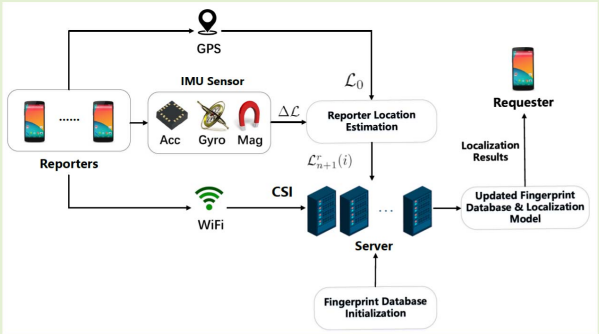
Crowdsourcing-Based Indoor Localization With Knowledge-Aided Fingerprint Transfer

Chenlu Xiang, *Graduate Student Member, IEEE*, Shunqing Zhang, *Senior Member, IEEE*, Shugong Xu, *Fellow, IEEE*, and Guoqiang Mao, *Fellow, IEEE*

***Abstract*—Precise indoor localization is a key requirement for the fifth-generation (5G) and beyond wireless communication systems with applications. To that end, many high accuracy signal fingerprint-based localization algorithms have been proposed. Most of these algorithms, however, face the problem of performance degradation in indoor environments, when the propagation environment changes with time. In order to address this issue, the crowdsourcing approach has been recently adopted, where the fingerprint database is frequently updated via user reporting. These crowdsourcing techniques still require precise indoor floor plans and fail to provide satisfactory accuracy. In this paper, we propose a low-complexity self-calibrating indoor crowdsourcing localization system that combines historical fingerprints with frequently updated fingerprints for high precision user positioning. We present a multi-kernel transfer learning approach that exploits the inner relationship between the historical and updated channel state information (CSI). Our indoor laboratory experimental results using Nexus 5 smartphones at 2.4GHz with 20MHz bandwidth have shown the feasibility of the proposed approach to achieve about one-meter level accuracy with a reasonable fingerprint update overhead.**



***Index Terms*—Indoor localization, crowdsourcing, channel state information, fingerprint database, transfer learning.**

I. Introduction

AMONG the technical requirements for the fifth-generation (5G) wireless communications and beyond, there is an important requirement for the precise indoor localization, which can bring, as representative applications, accurate navigation experience [2] in shopping malls and seamless tracking in smart factories [3]. Different from the outdoor localization process, where the combination of the global navigation satellite system (GNSS) with the inertial navigation system (INS) [4] provides satisfactory accuracy, indoor localization solutions are quite diversified, such as those based on the low-cost Bluetooth Low Energy (BLE) [5], the increasingly popular 3GPP LTE/5G [6], and the widely deployed WiFi technologies [7]–[11]. Nevertheless, signal fingerprinting approaches, including large-scale received signal strength indicator (RSSI) [12], or reference signal received power (RSRP) [6], or small-scale channel state information (CSI) [8], [10], are usually recognized as the most efficient solutions for high accuracy indoor localization.

The availability of timely and accurate signal fingerprint maps is of paramount importance for the aforementioned fingerprint-based localization approaches. However, the collection of fingerprint maps is often considered as a labor-intensive task. Furthermore, fingerprint maps can be easily corrupted by the fluctuations in the wireless channels, due to human movement or time-varying scattering and reflections. It was shown in [13] that those factors can gradually, over time, degrade the localization accuracy. In order to solve this problem, reference [14] proposed to regularly update the fingerprint database in order to maintain the positioning accuracy without deterioration. However, the associated system maintenance costs were prohibitively expensive [15].

Manuscript received October 30, 2021; accepted January 7, 2022. Date of publication January 12, 2022; date of current version February 28, 2022. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 62071284, Grant 61871262, Grant 61901251, and Grant 61904101; in part by the National Key Research and Development Program of China under Grant 2017YEF0121400 and Grant 2019YFE0196600; in part by the Innovation Program of Shanghai Municipal Science and Technology Commission under Grant 20JC1416400; in part by the Pudong New Area Science & Technology Development Fund; and in part by the Research Funds from the Shanghai Institute for Advanced Communication and Data Science (SICS). An earlier version of this paper was presented at the IEEE ICC 2021 [1] [DOI: 10.1109/ICC42927.2021.9500623]. The associate editor coordinating the review of this article and approving it for publication was Dr. Yulong Huang. *(Corresponding author: Shunqing Zhang.)*

Chenlu Xiang, Shunqing Zhang, and Shugong Xu are with the Key Laboratory of Specialty Fiber Optics and Optical Access Networks, School of Information and Communication Engineering, Shanghai Institute for Advanced Communication and Data Science, Shanghai University, Shanghai 200444, China (e-mail: xcl@shu.edu.cn; shunqing@shu.edu.cn; shugong@shu.edu.cn).

Guoqiang Mao is with the State Key Laboratory of Integrated Services Networks, Xidian University, Xi'an 710071, China (e-mail: gqmao@xidian.edu.cn).

Digital Object Identifier 10.1109/JSEN.2022.3142261

To address the aforementioned indoor localization issues, we present in this paper a novel self-calibrating indoor fingerprint-based localization system. By efficiently utilizing the dynamically updated fingerprints from crowdsourcing, we present a localization scheme based on multi-kernel transfer learning. The proposed scheme strives to keep the WiFi fingerprint database as well as the neural network models updated, in order to efficiently track the variations of the wireless channel for a long time period, while exhibiting reasonable implementation complexity. Our experimental results in a laboratory have showcased the performance of the proposed indoor localization scheme, which can achieve mean localization errors as low as one meter. The main contributions of this paper are summarized as follows.

- **Knowledge-Aided Fingerprint Reporting.** The newly reported fingerprints from crowdsourcing reporters inevitably contain position errors. To solve this problem, we develop the fingerprint confidence weight according to prior knowledge about key acquisition parameters of the newly reported fingerprints. This process ensures the effective update of the fingerprint database. In this way, it prevents inaccurately reported samples from deteriorating the expected system positioning accuracy.
- **Transfer Learning with the Updated Fingerprint Database.** The limited localization accuracy of conventional localization schemes is mainly due to ignoring the relationship between outdated and updated fingerprint databases. To overcome this obstacle, we introduce the maximum mean discrepancy (MMD) metric that describes the differences of their distributions and further extend it to a multi-kernel MMD (MK-MMD) version by incorporating the conditional CSI distributions. Through this approach, the proposed localization scheme can provide a transfer learning-based adaption among different fingerprint databases and achieve a better localization accuracy.
- **Simultaneous Self-Calibrating and Classification.** In order to balance the localization accuracy and the MK-MMD distances of self-calibrated fingerprint databases, we design a combined loss function on top of the proposed deep transfer learning framework. With this design, the proposed framework can simultaneously update the fingerprint database and the corresponding localization function, which facilitates the utilization of historical as well as updated fingerprint information for high precision localization, and eventually addresses the aforementioned issues.

The rest of this paper is organized as follows. In Section II, we summarize the existing technologies related to our proposed localization system. We introduce our proposed crowdsourcing indoor localization system and the process of reporter location estimation in Section III and Section IV, respectively. We then present the considered problem formulation and the implementation details of the proposed deep transfer learning approach in Section V and Section VI, respectively. Our experimental results are discussed in Section VII, while Section VIII concludes this paper.

II. Related Work

A. Fingerprint-Based Localization

Many methods based on signal fingerprints, e.g., [6], [12] and [10], have achieved impressive localization results. Those methods aim to extract the intrinsic features of wireless signals in the training phase, which are then utilized in the online operating phase to predict the user location in conjunction with real-time measurements. The RSSI [12] and RSRP [6] metrics have been proven to be highly correlated with the spatial locations, and thus, adopted for improving the localization accuracy to the level of a meter in indoor and outdoor scenarios, respectively. Furthermore, CSI measurements, which are frequently reported in [7], [8], [10], [16], [17], have managed to further improve the localization accuracy. Those measurements have complex structures and more dimensions than traditional fingerprints. Probabilistic models between the collected CSI and the candidate locations are established through some classifiers, such as deterministic k-nearest neighbor (KNN) clustering [16], probabilistic Bayes rule algorithms [17], and deep learning-based algorithms [10], [18]. The above fingerprint-based solutions have been shown to be able to achieve sub-meter, even decimeter level accuracy, if CSI from multiple APs [7], multiple frequency bands [8], and/or multiple antennas [17] can be fused together.

B. Fingerprint Update

The key of fingerprint-based localization methods is to establish the relationship between the fingerprint and the corresponding position. The related work mentioned above assumes that the indoor wireless environment does not change, and then establishes the wanted relationship between the fingerprint and the corresponding position. However, if the modeled relationship are changed by human movement or time-varying scattering and reflections, the system localization accuracy will rapidly deteriorate. Although [19], [20] have studied adapting fingerprints to environmental changes. These works assume a certain trend in signals at neighboring areas, which may not hold with AP movement and power adjustment.

