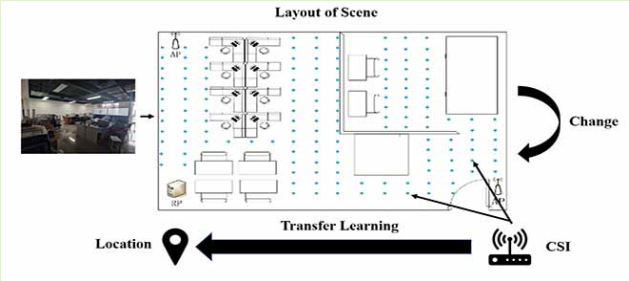
A Low-Overhead Indoor Positioning System Using CSI Fingerprint Based on Transfer Learning

Yong Zhang, Chengbin Wu, and Yang Chen

***Abstract*—With the rapidly growing demand for Location-Based Services in indoor environments, fingerprint-based indoor positioning has caused great interest due to its high positioning accuracy and low equipment cost. However, the standard signal radio map cannot provide consistent high positioning accuracy under environmental changes and new scenarios. To address this problem, we present a novel indoor positioning Transfer Learning(TL) system based on improved TrAdaBoost. First perform phase correction on the raw CSI(Channel State Information) phase, then use One-vs-Rest algorithm and One-Hot coding, which can realize the multi-classification ability of the TrAdaBoost algorithm. Meanwhile, optimize the iterative process of the algorithm by using a correction factor. In addition, Confidence regression is used to obtain the final estimated position. Experimental results show that the positioning accuracy can be improved by 35% in dynamic environment conditions. The proposed method can improve the positioning accuracy by an average of 30% in new scenes and the Site Survey Overhead(SSO) is reduced by 40%. Compared to other position b algorithm, our proposed method has more robustness in time and space.**



***Index Terms*—CSI, WiFi, transfer learning, indoor location.**

I. Introduction

POSITIONING service has become indispensable in our daily life. Location-Based Services(LBS) require users to provide accurate outdoor and indoor geographic location information that can be combined with a variety of exact location-related applications and content services [1]. Compared to the outdoor positioning system GPS(Global Positioning System) [2], indoor positioning still does not have a mainstreamed system. Besides high accuracy requirements, indoor positioning applications should also have a shorter estimated process time and lower mobile device complexity. Researchers have developed various indoor positioning technologies such as Infrared [3], Ultra-wideband [4], Cellular radios [5], Ultrasonic wave [6], WiFi(Wireless Fidelity) [7] and *et al.* In addition, the positioning method based on information fusion has also attracted the attention of many researchers [8], [9].

Due to widely installed network infrastructure, WLAN-enabled wireless terminal devices have been increasingly deployed in various public places, such as shopping malls, offices, airport and train station. With low deployment cost and open access, Wi-Fi-based wireless positioning technology has become one of the most promising positioning methods in the field of indoor positioning [10].

Received Signal Strength Indication(RSSI) which represents the aggregated signal strength of multiple signal paths is widely used in WiFi-based indoor positioning [11]. Radar is the first fingerprinting system based on RSSI with a deterministic method for location estimation [12]. Horus is another RSSI-based fingerprinting scheme, which utilizes a probabilistic location estimation method based on K-Nearest Neighbor [13]. In addition, a variety of machine learning methods have been widely applied to indoor positioning process, such as Neural Networks(NN) [14], Support Vector Machines (SVM) [15].

Although RSSI was widely used in previous researchs, it is only a rough representation of the wireless channel which does not extract richer multipath information from the subcarriers in Orthogonal Frequency-Division Multiplexing (OFDM) [16]. Compared with RSSI, Channel State

Manuscript received April 22, 2021; accepted May 17, 2021. Date of publication May 21, 2021; date of current version August 13, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 61801162 and in part by the Anhui Provincial Natural Science Foundation under Grant 2008085MF214. The associate editor coordinating the review of this article and approving it for publication was Dr. Qammer H. Abbasi. *(Corresponding author: Chengbin Wu.)*

The authors are with the School of Computer and Information, Hefei University of Technology, Hefei 230009, China (e-mail: yongzhang@hfut.edu.cn; 964016077@qq.com; chenyang@163.com).

Digital Object Identifier 10.1109/JSEN.2021.3082553

Information (CSI) can better reflect fine-grained channel information because of characteristics based the physical layer and description of amplitude and phase information. Therefore, CSI based positioning attracts attention from researchers. For example, CSI amplitude and calibrated phase information are both utilized in DeepFi [17] and PhaseFi [18], which collects CSI from the subcarriers to generate fingerprints through a deep autoencoder network. Zhang *et al.* proposes the Adaboost positioning system that uses the phase information in CSI [19]. Shao *et al.* designed a novel hybrid location image using Wi-Fi and magnetic field fingerprints for the indoor position system [20]. Schmidt at el. presented SDR-Fi which uses Wi-Fi software-defined radio (SDR) receiver for indoor positioning using CSI measurements as features for deep learning classification [21].

The practical CSI-based indoor positioning system faces two main challenges: one is the severe multipath and shadow fading influences, and another is the vulnerability to dynamic environment. The short-term interference (e.g. door opening, closing and movement of tables, chairs and other furniture) and long-term interference (e.g. humidity, temperature and light changes) will cause signal unavailabe. Therefore, the real-time CSI data will be quite different from the values in the fingerprint library [22], [23]. If the fingerprint library is not updated in time, the positioning accuracy will be reduced.

To adapt to the changes in the environment, a simple solution for indoor location is to re-collect data and supplement the fingerprint library. However, this method is very impractical because this calibration process is time-consuming and laborious. Some works deployed fixed hardware to get new CSI for modification [24], but additional hardware implementations would incur additional costs.

In this paper, we proposed a TL indoor positioning method based on improved TrAdaBoost. Through this transfer learning method, which uses a small number of positioning points can obtain better positioning accuracy. During our indoor location system, the original scene in which the fingerprint database is collected as the source domain, and the new scene or the scene with environment change is defined as the target domain. In this article, we make the following major contributions.

1) In order to cope with changes in positioning environment and scenes, we introduce the TrAdaBoost TL method for indoor positioning. The experimental results show that the positioning accuracy of the proposed method can be improved by 35% and 30% in the same scene with environment changes and in the new scenes, respectively. The experimental results also show that under the same positioning accuracy requirements, the SSO of the proposed method in this paper is reduced by 40%.

