Train Once, Locate Anytime for Anyone: Adversarial Learning-based Wireless Localization

DANYANG LI, JINGAO XU, and ZHENG YANG, School of Software and BNRist, Tsinghua University, China
CHENGPEI TANG, School of Intelligent Systems Engineering, Sun Yat-sen University, China

Among numerous indoor localization systems, WiFi fingerprint-based localization has been one of the most attractive solutions, which is known to be free of extra infrastructure and specialized hardware. To push forward this approach for wide deployment, three crucial goals on high deployment ubiquity, high localization accuracy, and low maintenance cost are desirable. However, due to severe challenges about signal variation, device heterogeneity, and database degradation root in environmental dynamics, pioneer works usually make a trade-off among them. In this article, we propose iToLoc, a deep learning-based localization system that achieves all three goals simultaneously. Once trained, iToLoc will provide accurate localization service for everyone using different devices and under diverse network conditions, and automatically update itself to maintain reliable performance anytime. iToLoc is purely based on WiFi fingerprints without relying on specific infrastructures. The core components of iToLoc are a domain adversarial neural network and a co-training-based semi-supervised learning framework. Extensive experiments across 7 months with eight different devices demonstrate that iToLoc achieves remarkable performance with an accuracy of 1.92 m and >95% localization success rate. Even 7 months after the original fingerprint database was established, the rate still maintains >90%, which significantly outperforms previous works.

37

CCS Concepts: • **Human-centered computing → Ubiquitous and mobile computing**; • **Networks → Location-based services;**

Additional Key Words and Phrases: Indoor localization, RSS fingerprint, adversarial learning, fingerprint database update

ACM Reference format:
Danyang Li, Jingao Xu, Zheng Yang, and Chengpei Tang. 2024. Train Once, Locate Anytime for Anyone: Adversarial Learning-based Wireless Localization. *ACM Trans. Sensor Netw.* 20, 2, Article 37 (January 2024), 21 pages.
https://doi.org/10.1145/3614095

A preliminary version [26] of this article appeared in International Conference on Computer Communications (IEEE INFOCOM 2021).
This work is supported in part by the National Key Research Plan under Grant No. 2021YFB2900100 and the NSFC under Grants No. 61832010, No. 62202262, and No. 61972131.
Authors' addresses: D. Li (Co-primary author), J. Xu (Co-primary author), and Z. Yang (Corresponding author), School of Software and BNRist, Tsinghua University, China; e-mails: {lidanyang1919, xujingao13, hmilyyz}@gmail.com; C. Tang, School of Intelligent Systems Engineering, Sun Yat-sen University, China; e-mail: tchengp@mail.sysu.edu.cn.

1550-4859/2024/01-ART37 $15.00
https://doi.org/10.1145/3614095

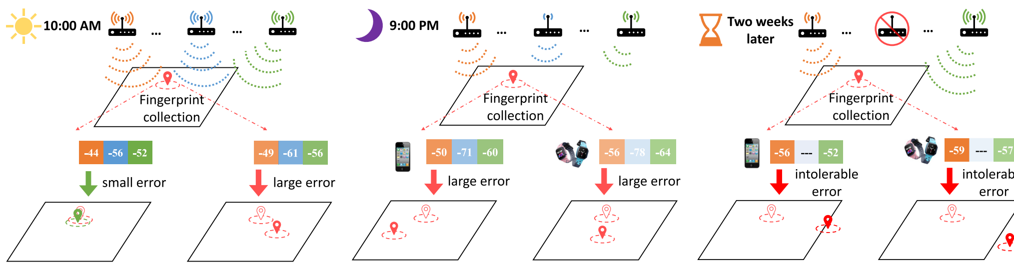
1 INTRODUCTION

Accurate and stable **Indoor Location-based Service (ILBS)** is a key enabler for many ubiquitous applications. To provide ILBS, various wireless indoor localization techniques have been proposed in the past decade, including WiFi [12, 48, 52, 53, 62], RFID [39, 45], acoustic signals [30], and visual images [13, 25, 55, 57]. Among them, due to the wide deployment and availability of WiFi infrastructure, WiFi **Received Signal Strength (RSS)** fingerprint-based indoor localization has become one of the most attractive solutions [37, 40, 49, 59]. This approach generally has a training stage, in which RSS fingerprints with location labels are collected by manual site survey or leveraging crowdsourcing scheme [37, 59] to automatically form a fingerprint database (a.k.a. radio map). Then, users are located by matching their fingerprint observation against the fingerprint database. Such a method has attracted attention from both academic and industrial communities. For instance, Microsoft hosts indoor localization competitions based on RSS fingerprints [31]; XiaoTianCai and HUAWEI develop smartwatches integrated with such modules to locate and protect children [21]; Baidu and Google deploy about 4,000 buildings to provide fingerprint-based ILBS.

Despite extensive research, RSS fingerprint-based indoor localization frequently yields large localization errors and has not yet stepped in the prime time for wide deployment. We evaluated the performance of the RSS fingerprint-based localization system in real business environments across 7 months and finally found that the primary hurdles are *signal variation*, *device heterogeneity*, and *database deterioration*, as illustrated in Figure 1. On the one hand, diverse RSS will be encountered at different times by different devices, which will lead to query fingerprint's mismatch against the database. Typically, location errors sometimes increase up to 10 m. On the other hand, considering severe RSS variations and environmental changes, an initial fingerprint database may gradually deteriorate, leading to grossly inaccurate location estimations. The fingerprint database may need to be periodically calibrated, even reconstructed, which induces expensive maintenance costs.

Recent efforts attempt to overcome the above challenges by leveraging: (1) robust fingerprint constructor and learning-based classifier: these works generate robust forms based on multiple RSS fingerprints as new representations for each location [48, 58], and further employ deep neural networks for classification [1]. And (2) additional information: inertial sensing [50], image matching [54], and even physical layer **Channel State Information (CSI)** [10] have been recently incorporated for improving performance. Additionally, some recent approaches also leverage the extra geometry constraints or user motion patterns provided by the above techniques to occasionally update the fingerprint database [51], aiming to ease the maintenance cost.

Albeit inspiring, previous works make a sacrifice to overcome the above challenges and thus face severe limitations. First, the *localization accuracy and cross-device robustness remain low* in practice because of severe environment dynamics and device heterogeneity [56]. Second, we find a theoretical gap between reliable localization and radio-map updates. A so-called chicken-egg problem can explain it: reliable radio-map update depends on accurate localization of unlabeled fingerprints. However, the localization performance is exactly influenced by the quality of pre-updated radio-map [51, 59]. As a consequence, *maintenance overhead has not been obviously reduced*, and we still need to recollect the radio map frequently. Last but not least, although some works may achieve enhanced accuracy, they also *degrade the deployment ubiquity*. For example, a user needs to placidly hold smartphones horizontally or vertically for precisely collecting inertial sensor data and images. However, this requirement is impractical [54, 60]. It is also required to deploy extra infrastructures, including surveillance cameras, ultrasonic beacons, or specific WiFi devices to collect CSI. Figure 2 illustrates qualitative comparison of state-of-the-art works. None of the previous localization systems can simultaneously solve the above issues and achieve three goals: high localization accuracy, low maintenance cost, and high deployment ubiquity.



(a) Localization at 10:00 AM (b) Localization at 9:00 PM (c) Localization after two weeks

Fig. 1. Illustration of three key reasons that lead to frequent large localization bias: *signal variation*, *device heterogeneity*, and *database deterioration*. Panels (a) and (b) show that RSS values are vulnerable to environment dynamics. In addition, automatic power adjustment strategy implemented in modern APs exaggerates the RSS temporal fluctuations; due to inherent hardware characteristics, different devices may encounter diverse RSSs of a specific Access Point (AP) under even identical wireless conditions. (c) Because of environmental changes, an initial fingerprint database may gradually deteriorate, leading to inaccurate location estimations.

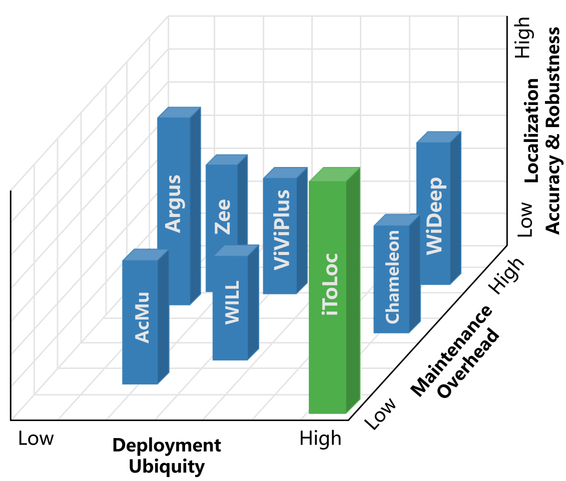


Fig. 2. Comparison of state-of-the-art works.