Recently, a low-cost alternative scheme named *crowdsourcing* [21] has been proposed to keep the fingerprint database up-to-date via collaborative user reporting. With the aid of floor plans, the existing crowdsourcing systems [22] can update the fingerprint database by matching the mobile users' zigzag routing, estimated by inertial measurement unit (IMU) results for a period of time with the pre-acquired floor plan. However, the above method suffers from the inaccurate location information of crowdsourcing users, which often results in error accumulating events, as shown in [22]. In addition, the localization approaches only update the fingerprint database by newly collected fingerprints using conventional multi-dimensional scaling (MDS) [23], marginalized particle filtering (MPF) [24] or Gaussian process regression (GPR) [25] schemes, which have been shown to exhibit poor localization accuracy.

C. Transfer Learning

Transfer learning methods have been proven to be an efficient method to reduce the labeling cost by transferring the knowledge from the source domain to the target domain in computer vision tasks [26]. As a typical problem situation in transfer learning, domain adaptation addresses the problem that we have data from two related domains but under different distributions, and the fingerprint data from different time periods can be viewed as two related domains with different distributions. Recently, some domain adaptation methods, like transfer component analysis (TCA) [27] and joint distribution adaptation (JDA) [28], have been proposed to mitigate the impact of RSSI variation and help to solve the RSSI based localization problem. However, most existing methods show poor performance in severe RSSI variations caused by changing environment and heterogeneous hardware, because most of them learn new shallow representation features due to limited representation ability, which can only slightly reduce the domain discrepancy. Deep neural networks can learn deep transferable features to manifest invariant factors underlying different domains, making hidden representations robust to noise. Thus, some deep transfer learning methods [29], [30]. apply deep neural networks for domain adaptation, which can learn more transferable features. However, these methods either aim for the binary classification problem or require a certain amount of samples in the target domain, and thus cannot be applied to fingerprint-based indoor localization directly.

Different from the existing methods, our proposed transfer learning-based algorithm utilizes sparse crowdsourcing data from reporters to establish fingerprint database relationships on different time scales, thereby improving the online localization accuracy.

III. System Overview

In this section, we present the proposed crowdsourcing system architecture, which is depicted in Fig. 1, and discuss the initialization of the architecture's fingerprint database as well as its updating procedure via crowdsourcing.

A. Fingerprint Database Initialization

The initial fingerprint database, whose content is denoted as $\mathcal{DB}_0$, stores the CSI samples from N_R discrete grid locations, e.g., $\{\mathcal{A}_m, \forall m \in [1, \ldots, N_R]\}$. We make use of the notation $\mathbf{H}(\mathcal{A}_m, T_0) \in \mathbb{C}^{N_{sc} \times N_s}$ for the aggregated channel response of N_{sc} subcarriers and N_s consecutive orthogonal frequency division multiplexing (OFDM) symbols in a grid area $\mathcal{A}_m$ at the initial time instant T_0. The entire fingerprint database is initialized as follows,

$$\mathcal{DB}_0 = \{(m, \mathbf{H}(\mathcal{A}_m, T_0)), \forall m \in [1, \ldots, N_R]\}. \quad (1)$$

In practice, it is in general hard to obtain the channel responses of the entire grid area. For our fingerprint database initialization, the CSI samples at the center locations $\{\mathcal{L}_m, \forall m \in [1, \ldots, N_R]\}$ of each grid area (see the red points in our laboratory experimental results, as illustrated later in Fig. 8), are considered as the regional CSI, $\mathbf{H}(\mathcal{A}_m, T_0)$, and CSI at the reference point $\mathcal{L}_m$, e.g., $\mathbf{H}(\mathcal{L}_m, T_0)$. These measurements can be obtained via a standard minimum mean square error (MMSE) detection algorithm, as discussed in [31].

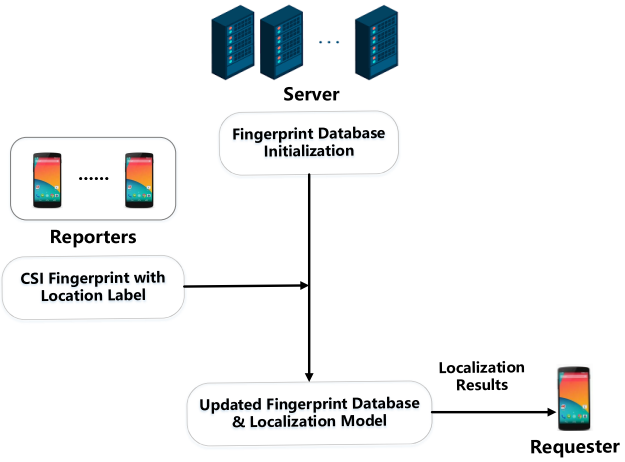


Fig. 1. The proposed crowdsourcing system includes one or more reporters, a server, and a requester. The system operator designs the initial fingerprint database at the server, and the reporter provides frequent updates on its fingerprints with location labels. Based on this information, the server keeps updating the relationship between the database with the fingerprints and the reporter locations. In the online phase, the requester leverages the latter updated mapping to obtain its precise localization.

B. Fingerprint Database Update

For each duration between the time instants T_n and T_{n+1}, we assume N^r_{n+1} reporters are transmitting their respective location information $\{\mathcal{L}^r_{n+1}(i)\}$ and collected CSI $\{\mathbf{H}(\mathcal{L}^r_{n+1}(i), T_{n+1})\}$, where $i \in [1, \ldots, N^r_{n+1}]$, to the central server. In order to obtain the accurate location $\mathcal{L}^r_{n+1}(i)$, we utilize the WiFi, GPS receiver, and IMU sensors, as shown in Fig. 2. Let us denote by $\mathbf{\Omega}_i(\mathcal{A}_m)$ the set of reports in the area $\mathcal{A}_m$, e.g., $\mathbf{\Omega}_i(\mathcal{A}_m) = \{i, \forall i \text{ satisfying } \mathcal{L}^r_{n+1}(i) \in \mathcal{A}_m\}$. We can update the fingerprint $\mathbf{H}(\mathcal{A}_m, T_{n+1})$ by averaging the collected CSI within the area $\mathcal{A}_m$ in order to eliminate occasional measurement errors,[1] which is given by

$$\mathbf{H}(\mathcal{A}_m, T_{n+1}) = \begin{cases} \dfrac{\omega_{n+1}}{|\mathbf{\Omega}_i(\mathcal{A}_m)|} \displaystyle\sum_{i \in \mathbf{\Omega}_i(\mathcal{A}_m)} \mathbf{H}(\mathcal{L}^r_{n+1}(i), T_{n+1}) \\ \quad +(1-\omega_{n+1}) \cdot \mathbf{H}(\mathcal{A}_m, T_n), & \mathbf{\Omega}_i(\mathcal{A}_m) \neq \emptyset, \\ \mathbf{H}(\mathcal{A}_m, T_n), & \mathbf{\Omega}_i(\mathcal{A}_m) = \emptyset, \end{cases}$$

where $|\cdot|$ denotes the cardinality of the inner set, $\emptyset$ denotes the empty set, and $\omega_{n+1} \in [0, 1]$ denotes the confidence weight of the collected fingerprint at T_{n+1}, which will be introduced in Section IV. Then, the fingerprint database at T_{n+1}, i.e., $\mathcal{DB}_{n+1}$, is given by

$$\mathcal{DB}_{n+1} = \{(m, \mathbf{H}(\mathcal{A}_m, T_{n+1})), \forall m \in [1, \ldots, N_R]\}. \quad (2)$$

[1]If the newly collected CSI is not labeled correctly (poison added), the relationship between the fingerprint and the corresponding location at T_{n+1} cannot be obtained correctly. Then the accuracy of the position estimation in the online phase cannot be guaranteed as well.

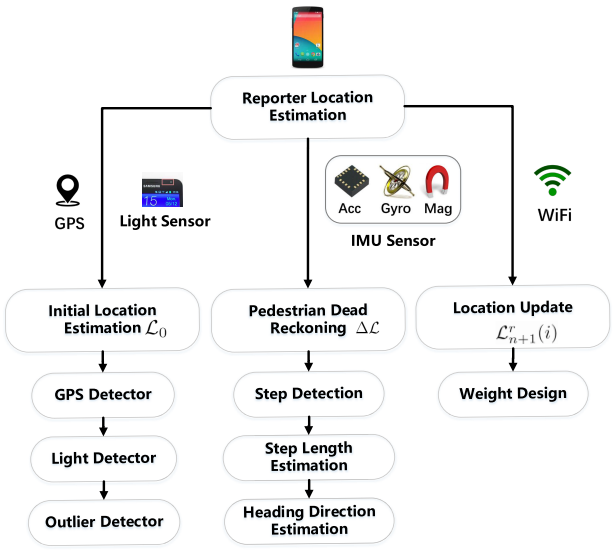


Fig. 2. The proposed reporter location estimation process consists of three parts, including the initial position estimation and pedestrian dead reckoning, and location update. Each part is composed of several steps, which involve multiple sensors such as GPS, light sensor, IMU sensor, and WiFi on the smartphone.

After the location fingerprint database is updated, the corresponding localization model is required to be updated as well to provide requesters with more accurate localization services, which will be introduced in Section VI.

IV. Reporter Location Estimation

In this section, we will introduce the process of obtaining the reporters' indoor position, including the initial position estimation and pedestrian dead reckoning, and location update. The whole process is illustrated in Fig. 2. It is worth noting that we are required to assess the accuracy of the newly reported fingerprints, to avoid contaminating the original fingerprint database or deteriorating the system positioning accuracy.

