pp. 1–6.

[27] D. Li, B. Zhang, and C. Li, "A feature scaling-based K-nearest neighbor algorithm for indoor positioning systems," *IEEE Internet Things J.*, vol. 3, no. 4, pp. 590–597, Aug. 2016.

[28] Z. Wu, Q. Xu, J. Li, C. Fu, Q. Xuan, and Y. Xiang, "Passive indoor localization based on CSI and naive Bayes classification," *IEEE Trans. Syst. Man, Cybern., Syst.*, vol. 48, no. 9, pp. 1566–1577, Sep. 2018.

[29] M. Z. Win, Y. Shen, and W. Dai, "A theoretical foundation of network localization and navigation," *Proc. IEEE*, vol. 106, no. 7, pp. 1136–1165, Jul. 2018.

[30] Z. Wu, C.-H. Li, J. K.-Yin Ng, and K. R. P. H. Leung, "Location estimation via support vector regression," *IEEE Trans. Mobile Comput.*, vol. 6, no. 3, pp. 311–321, Mar. 2007.

[31] R. Zhou, X. Lu, P. Zhao, and J. Chen, "Device-free presence detection and localization with SVM and CSI fingerprinting," *IEEE Sensors J.*, vol. 17, no. 23, pp. 7990–7999, Dec. 2017.

[32] A. Rojarath, W. Songpan, and C. Pong-inwong, "Improved ensemble learning for classification techniques based on majority voting," in *Proc. IEEE Int. Conf. Softw. Eng. Service Sci. (ICSESS)*, Beijing, China, Aug. 2016, pp. 107–110.

[33] P.-B. Zhang and Z.-X. Yang, "A novel adaboost framework with robust threshold and structural optimization," *IEEE Trans. Cybern.*, vol. 48, no. 1, pp. 64–76, Jan. 2018.

[34] T. Yang, P. G. Mehta, and S. P. Meyn, "Feedback particle filter," *IEEE Trans. Autom. Control*, vol. 58, no. 10, pp. 2465–2480, Oct. 2013.

[35] C. Chen, Y. Wang, Y. Zhang, and Y. Zhai, "Indoor positioning algorithm based on nonlinear PLS integrated with RVM," *IEEE Sensors J.*, vol. 18, no. 2, pp. 660–668, Jan. 2018.

[36] X. Wang, L. Gao, S. Mao, and S. Pandey, "CSI-based fingerprinting for indoor localization: A deep learning approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 763–776, Jan. 2017.

Yong Zhang was born in Anqing, Anhui, China, in 1973. He received the Ph.D. degree in signal and information processing from the University of Science and Technology of China in 2007. He is an Associate Professor with the Hefei University of Technology. His research interests include intelligent information processing, hand gesture recognition, and indoor positioning.

Dapeng Li was born in Rizhao, Shandong, China, in 1995. He received the B.Eng. degree from the Hefei University of Technology, where he is currently pursuing the M.Sc. degree. His research interests include intelligent information processing and indoor positioning.

Yujie Wang was born in Zaozhuang, Shandong, China, in 1980. She received the Ph.D. degree in circuits and systems from the University of Science and Technology of China in 2011. She is currently a Lecturer with the Hefei University of Technology. Her research interests include intelligent information processing, indoor positioning, and audio signal processing.