In this work, we aim to achieve all the above three goals and propose iToLoc, a fine-grained deep learning-based **i**ndoor localization system that is able to *Train once, update automatically, Locate anytime for anyone*. Specifically, once trained, iToLoc will extract device-independent and environmental dynamics-resistant features for accurate and robust localization. In addition, iToLoc will automatically update itself, keep high localization accuracy, and ease maintenance costs for the long-term. Our design and implementation of iToLoc excel in three unique aspects:

First, to achieve accurate and robust localization performance, we design a deep learning-based framework that can *remove the influence of signal variation and device heterogeneity* contained in collected RSS fingerprints and *extract device-independent and dynamics-resistant* features. The core of the framework is a **Domain Adversarial Neural Network (DANN)** [15, 23], which consists of four main components: fingerprint-image transformer, feature extractor, location predictor, and domain discriminator. The feature extractor, which is a **Convolutional Neural Network (CNN)**, cooperates with the enhanced location predictor to carry out the major task of localization, and simultaneously, tries to fool the domain discriminator to learn the dynamics-resistant and device-independent representations.

Second, to ease the system maintenance cost, we design a reliable model update framework, unlike the recent works mentioned above. Our key insight is that query fingerprints can be seen as unlabeled data; hence, we can treat radio-map adaption as model fine-tune on this unlabeled dataset, which is a classical problem in the scope of *semi-supervised learning*. In this work, we adopt the concept of *co-training* [14, 64] to fill the gap between robust localization and reliable model update. Specifically, the location predictor in the DANN mentioned above is consists of three modules with diverse network structures to enhance the diversity of classification view. Upon receiving unlabeled fingerprints, three modules will co-determine the localization result and co-refine the network according to the confidence.

Third, to ensure the system ubiquity for wide deployment, iToLoc is purely based on RSS fingerprints without inducing additional costs or introducing extra information.

We have fully implemented iToLoc on six different types of commodity smartphones and two types of smartwatches. Comprehensive experiments are carried out in three buildings with various conditions. We deploy iToLoc in real business environments, and continuously evaluate the system performance across 7 months. The results demonstrate that iToLoc achieves reliable performance with an average accuracy of 1.92 m and a 95th percentile accuracy of 3.4 m, outperforming even the best among four comparative approaches by >30%. The localization success rate of iToLoc is >95%. Even 7 months after the fingerprint database is established, the localization success rate still maintains >90%, outperforming other works by more than 20%. Achieving truly high precision, low maintenance cost, and high deployment ubiquity, iToLoc takes an essential step toward practical indoor localization for mobile users.

The core contributions are summarized as follows:

(1) We design a novel adversarial network-based localization framework. Based on the in-depth understanding of RSS fingerprints and efficient design of the network model, the proposed framework is able to extract device-independent and dynamics-resistant feature for robust localization.
(2) We provide a fresh perspective to solve the radio-map automatic adaption problem based on semi-supervised learning, which requires no additional hardware or extra user intervention. Compared with existing methods, we first fill the gap between robust localization and reliable model update.
(3) We prototype iToLoc on eight different types of devices (including two smartwatches) in real environments for 7 months. Encouraging results demonstrate that iToLoc makes a great progress toward fortifying WiFi fingerprint-based localization to an entirely practical service for wide deployment.

The rest of article is organized as follows. We present an overview in Section 2 and introduce *adversarial learning-based robust localization* in Section 3. *Co-training-based reliable model update* is provided in Section 4, followed by implementation experiments in Section 5. We review related works in Section 6 and conclude this article in Section 7.

2 SYSTEM OVERVIEW

As illustrated in Figure 3, the design of iToLoc follows the classical fingerprint framework, with no more inputs than any existing fingerprint-based systems. Benefiting from this, we retain the elegant ubiquity of WiFi fingerprinting. iToLoc contains two unique modules: *adversarial learning-based robust localization* and *co-training-based reliable model update*. The adversarial network

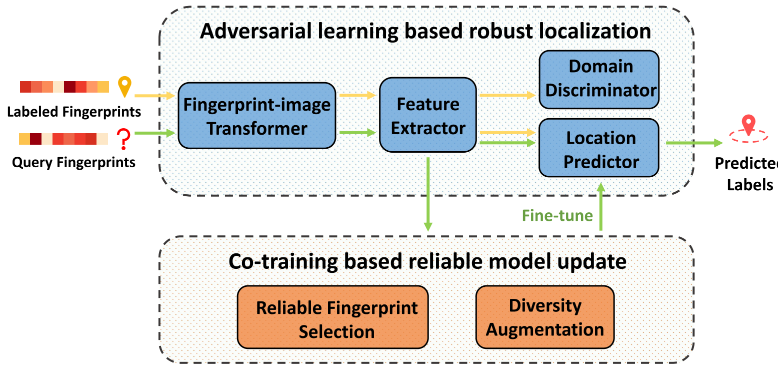


Fig. 3. System overview.

consists of four components: *fingerprint-image transformer*, *feature extractor*, *domain discriminator*, and *location predictor* (detailed design of the adversarial network with four components will be described in Section 3). Same as traditional fingerprint-based systems, in the offline training stage, the collected fingerprint database is leveraged to train the network. Afterward, the adversarial network is able to extract robust *device-independent and dynamics-resistant* features based on original RSS fingerprints, which improve the localization performance.

Upon receiving query fingerprints, *location predictor* will finally calculate the localization results. In the meantime, the *co-training-based reliable model update* module will also be triggered to fine-tune the pre-trained model leveraging these unlabeled fingerprints. The design of this module is followed by the concept of semi-supervised learning (detailed description in Section 4). Based on this strategy, iToLoc will automatically and reliably update the model. The updated model, which has been adapted to the environmental changes, is then used for online localization for further location queries.

3 ADVERSARIAL LEARNING-BASED ROBUST LOCALIZATION

An overview of the proposed adversarial learning model is shown in Figure 4. As aforementioned, the goal of the proposed model is to extract *device-independent and dynamics-resistant* representations based on original RSS fingerprints.

Toward this goal, the 1D RSS fingerprint vectors are first transformed into 2D fingerprint images as the input data format by the component of **fingerprint-image transformer** for better expressive ability, which will promote following components to extract more robust and stable features. Afterward, the input images are transformed into latent representations by the component of **feature extractor**. Using the learned feature representations, the **location predictor** is leveraged to maximize the localization accuracy and obtain the location predictions. To remove domain specific features, a **domain discriminator** is designed to label each domain (specifically, to identify *when the fingerprints are collected* by *what type of devices*). The goal of domain discriminator is to maximize the domain labeling accuracy, which seemingly contradicts with our ultimate goal of extracting domain-independent features. However, in our design, the feature extractor tries its best to cheat the domain discriminator, and at the same time, boost the accuracy of the localization results, which is termed as a minimax game [63]. Through the game, the feature extractor can finally learn the common domain-independent features for all fingerprints. Besides, we design a spatial constraint that can significantly reduce the appearance of large localization errors. The details of our model will be elaborated in the rest of this section.

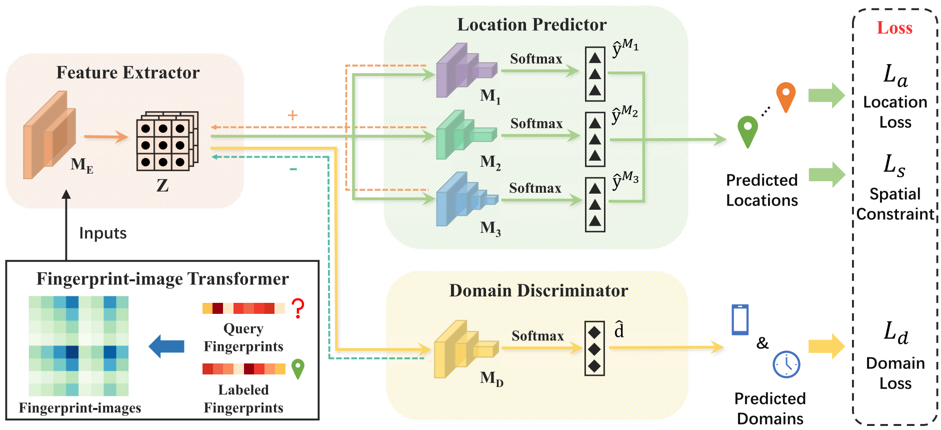


Fig. 4. Overview of domain adversarial learning-based model.