A. Initial Location Estimation

It is well known that GPS is currently the most widely applied GNSS service in the world, and is deployed on most mobile terminals. Actually, our smartphones often receive strong GPS signals and acquire accurate positioning coordinates in some open outdoor areas, while indoor GPS signals are usually considered too weak for indoor localization due to signal blockage. However, opportunistic GPS localization can be accessible to annotate the newly collected fingerprint at building entrances, windows or balconies. In order to improve the accuracy of the prior position $\mathcal{L}_0$ precision, we have designed the following detectors to help assess the GPS signal quality.

1) GPS Detector: We perform a GPS signal strength test at various locations through smartphones indoors and outdoors. Based on the test results, we have found that when the mean signal-to-noise ratio (SNR) of GPS signals from all the detected satellites exceeds 26 dBm, GPS signals can give one meter level localization results. Note that if the number of the detected satellites is smaller than 3, the localization results are considered inaccurate.[2] Furthermore, the prior positions given by GPS signals are expected to be acquired in the case of stationary humans or terminals. To judge the indoor human movement and confirm the stability of the received GPS signal received, we require the GPS signal to fluctuate within 5 dbm during the localization process.

2) Light Detector: As mentioned before, the mean SNR of the indoor GPS signal is obviously smaller than that of the outdoor. We believe that only the locations obtained by outdoor GPS signals are usable. Therefore, we utilize the existing light sensors on smartphones to help calculate the user switching time between indoor and outdoor.

Our proposed light detector can be applied to both daytime and night scenarios, considering that smartphones can obtain the users' local time. We firstly utilize the proximity sensor on the smartphone to judge whether the light sensor is available. If the light sensor is available during the daytime and the light intensity is greater than 1000 lx (light intensity), the user is detected outdoor, otherwise indoor.[3] If the light sensor is available at night and the light intensity is greater than 500 lx, the user is detected indoor, considering the indoor intensity is usually larger than that of the outdoor at night. Using this approach, we accurately obtain the time point and corresponding locations when users switch between indoor and outdoor scenarios.

3) Outlier Detector: Due to measurement uncertainty of GPS signals, the smartphones will report a set of test results $\{\hat{\mathcal{L}}_m\}$, resulting in unsatisfactory localization accuracy. Hence, the outlier points of the observations may exist in the set $\{\hat{\mathcal{L}}_m\}$ and an outliers removal scheme is needed to rule out the outlier points, which are far away from the clustering set center in the decision process. We denote $\overline{\mathcal{L}_m}$ as the average point and $\text{std}_{\mathcal{L}}$ as the standard deviation of set $\{\hat{\mathcal{L}}_m\}$, which is defined as,

$$\text{std}_{\mathcal{L}} = \sqrt{\frac{1}{N_{gs}-1}\sum_{k=1}^{N_{gs}}(\hat{\mathcal{L}}_m - \overline{\mathcal{L}_m})^2}, \tag{3}$$

where N_{gs} is the sample group size. If the following condition:

$$\delta_{\mathcal{L}} = \frac{|\hat{\mathcal{L}}_m - \overline{\mathcal{L}_m}|}{\text{std}} > \delta_{th}, \tag{4}$$

is held, then $\hat{\mathcal{L}}_m$ is considered as an outlier point and removed from the set $\{\hat{\mathcal{L}}_m\}$, in which $\delta_{th} > 0$ is the designed rejection threshold. When we need more accurate GPS positions accuracy, we can reduce the parameter value at the cost of reducing the samples number.

With the help of the three proposed detectors, the localization accuracy provided by the collected indoor GPS signal is greatly improved. We find it accurate enough for the modified GPS signals to give the prior locations $\mathcal{L}_0$ at the building

[2]These values vary with different environments of buildings, and can be adjusted according to the actual situation [32].

[3]In the daytime, the light intensity outdoor is greater than indoors, even if there are lamps indoors. Considering the low light outdoor in extreme weather, our detector threshold is set to 1000 lx.

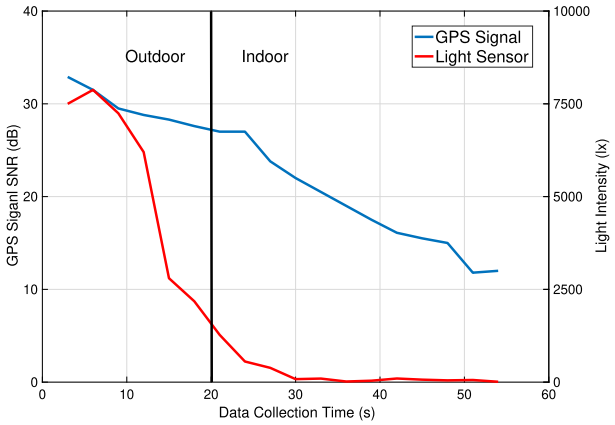


Fig. 3. The changes in the SNR of the GPS signal and in the light sensor readings during the daytime from outdoor to indoor positions. We can accurately obtain the timing when the user enters the room through the light sensor reading variance and estimate the initial location information.

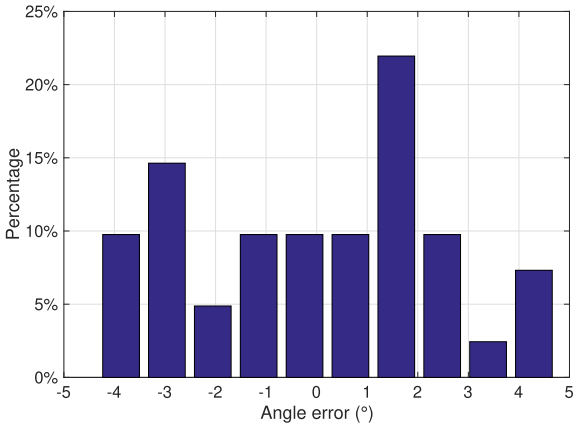


Fig. 4. In order to evaluate the performance of the heading direction estimation, we compare the calculated angles with the ground truth angles within one minute, and we find that the angle error can be controlled within 5°.

entrance, compared with meters-level WiFi localization errors. Here is a simple example as shown in Fig. 3.

B. Pedestrian Dead Reckoning

To further obtain the user traces, we apply an offline pedestrian dead reckoning (PDR) algorithm on the collected IMU sensor data, including the accelerometer, gyroscope, and magnetometer, as also performed in [32]. The whole process consists of step detection, step length estimation, heading direction estimation, and location update.

1) Step Detection: Due to the periodic fluctuations of the measured accelerometer data during the walk, we conduct step detection by monitoring the peaks and valleys of the triaxial accelerometer readings. We can obtain the magnitude of the triaxial accelerometer reading by

$$A = \sqrt{A_x^2 + A_y^2 + A_z^2}, \tag{5}$$

where A_x, A_y, and A_z are the acceleration in x, y and z axes.

In order to eliminate the influence of noisy accelerometer data, we utilize a low pass filter to filter out the high-frequency noise, which is caused by the random movement of the smartphone. Afterwards, we detect the peaks and valleys of the accelerometer reading magnitudes, which are marked as A^p and A^v expressed in m/sec^2. The corresponding detection time are marked as T^p (s) and T^v (s). We set the following conditions for step detection considering the average walking speed of pedestrians, which is shown as,

$$|A^p - A^v| > 0.5, \tag{6}$$

$$|T^p - T^v| > 0.15. \tag{7}$$

In this way, we manage to detect every user step and record the step number N_L accurately.

2) Step Length Estimation: The user's step length is closely related to his/her height and weight. If the user's height h (in meters) and weight w (in kilogram) information are already known, we can directly estimate the step length L (in meters) according to the following empirical formula in [32]:

$$L = 0.4\ h + 0.001\ w. \tag{8}$$

For example, we can calculate the step length of an adult male with a height of 1.73m and a weight of 70kg, which is $0.4 * 1.73 + 0.001 * 70 = 0.762m$ according to the formula.

However, it is not practical to ask each user about the body profile information, like height and weight. If no prior knowledge of user features is provided, we apply the personalization algorithm, as proposed in [33]. By analyzing the walk data from various volunteers, a linear relationship between step length and step frequency has been discovered. Therefore, the step length L (m) can be estimated as follows:

$$L = 0.25 f_s + 0.25, \tag{9}$$

where f_s (Hz) is the step frequency.

3) Heading Direction Estimation: Heading direction estimation is an important part of accurate path estimation. Though a magnetometer can provide the heading direction, the severe distortion of magnetic signals brings errors to direction estimation. Considering that even small direction errors can greatly affect the final results, we adopt a gyroscope and accelerometer fusion-based method to estimate the direction angle α_k.

Specifically, we measure the gravity value of the 3-axis by the accelerometer to determine the orientation of the smartphone. Then we use the gyroscope to measure the angular velocities (rad/sec) around x, y, and z, and obtain the degree of rotation. At last, we fuse the gyroscope and accelerometer results to estimate the Euler angles, and the angle error histogram is shown in Fig. 4.

After completing the process of step detection, step length estimation, and heading direction estimation, the PDR displacement within N_L steps is described as,

$$\Delta\mathcal{L} = \sum_{k=1}^{N_L} \alpha_k \cdot L_k, \tag{10}$$

where L_k is the k^{th} step length of reporter. A particle filter [24] is utilized to reduce IMU distance errors during this process.