2) We use One-Hot encoding and One-vs-Rest algorithm in the TrAdaBoost algorithm to solve the problem of binary classification effectively, which makes it possible to classify multi-label indoor positioning fingerprint points. We utilize the improved TrAdaBoost TL method to build a fingerprint database which combines source domain and target domain, then use the decision tree as basic classifier. The phase of the CSI after eliminating phase offset is used as the fingerprint feature.

3) The adaptive penalty weight is added to the category of misclassification in source data. We also add a weight correction factor to the weight iteration process to effectively alleviate the problem of weight descending too quickly, which can reduce the misclassification and improve the classification efficiency.

The rest of this article is arranged as follows: Section II introduces CSI data format and indoor positioning based on dynamic environment. Section III describes the system structure including One-Hot coding, One-Vs-Rest algorithm and improved TrAdaBoost. In Section IV, we performed experiments to evaluate the performance of the system. Finally, we conclude in Section V.

II. Related Work

In this section, we introduce the indoor positioning based on dynamic environment, and the composition of CSI data.

A. Indoor Positioning Based on Dynamic Environment

The application of TL to indoor positioning based on dynamic environment has attracted the attention of researchers. Liu Kai *et al.* [22] proposed an indoor positioning algorithm based on TL framework, which is divided into metric learning and metric Transfer. These two parts are used to learn the distance metrics from source domains and identify the most suitable metric for the target domain, respectively. Zou *et al.* [25] recommend the use of Transfer kernel learning to design robust localization models. It can learn domain-invariant kernels by directly matching the source and target distributions in the replicated kernel Hilbert space instead of the original noise signal space. The generated kernel can be used as input for the Support Vector Regression(SVR) training process. The trained localized model can inherit information from the online phase to adaptively enhance standard calibrated radio maps. Recently, Pan *et al.* [26] used transfer learning to transfer the temporal fluctuation of RSS by transferring knowledge from a source domain (offline database) to a target domain (online RSS sample). Wang *et al.* [27] proposed a Domain Adaptation Network(DAN) to reduce the domain discrepancy efficiently. DAN learn transferable features from domain shift minimization by aligning the mean embeddings of the hidden representation in a reproducing kernel Hilbert space, and aligning the second order statics of hidden representation of different domains.

The TL methods based on features, models and kernel learning also have achieved good positioning effects in various scenes but the algorithm complexity is pretty high. Existing work such as [22], [25], [28] only considers the robustness in space, and [29] only considers the robustness in time. Instead of these solutions our system is based on the positioning method of transfer learning and the purpose is to achieve lower accuracy under the same accuracy, and have good robustness both in time and space. Compared with the above methods, the proposed method could achieve high indoor positioning accuracy with low algorithm complexity. In addition, our system reduces the overhead caused by recollecting fingerprint point data.

B. Channel State Information

CSI is the channel attribute of the communication link, which describes the channel state information, such as scattering, fading, multipath fading or shadowing fading, power decay, and other information. Moreover, CSI can adapt the communication system to current channel conditions, providing high reliability and high rate communication in multi-antenna systems [18]. The received signal after the multipath channel is expressed as

$$\boldsymbol{Y} = H\boldsymbol{X} + N \tag{1}$$

where $\boldsymbol{X}$ and $\boldsymbol{Y}$ are the vectors of the receiving and the transmitting end respectively, H and N are the channel matrix and the pseudo-Gaussian white noise, respectively. H can be expressed as

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

Each element in the matrix is a complex number indicating the amplitude and phase of the channel state, where M represents a different subcarrier number, m and n represent the number of transmitted and received antennas, respectively. Therefore, $\boldsymbol{Y}$ can also be expressed as:

$$\boldsymbol{Y} = \|\boldsymbol{Y}\| e^{j\times\angle\boldsymbol{Y}} \tag{3}$$

where $\|\boldsymbol{Y}\|$ represents the amplitude of the signal and $\angle Y$ represents the phase of the signal.

III. System Structure

In this section, we introduce structure of our proposed system.

A. Data Collection and Pre-Processing

Since the amplitude information is greatly hindered by other facilities while the phase information is less affected [30], which means the phase is more robust than the amplitude. Therefore we choose CSI phase to build a fingerprint database.

The proposed system consists of two domains, the source domain D_s and the target domain D_t. Data consists of source domain fingerprint data T_s, target domain fingerprint data T_d, and the test data S in target domain, which been shown in Fig. 1.

The raw CSI data has the characteristics of high dimension, which cause difficulty in parameter estimation and calculation and infeasible to be directly used in the positioning process. Thereby, we use the PCA to extract the main features and eliminate the noise in the original CSI data during the data pre-processing process. Because of Carrier Frequency Offset(CFO) and Sample Frequency Offset(SFO) in the process of data collection, the phase data need to be calibrated before using PCA to reduce dimension and de-noise. CFO is generated by the down converter because the center frequency between the receiver and the transmitter cannot be completely synchronized. The SFO is caused by the ADC due to the clocks not being synchronized. Moreover, for SFO, the measured phase error is different for every subcarrier. For residual errors in the phase, it can be eliminated by linear transformation. The processed phase can be obtained by the following method:

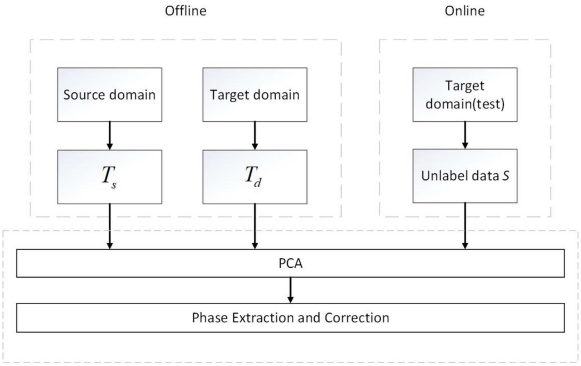


Fig. 1. Data collection and Pre-processing.

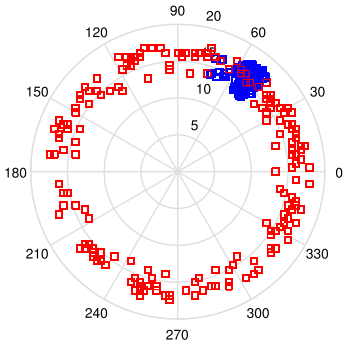


Fig. 2. The raw and processed phase of the same subcarrier.

$$\angle Y_i' = \angle Y_i - am_i - b \tag{4}$$

where $\angle Y_i'$ and $\angle Y_i$ represent the calibrated and uncalibrated phase of the ith subcarrier, respectively. a and b are the two parameters of the linear transformation. m_i represents the index value of the ith subcarrier.