3.1 Fingerprint-image Transformer

Suppose that each user receives the RSS values from neighboring N APs, namely, the data $\boldsymbol{f} = \{f_1, f_2, \ldots, f_N\}$ is a 1D fingerprint vector of length N where f_i denotes the RSS value obtained from the AP_i. It is straightforward that we can directly take the above 1D fingerprint vector as the input, just like recent CNN-based localization systems [22, 42]. Nevertheless, this approach may not yield optimal performance for the adversarial network. The main reason is that the *feature extractor* in mainstream domain adversarial networks is composed of only a few layers (typically no more than three layers) for a balance minimax game against *domain discriminator* without overfitting [23, 63]. Further, the convolution kernel in CNN is leveraged based on the assumption that adjacent elements enjoy a certain spatial relationship (i.e., in an image, one pixel's color is relevant to its adjacent pixels) [16, 24]. To this end, the original 1D fingerprint vector, which is composed of irrelevant RSS values, has limited information expressive ability as network input and can not promote the adversarial network to extract domain-independent features.

To solve this drawback, we design a fingerprint-image transformer to enhance the expressive ability of original RSS fingerprints and fulfill the characteristics of adversarial networks. Specifically, as illustrate in Figure 5, we first consider the **Log-distance Path Loss (LDPL)** model, which describes the relationship between the RSS f and the physical distance d between transceivers:

$$f = R - \eta 10 \lg d, \tag{1}$$

where R and η are predefined reference RSS received at a distance of 1 m and path loss exponent, respectively. According to Equation (1), the distance d can be calculated as follows:

$$d = 10^{\frac{R-f}{10\eta}}, \tag{2}$$

here, the d cannot fully represent the physical distance between the sampling point and AP location, but it reflects the relative signal propagation distance, which considering the signal occlusion and multipath effects. We transform the RSS f_i in $\boldsymbol{f}$ to the distance d_i according to Equation (2), and obtain the distance vector $\boldsymbol{d} = \{d_1, d_2, \ldots, d_N\}$.

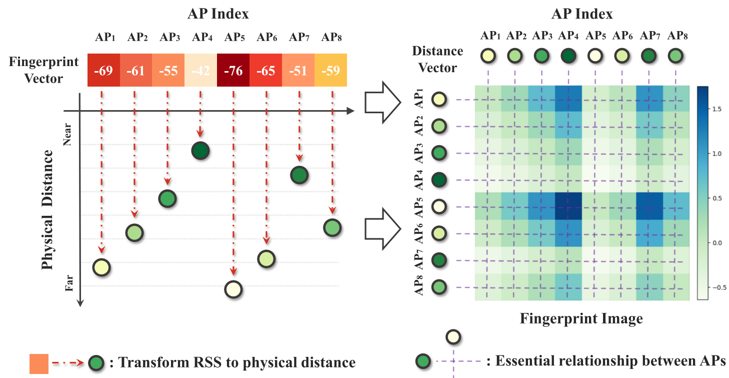


Fig. 5. Fingerprint-image transformer: The left panel illustrates the transformation of the RSS vector f into the physical distance vector d using the LDPL model. The right panel depicts the subsequent conversion of the distance vector into a fingerprint image $\mathbf{x}$. The elements in $\mathbf{x}$ range from -1 to positive infinity, with the main diagonal elements inherently being zero.

Afterwards, we leverage the fingerprint image $\mathbf{x}$ (a 2D $N \times N$ matrix) as the input of the adversarial network:

$$\mathbf{x} = \left\{ \begin{array}{cccc} f^*_{1,1} & f^*_{1,2} & \cdots & f^*_{1,N} \\ f^*_{2,1} & f^*_{2,2} & \cdots & f^*_{2,N} \\ \cdots & \cdots & \cdots & \cdots \\ f^*_{N,1} & f^*_{N,2} & \cdots & f^*_{N,N} \end{array} \right\}, \tag{3}$$

where $f^*_{j,k} = (d_j - d_k) * \frac{1}{d_k}$. Through the above steps, the fingerprint image $\mathbf{x}$ contains the basic relationships between any two APs. Compared to the one-dimensional fingerprint vector, each element in $\mathbf{x}$ is related to its adjacent elements. Additionally, all the ratios of the RSS in the original fingerprint can be recalculated from $\mathbf{x}$. Effectiveness of the proposed fingerprint-image transformer will be evaluated in Section 5.3.

We let the fingerprint image $\mathbf{x}$ be the input data of the proposed model. Each $\mathbf{x}_i$ has a corresponding domain label $\mathbf{d}_i \in \mathcal{D}$, where $\mathcal{D}$ denotes the set of all the domains. Here, we refer to the domains with and without label information as the source and target domain. Each labeled fingerprint $\mathbf{x}_i$ also has a location label $\mathbf{y}_i \in \mathcal{Y}$, where $\mathcal{Y}$ is the set of all the location in the area of interests. Let $\mathbf{d}$ denotes the domain label vector of $\mathbf{x}$, and $\mathbf{y}$ be the ground truth vector of $\mathbf{x}$. Thus, the inputs of our model are the 2D fingerprint image $\mathbf{x}$, the domain label vector $\mathbf{d}$, and the ground truth location $\mathbf{y}$.

3.2 Feature Extractor

In this work, a two-layer CNN module denoted as M_E is employed as the feature extractor to extract latent features. Let Θ_{M_E} denote the set of parameters associated with M_E. Given input data $\mathbf{x}_i$, the corresponding latent feature Z_i is obtained as follows:

$$Z_i = M_E(\mathbf{x}_i; \Theta_{M_E}). \tag{4}$$

3.3 Location Predictor

We design three different CNN modules M_1, M_2, M_3 and integrate them into a location predictor. These three modules are used to learn three different representations $\boldsymbol{V}_i^{M_1}$, $\boldsymbol{V}_i^{M_2}$, and $\boldsymbol{V}_i^{M_3}$ of $\mathbf{x}_i$

based on $\boldsymbol{Z}_i$:

$$\boldsymbol{V}_i^{M_k} = M_k(\boldsymbol{Z}_i; \Theta_{M_k}); k = 1, 2, 3, \tag{5}$$

where Θ_{M_k} are the parameters of M_k to be learned. To predict the labels, we map the feature representation $\boldsymbol{V}_i$ to the latent space $\mathbb{R}^C$, where C is the number of locations in the area of interests. Moreover, a softmax layer is used to obtain the probability vector as follows:

$$\hat{\mathbf{y}}_i^{M_k} = \text{Softmax}\left(\boldsymbol{W}_v^{M_k}\boldsymbol{V}_i^{M_k} + \boldsymbol{b}_v^{M_k}\right); k = 1, 2, 3, \tag{6}$$

where $\boldsymbol{W}_v^{M_k}$ and $\boldsymbol{b}_v^{M_k}$ are the parameters. $\hat{\mathbf{y}}_i^{M_k}$ denotes the predicted probabilities of labeled data by module M_k. The loss of the location predictor, L_a, is defined as the cross-entropy as follows:

$$L_a = -\frac{1}{|\mathbf{X}|}\sum_{i=1}^{|\mathbf{X}|}\sum_{k=1}^{3}\sum_{c=1}^{C}\mathbf{y}_{ic}\log\left(\hat{\mathbf{y}}_{ic}^{M_k}\right), \tag{7}$$

where $|\mathbf{X}|$ is the number of fingerprints in the training set. During the training, the feature extractor and location predictor play a cooperative game to minimize the L_a.

3.4 Domain Discriminator

In our adversarial network, the domain is defined as *a pair of device and time*. The rationale behind making this definition are twofold: On the one hand, mobile devices with diverse types of wireless cards have distinct capacities of sensing Wi-Fi signals, some are especially sensitive but some are not. Taking *device* variety into consideration will help adversarial network learn device-independent representations. On the other hand, RSS values are very noisy and fluctuant due to multi-path and shadow fading effects. Superadded with the affection to signal propagation caused by temperature, humidity, and movement of people, the distributions of RSS values of different time periods vary a lot. Given the difficulty to accurately profile environmental dynamics or RSS variance, it is of great help to treat *time* as another attribute to enable the adversarial network to extract dynamics-resistant features [43].