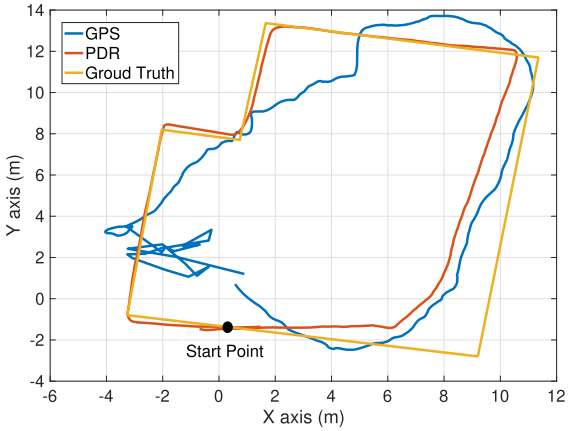


Fig. 5. Compared with the position provided by the GPS signal, the proposed PDR estimation algorithm provides a more accurate position estimation around the start point. But the localization errors of PDR estimation gradually increase over time.

With help of these three steps, we find it accurate enough to estimate the user PDR displacement $\Delta\mathcal{L}$ in a relatively short time period.[4] We test our proposed PDR estimation algorithm in an open space and the localization results are given in Fig. 5.

C. Location Update

By combing the prior location $\mathcal{L}_0$ and user PDR displacement $\Delta\mathcal{L}$, we can obtain the reporter's location by,

$$\mathcal{L}^r_{n+1}(i) = \mathcal{L}_0 + \Delta\mathcal{L}, \tag{11}$$

and record the corresponding CSI, $\mathbf{H}(\mathcal{L}^r_{n+1}(i), T_{n+1})$, at the same time. As mentioned before, we design the confidence weight ω_{n+1} for the newly collected fingerprint $\mathbf{H}(\mathcal{L}^r_{n+1}(i), T_{n+1})$ at the time instant T_{n+1} to represent the reported positions accuracy. Intuitively, the reported position accuracy is highly related to the implementation processes of initial location estimation and pedestrian dead reckoning, which involve multiple parameters such as the number of detected satellites, the mean SNR of the GPS signal, and so on. In order to describe the impact of each parameter on the final reported results, we regress an error model by collecting experimental data in an open space as described in Fig. 5. We adopt the procedure suggested in [34] to estimate each coefficient, and then the localization error e_n can be formulated as follows:

$$\begin{aligned} e_n = {} & \beta_0 + \beta_1 x_{1,n} + \beta_2 x_{2,n} + \cdots \\ & + \beta_i x_{i,n} \cdots + \beta_k x_{k,n} + \varepsilon_n, \end{aligned} \tag{12}$$

where $x_{i,n}$ is the i^{th} factor parameter, β_i is the coefficient of $x_{i,n}$, and ε_n is the residual term after regression. To be more specific, we choose several important factor parameters and learn the corresponding coefficients of each parameters, which are summarized in Table I. In order to obtain the corresponding coefficient, we have collected multiple sets of experimental parameters and corresponding real and estimated localization results. Considering that we need to fit five parameters β_0, β_1, β_2, β_3, β_4, we need more than 5 sets of experimental results and establish five equations to solve them. We utilize the least square (LS) method to solve the parameters, and we can obtain more optimized parameters if more sets of experimental data are given.

[4]To avoid the problem of IMU error accumulation, N_L values will not exceed 10 for continuous detection in our experimental settings.

TABLE I
ERROR MODEL COEFFICIENTS FOR TYPICAL LOCALIZATION PARAMETERS

Module	Parameters	Coefficient	Value
GPS	Number of detected satellites $x_{1,n}$	β_1	-0.05
	Mean SNR of GPS signals $x_{2,n}$	β_2	-0.001
IMU	Cumulative steps of PDR process $x_{3,n}$	β_3	0.03
	Distance from the last landmark $x_{4,n}$	β_4	0.1

Once the coefficients of the chosen parameters are given, we can obtain the confidence weight ω_{n+1} according to the calculated e_n, yielding:

$$\omega_{n+1} = \begin{cases} 1 - e_n/e_{val}, & e_n \le e_{val}, \\ 0, & e_n > e_{val}, \end{cases}$$

where e_{val} is the preset threshold used to indicate the system accuracy requirement.

V. PROBLEM FORMULATION

In this section, we present the considered crowdsourcing localization problem formulation. We apply a general optimization framework to describe the localization problem. Like some existing literature, such as [10], we formulate the localization problem as a common classification problem. Let $\mathcal{L}^k_n$ and $\hat{\mathcal{L}}^k_n$ be the ground truth and the predicted locations of the k^{th} target at T_n respectively, and the corresponding mean distance error (MDE) performance over K sampling positions is given by $\frac{1}{K}\sum_{k=1}^{K} \|\hat{\mathcal{L}}^k_n - \mathcal{L}^k_n\|_2$, where $\|\cdot\|_2$ represents the vector l_2 norm. With the above notation, we can describe the MDE minimization problem using the following optimization framework.

Problem 1 (MDE Minimization):

$$\underset{\{g_n(\cdot)\}}{\text{minimize}} \ \frac{1}{N}\frac{1}{K}\sum_{n=1}^{N}\sum_{k=1}^{K} \|\hat{\mathcal{L}}^k_n - \mathcal{L}^k_n\|_2 \tag{13}$$

$$\text{subject to } \hat{\mathcal{L}}^k_n = \left(\mathbf{p}^k_n\right)^T \cdot \mathcal{L}_c, \tag{14}$$

$$\mathbf{p}^k_n = g_n\left(\mathbf{H}(\mathcal{L}^k_n), \mathcal{DB}_n\right), \quad \forall n, \tag{15}$$

$$\mathbf{p}^k_n \in [0,1]^{N_R}, \quad \forall n, k, \tag{16}$$

where N is the total number of localization time instants, as well as $\mathcal{L}_c = [\mathcal{L}^1_c, \ldots, \mathcal{L}^m_c, \ldots, \mathcal{L}^{N_R}_c]$ and $\mathbf{p}^k_n$ denote the central grid positions and the position likelihood distribution of the k^{th} target at T_n with respect to all the possible $\mathcal{A}_m$,

respectively. The function g_n represents the unknown mapping relationship between the measured CSI $\mathbf{H}(\mathcal{L}_n^k)$ and $\mathbf{p}_n^k$.

In order to minimize the localization errors, it is necessary to estimate $\mathbf{p}_n^k$ by fitting the $g_n(\cdot)$ function, and further obtain a recursive relation with the new function $g_{n+1}(\cdot)$, which is represented as:

$$\mathbf{p}_{n+1}^k = g_{n+1}\left(\mathbf{H}(\mathcal{L}_{n+1}^k), \mathcal{DB}_{n+1}\right). \tag{17}$$

Considering that part of fingerprints in $\mathcal{DB}_{n+1}$ are updated compared with $\mathcal{DB}_n$, it is unnecessary to fit the $g_{n+1}(\cdot)$ function using the entire $\mathcal{DB}_{n+1}$. In order to reduce the computational complexity of the fingerprint database update, we propose to apply transfer learning for fingerprint transfer. Due to the difference between $\Pr(\mathbf{H}(\mathcal{A}_m, T_n))$ and $\Pr(\mathbf{H}(\mathcal{A}_m, T_{n+1}))$, the self-calibrating localization system needs to utilize a transfer mapping function $\Phi(\cdot)$ in order to model the difference in reproducing kernel Hilbert space (RKHS) instead, where $\Pr(\cdot)$ represents the channel distribution property. The corresponding MMD measure is thus given by

$$\begin{aligned}&\mathcal{D}_{\text{MMD}}(\mathcal{DB}_n, \mathcal{DB}_{n+1})\\ &= \sum\nolimits_{m=1}^{N_R} \left\|\Phi\left(\mathbf{H}(\mathcal{A}_m, T_n)\right)) - \Phi\left(\mathbf{H}(\mathcal{A}_m, T_{n+1})\right))\right\|_{\mathcal{H}}^2,\end{aligned} \tag{18}$$

where $\|\cdot\|_{\mathcal{H}}$ denotes the vector norm operation in RKHS. With the above manipulation, we are required to fit the optimal mapping function $\Phi(\cdot)$ to update the function $g_{n+1}(\cdot)$, that is, minimizing MMD. Then, we can transform the original MDE minimization problem into the following MMD minimization problem.