According to the IEEE 802.11n standard, the index $i \in (1, 30)$, and m_i ranging from -28 to 28. Let:

$$a = \frac{\angle CSI_{30} - \angle CSI_1}{m_{30} - m_1} \tag{5}$$

$$\mathrm{b} = \frac{1}{30}\sum_{n=1}^{30} \angle Y_i \tag{6}$$

As shown in Fig. 2, we plotted the original phase (red square) and the calibrated phase (blue square) of the 200 CSI data in the third subcarrier of the second channel in the polar coordinate system. The original phase is randomly distributed in the range of 0° to 360°, while the calibrated phase is concentrated in the sector of 30° to 90°. It illustrates that the calibrated phase eliminates the errors caused by CFO and SFO, which make it can be applied to indoor positioning.

B. Preparation Works for Tradaboost

In the online phase of fingerprint localization, test points and fingerprint points need to be multi-classified, while the

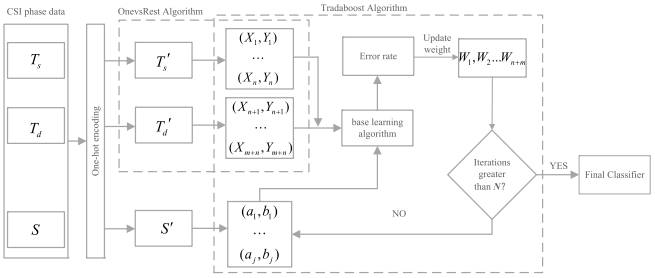


Fig. 3. The flow chart of the improved TrAdaBoost algorithm.

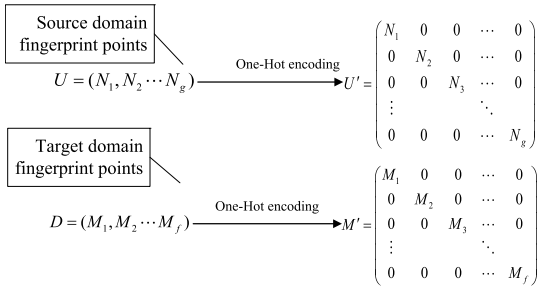


Fig. 4. One-hot coding.

TABLE I
SUMMARY OF THE NOTATIONS

Symbol	Description
A	feature space, $A \in R^d$
B	label space = {0,1}
D_s	source domain
D_t	target domain
T_s	source domain fingerprint data
T_d	target domain fingerprint data
ϵ_t	classifier error at boosting iteration t
W	weight vector
N	number of iterations
g	number of source instances
f	number of target instances
m	phase data of one fingerprint point in the source domain
n	phase data of one fingerprint point in the target domain
gf	total number of categories after cross matching
t	index for boosting iteration
h_t	weak classifier at boosting iteration t
CSI_x	fingerprint points data
X_i	CSI phase data set
Y_i	fingerprint points data
C^t	correction factor

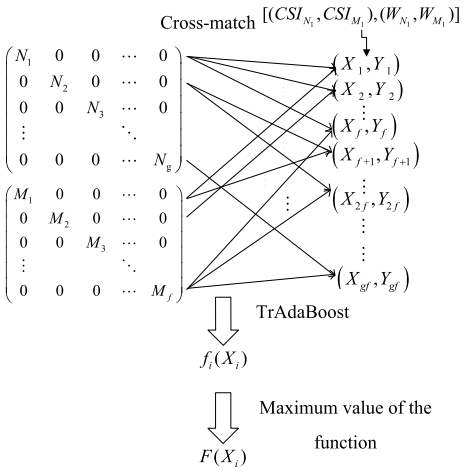


Fig. 5. One-vs-Rest Algorithm.

basic TrAdaBoost algorithm [31] can only perform two-class classification. Besides these, small amount of processed phase data is not enough to get the multi-classifier with strong generalization ability. Therefore, we proposed an improved TrAdaBoost TL algorithm which can solve this problem effectively. The improved TrAdaBoost algorithm flowchart is shown in Fig. 3. The description of the corresponding symbols can be seen in Table I.

C. One-Hot Coding

In order to increase the non-linear capability of our model, we first perform One-Hot coding [32] for T_s and T_d, which can reduce the impact of outliers of fingerprint data on the model, thereby increasing the stability of the model.

Supposed that the source domain data $T_S = \left(CSI_{N_1}, C\ SI_{N_2} \ldots CSI_{N_g}\right)$ has g fingerprints or classes and target domain data $T_D = \left(CSI_{M_1}, CSI_{M_2} \ldots CSI_{M_f}\right)$ also has f fingerprints or classes. Each fingerprint point of the source and target domains includes n and m CSI phase data respectively. Accordingly, these fingerprint points are labeled as $U = \left(N_1, N_2 \ldots N_g\right), D = \left(M_1, M_2 \ldots M_f\right)$, respectively.

We use One-Hot Coding for U and D to convert the original one-dimensional label data into a two-dimensional matrix U' and D', which is shown in Fig. 4. The encoded fingerprint points' label becomes a matrix whose element is zero except for the diagonals. In this way, the label of the first fingerprint point in the source domain is encoded as $(N_1, 0 \ldots 0)$.

Encoding the label of these fingerprints can increase the data dimension to improve the stability of the model, and it is also beneficial to the One-vs-Rest algorithm for cross-matching and the process of finding the location point of the maximum probability.

D. One-vs-Rest Algorithm

Since the TrAdaBoost algorithm only supports binary classification, we use the One-vs-Rest algorithm to process T_s and T_d which are encoded source domain data and target domain data to enhance its multi-classification ability.

The One-vs-Rest algorithm assumes that the sample is divided into k classes, then constructs k subclassifiers to separate each class from all the other classes, and its classification output is the category corresponding to the maximum value in the output function of each subclassifies [33].

In this paper, One-vs-Rest algorithm matches each fingerprint point of the source domain with all the fingerprint points data of the target domain through U' and M' to obtain $X_i,\ i = 1, \ldots, gf$ where T_i include n source data and m target data as shown in Fig. 5. Then gf sub-classifiers are constructed to separate each class from all other classes. Assigning different weight Y_i to X_i in TrAdaBoost algorithm, Where Y_i contains the weights of the source domain data and the target domain data, respectively, which is represented by

Algorithm 1 The Improved TrAdaBoost

Require: We get X_i by One-vs-Rest algorithm, then choose decision tree algorithm (Learner) and set the maximum number of iterations to N.

Ensure: the final hypothesis.