For the domain discriminator to identify the domain labels of the input fingerprints, we design a CNN module M_D to learn the representation $\boldsymbol{U}_i$ of $\mathbf{x}_i$ based on $\boldsymbol{Z}_i$:

$$\boldsymbol{U}_i = M_D(\boldsymbol{Z}_i; \Theta_{M_D}), \tag{8}$$

and map $\boldsymbol{U}_i$ into domain distribution $\hat{\mathbf{d}}_i$:

$$\hat{\mathbf{d}}_i = \text{Softmax}(\mathbf{W}_u\mathbf{U}_i + \mathbf{b}_u). \tag{9}$$

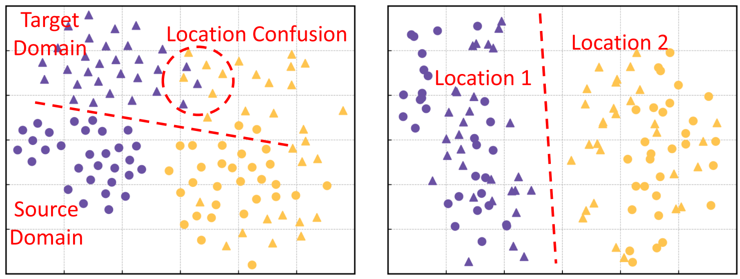
We define the loss as the cross-entropy between the domain distribution and true domain labels:

$$L_d = -\frac{1}{|\mathbf{X}|}\sum_{i=1}^{|\mathbf{X}|}\sum_{j=1}^{|\mathcal{D}|}\mathbf{d}_{ij}\log(\hat{\mathbf{d}}_{ij}), \tag{10}$$

where $\mathbf{d}_i$ is the one-hot vector of true domain labels. The goal of the domain discriminator is to minimize L_d to maximize the performance of domain prediction, which contradicts with our ultimate goal of learning domain-independent features. To address this contradiction, we maximize the domain discriminator loss L_d in final objective function. Based on Equations (7) and (10), we can obtain the loss function as follows:

$$L = L_a - \lambda L_d \ (\lambda > 0). \tag{11}$$

Through this minimax game, we can learn the common domain-independent features for all the fingerprints.



(a) (b)

Fig. 6. Learned representations by different frameworks: (a) In baseline CNN framework, the features are separated according to different domains, however mixed under different locations, which will lead to localization bias. (b) In iToLoc, features are clustered corresponding to different locations, and mixed under different domains, which will increase localization robustness.

3.5 Spatial Constraint

We design a spatial constraint on the estimated location $\hat{\mathbf{y}}_i$, which penalizes $\hat{\mathbf{y}}_i$ when it is inconsistent with and far away from the ground truth $\mathbf{y}_i$. The loss of the spatial constraint is defined as follows:

$$L_s = \frac{1}{|\mathbf{X}|}\sum_{i=1}^{|\mathbf{X}|}\sum_{c=1}^{C} w_{y_i c}\hat{\mathbf{y}}_{ic}, \tag{12}$$

where $\hat{\mathbf{y}}_{ic}$ denotes the predicted probability of the fingerprint at the cth location, while $w_{y_i c}$ signifies a normalized weight that encapsulates the physical distance between the cth location and the ground truth. As the cth location approaches the ground truth, the corresponding weight $w_{y_i c}$ diminishes toward zero. This weight is normalized by the maximum distance within the space, ensuring that all weights fall within $[0, 1]$. Such normalization preserves the integrity of relative distances and mitigates the potential adverse impact of excessively unbalanced weight values on the training process. In most existing localization applications, the physical coordinates of each sampling location are carefully recorded during the fingerprints collection stage, therefore $w_{y_i c}$ can be obtained without any additional effort.

3.6 Objective and Training

With the spatial constraint, we can finally give the overall loss function as follows:

$$L = L_a + \gamma L_s - \lambda L_d \ (\gamma, \lambda > 0). \tag{13}$$

Following the training process of DANN [15], we iteratively update the parameters. Let $\Omega = \{\Delta, \Gamma\}$ be the set of all the parameters, where Δ denotes the parameters in the domain discriminator. We first fix Δ and update the remaining parameters $\Gamma = \Omega - \Delta$, and then fix Γ to update Δ.

The ultimate goal of the proposed adversarial learning-based framework is to learn device-independent and dynamics-resistant representations of locations. To qualitatively evaluate the learned representations, we conduct the following experiment. We first select fingerprints collected at two different locations under two different domains (when to collect and by which device), i.e., four location and domain pairs. Then, we randomly select 30 fingerprint samples for each pair and extract features according to Equation (4) by baseline CNN frameworks and iToLoc, respectively. Finally, we plot the learned representations of these samples on a 2D space with a manifold learning algorithm t-SNE [32]. In Figure 6, we use orange and purple colors to represent different locations where the fingerprints are collected, circle and triangle markers to represent

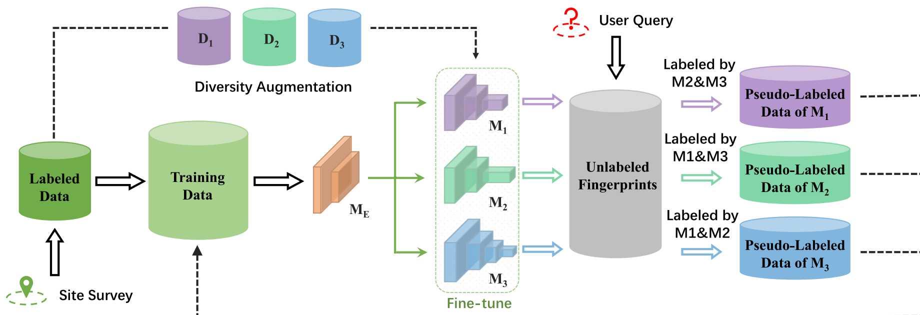


Fig. 7. Training process of model update.

source domain (time and device labels are known) and target domain (unknown time and device), respectively. As shown in Figure 6(a), the features learned from the baseline method are clustered according to different domains. What's worse, in the target domain (triangle markers), the features corresponding to different locations are mixed together, which will lead to various localization bias. In contrast, as shown in Figure 6(b), features extracted by iToLoc can form two clearly separate clusters, where each cluster corresponds to the different locations. Moreover, we can observe that within each location cluster, samples from different domains have almost the same distribution. This result demonstrates the effectiveness of the proposed domain adversarial training to learn *device-independent and dynamics-resistant* localization features.

4 CO-TRAINING-BASED RELIABLE MODEL UPDATE

The training process of model update is shown in Figure 7. M_E denotes the well-trained feature extractor, M_1, M_2, and M_3 are the three modules in the location predictor. During the model update, a part of unlabeled fingerprints will be labeled and added to the training set. With three different modules M_1, M_2, and M_3, if two modules agree on the prediction of unlabeled fingerprint and the prediction is confident and stable, this reliable fingerprint with the pseudo-label predicted by the two modules is added into the training sets of the third module. The third module is refined with the augmented training set. The details of fingerprint selection will be presented in Section 4.1. However, the three modules will be more similar, since they augment the training sets of one another. To tackle this problem, we fine-tune the modules by smearing labeled data to augment the diversity among them in some specific rounds. The details of diversity augmentation will be presented in Section 4.2. With the above techniques for the model update, iToLoc can adapt to the unpredictable environmental dynamic and provide long-term services.

4.1 Reliable Fingerprint Selection

It is possible that the pseudo-labels assigned to newly labeled fingerprints may be inaccurate, and these inaccuracies can have a negative impact on performance. Therefore, we must carefully select confident and stable fingerprints. Here, a confident prediction is characterized by the condition where the average maximum posterior probability of the two modules surpasses a threshold σ. A larger σ value demands a higher degree of confidence for the inclusion of new fingerprints. Here, confident prediction means that the average maximum posterior probability of the two modules is larger than a threshold σ. Stable prediction means that the pseudo-label should not change much when the modules predict the fingerprint repeatedly. We leverage a *data-editing* method [14] with dropout to determine the stability of the selected fingerprints. Generally, dropout works in two modes: at training mode, the connections of the network are different in every forward pass; at test mode, the connections are fixed. This means that the prediction for dropout working in training mode may change. For each $(\mathbf{x}_i, \bar{\mathbf{y}}_i)$, $\bar{\mathbf{y}}_i$ is the pseudo-label predicted by the modules working in test mode. And we use dropout working in train mode to measure the stability of the pseudo-labeled data, i.e., we use the modules to predict the label of $\mathbf{x}_i$ for K times in training mode and record the frequency k that the prediction is different from $\bar{\mathbf{y}}_i$. If $k > \frac{K}{3}$, then we regard the pseudo-label $\bar{\mathbf{y}}_i$ of $\mathbf{x}_i$ as an unstable pseudo-label. For these unstable pseudo labels, we will eliminate them.