Problem 2 (MMD Minimization):

$$\begin{aligned}&\underset{\Phi(\cdot)}{\text{minimize}}\ \mathcal{D}_{\text{MMD}}(\mathcal{DB}_n, \mathcal{DB}_{n+1})\\ &\text{subject to (15), (17).}\end{aligned}$$

Although the aforementioned MMD considers the distribution differences of CSI between $\Pr(\mathbf{H}(\mathcal{A}_m, T_n))$ and $\Pr(\mathbf{H}(\mathcal{A}_m, T_{n+1}))$, the conditional distributions of specific areas, e.g., $\Pr(\mathbf{H}(\mathcal{A}_m, T_n)|\mathcal{A}_m)$ and $\Pr(\mathbf{H}(\mathcal{A}_m, T_{n+1})|\mathcal{A}_m)$, are ignored. Since this feature provides additional correlation information in different areas, we propose to use an improved multi-kernel solution, namely *MK-MMD* [29], defined as follows:

$$\begin{aligned}&\mathcal{D}_{\text{MK-MMD}}(\mathcal{DB}_n, \mathcal{DB}_{n+1})\\ &= \sum\nolimits_{m=1}^{N_R} \lambda \left\|\Phi\left(\mathbf{H}(\mathcal{A}_m, T_n)\right) - \Phi\left(\mathbf{H}(\mathcal{A}_m, T_{n+1})\right)\right\|_{\mathcal{H}}^2\\ &\quad + \mu \left\|\Phi\left(\mathbf{H}(\mathcal{A}_m, T_n)|\mathcal{A}_m\right) - \Phi\left(\mathbf{H}(\mathcal{A}_m, T_{n+1})|\mathcal{A}_m\right)\right\|_{\mathcal{H}}^2,\end{aligned} \tag{19}$$

where $\lambda, \mu \in [0, 1]$ denote a fine-tuning coefficient indicating the data similarity between $\mathcal{DB}_n$ and $\mathcal{DB}_{n+1}$, for which it holds $\lambda + \mu = 1$. With the proposed MK-MMD metric, we define the MK-MMD minimization problem to better fit the optimal mapping function $\Phi(\cdot)$ and update the function $g_{n+1}(\cdot)$ as follows.

Problem 3 (MK-MMD Minimization):

$$\begin{aligned}&\underset{\Phi(\cdot)}{\text{minimize}}\ \mathcal{D}_{\text{MK-MMD}}(\mathcal{DB}_n, \mathcal{DB}_{n+1})\\ &\text{subject to } \lambda + \mu = 1,\\ &\qquad (15),\ (17).\end{aligned}$$

It is noted that in the conventional methods, such as JDA [28], usually the objective is to find a transformation matrix to represent the transfer mapping function $\Phi(\cdot)$. In the above MK-MMD formulation, a simple transformation matrix is insufficient, as mentioned in [29], which motivates us to apply deep neural networks for the transfer function representation, as described in the sequel.

VI. Proposed Deep Transfer Learning Scheme

In this section, we devise a deep transfer learning scheme for the fingerprint database update. In particular, we present a MK-MMD minimization framework, based on which a novel neural network structure and loss function are designed to exploit the inner relationship between the original and updated CSI measurements.

A. Data Collection and Cleaning

Network Interface Cards (NICs) like Qualcomm Atheros AR series and Intel 5300 make it possible to collect CSI data. Rather than generating synthetic data from numerical simulations, we have implemented our localization system using a TP-LINK wireless router as the AP and two Nexus 5 smartphones as the reporter and requester, respectively. The whole system works at 2.4 GHz with a bandwidth of 20MHz. Nexus 5 with Nexmon [35] software installed overhears the user datagram protocol (UDP) frames transmitted by the AP and then extracts the CSI from them. CSI knowledge is computed on a WiFi OFDM symbol basis with duration 3.2μs according to IEEE 802.11n standards. The training dataset contains 60000 transmitted packets,[5] and the packet interval is 4 ms, that is various channel situations of 4 minutes duration in the experimental environment are logged in the training dataset. Furthermore, CSI data extracted by the Nexmon CSI Tool is transformed into polar coordinates for convenient data processing, i.e. $\mathbf{h}_i(\mathcal{L}, n) = |\mathbf{h}_i(\mathcal{L}, n)|e^{j\theta_i(\mathcal{L},n)}$, where $|\mathbf{h}_i(\mathcal{L}, n)|$ and $\theta_i(\mathcal{L}, n)$ denote the corresponding amplitude and phase information, respectively and j represents the imaginary unit.

In practical systems, the measured phase information, e.g. $\hat{\theta}_i(\mathcal{L}, n)$ for subcarrier i, cannot be directly used for high accurate localization due to random jitters and noises caused by imperfect hardware components. In order to eliminate this effect, we adopt the common phase calibration algorithm proposed in [36], and then obtain:

$$\theta_i(\mathcal{L}, n) = \hat{\theta}_i(\mathcal{L}, n) + \frac{2\pi i}{N_{FFT}}\delta - Z, \tag{20}$$

where N_{FFT} denotes the size of the Fast Fourier Transform (FFT), δ means the time lag at the receiver side, and Z is the unknown random measurement noise. Nexmon CSI Tool

[5]To accelerate the model training, we install Pytorch on our server with Intel(R) Xeon(R) CPU E5-3680 and NVIDIA Tesla P100 GPU.

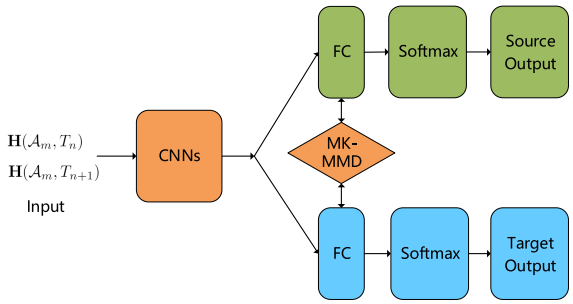


Fig. 6. The architecture of the proposed deep transfer learning network, which consists of convolution layers, FC layers, and average pooling layers. The input of the network are the channel responses at the time instants T_n and T_{n+1} with respect to the grid area $\mathcal{A}_m$, which are $\mathbf{H}(\mathcal{A}_m, T_n)$ and $\mathbf{H}(\mathcal{A}_m, T_{n+1})$, respectively. The output of the network is the estimated position likelihood distribution at T_n and T_{n+1}, which are $\mathbf{p}_n^k$ and $\mathbf{p}_{n+1}^k$, respectively.

is designed according to IEEE 802.11n protocol, and the FFT size is 64.

B. Neural Network Design

To address the mentioned Problem 3, we design a deep transfer learning network for fingerprint adaptation. As shown in Fig. 6, we start with a deep convolutional neural network (CNN), which is a common structure to fulfill the complex signal feature extraction and dimension reduction tasks. Since the convolutional layers only learns generic features in the related data sets, we only impose the MK-MMD domain adaptation in the final fully connected (FC) layer. This is because the unique characteristics of different data sets begin to appear when the structure of neural networks is deep enough [29].

Motivated by this fact, we design the neural network structure with five convolutional layers and one average pooling layer to obtain the feature vectors. An FC layer with softmax output [37] is used to provide the normalized probability. The detailed configuration and parameters of the proposed neural networks are listed in Table II, which is designed with reference to the commonly used transfer learning neural network. Meanwhile, in the neural network design, we also propose to use a joint loss measure, which considers both the *cross-entropy* measure to describe the differences between the normalized output probability $\hat{\mathbf{p}}_n^k$ and the ground true label vector $\mathbf{p}_n^k$, and the MK-MMD based loss, $\mathcal{D}_{\text{MK-MMD}}(\mathcal{DB}_n, \mathcal{DB}_{n+1})$. The parameters in the network can be adjusted by minimizing the loss function, which can be written as:

$$\mathbb{L} = -\sum_{m=1}^{N_R} \mathbf{p}_{n,m}^k \log \hat{\mathbf{p}}_{n,m}^k + \mathcal{D}_{\text{MK-MMD}}(\mathcal{DB}_n, \mathcal{DB}_{n+1}), \quad (21)$$

where $\mathbf{p}_{n,m}^k$ and $\hat{\mathbf{p}}_{n,m}^k$ are the normalized probability for the m^{th} grid area. In addition, we train the parameters of deep neural networks with Adam optimizer to minimize the above loss function in the training stage.

TABLE II
AN OVERVIEW OF THE CONSIDERED NEURAL NETWORK CONFIGURATION AND PARAMETERS

Module	Layers	Output Size
CNNs	conv1	$112 \times 112 \times 1$
	conv2_x	$56 \times 56 \times 64$
	conv3_x	$28 \times 28 \times 128$
	conv4_x	$14 \times 14 \times 256$
	conv-5_x	$7 \times 7 \times 512$
	average pooling	$1 \times 1 \times 512$
	FC	$1 \times 1 \times 512$
Output	Softmax	$N_R \times 1$

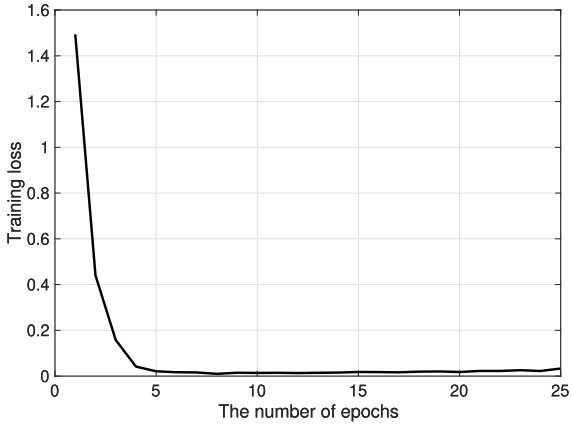


Fig. 7. The training loss gradually decreases with respect to the number of epochs. Within less than 20 epochs, the training losses will converge based on our experimental results, which demonstrates the efficiency of our proposed neural networks.