1: Initialize the weight $W^1 = \left(W_1^1, \cdots, W_{n+m}^1\right)$. Set weight $1/n$ and $1/m$ for W_1^1 to W_n^1 and W_{n+1}^1 to W_{n+m}^1, respectively;

2: **for** t $= 1, \ldots, N$ **do**

3: Set $P^t = W^t / \left(\sum_{i=1}^{n+m} W_i^t\right)$;

4: Call Learner, providing it the combined training set $T = T_s' \cup T_d'$ with the distribution P^t over T and the test data $S' = \left(a_j, b_j\right)$, where a_j represents the test fingerprint point after One-hot encoding and b_j represents CSI fingerprint data. Then, get back a hypothesis $h_t : A \rightarrow B$(or [0, 1] by confidence);

5: Calculate the error of h_t on T_t:

$$\epsilon_t = \sum_{i=n+1}^{n+m} \frac{w_i^t \cdot \left|h_t\left(x_i\right) - c\left(x_i\right)\right|}{\sum_{i=n+1}^{n+m} w_i^t} \tag{7}$$

where x_i represents the phase data in X_i, $c(x_i)$ represents actual correct classification results;

6: Set $\beta_t = \epsilon_t / \left(1 - \epsilon_t\right)$ and $\beta = 1/(1 + \sqrt{2 \ln n / N})$.In which, ϵ_t is required to be less than 1/2;

7: Set $C^t = 1.8\left(1 - \epsilon_t\right)$;

8: Update the new weight vector:

$$W_i^{t+1} = \begin{cases} C^t W_i^t \beta^{\left|h_t(x_i) - c(x_i)\right|}, & 1 \le i \le n \\ W_i^t \beta_t^{-\left|h_t(x_i) - c(x_i)\right|}, & n+1 \le i \le n+m \end{cases} \tag{8}$$

9: **end for**;

10: output the hypothesis:

$$f_i(x_i) = \begin{cases} 1, & \prod_{t=[N/2]}^{N} \beta_t^{-h_t(x_i)} \ge \prod_{t=[N/2]}^{N} \beta_t^{-\frac{1}{2}} \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

W_j, $j = 1, \ldots, n+m$. Finally, we get multi-class training samples(X_i, Y_i).

We use TrAdaBoost algorithm for the separation process to get the output $f_i\left(X_i\right)$, which is accumulated by classification result $f_i\left(X_i\right)$, as shown in Algorithm 1. By using the binary classification function $f_i\left(X_i\right)$, $i = 1, \ldots, gf$, the i-th sample is separated from other training samples. And by selecting the class $F\left(X_i\right)$ corresponding to the maximum value of the function $F\left(X_i\right) = \arg\max\left\{f_1\left(X_1\right), \ldots, f_{qf}\left(X_{qf}\right)\right\}$ to obtain the final classification result as the input for classification regression.

E. The Improved TrAdaBoost Algorithm

After the cross-matched T_s' and T_d' by One-vs-Rest algorithm, we assign weight Y_i to U' and M' as the input of the TrAdaBoost algorithm. The detailed steps are as Algorithm 1.

In the original TrAdaBoost algorithm, the classification error of the source domain training data means that the data conflicts with the target training data during each iteration of the round. Therefore, it is necessary to reduce the weight of the misclassified data so that the impact of several misclassified samples on the classification model in the next iteration will be less than the previous iteration. In the end, data that matches the source data in the target domain data will have a higher weight, and those that do not match will have a lower weight.

But when the number of iterations $N \rightarrow \infty$, the classification error of each basic classifier in the source domain is ignored, meaning that all samples in the source domain are correctly classified [34]. Since the source domain uses the weighted moving average algorithm to adjust the weights, the sum of sample weights in the source domain on the t+1th iteration:

$$S_s = \sum_{i=1}^{n} W_{S_i}^{t+1} = \sum_{i=1}^{n} W_{S_i}^t = n W_S^t \tag{10}$$

The sum of the target domain sample weights is:

$$S_T = m W_t^t \left(1 - \epsilon_t^t\right) + m W_t^t \epsilon_t^t \frac{1 - W_t^t}{\epsilon_t^t} = 2m W_t^t \left(1 - \epsilon_t^t\right) \tag{11}$$

Therefore, the sample weight distribution of the source domain on the $t+1$th iteration is:

$$W_S^{t+1} = \frac{w_S^t}{S_s + S_T} = \frac{W_S^t}{n W_S^t + 2m W_t^t \left(1 - \epsilon_t^t\right)} \tag{12}$$

Set C^t as the correction factor. Then the weights of the source domain samples need to be stable after t iterations that is:

$$w_S^t = \frac{C^t W_S^t}{n C^t W_S^t + 2m W_t^t \left(1 - \epsilon_t^t\right)} \tag{13}$$

Considering that the fingerprint points in the source domain generally have about 10% outliers, that is to say, not all the source domain data is valuable to the target domain in the transfer learning which means $W_S^{t+1} = 0.9\ W_S^t$. So in the process of weight iteration:

$$0.9 W_S^t = \frac{C^t W_S^t}{n C^t W_S^t + 2m W_t^t \left(1 - \epsilon_t^t\right)} \tag{14}$$

Then combine (7) and (8):

$$C^t = \frac{1.8\left(1 - \epsilon_t^t\right)}{1 + 0.1 S_S / S_T} \tag{15}$$

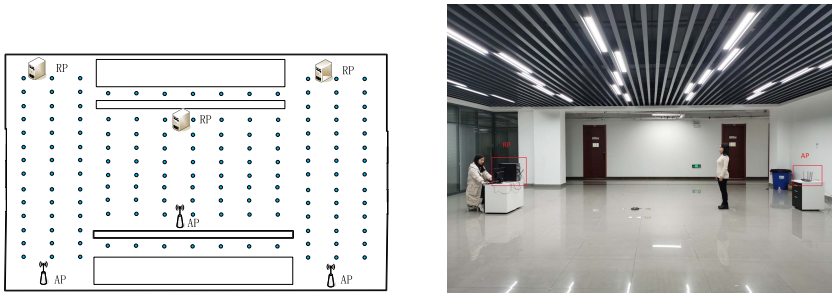
Finally, when $N \rightarrow \infty$, $S_S \ll S_T$ correction factor is:

$$C^t \approx 1.8\left(1 - \epsilon_t^t\right) \tag{16}$$

F. Confidence Regression

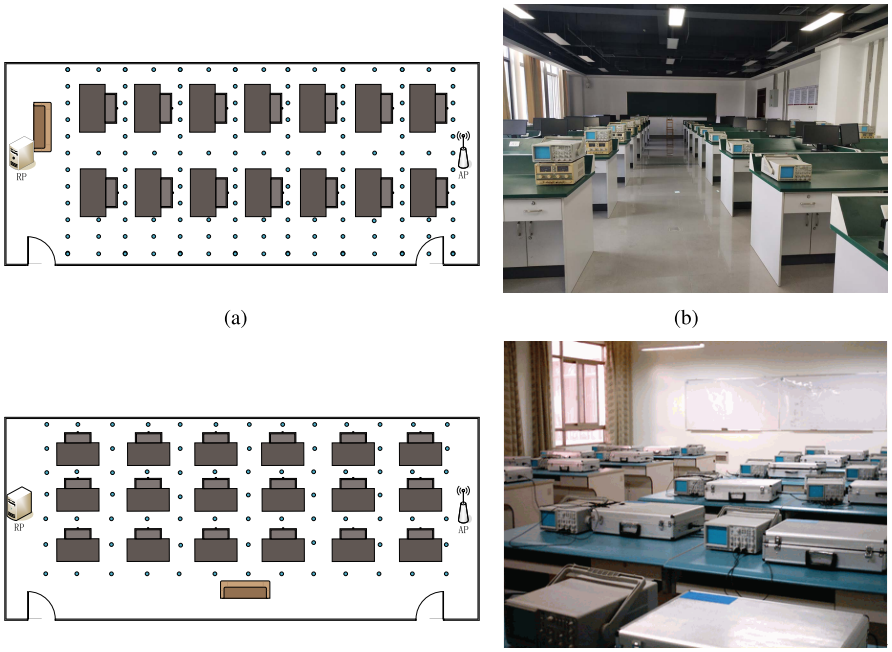
We choose a method based on confidence regression for the final regression, and use $Softmax$ to process different classification results to get the probability value of the test point.

$$p = \frac{e^{z_i}}{\sum_{c=1}^{C} e^{z_c}} \tag{17}$$



(a) (b)

Fig. 6. Scene 1. (a) Layout of scene 1. (b) True condition of scene 1.



(a) (b)

(c) (d)

Fig. 7. Scene 2 and scene 3. (a) Layout of scene 2. (b) True condition of scene 2. (c) Layout of scene 3. (d) True condition of scene 3.

where z_i is the output value of the i-th fingerprint point category, and C is the number of output nodes, that is, the number of categories. Through the $Softmax$ function, the output value of the multi-classification can be converted into a probability distribution in the range of [0,1].

Considering the calculation error and the environmental interference, we choose to calculate the five highest probability tags, which are respectively m_1, m_2, m_3, m_4, m_5, and the corresponding tag values are expressed as c_1, c_2, c_3, c_4, c_5, respectively.

$$c_i = (x_i, y_i) \tag{18}$$

where x_i and y_i are the coordinate values of the point. The final position of the predicted point is

$$x_{\text{final}} = \sum_{i=1}^{5} p_i x_i \tag{19}$$

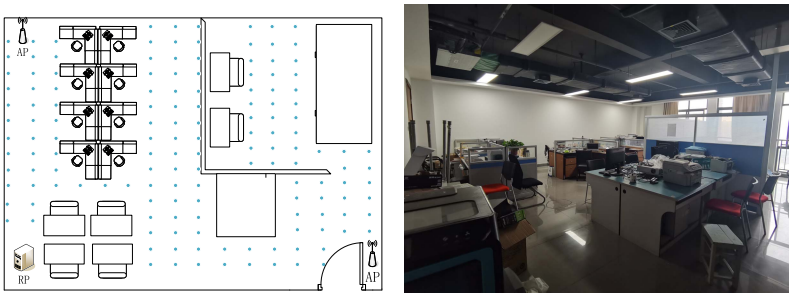
$$y_{\text{final}} = \sum_{i=1}^{5} p_i y_i \tag{20}$$

IV. Experiments and Discussion

In this section, we will explain the experimental environment setup and evaluate our proposed TL system in the context of environment changes including the effectiveness in the same scene with environment changes and new scenes. And we will discuss SSO and the impact of different parameters on positioning accuracy.

A. Experimental Condition

In order to verify the effectiveness and robustness of the proposed method, the effect is verified in four scenarios, which are shown in Fig.7 to Fig.9. Sence 1 is an open environment with an area of 20m*15m. Sence 2 and Sence 3 are closed laboratory with size of 10m*15m. Sence 4 is a closed laboratory with size of 10m*10m, in where the blue points are the fingerprints. TL-WDR6500 router was used as the AP, and the Intel 5300 network card was used as the RP. AP and RP have two transmit antennas and three receive antennas, respectively. The transmission frequency is 5 GHz. Meanwhile, we set the sampling rate to 100 packets per second

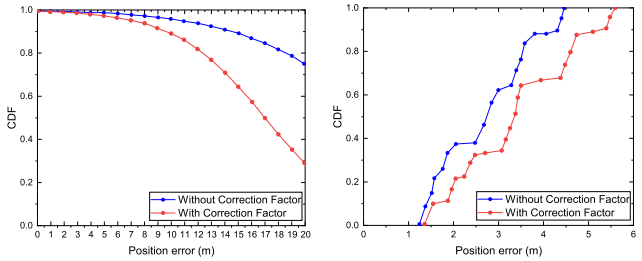


(a) (b)

Fig. 8. Scene 1. (a) Layout of scene 4. (b) True condition of scene 4.

TABLE II
CLASSIFICATIONRESULTS OF DIFFERENTPARAMETERS IN THE OPEN ENVIRONMENT

N	max_depth	min_samples_split	max_features	accurancy
	5			70%
20	10	5	10	78%
	5			76%
	5			75%
40	10	5	10	84%
	15			80%
	5			73%
60	10	5	10	82%
	15			78%



(a) (b)

Fig. 9. Influence of correction factor. (a) For 20 iterations with or without correction factor. (b) CDF of the absolute error in scene 1 with or without correction factor.

and the format of the received CSI data was $2 \times 3 \times 30$. We perform all of our experiments on a computer equipped with Intel i7-7700K CPU and an NVIDIA GTX 1080 GPU.

To measure the performance of the system, we will consider the following indicators:

1) Accuracy Ratio:the accuracy of location is:

$$Ar = m/n \tag{21}$$

where Ar represents accuracy of location, and the n represent the total test positioning point, m represent the correct result of the final system positioning.

2) Site Survey Overhead(SSO):

$$SSO = a * (r + d) \tag{22}$$

where SSO represents the total time for Site Survey Overhead, and the time to re-measure is r, the time to deploy points is d, and a is the number of points.