4.2 Diversity Augmentation

Diversity among the three modules plays a vital role in the training process of model update. Although we use different network structures to enhance the diversity of classification views, when three modules label unlabeled data to augment the training sets of one another, they become more and more similar. To maintain the diversity, we use *output smearing* [8] to generate three different labeled data sets (i.e., D_1, D_2, and D_3) from the labeled training set D. Here, output smearing constructs diverse training sets by injecting random noise into true labels and generating modules from the diverse training sets. We apply this technique to fine-tune our modules M_1, M_2 and M_3. For example $(\mathbf{x}_i, \mathbf{y}_i)$, where $\mathbf{y}_i = (\mathbf{y}_{i1}, \mathbf{y}_{i2}, \ldots, \mathbf{y}_{ic})$, $\mathbf{y}_{ic} = 1$ if the example belongs to the cth class otherwise $\mathbf{y}_{ic} = 0$. In output smearing, we add noise into every component of $\mathbf{y}_i$:

$$\mathbf{y}'_{ic} = \mathbf{y}_{ic} + \mathrm{ReLU}\,(z_{ic} \times std)\,, \tag{14}$$

where z_{ic} is independently sampled from the standard normal distribution. We use std to scale the amplitude of z_{ic}, which represents the standard deviation of the noise. The ReLU function is applied to ensure that the output values $\mathbf{y}'_{ic}$ are non-negative. Then, we normalize $\mathbf{y}'_{ic}$ as follows:

$$\mathbf{y}'_i = (\mathbf{y}'_{i1}, \mathbf{y}'_{i2}, \ldots, \mathbf{y}'_{ic}) \,/\, \sum_{c=1}^{C} \mathbf{y}'_{ic}. \tag{15}$$

With output smearing, we construct three diverse training sets D_1, D_2, and D_3. Then, we fine-tune three modules M_1, M_2, and M_3 on the diverse training set in specific rounds.

4.3 Rationale Behind Reliable Model Update

In this subsection, we explain the rationale behind the proposed strategy for the reliable model update. First, the fundamental of the strategy is the *disagreement-based semi-supervised learning* whose basic idea is to train multiple learners for the same task but exploit the disagreements during the learning process. Many theoretical studies have explained why unlabeled data can improve learning performance and efficiently fine-tune models based on such strategy [5–7, 46]. In our proposed co-training-based model update, the disagreement is exactly reflected in different views: Three modules are constructed from different network structures, and the diversity augmentation method is leveraged to enhance diversity among different training data for each module. We plot the learned views of three modules (M_1, M_2, and M_3). As shown in Figure 8, the disagreement generated by different views on the fingerprints guides the model to update the classification boundaries effectively. Second, unlike previous works that the original unlabeled fingerprints are directly leveraged to update radio-map based on localization results, iToLoc first removes the device-associated and dynamics-influenced noises contained in unlabeled fingerprints and focuses on the essential information to simultaneously predict locations and update the model. Compared with related works, our model update strategy is more reliable and can further ease the system maintenance for the long term.

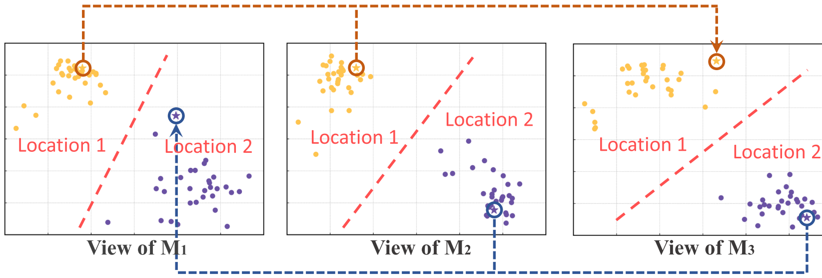
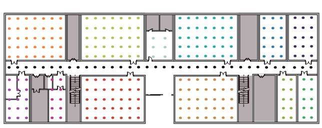
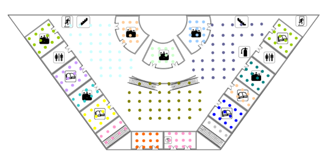


Fig. 8. The orange and purple colors represent different locations. The star markers in different views represent latent features of the same input fingerprint. The latent features of location 1 marked in the M_1 and M_2 views are at the center of the cluster, while the latent feature of the same input in the view of M_3 are away from the cluster. Similarly, the disagreement between the different views is also reflected in the samples of location 2.



(a) Office building



(b) Classroom building

(c) Shopping mall

Fig. 9. Experimental areas.

5 IMPLEMENTATION AND EVALUATION

5.1 Experimental Methodology

We prototype iToLoc on the popular Android OS and conduct experiments using 8 different devices over various scenarios. Furthermore, we deploy iToLoc in real business environments, and continuously evaluate the system performance across 7 months. In this section, we first introduce the experimental settings and then present the detailed evaluation.

5.1.1 Experimental Scenarios and Datasets. We carry out experiments in three different buildings with various floor layouts and diverse wireless environments. Figure 9 shows the floor plans of experimental areas. The wireless conditions and environmental dynamism are pretty different in the three buildings. Office building enjoys the most stable wireless environment; the classroom building is crowded or empty to different extents depending on the course schedule, which influence the wireless condition. As for the shopping mall, both the wireless condition and environmental layout change frequently, which may lead to short life cycles of the collected fingerprint database.

The data collection details are summarized in Table 1. The training samples are collected once at the beginning, while test samples are collected multiple times at intervals. When collecting fingerprints for training, we employ a typical sampling frequency of around 1Hz. We employ 8 phones of 6 different models that are manufactured by different companies for data collection, including two HUAWEI P10, one Lenovo Phab2 pro, two Google Nexus 6p, one Google Nexus 7, one Millet 6 and one Millet 9, which are equipped with different WiFi chips. To further evaluate

Table 1. Data Collection in Different Scenarios

#	Building type	Size(m^2)	Density	Devices	Region	Samples	Duration
1	Office	600	1m × 1m	HUAWEI P10 * 2, Phab2, Nexus 6p * 2/7, Millet 6/9	13	72K	2 weeks
2	Classroom	1,360	1.5m × 1.5m	HUAWEI P10 * 2, Phab2, Nexus 6p * 2/7, Millet 6/9	18	96K	2 weeks
3	Shopping mall	2,130	—	HUAWEI P10 * 2, Phab2, Nexus 6p * 2/7, Millet 6/9, imoo Z5/Z6	30	288K	7 months

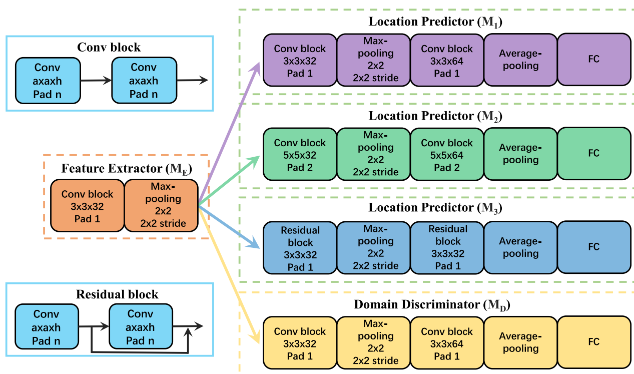


Fig. 10. The architecture of iToLoc.

the robustness of iToLoc in device diversity, we additionally employ two kinds of smartwatches, imoo Z5 and imoo Z6, to collect wireless fingerprints during the testing process.

5.1.2 Network Architectures and Parameters. The network architecture of iToLoc is illustrated in Figure 10. Different convolution kernel sizes and network units (with/without residual block) are integrated into M_1, M_2, and M_3 to enhance the diversity of the learned views among the three modules. The backbone network structure of iToLoc is consistent in all scenarios. However, the size of the input layer varies according to the number of APs in the fingerprint.

We use dropout ($p = 0.5$) after each max-pooling layer with Leaky-ReLU ($\alpha = 0.1$) as activate function except the FC layer, which is soft-max. Batch-Normalization is equipped for all layers. The learning rate starts from 0.1 in adversarial training (0.01 in model update) and is divided by 10 when the error plateaus. We maintain a batch size of 32 across the whole set of experiments. We use SGD optimizer schedule for gradient descent optimizer with weight decay of 0.0001 and a momentum of 0.9.