In the operating stage, users ask for their position information by reporting their real-time CSI to the server. The trained neural network with the updated parameters outputs a probability vector $\hat{\mathbf{p}}_{n+1,m}^k$, which is utilized to obtain the final estimation location $\hat{\mathcal{L}}_n^k$. The corresponding mathematical expression is given by:

$$\hat{\mathcal{L}}_{n+1}^k = \sum_{m=1}^{N_R} \hat{\mathbf{p}}_{n+1,m}^k \cdot \mathcal{L}_c^m. \quad (22)$$

C. Convergence and Complexity

To design efficient neural network architectures, another important dimension to be considered is the convergence property and the implementation complexity, where the convergence speed greatly impacts the offline training cost and the implementation complexity determines the cost of computational resources. We improve the convergence speed in the proposed method by adding early stopping points in the training stages. Specifically, the training stages will be ended at the i^{th} epoch if the associated training loss becomes larger, e.g. $\mathbb{L}^{(i)} \geq \mathbb{L}^{(i-1)}$. In Fig. 7, we plot the training loss concerning the number of epochs. We find that the training loss will converge based on our experimental results within less than 20 epochs, which demonstrates the efficiency of our proposed neural networks.

TABLE III
MEAN LOCALIZATION ERRORS OF REPORTERS FOR DIFFERENT METHODS

Solution	GPS Only	PDR Only	Proposed Solution
Mean error	3.82 m	1.35 m	0.37 m

The implementation complexity of neural networks is determined by the sizes of neural networks, which corresponds to the number of neurons inside each neural network. According to the empirical formula, as shown in [38], the maximum number of neurons N_h shall satisfy the following expression,

$$N_h = 2\sqrt{(N_{sc} \times N_s \times N_{RP} + 2) \times N_R}, \tag{23}$$

where N_{RP} is the number of reference points in training dataset. We summarize the above results in the following remark and the detailed numerical comparisons are performed in Section VII.

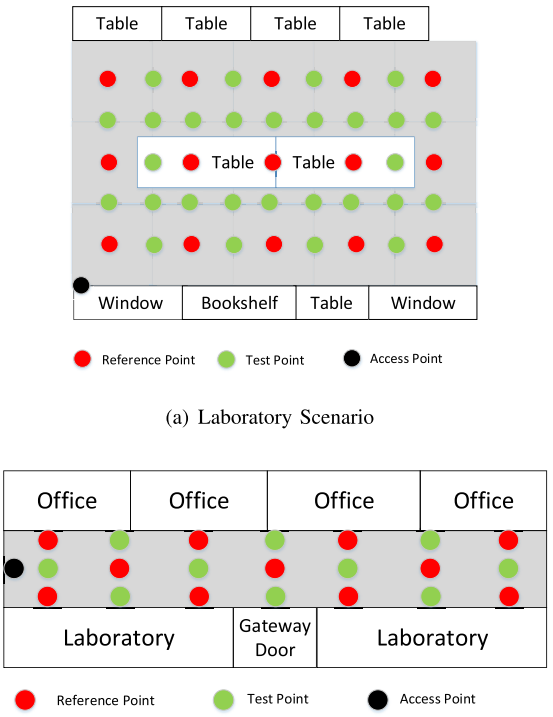
Remark 1: The implementation complexity scales with the number of neurons in the neural networks linearly, which is in the order of $\mathcal{O}(\sqrt{N_{sc} \times N_s \times N_{RP}})$.

VII. EXPERIMENT RESULTS

In this section, we provide some numerical results to show the effectiveness of the proposed deep transfer learning based approach for neural networks update in terms of effectiveness and complexity. Firstly, we have collected CSI data at all the reference points at time instant T_0 and constructed the database $\mathcal{DB}_0$, and then trained the neural network. In order to ensure the localization performance of the proposed system, we label the newly collected CSI fingerprint at time instant T_n by the method mentioned in Section IV and update the fingerprint database and neural network parameters secondly. So these data are not used in evaluating the position accuracy. Thirdly, we have collected some CSI data with known locations to verify the localization performance after fingerprint transfer. In order to demonstrate the robustness of the system, we verify the proposed scheme in both laboratory and corridor environments, whose layouts are shown in Fig. 8. With laboratory equipment, furniture, and people movements in the real situation, the tested wireless fading conditions cover most of the daily indoor scenarios with mixed LOS and NLOS paths.

A. Accuracy of Reporter Location Estimation

In this part, we investigate the accuracy of the reporter location estimation as we proposed in SectionIV. Considering even small errors in reporter location will contaminate the fingerprint database, we are required to verify the reliability of the proposed solution by comparing with GPS only and PDR only based solution in the scenario shown in Fig. 8. Table III shows the mean localization errors provided by reported by different methods. We can find that the proposed solution of reporter location estimation with more accurate initial location estimation and pedestrian dead reckoning can outperform the others, considering the weak signal of GPS indoors and the accumulation of PDR errors.



(a) Laboratory Scenario

(b) Corridor Scenario

Fig. 8. A sketch map of the experimental environment, in which the red, green, and black spots represent the location of reference points of the initial fingerprint database, test points of requesters, and that of the access point, respectively. The distance between two adjacent reference points is about 1.2m. We collect 1500 CSI samples for the training dataset and 750 CSI samples for the test dataset.

B. Effect of Fingerprint Transfer

In this experiment, we investigate the effect of fingerprint database update. We compared the proposed schemes with the following baseline approaches by measuring the cumulative distribution function (CDF) [39] of distance error in the test scenario. To be more specific, we compare the proposed schemes with two baselines, e.g., *Baseline 1*: non-updated KNN based scheme and *Baseline 2*: JDA based transfer scheme. For *Baseline 1*, we do not use the newly collected CSI data and obtain the position estimation by the original fingerprint database $\mathcal{DB}_0$. For *Baseline 2*, we use the newly collected CSI data labeled by the method described in Section IV to update the fingerprint database.

Fig. 9 describes the CDF of the localization distance error during the operating stage. The proposed regression based algorithms show superior localization accuracy over *Baseline 1* and *Baseline 2*. By comparing JDA based approach (red solid curves) and the deep transfer learning based approach (black solid curves) with a non-update KNN based scheme, we can find that the fingerprint transfer based schemes exhibit better localization performance. By comparing the JDA based and the deep transfer learning based approaches, the latter one achieves a mean error of 1.08 m for the test scenarios, which shows better localization accuracy than 1.37 m of the former one. This is due to the fact that the deep transfer learning based approach is able to utilize the complex structure of

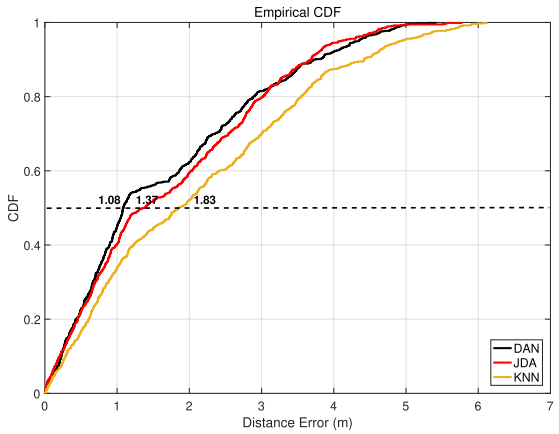


Fig. 9. CDF of the localization distance errors for different algorithms in the test scenarios. The proposed deep transfer learning based approach is compared with two baselines to test the algorithm effectiveness.

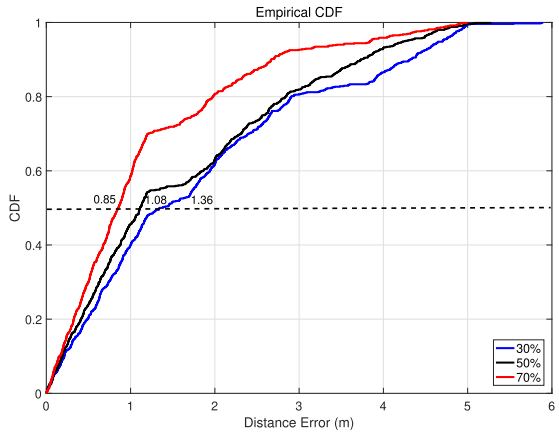


Fig. 10. CDF of the localization errors for different percentage of newly collected samples to explore the time period of fingerprint transfer.

TABLE IV
MEAN LOCALIZATION ERRORS OF REQUESTER FOR DIFFERENT METHODS

Methods	Mean Localization Error
Baseline 1	1.83 m
Baseline 2	1.37 m
MDS [23]	1.23 m
MPF [24]	1.35 m
GPR [25]	2.30 m
DAN with FC feature extractor	1.17 m
DAN with CNN feature extractor	1.08 m

the neural network to better minimize MK-MMD and build the relationship between the updated fingerprint database and locations, while the transfer ability of the transfer matrix in JDA based approach is very limited.

Moreover, the following methods are implemented and compared: MDS [23], MPF [24], GPR [25], and DAN with FC layer architecture. We summarize the comparisons of these methods in Table IV. We can find conclude that the proposed DAN with CNN feature extractor reduces the mean localization errors by 41%, 21%, 12%, 20%, 53% and 8% as compared with *Baseline 1*, *Baseline 2*, MDS, MPF, GPR and DAN with FC feature extractor, respectively. The above results demonstrate the superiority of the proposed DAN based method regardless of the feature extractor architecture for the transfer learning neural network.