B. Influence of Different Parameters

1) Influence of System Parameters: In this system, the most important parameters are the number of iterations and max_depth which represents the depth of the decision tree. The size of the weight would be affected by modifying the number of iterations N. In addition, among other parameters of basic classifier, the biggest influence is max_depth. Table II shows the classification accuracy of different parameters in the scene 1. Constantly adjusting the parameters $max\_features$ and $min\_samples\_split$, we found that when $max\_features$ is set to 10 and $min\_samples\_split$ is set to 5, the best classification accuracy can be achieved while keeping other parameters consistent. When max_depth is 10, the classification accuracy is the highest compared to 5 and 15. Therefore, we set max_depth to 10, the number of iterations N to 40, and $max\_features$ and $min\_samples\_split$ respectively into 10 and 5.

2) Influence of the Correction Factor: We evaluated the influence of the correction factor on the weight dropping and positioning accuracy of the Tradaboost algorithm in scene 1. It can be seen in Fig. 9(a) that the decline rate of sample weights in the source domain is effectively slowed down by the correction factor. And Fig. 9(b) shows that the Tradaboost algorithm with a correction factor has better CDF of the absolute error. This proves that the correction factor can effectively reduce the decline speed of the source domain sample weight and improve the accuracy of positioning.

C. Location Estimation Accuracy

1) Influence of Environment Changes in Same Scene: To verify the effectiveness of the algorithm, we perform positioning experiments in the same scene as the environment changes, as shown in Fig. 10 (target domain). We added 2 experimenters and 4 chairs compared to sence 1(source domain).

Figure 11 shows the changes in positioning accuracy before and after the environmental changes in the scene. It can be seen that after the scene changes, the positioning accuracy without the transfer is significantly lower than the positioning accuracy using TL. That is because the data in the fingerprint database was not updated in time, which caused low accuracy when the scene changed.

Fig. 10. Environment changes in scene 1.

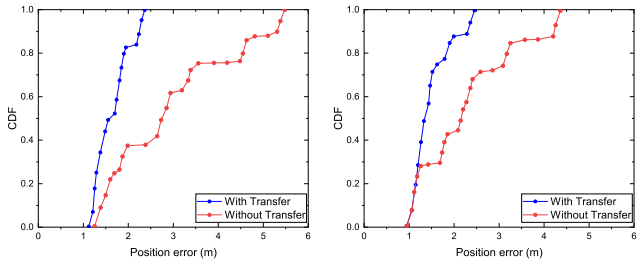


Fig. 11. CDF of the absolute error. (a) Scene 1. (b) Scene 2.

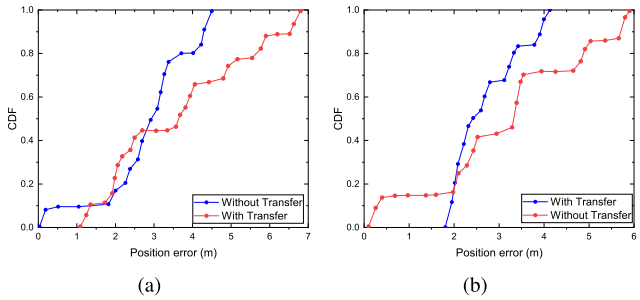


Fig. 12. CDF of the absolute error. (a) Scene 2 as source domain, Scene 1 as target domain. (b) Scene 1 as source domain, Scene 2 as target domain.

2) Influence of Different Scenes on Positioning Accuracy: We also evaluated the CDF of positioning accuracy errors in different scenarios. In Fig. 12(a), we use scenario 1 as the source domain and scenario 2 as the target domain. Correspondingly, in Fig. 12(b), we use scenario 2 as the source domain and scenario 1 as the target domain. As can be seen from the Figure 13, the average positioning accuracy using the Transfer algorithm is better than that without the Transfer algorithm.

D. Site Survey Overhead

To compare the survey overhead during the actual positioning process, we designed an experiment in which we deployed 11 training points in different source domain(Sence 1, Sence 2). In this experiment, we set the time for marking fingerprint points to $1min$, and the time for collecting data for each fingerprint point to $3min$.

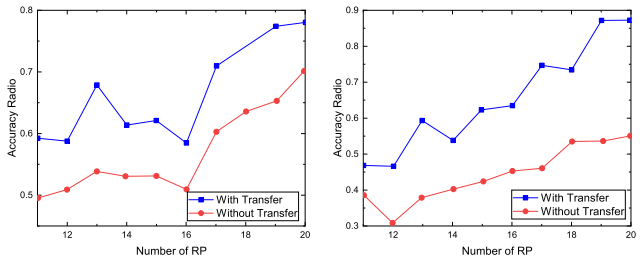


Fig. 13. The impact of different number of RP points on AR. (a) Sence 1. (b) Sence 2.

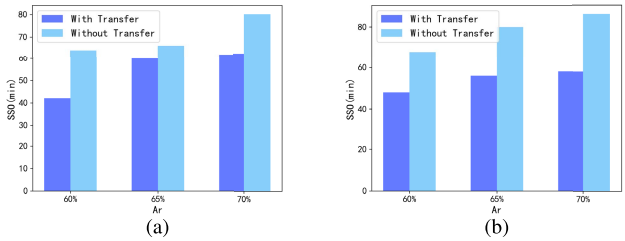


Fig. 14. *SSO* under different accuracy requirements. (a) Scene 1. (b) Scene 2.

Therefore, the SSO is $11 * (3 + 1) = 44mins$. It should be noted here that this is the necessary SSO of whether using the Transfer or not using the Transfer, so it is not included in the cost. For the target domain, during the regular training phase, we gradually increase the number of training points and collect 3mins of data on it until the required accuracy is achieved. The marking time of each point is also considered to be 1min. In the online positioning phase, 10 points in the target domain are randomly tested and the accuracy of location is calculated.

In Fig. 13(a) scene 1 serves as the target domain, and accordingly, scene 2 serves as the source domain, and vice versa in Fig. 13(b). It can be seen that with the increasing number of training points in the training process, in scene 1, the positioning accuracy of not using the Transfer is always lower than that of using the Transfer, and it is always below the growth of using TL for indoor positioning. In scene 2, with the increase of points, the growth rate of the positioning accuracy using Transfer is higher than that without using Transfer. This shows that the TL can be well used in the online training process, which means small number of fingerprint points can achieve higher precision.

In Fig. 14, we compare SSO of using transfer and not using transfer at the same accuracy. Specifically, the X-axis shows the positioning accuracy achieved, and the Y-axis shows the minimum overhead of site survey required to achieve the corresponding accuracy. As shown in Fig. 14, as the accuracy requirements increase, both the transfer and the non-use Transfer require more SSO.