During the offline stage, all modules are trained for 200 epochs for initialization. Subsequently, modules M_1, M_2, M_3 are trained for 60 epochs for updating. During the model update, to prevent the network from overfitting, the reliable unlabeled fingerprints are selected from a pool, and we gradually increase the pool size $N = 60 \times 2^t$ up to the size of the unlabeled dataset U [38], where $t = 24$ denotes the learning round in all experiments. In the training process, we fine-tune the three modules M_1, M_2, and M_3 on the diverse training sets D_1, D_2, and D_3 every three rounds after $N = |U|$ to maintain diversity. As previously mentioned, since D_1, D_2, and D_3 are artificially fused with random noise, we progressively lower the confidence threshold σ by decreasing σ_{os} from its initial value σ_0 once $N = |U|$, allowing more unlabeled data to be labeled. Specifically, in the shopping mall dataset, we set $\sigma_0 = 0.96$ and $\sigma_{os} = 0.1$; in the office and classroom building dataset, we set $\sigma_0 = 0.98$ and $\sigma_{os} = 0.05$. Empirical hyper-parameters in adversarial training (Equation (13))

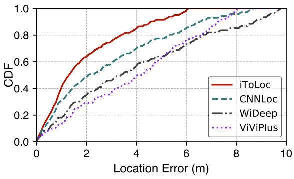


Fig. 11. Accuracy comparison.

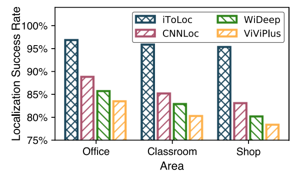


Fig. 12. Comparison in different areas.

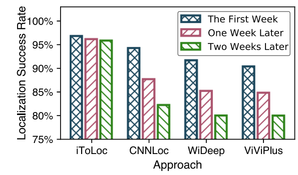


Fig. 13. Comparison in temporal robustness.

are set as follows: $\gamma = 0.2$, and $\lambda = 0.4$. In output smearing (Equation (14)), we set $std = 0.05$ for all three datasets.

5.1.3 Comparative Methods. To extensively evaluate the performance of iToLoc, we implement seven state-of-the-art approaches for comparison. We compare the localization accuracy and robustness of iToLoc without model update for short-term (within 2 weeks) with *ViViPlus* [58], *WiDeep* [1], *CNNLoc* [22], and *GIFT* [41]. And we further evaluate the automatic maintainability of iToLoc as well as *AcMu* [51], *Chameleon* [18], *WILL* [50], and *CapsLoc* [61] with model update for long-term (across 7 months).

5.1.4 Evaluation Metrics. In line with prior research, we adopt two methods to evaluate the localization performance. (1) *Distance-level localization bias*, which serves as a fine-grained indicator of performance. To achieve this, RSS fingerprints are collected at densely set checkpoints, as depicted in Figures 9(a) and 9(b), and the locations of these checkpoints are used as ground truth. The localization bias is defined as the Euler-distance between the localization result and the ground truth. (2) *Region-level localization success rate*, which is a relatively coarse-grained but intuitive and meaningful performance indicator. Here, we determine the rate at which a system correctly locates a user in the designated room or region that has been segmented. For the purposes of our study, we did not consider whether specific points within a room were accurately located. Typically, region-level localization metrics are more appropriate for coarser-grained indoor localization applications, while distance-level metrics are better suited for precise indoor navigation.

Both indicators are leveraged in experiments in the office and classroom buildings. However, due to the vast area of the shopping mall and the irregular shape of its public areas, a straightforward partitioning strategy was employed to divide the mall into distinct regions, wherein fingerprint data were collected and labeled for each region. The rationale behind this strategy was to avoid the complexity and time-consuming process of meticulously recording the physical coordinates of every point within such a large area. Consequently, the localization success rate was used as the sole performance metric to evaluate the localization performance, without taking into account the exact physical coordinates of each point.

5.2 Performance Evaluation

5.2.1 Overall Performance Comparison. We first evaluate the performance of iToLoc without the model update in short-term localization scenarios. Figure 11 depicts the performance of the proposed iToLoc as well as three other comparative systems. As shown, iToLoc achieves the best performance among all comparative systems. The average accuracy of iToLoc is 1.92 m, which outperforms CNNLoc by 32.1%, WiDeep by 51.7%, and ViViPlus by 52.1%. The 95th percentile accuracy of iToLoc outperforms these systems by 24.6%, 40.2%, and 29.5%, respectively. The results demonstrate iToLoc achieves remarkable performance based on only RSS fingerprints without introducing extra information or constraints.

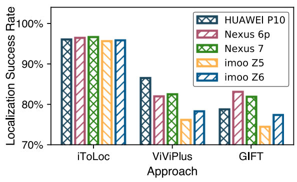


Fig. 14. Comparison in cross-device robustness.

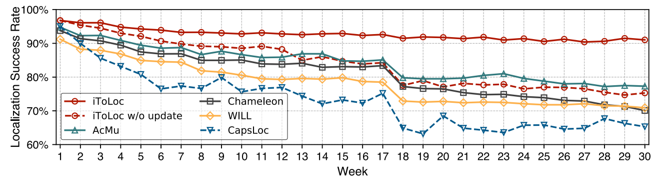


Fig. 15. Comparison in model update.

5.2.2 Performance Comparison in Different Areas. We further evaluate the performance in three different environments, as shown in Figure 9. In each environment, we send more than 1,000 queries and calculate the localization success rate. As depicted in Figure 12, iToLoc yields an average localization success rate of 96.8% in the office building, 95.8% in the classroom, and 95.3% in the shopping mall. And outperforms the other approaches by over 8% in all environments. Compared with the other approaches that the success rate varied by more than 5%, iToLoc yields similar performance (the differences in success rate are within 1.5%) regardless of the environments.

5.2.3 Performance Comparison with Different Time Interval. We also examine localization robustness in terms of temporal stability. We recollect fingerprints during different time intervals and use the original fingerprint database without update for localization. Figure 13 depicts the performance of iToLoc as well as the three approaches. iToLoc yields a similar success rate of more than 95% even after two weeks. Compared with related works where the success rates decrease more than 9%, the decrease in success rate for iToLoc is within 1%. The results demonstrate that based on pure RSS fingerprints, iToLoc extracts robust features that resist the fingerprint temporal instability caused by wireless signal fluctuation.

5.2.4 Performance Comparison with Device Diversity. We evaluate the cross-device robustness of iToLoc by involving various devices in training and testing stages. In the training stage, we use the fingerprints collected by five different devices, including Phab2 pro, Nexus 6p, Nexus 7, Millet 6, and Millet 9. And five other devices are used in the test stage. As shown in Figure 14, all comparative approaches severely suffer from device heterogeneity. Compared with these works, the variety of localization success rate of iToLoc is within 1%. Note that the WiFi chips for imoo Z5 and Z6 smartwatches vary considerably from those for the above smartphones. iToLoc still achieves outstanding performance under such a strict condition, demonstrating that iToLoc can tolerate device diversity much better than comparative methods.

5.2.5 Long-term Performance Comparison. We finally compare iToLoc with related systems equipped with radio-map automatic update algorithms to verify the effectiveness of the proposed semi-supervised learning-based model update framework. We continuously collect RSS fingerprints and evaluate the performance of systems over 7 months. Throughout the given period, our dependence on ubiquitously deployed WiFi devices prevents us from obtaining knowledge on potential infrastructure modifications, including the replacement, addition, or removal of one or more APs. Hence, we employ the auto-update algorithm every week to facilitate the model's adaptability to any undetermined environmental changes. Figure 15 records performance variation during such a long period. As shown, iToLoc achieves the best performance among all comparative systems at any given time. Even after seven months, the localization success rate of iToLoc still maintains 91%, which only decreases by 5.8% compared with other systems by at least 17%. Comparing the solid red line and dotted line, we can find that the proposed *co-training-based reliable model update* framework efficiently maintains the system performance. It is worth mentioning

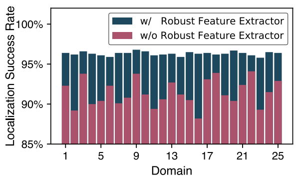


Fig. 16. Accuracy on each target domain.

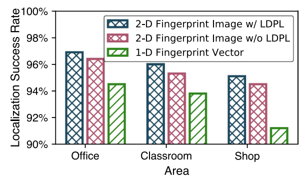


Fig. 17. Performance of 2D fingerprint image.