C. Effect of Updated Samples

In this experiment, we investigate the test localization accuracy with different percentages of newly collected samples in the original dataset, considering that not all the CSI samples can be updated through the crowdsourcing approach. Based on this consideration, we can figure out the time period of performing a fingerprint transfer to calculate the frequency needed to update the neural network. To this end, the percentage of newly collected samples is set to be 30%, 50%, and 70%, respectively.

Localization errors under different new CSI samples percentage are illustrated in Fig. 10, where the corresponding average localization errors are 1.36 m (blue solid curves), 1.08 m (black solid curves) and 0.85 m (red solid curves), respectively. It is worth noting that when the size is changed from 30% to 70%, the localization errors reduce to half at the cost of approximately two times the data collection and labeling. Based on the above results, we find that if half of the fingerprint database can be updated by the newly collected samples, the crowdsourcing localization system can provide an accuracy of about one meter.

TABLE V
TOTAL RUNNING TIME COMPARISON FOR THE CONSIDERED METHODS

Samples Number	Baseline 1	Baseline 2	Proposed Method
1	0.01s	13.9s	0.6s
100	0.02s	15.9s	4.7s
1000	0.1s	31.6s	5.61s

D. Computational Complexity

Although in terms of localization errors, the proposed schemes provide satisfactory performance, the implementation complexity is still uncertain. Therefore, in the second experiment, we show the effectiveness of the proposed schemes by comparing the computational time cost with the considered baseline schemes.

In Table V, we compare the total running time of different schemes with different numbers of test samples on the same experimental platform. As shown in Table V, KNN based scheme (*Baseline 1*) costs the least time, regardless of the number of samples, because only the matching and classification processes are conducted, instead of the complex fingerprint transfer process. For the JDA based scheme (*Baseline 2*), most of the calculation time is used to complete the iterative process to find the suitable transfer matrix. That's why the JDA algorithm requires the largest running time regardless of the sample number. Thanks to the parameter storage capacity of the neural network architecture, the trained network helps to quickly calculate the position information of the test signal in the online phase. Compared with two baseline methods,

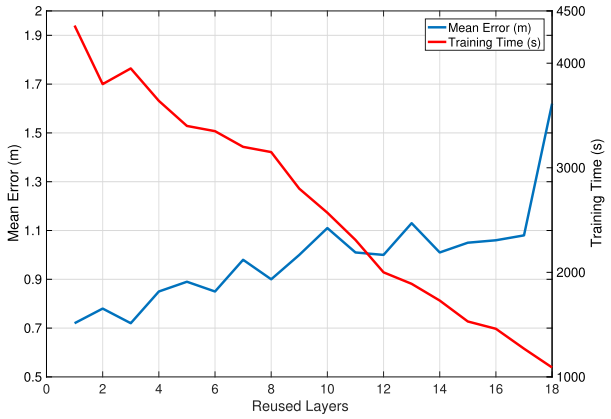


Fig. 11. The mean localization error and training time for the different reused layers of the original trained model.

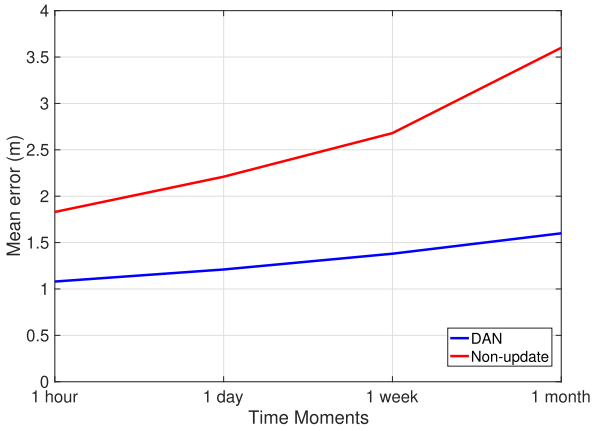


Fig. 12. The mean localization error for different fingerprint update schemes at different time moments after localization system deployed.

our proposed deep transfer learning based method provides high positioning accuracy with low computational complexity, which reduces the calculation overload of the localization system.

E. Effect of Reused Layers

We used the model trained by the original fingerprint database as the base model and reused a part of its layers during retraining. As mentioned in Table II, the proposed deep learning model has 18 layers and about 33 million parameters. The number of reused, or frozen, layers are gradually increased from only the first layer to all layers except for the 18^{th} layer. During this process, as we fed the newly collected CSI data to retrain the model, the localization accuracy and the training time of the model are shown in Fig. 11. Due to the inconsistent direction of the gradient descent every time the model is retrained, the time required for the new model to retrain, when some layers are reused has no accurate quantitative meaning. However, it can still be seen qualitatively that the turning point of the model training time appears to be the 17^{th} layer (the pooling layer), while the turning point of the model accuracy lies on the 18^{th} layer, the FC layer.

This observation indicates that, for a well-trained localization model, its layers before the flatten layer own certain common knowledge about the data and can be transferred to other models without retraining or tuning, whereas the fully connected layers' parameters cannot be utilized directly. In other words, the convolutional layers and the pooling layers play the role of feature extraction, while the fully connected layers further fit the data and perform the prediction. On the other hand, it can be seen that the majority of the training time consumption lies in the convolutional layers. The training of the fully connected layers only involves thousands of parameters and can save about 74% of the training time.

F. Effect of Operating Time

In this experiment, we test the localization accuracy performance of the deployed crowdsourcing-based localization system at different time moments to verify the system stability. We selected four different time moments: one hour, one day, one week, and one month after the system was deployed to test the positioning accuracy. It is worth noting that we assume that 50% of the fingerprint database has been updated by crowdsourcing reporters before each test. We compare the localization performance of the proposed crowdsourcing-based scheme with the non-update based scheme, and the experimental result is shown in Fig. 12.

From the given experimental results, we find that the localization accuracy of the proposed crowdsourcing system has been controlled within 1.6m during a month with a slight decline. As a comparison, the mean localization error of the non-updated based scheme has reached about 3m after one month due to the unusable fingerprint database caused by environmental changes, which is obviously unacceptable. As a result, we can conclude that the localization performance of the proposed crowdsourcing-based localization system does not degrade with time.

VIII. Conclusion

In this paper, we presented a self-calibrating crowdsourcing localization system that is based on multi-kernel deep transfer learning for efficiently exploiting the availability of frequently updated fingerprinting signals. The proposed system simultaneously updates the fingerprint database and the corresponding localization function, enabling the utilization of both historical, as well as updated fingerprint information for high precision localization. The presented indoor laboratory experimental results with the proposed system using two Nexus 5 smartphones at 2.4GHz with 20MHz bandwidth have shown that the proposed framework achieves about one meter level accuracy with relatively low computational complexity. This performance is achieved with 50% of the fingerprint update needed in the conventional transfer matrix based method.

References

[1] C. Xiang, S. Zhang, S. Xu, and G. C. Alexandropoulos, "Self-calibrating indoor localization with crowdsourcing fingerprints and transfer learning," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.

[2] B. Gozick, K. P. Subbu, R. Dantu, and T. Maeshiro, "Magnetic maps for indoor navigation," *IEEE Trans. Instrum. Meas.*, vol. 60, no. 12, pp. 3883–3891, Dec. 2011.

[3] E. Ahmed, I. Yaqoob, A. Gani, M. Imran, and M. Guizani, "Internet-of-Things-based smart environments: State of the art, taxonomy, and open research challenges," *IEEE Wireless Commun.*, vol. 23, no. 5, pp. 10–16, Oct. 2016.

[4] B. Hofmann-Wellenhof, H. Lichtenegger, and E. Wasle, *GNSS–Global Navigation Satellite Systems: GPS, GLONASS, Galileo, and More*. Cham, Switzerland: Springer, 2007.

[5] R. Faragher and R. Harle, "Location fingerprinting with Bluetooth low energy beacons," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 11, pp. 2418–2428, Nov. 2015.

[6] H. Zhang, Z. Zhang, S. Zhang, S. Xu, and S. Cao, "Fingerprint-based localization using commercial LTE signals: A field-trial study," in *Proc. IEEE 90th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2019, pp. 1–5.

[7] M. Kotaru, K. Joshi, D. Bharadia, and S. Katti, "SpotFi: Decimeter level localization using WiFi," in *Proc. ACM Conf. Special Interest Group Data Commun.*, Aug. 2015, pp. 269–282.

[8] D. Vasisht, S. Kumar, and D. Katabi, "Decimeter-level localization with a single WiFi access point," in *Proc. Usenix Conf. Netw. Syst. Des. Implement.*, 2016, pp. 165–178.

[9] C. Xiang, Z. Zhang, S. Zhang, S. Xu, S. Cao, and V. Lau, "Robust sub-meter level indoor localization—A logistic regression approach," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2019, pp. 1–6.

[10] C. Xiang *et al.*, "Robust sub-meter level indoor localization with a single WiFi access point—Regression versus classification," *IEEE Access*, vol. 7, pp. 146309–146321, 2019.

[11] Y. Wang, G. Zhou, C. Xiang, S. Zhang, and S. Xu, "Joint visual and wireless signal feature based approach for high-precision indoor localization," in *Proc. GLOBECOM IEEE Global Commun. Conf.*, Dec. 2020.

[12] A. S. Paul and E. A. Wan, "RSSI-based indoor localization and tracking using sigma-point Kalman smoothers," *IEEE J. Sel. Topics Signal Process.*, vol. 3, no. 5, pp. 860–873, Oct. 2009.