This is reasonable because more training data is needed to achieve higher accuracy. However, we can also see that, under the same accuracy conditions, the transfer can achieve lower SSO compared to no Transfer. Moreover, it can be seen from

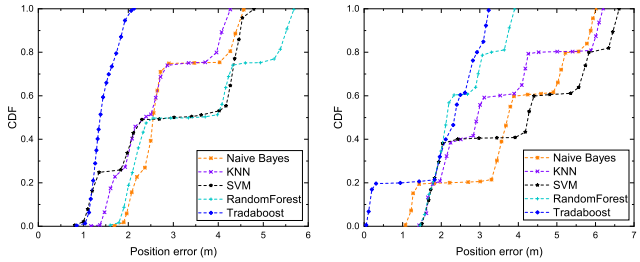


Fig. 15. Comparison with other machine learning methods for positioning accuracy. (a) Scene 1. (b) Scene 2.

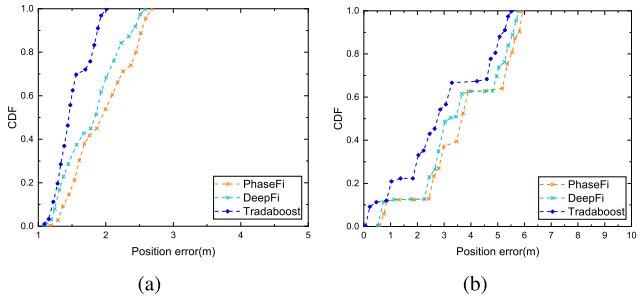


Fig. 16. Comparison with DeepFi and PhaseFi for positioning accuracy in dynamic environment. (a) Scene 1. (b) Scene 2.

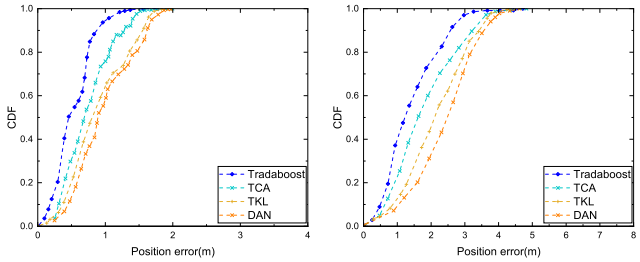


Fig. 17. Comparison with indoor position methods based TL in dynamic environment. (a) Scene 1. (b) Scene 3.

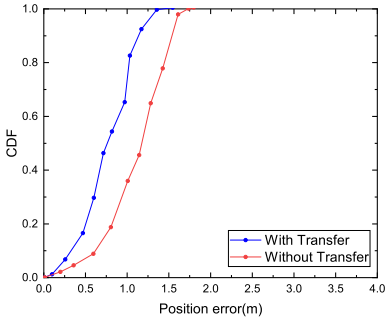


Fig. 18. CDF of the absolute error after one month.

the Fig. 15 that whether scene 1 or scene 2 is used as the target domain can reduce the field overhead, which verifying the universality of the Transfer framework.

E. Comparison of Different Methods

The comparison of different methods is divided into four parts. The first is the traditional machine learning methods, including Naive Bayes, KNN [14], SVM [35] and RandomForest. Then there are different positioning methods including DeepFi [17] and PhaseFi [18]. Followed by the comparison of positioning methods based on transfer learning, including TKL [25], DAN [27], and TCA [26]. The last is a comparison of two different regression methods.

1) Comparison of Different Machine Learning Methods: We use other machine learning methods for the indoor positioning to compare with the TrAdaBoost algorithm. As can be seen from Fig. 15, in scene 1, using TrAdaBoost for positioning, 70% of the test points have a positioning error of less than 1.5m, and 50% of the positioning error is less than 1.3m which is lower than other machine learning methods. In the larger indoor scene 2, using TrAdaBoost for positioning and using KNN($k = 5$), SVM($kernel = rbf$, $c = 1$) and Random Forest ($max\_features = 5$, $max\_depth = 10$, $max\_sample\_split = 5$) have 40% test point error of less than 2m, but 70% of TrAdaBoost's positioning error is only 2.5m. Overall test points have a positioning error of less than 3.2m, which is superior to other machine learning methods. This also demonstrates the advantages of the TrAdaBoost algorithm versus traditional machine learning methods in the indoor positioning.

2) Comparison of Different Indoor Position Methods: We also compared with the DeepFi and PhaseFi. As shown in Fig. 16, the average positioning error of our proposed method is better than the other two methods in both scenarios, which demonstrates the advantages of the TrAdaBoost algorithm in the indoor positioning.

3) Comparison of Different Transfer Learning Indoor Position Methods: We compared TCA, DAN and TKL three transfer learning methods used in indoor positioning, and the positioning results are shown in the Fig. 17. It can be seen that the positioning effect of Tradaboost is better than the other positioning methods.

F. Robustness of System

In order to verify the robustness of the system, we compare the positioning effect with and without TL in our laboratory scene(scene 4). As can be seen from the Fig. 18, the average positioning accuracy using TL is 0.75m. In contrast, the average accuracy without TL is 1.42m, which shows that the proposed method is still effective and has robustness in time. This experiment shows that the positioning system of TradaBoost proposed by us has a good positioning effect compared with non-transfer positioning under dynamic environment changes, especially long-term changes.

V. Conclusion

In this paper, we propose a low-overhead indoor positioning system based on TrAdaBoost. We use CSI phase data that has been corrected and reduced dimension as fingerprint features and input them into the improved TrAdaBoost system, which combines the One-Hot encoding and One-vs-Rest algorithms. Compared with the image and speech data, the phase data based fingerprint point location method has less features. Therefore, the TL algorithm based on NN, such as TCA and

DAN, cannot significantly improve the positioning effect. The proposed method is tested in different space and time, and the experimental results verify the effectiveness of this method and the advantages of low SSO and algorithm complexity.

REFERENCES

[1] R. Tan, Y. Tao, W. Si, and Y.-Y. Zhang, "Privacy preserving semantic trajectory data publishing for mobile location-based services," *Wireless Netw.*, vol. 26, no. 8, pp. 1–10, Jun. 2019.

[2] 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.

[3] R. Want, A. Hopper, V. Falcão, and J. Gibbons, "The active badge location system," *ACM Trans. Inf. Syst.*, vol. 10, no. 1, pp. 91–102, Jan. 1992.

[4] X.-H. Li and T. Zhang, "Research on improved UWB localization algorithm in NLOS environment," in *Proc. Int. Conf. Intell. Transp., Big Data Smart City (ICITBS)*, Jan. 2018, pp. 707–711.

[5] P.-H. Tseng and K.-T. Lee, "A femto-aided location tracking algorithm in LTE-A heterogeneous networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 748–762, Jan. 2017.