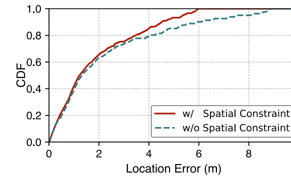


Fig. 18. Performance of spatial constraint.

that in the 18th week, due to the redecoration of the shopping mall, the performance of four comparative systems decreased sharply. In particular, CapsLoc, which lacked the ability to adapt to environmental changes, decreased by more than 10%. However, the impact on the performance of the proposed system iToLoc was negligible, with a localization accuracy decrease of no more than 2%. The above remarkable results demonstrate that the proposed iToLoc is qualified for locating in dynamic environments, and the semi-supervised learning-based model update framework is able to maintain the system's performance for a long time. In addition, unlike AcMu and WILL, iToLoc does not incorporate extra information (IMU, floor plan, etc.). Thus, the ubiquitousness of iToLoc retains the potential for wide deployment in practical service.

5.3 Study of Core Components

5.3.1 Impact of Robust Feature Extractor. We quantitatively analyze the performance of the DANN-based robust feature extractor with the baseline method without domain adversarial training strategy. We divide the training dataset into 22 source domains (5 devices with 6 time periods) and test set into 25 target domains (5 devices with 5 time periods) for evaluation. Figure 16 shows the accuracy on each target domain. The standard deviation of the localization success rate using the robust feature extractor is only 0.24%, while the baseline method is up to 1.62%. The average accuracy of our approach is also higher than the baseline by 4.9%. The above remarkable performance indicates that our model is capable of extracting device-independent and dynamics-resistant features.

5.3.2 Impact of Fingerprint-image Transformer. In this experiment, we aim to validate the advantages of the proposed fingerprint-image transformer. We focus on the relative accuracy other than the absolute improvement achieved by our approach. In each experimental scenario, we use the 2D fingerprint image with LDPL, 2D fingerprint image without LDPL and 1D fingerprint vector as their inputs, respectively. As shown in Figure 17, by leveraging the fingerprint image generated without LDPL as input, the performance gain is 2.0%, 1.6%, and 3.5% in the office building, classroom building, and shopping mall, respectively. Meanwhile, the 2D fingerprint image generated with LDPL is about 0.5% better than the one without LDPL in terms of overall performance. This results demonstrate that the proposed fingerprint-image transformer will further promote iToLoc to extract domain-independent robust features.

5.3.3 Impact of Spatial Constraint. We further evaluate the spatial constraint, which is proposed to solve the frequent significant localization bias. As shown in Figure 18, the average accuracy outperforms the method without spatial constraint by 18.3%, and the improvement of 95th percentile location error is 32.4%, achieving a mean accuracy of 1.86 m and 95th percentile accuracy of 5.41 m. The results demonstrate that the spatial constraint can effectively reduce large errors and improve overall localization performance.

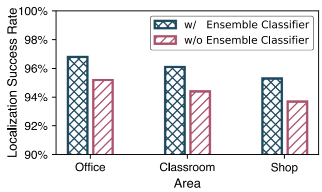


Fig. 19. Performance of ensemble classifier.

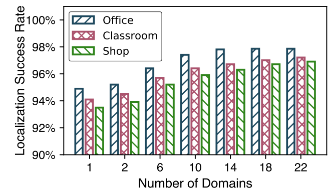
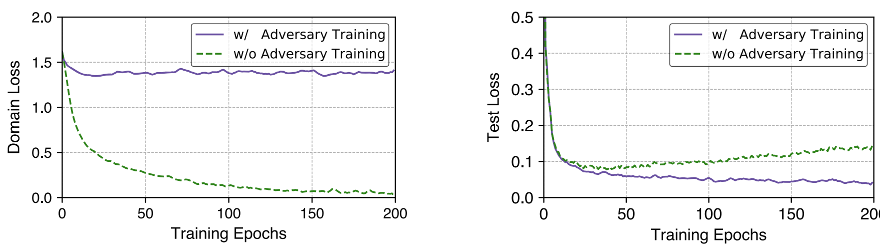


Fig. 20. Number of domains used for training.



(a) Domain loss

(b) Test loss

Fig. 21. The models with and without adversarial learning are evaluated on the same dataset. A higher domain loss indicates the removal of domain specific information, and a lower test loss shows that the proposed DANN can better avoid overfitting.

5.3.4 Impact of Ensemble Classifier. In this experiment, we evaluate the performance of ensemble classifier (M_1, M_2, and M_3) in location predictor. We compare with a baseline model, which has only one classifier M_1 in location predictor. As shown in Figure 19, our method outperforms the baseline by 1.65%, 1.77%, and 1.68% in office building, classroom building, and shopping mall, respectively. The results demonstrate that ensemble classifier can effectively leverage multiple views from three classifiers with different architecture for localization.

5.4 Training of DANN

5.4.1 Number of Domains Used for Training. As shown in Figure 20, we gradually increase the number of source domains from 1 to 22 while keep the total amount of data in the training set unchanged. When there are only 1 or 2 source domains, iToLoc has relatively low accuracy, because most of the fingerprints have no domain label so that the discriminator is difficult to extract domain independent features. As the number of source domains increases, iToLoc is able to extract the common features shared by both source and target domains. It is worth mentioning that when the number of domains reaches 14, the performance of iToLoc in office building will no longer improve, but in the more complex environments, especially the shopping mall, the performance is still improved even if the number of domains is large enough.

5.4.2 Qualitative Analysis of Training Loss. To qualitatively evaluate the learned robust feature, we look at the losses on the training set and test set as training progresses to check whether the extraneous information specific to the devices and time periods can be removed. We compare with a version of our model without the domain discriminator. For this baseline, we train a (non-adversarial) discriminator to determine the domain of features. Figure 21(a) shows that the loss of the domain discriminator in the baseline model decreases very quickly while ours stays high, suggesting that our learned robust feature is invariant across source domain. Figure 21(b) also shows that adding an domain discriminator increases the performance on the test set and can be helpful in reducing over-fitting.

6 RELATED WORK

Indoor localization has attracted vast research efforts during the past decades. We briefly review the most related latest works in the following.

Fingerprint-based indoor localization. Fingerprint-based indoor localization is a commonly used approach that utilizes radio-map to determine the location of mobile devices in indoor environments. The mainstream of existing methods rely on RSS observations from wireless signals, such as Wi-Fi and Bluetooth, at known locations to generate fingerprints [27, 29, 59]. These systems typically involve two phases: during the offline training phase, an RSS fingerprint database is constructed with labeled fingerprints and their corresponding physical locations. In the online localization phase, the location of the mobile device is determined through fingerprint matching [4, 27]. Despite being a well-established localization paradigm, fingerprint-based localization encounters challenges in today's dynamic indoor environments due to signal fluctuations, device heterogeneity, and degradation of the radio-map [26, 36, 48].

Robust wireless fingerprint extractor. To enhance the localization robustness without compromising the ubiquity of fingerprint-based approaches, various subsequent works have sought to find a robust and expressive form of RSS fingerprints [11, 17, 28, 33, 40]. In particular, recent innovations explore the spatial/temporal properties of fingerprints for localization. GIFT [41] defines a metric of binary differential value between RSS observed at two adjacent locations as replacements to the original RSS values as fingerprints; ViVi [48] and ViViPlus [58] explore the spatial gradient of selected multiple neighboring fingerprints to deal with the device heterogeneity as well as spatial ambiguity. However, the localization accuracy and cross-device robustness still remain low in practice due to frequent environmental changes and device heterogeneity. Compared with these works, iToLoc extracts device-independent and dynamics-resistant features, which have been demonstrated to be more robustness.

Deep learning assisted localization. There has been some recent work in localization using deep neural networks [2, 3, 35, 47]. Some works are also equipped with deep learning models to enhance fingerprint matching. In References [9, 20, 22, 34, 42], the CNN-based WiFi fingerprinting method was presented to pursue better performance. WiDeep [1] integrates a stacked autoencoders deep learning model with a probabilistic framework to handle the noise and capture the complex relationship between the WiFi APs signals. The previous approaches [36, 43] have shown that considering fingerprint collection time and device as domains and utilizing domain-adaptive transfer learning-based frameworks can significantly improve the accuracy of localization. Despite the increased offline training work and online inference time required by deep learning-based methods, which are considered secondary to localization accuracy, their significant performance improvements cannot be denied. Such advancements have further propelled the development of high-resolution indoor localization applications. Inspired by recent works, iToLoc leverages adversarial network to extract robust domain-independent representations.