[13] B. Wang, Q. Chen, L. T. Yang, and H.-C. Chao, "Indoor smartphone localization via fingerprint crowdsourcing: Challenges and approaches," *IEEE Wireless Commun.*, vol. 23, no. 3, pp. 82–89, Jun. 2016.

[14] A. A. C. M. Kalker and J. A. Haitsma, "Fingerprint database updating method, client and server," U.S. Patent 7 523 312, Apr. 21, 2009.

[15] R. Liu, Z. Yin, W. Jiang, and T. He, "WiBeacon: Expanding BLE location-based services via WiFi," in *Proc. 27th Annu. Int. Conf. Mobile Comput. Netw.*, Sep. 2021, pp. 83–96.

[16] S. Sen, B. Radunovic, R. R. Choudhury, and T. Minka, "You are facing the Mona Lisa: Spot localization using PHY layer information," in *Proc. 10th Int. Conf. Mobile Syst., Appl., Services (MobiSys)*, 2012, pp. 183–196.

[17] Y. Chapre, A. Ignjatovic, A. Seneviratne, and S. Jha, "CSI-MIMO: An efficient Wi-Fi fingerprinting using channel state information with MIMO," *Pervasive Mobile Comput.*, vol. 23, pp. 89–103, Oct. 2015.

[18] C. Huang, G. C. Alexandropoulos, C. Yuen, and M. Debbah, "Indoor signal focusing with deep learning designed reconfigurable intelligent surfaces," in *Proc. IEEE 20th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Jul. 2019, pp. 1–5.

[19] 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, Sep. 2018.

[20] H. Zou, M. Jin, H. Jiang, L. Xie, and C. J. Spanos, "WinIPS: WiFi-based non-intrusive indoor positioning system with online radio map construction and adaptation," *IEEE Trans. Wireless Commun.*, vol. 16, no. 12, pp. 8118–8130, Dec. 2017.

[21] Y. Liu, L. Kong, and G. Chen, "Data-oriented mobile crowdsensing: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2849–2885, 3rd Quart., 2019.

[22] M. Murata, D. Ahmetovic, D. Sato, H. Takagi, K. M. Kitani, and C. Asakawa, "Smartphone-based localization for blind navigation in building-scale indoor environments," *Pervas. Mobile Comput.*, vol. 57, pp. 14–32, Jul. 2019.

[23] Y. Zhao, W.-C. Wong, T. Feng, and H. K. Garg, "Calibration-free indoor positioning using crowdsourced data and multidimensional scaling," *IEEE Trans. Wireless Commun.*, vol. 19, no. 3, pp. 1770–1785, Dec. 2019.

[24] B. Huang, Z. Xu, B. Jia, and G. Mao, "An online radio map update scheme for WiFi fingerprint-based localization," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6909–6918, Aug. 2019.

[25] S. Kumar, R. M. Hegde, and N. Trigoni, "Gaussian process regression for fingerprinting based localization," *Ad Hoc Netw.*, vol. 51, pp. 1–10, Nov. 2016.

[26] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, "A survey on deep transfer learning," in *Proc. Int. Conf. Artif. Neural Netw.* Cham, Switzerland: Springer, 2018, pp. 270–279.

[27] 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.

[28] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2200–2207.

[29] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 97–105.

[30] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. Eur. Conf. Comp. Vis.* Cham, Switzerland: Springer, 2016, pp. 443–450.

[31] B. Yang, Z. Cao, and K. B. Letaief, "Analysis of low-complexity windowed DFT-based MMSE channel estimator for OFDM systems," *IEEE Trans. Commun.*, vol. 49, no. 11, pp. 1977–1987, Nov. 2001.

[32] Z. Li *et al.*, "SoiCP: A seamless outdoor–indoor crowdsensing positioning system," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 8626–8644, Oct. 2019.

[33] F. Li, C. Zhao, G. Ding, J. Gong, C. Liu, and F. Zhao, "A reliable and accurate indoor localization method using phone inertial sensors," in *Proc. ACM UbiComp*, 2012, pp. 421–430.

[34] D. C. Montgomery, E. A. Peck, and G. G. Vining, *Introduction to Linear Regression Analysis*. Hoboken, NJ, USA: Wiley, 2021.

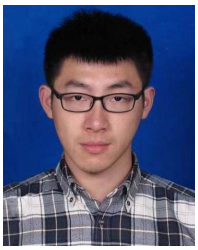
[35] F. Gringoli, M. Schulz, J. Link, and M. Hollick, "Free your CSI: A channel state information extraction platform for modern Wi-Fi chipsets," in *Proc. 13th Int. Workshop Wireless Netw. Testbeds, Experim. Eval. Characterization (WiNTECH)*, 2019, pp. 21–28.

[36] X. Wang, L. Gao, and S. Mao, "CSI phase fingerprinting for indoor localization with a deep learning approach," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1113–1123, Dec. 2016.

[37] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," 2016, *arXiv:1611.01144*.

[38] G.-B. Huang, "Learning capability and storage capacity of two-hidden-layer feedforward networks," *IEEE Trans. Neural Netw.*, vol. 14, no. 2, pp. 274–281, Mar. 2003.

[39] M. P. Deisenroth, A. A. Faisal, and C. S. Ong, *Mathematics for Machine Learning*. Cambridge, U.K.: Cambridge Univ. Press, 2020.



Chenlu Xiang (Graduate Student Member, IEEE) received the B.E. degree in communication engineering from Shanghai University, China, in 2017, where he is currently pursuing the Ph.D. degree in information and communication engineering. His research interests include indoor localization, advanced signal processing, and deep learning.

Shunqing Zhang (Senior Member, IEEE) received the B.S. degree from the Department of Microelectronics, Fudan University, Shanghai, China, in 2005, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Hong Kong University of Science and Technology, Hong Kong, in 2009.

He was with the Communication Technologies Laboratory, Huawei Technologies, as a Research Engineer and then a Senior Research Engineer from 2009 to 2014, and a Senior Research Scientist of the Intel Collaborative Research Institute on Mobile Networking and Computing, Intel Labs, from 2015 to 2017. Since 2017, he has been with the School of Communication and Information Engineering, Shanghai University, Shanghai, as a Full Professor. He has published over 60 peer-reviewed journals and conference papers, and over 50 granted patents. His current research interests include energy-efficient 5G/5G+ communication networks, hybrid computing platforms, and joint radio frequency and baseband design. He has received the National Young 1000-Talents Program and won the Paper Award for Advances in Communications from IEEE Communications Society in 2017.

Shugong Xu (Fellow, IEEE) received the Graduate degree from Wuhan University, China, in 1990, and the master's degree in pattern recognition and intelligent control and the Ph.D. degree in EE from the Huazhong University of Science and Technology (HUST), China, in 1993 and 1996, respectively. He is a Professor with Shanghai University, the Head of the Shanghai Institute for Advanced Communication and Data Science (SICS). He was the Center Director and an Intel Principal Investigator of the Intel Collaborative Research Institute for Mobile Networking and Computing (ICRI-MNC), prior to December 2016, where he joined Shanghai University. Before joining Intel in September 2013, he was the Research Director and the Principal Scientist with the Communication Technologies Laboratory, Huawei Technologies. Among his responsibilities at Huawei, he founded and directed Huawei's Green Radio Research Program, Green Radio Excellence in Architecture and Technologies (GREAT). He was also the Chief Scientist and PI for the China National 863 Project on End-to-End Energy Efficient Networks. He was one of the co-founders of the Green Touch consortium together with Bell Labs. He served as the Co-Chair of the Technical Committee for three terms in this international consortium. Prior to joining Huawei in 2008, he was with Sharp Laboratories of America as a Senior Research Scientist. Before that, he conducted research as a Research Fellow with the City College of New York, Michigan State University, and Tsinghua University. Dr. Xu has published over 100 peer-reviewed research papers in top international conferences and journals. His current research interests include wireless communication systems and machine learning. One of his most referenced papers has over 1400 Google Scholar citations, in which the findings were among the major triggers for the research and standardization of the IEEE 802.11S. He has over 20 U.S. patents granted. Some of these technologies have been adopted in international standards, including the IEEE 802.11, 3GPP LTE, and DLNA. He was awarded "National Innovation Leadership Talent" by China Government in 2013 and elevated to IEEE Fellow in 2015 for contributions to the improvement of wireless networks efficiency. He is also the winner of the 2017 Award for Advances in Communication from IEEE Communications Society.

Guoqiang Mao (Fellow, IEEE) is a Distinguished Professor and the Dean of the Research Institute of Smart Transportation, Xidian University. Before that, he was with the University of Technology Sydney and the University of Sydney. He has published over 200 papers in international conferences and journals, which have been cited more than 9000 times. His research interests include intelligent transport systems, applied graph theory and its applications in telecommunications, the Internet of Things, wireless sensor networks, wireless localization techniques, and network modeling and performance analysis. He is a Fellow of IET. He has been an Editor of the IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS since 2018, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS (2014–2019), and IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY since 2010, and received the "Top Editor" Award for outstanding contributions to the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY in 2011, 2014, and 2015. He was the Co-Chair of IEEE Intelligent Transport Systems Society Technical Committee on Communication Networks. He has served as the chair, the co-chair, and a TPC member for a number of international conferences.