[6] D. Wang, M. Fattouche, and F. M. Ghannouchi, "Bounds of mmWave-based ranging and positioning in multipath channels," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2017, pp. 1–6.

[7] C. Luo, L. Cheng, M. C. Chan, Y. Gu, J. Li, and Z. Ming, "Pallas: Self-bootstrapping fine-grained passive indoor localization using WiFi monitors," *IEEE Trans. Mobile Comput.*, vol. 16, no. 2, pp. 466–481, Feb. 2017.

[8] M. Kordestani, A. Chibakhsh, and M. Saif, "A control oriented cyber-secure strategy based on multiple sensor fusion," in *Proc. IEEE Int. Conf. Syst., Man Cybern. (SMC)*, Oct. 2019, pp. 1875–1881.

[9] M. Kordestani, M. Dehghani, B. Moshiri, and M. Saif, "A new fusion estimation method for multi-rate multi-sensor systems with missing measurements," *IEEE Access*, vol. 8, pp. 47522–47532, 2020.

[10] 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. 2016.

[11] Q. Tran, J. Tantra, C. Foh, A.-H. Tan, K. Yow, and D. Qiu, "Wireless indoor positioning system with enhanced nearest neighbors in signal space algorithm," in *Proc. IEEE Veh. Technol. Conf.*, Sep. 2006, pp. 1–5.

[12] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system," in *Proc. IEEE INFOCOM Conf. Comput. Commun., 19th Annu. Joint Conf. IEEE Comput. Commun. Societies*, Mar. 2000, pp. 775–784.

[13] M. Youssef and A. Agrawala, "The Horus WLAN location determination system," in *Proc. 3rd Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, 2005, pp. 205–218.

[14] C. Laoudias, D. G. Eliades, P. Kemppi, C. G. Panayiotou, and M. M. Polycarpou, "Indoor localization using neural networks with location fingerprints," in *Proc. Int. Conf. Artif. Neural Netw.* Springer, 2009, pp. 954–963.

[15] 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.

[16] Z. Yang, Z. Zhou, and Y. Liu, "From RSSI to CSI: Indoor localization via channel response," *ACM Comput. Surv.*, vol. 46, no. 2, pp. 1–32, Nov. 2013.

[17] X. Wang, L. Gao, S. Mao, and S. Pandey, "DeepFi: Deep learning for indoor fingerprinting using channel state information," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Mar. 2015, pp. 1666–1671.

[18] 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)*, Dec. 2014, pp. 1–6.

[19] Y. Zhang, D. Li, and Y. Wang, "An indoor passive positioning method using CSI fingerprint based on AdaBoost," *IEEE Sensors J.*, vol. 19, no. 14, pp. 5792–5800, Jul. 2019.

[20] W. Shao, H. Luo, F. Zhao, Y. Ma, Z. Zhao, and A. Crivello, "Indoor positioning based on fingerprint-image and deep learning," *IEEE Access*, vol. 6, pp. 74699–74712, 2018.

[21] E. Schmidt, D. Inupakutika, R. Mundlamuri, and D. Akopian, "SDR-Fi: Deep-learning-based indoor positioning via software-defined radio," *IEEE Access*, vol. 7, pp. 145784–145797, 2019.

[22] K. Liu *et al.*, "Toward low-overhead fingerprint-based indoor localization via transfer learning: Design, implementation, and evaluation," *IEEE Trans. Ind. Informat.*, vol. 14, no. 3, pp. 898–908, Mar. 2018.

[23] S. D. Regani, Q. Xu, B. Wang, M. Wu, and K. J. R. Liu, "Driver authentication for smart car using wireless sensing," *IEEE Internet Things J.*, vol. 7, no. 3, pp. 2235–2246, Mar. 2020.

[24] Z. Xiao, H. Wen, A. Markham, N. Trigoni, P. Blunsom, and J. Frolik, "Non-line-of-sight identification and mitigation using received signal strength," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1689–1702, Mar. 2015.

[25] H. Zou, Y. Zhou, H. Jiang, B. Huang, L. Xie, and C. Spanos, "A transfer kernel learning based strategy for adaptive localization in dynamic indoor environments: Poster," in *Proc. 22nd Annu. Int. Conf. Mobile Comput. Netw.*, Oct. 2016, pp. 462–464.

[26] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011.

[27] L. Wang, Y. Shao, and X. Guo, "An adaptive localization approach based on deep adaptation networks," in *Proc. Int. Conf. Control, Autom. Inf. Sci. (ICCAIS)*, Oct. 2019, pp. 1–5.

[28] S. J. Pan, D. Shen, Q. Yang, and J. T. Kwok, "Transferring localization models across space," in *Proc. AAAI*, 2008, pp. 1383–1388.

[29] V. W. Zheng, E. W. Xiang, Q. Yang, and D. Shen, "Transferring localization models over time," in *Proc. AAAI*, 2008, pp. 1421–1426.

[30] 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. Commun. (INFOCOM)*, Apr. 2015, pp. 2038–2046.

[31] W. Dai, Q. Yang, G.-R. Xue, and Y. Yu, "Boosting for transfer learning," in *Proc. 24th Int. Conf. Mach. Learn. (ICML)*, 2007, pp. 193–200.

[32] Y. Matsunaga, "Accelerating SAT-based Boolean matching for heterogeneous FPGAs using one-hot encoding and CEGAR technique," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E99.A, no. 7, pp. 1374–1380, 2016.

[33] J.-H. Hong and S.-B. Cho, "A probabilistic multi-class strategy of one-vs.-rest support vector machines for cancer classification," *Neurocomputing*, vol. 71, nos. 16–18, pp. 3275–3281, Oct. 2008.

[34] S. Al-Stouhi and C. K. Reddy, "Adaptive boosting for transfer learning using dynamic updates," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*. Springer, 2011, pp. 60–75.

[35] Z.-L. Wu, C.-H. Li, J. K.-Y. 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.

Yong Zhang was born in Anhui, China, in 1973. He received the doctor's degree in signal and information processing from the University of Science and Technology of China in 2007. He is an Associate Professor. His research interests include intelligent information processing, wireless sensor networks, and indoor positioning.

Chengbin Wu was born in Anhui, China, in 1996. He received the B.S. degree from the Hefei University of Technology in 2018, where he is currently pursuing the M.Sc. degree. His research interest includes indoor positioning.

Yang Chen was born in Anhui, China, in 1997. He received the B.S. degree from the Hefei University of Technology in 2019, where he is currently pursuing the M.Sc. degree. His research interest includes indoor positioning.