Fingerprint database automatic update. Considering environmental dynamics, pioneering works, including LiFs [59], UnLoc [44], WILL [50], and AcMu [51], leverage the various built-in sensors in smartphones to provide extra range constraints for accurate localization results and further conditionally update the fingerprint database. However, they induce extra constraints to user behaviors and degrade the system ubiquity. Some recent transfer learning-based techniques, such as manifold alignment, are also applied to correct RSS measurements and update radio-map over time [18, 19]. However, the fingerprint database will also gradually deteriorate, because there is still a gap between reliable adaption and robust localization. In contrast, iToLoc treats the radio-map adaption problem as a semi-supervised learning task and leverages co-training-based framework to automatically update the model, which has been demonstrated more effective to underpin a long-term localization service.

7 CONCLUSION

In this article, we propose iToLoc, a deep learning-based localization system that achieves all three goals on high deployment ubiquity, high localization accuracy and robustness, and low maintenance overhead simultaneously. We prototype iToLoc and evaluate it by extensive experiments across 7 months and by eight different devices. The results demonstrate its superior performance over previous schemes. Trained once, locate anytime for anyone: iToLoc makes a great progress toward fortifying WiFi fingerprint-based localization to a fully practical service for wide deployment.

REFERENCES

[1] Moustafa Abbas, Moustafa Elhamshary, Hamada Rizk, Marwan Torki, and Moustafa Youssef. 2019. WiDeep: WiFi-based accurate and robust indoor localization system using deep learning. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom'19)*. IEEE, 1–10.

[2] Maximilian Arnold, Jakob Hoydis, and Stephan ten Brink. 2019. Novel massive MIMO channel sounding data applied to deep learning-based indoor positioning. In *Proceedings of the 12th International ITG Conference on Systems, Communications and Coding (SCC'19)*. VDE.

[3] Roshan Ayyalasomayajula, Aditya Arun, Chenfeng Wu, Sanatan Sharma, Abhishek Rajkumar Sethi, Deepak Vasisht, and Dinesh Bharadia. 2020. Deep learning-based wireless localization for indoor navigation. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*.

[4] Paramvir Bahl and Venkata N. Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. In *Proceedings IEEE Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'00)*, Vol. 2. IEEE, 775–784.

[5] Maria-Florina Balcan and Avrim Blum. 2010. A discriminative model for semi-supervised learning. *J. ACM* 57, 3 (2010), 1–46.

[6] Maria-Florina Balcan, Avrim Blum, and Ke Yang. 2005. Co-training and expansion: Towards bridging theory and practice. In *Advances in Neural Information Processing Systems*. 89–96.

[7] Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the 11th Annual Conference on Computational Learning Theory*. 92–100.

[8] Leo Breiman. 2000. Randomizing outputs to increase prediction accuracy. *Mach. Learn.* (2000).

[9] Hao Chen, Yifan Zhang, Wei Li, Xiaofeng Tao, and Ping Zhang. 2017. ConFi: Convolutional neural networks based indoor Wi-Fi localization using channel state information. *IEEE Access* (2017).

[10] Xi Chen, Hang Li, Chenyi Zhou, Xue Liu, Di Wu, and Gregory Dudek. 2020. FiDo: Ubiquitous Fine-Grained WiFi-based Localization for Unlabelled Users via Domain Adaptation. In *Proceedings of the Web Conference 2020*. 23–33.

[11] Yiqiang Chen, Qiang Yang, Jie Yin, and Xiaoyong Chai. 2006. Power-efficient access-point selection for indoor location estimation. *IEEE Trans. Knowl. Data Eng.* 18, 7 (2006), 877–888.

[12] Guoxuan Chi, Zheng Yang, Jingao Xu, Chenshu Wu, Jialin Zhang, Jianzhe Liang, and Yunhao Liu. 2022. Wi-drone: WIFI-based 6-DoF tracking for indoor drone flight control. In *Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services*.

[13] Liang Dong, Jingao Xu, Guoxuan Chi, Danyang Li, Xinglin Zhang, Jianbo Li, Qiang Ma, and Zheng Yang. 2020. Enabling surveillance cameras to navigate. In *Proceedings of the IEEE International Conference on Computer Communications and Networks (ICCCN'20)*.

[14] Chen Dongdong, Wang Wei, Gao Wei, and Zhou Zhihua. 2018. Tri-net for semi-supervised deep learning. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*.

[15] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* 17, 1 (2016), 2096–2030.

[16] Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai et al. 2018. Recent advances in convolutional neural networks. *Pattern Recogn.* 77 (2018), 354–377.

[17] Suining He and S.-H. Gary Chan. 2015. Wi-Fi fingerprint-based indoor positioning: Recent advances and comparisons. *IEEE Commun. Surveys Tutor.* 18, 1 (2015), 466–490.

[18] Suining He, Bo Ji, and S.-H. Gary Chan. 2016. Chameleon: Survey-free updating of a fingerprint database for indoor localization. *IEEE Pervas. Comput.* 15, 4 (2016), 66–75.

[19] Suining He, Wenbin Lin, and S.-H. Gary Chan. 2016. Indoor localization and automatic fingerprint update with altered AP signals. *IEEE Trans. Mobile Comput.* 16, 7 (2016), 1897–1910.

[20] Mai Ibrahim, Marwan Torki, and Mustafa ElNainay. 2018. CNN based indoor localization using RSS time-series. In *Proceedings of the IEEE Symposium on Computers and Communications (ISCC'18)*. IEEE.

[21] IMOO. 2023. Retrieved from https://www.imoo.com/

[22] Jin-Woo Jang and Song-Nam Hong. 2018. Indoor localization with wifi fingerprinting using convolutional neural network. In *Proceeding of the IEEE International Conference on Ubiquitous and Future Networks (ICUFN'18).*

[23] Wenjun Jiang, Chenglin Miao, Fenglong Ma, Shuochao Yao, Yaqing Wang, Ye Yuan, Hongfei Xue, Chen Song, Xin Ma, Dimitrios Koutsonikolas, et al. 2018. Towards environment independent device free human activity recognition. In *Proceedings of the ACM Annual International Conference on Mobile Computing and Networking (MOBICOM'18).* 289–304.

[24] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems.* 1097–1105.

[25] Danyang Li, Yumeng Lu, Jingao Xu, Qiang Ma, and Zhuo Liu. 2019. iPAC: Integrate pedestrian dead reckoning and computer vision for indoor localization and tracking. *IEEE Access* (2019).

[26] Danyang Li, Jingao Xu, Zheng Yang, Yumeng Lu, Qian Zhang, and Xinglin Zhang. 2021. Train once, locate anytime for anyone: Adversarial learning based wireless localization. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM'21).*

[27] Danyang Li, Jingao Xu, Zheng Yang, Chenshu Wu, Jianbo Li, and Nicholas D. Lane. 2021. Wireless localization with spatial-temporal robust fingerprints. *ACM Trans. Sensor Netw.* (2021).

[28] Liqun Li, Guobin Shen, Chunshui Zhao, Thomas Moscibroda, Jyh-Han Lin, and Feng Zhao. 2014. Experiencing and handling the diversity in data density and environmental locality in an indoor positioning service. In *Proceedings of the ACM Annual International Conference on Mobile Computing and Networking (MOBICOM'14).* 459–470.

[29] Hongbo Liu, Yu Gan, Jie Yang, Simon Sidhom, Yan Wang, Yingying Chen, and Fan Ye. 2012. Push the limit of WiFi based localization for smartphones. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking.* 305–316.

[30] Kaikai Liu, Xinxin Liu, and Xiaolin Li. 2013. Guoguo: Enabling fine-grained indoor localization via smartphone. In *Proceedings of the ACM Annual International Conference on Mobile Systems, Applications and Services (MobiSys'13).* 235–248.

[31] Dimitrios Lymberopoulos and Jie Liu. 2017. The microsoft indoor localization competition: Experiences and lessons learned. *IEEE Signal Process. Mag.* 34, 5 (2017), 125–140.

[32] Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *J. Mach. Learn. Res.* 9 (Nov. 2008), 2579–2605.

[33] Piotr Mirowski, Philip Whiting, Harald Steck, Ravishankar Palaniappan, Michael MacDonald, Detlef Hartmann, and Tin Kam Ho. 2012. Probability kernel regression for WiFi localisation. *J. Loc. Based Serv.* 6, 2 (2012), 81–100.

[34] Wafa Njima, Iness Ahriz, Rafik Zayani, Michel Terre, and Ridha Bouallegue. 2019. Deep CNN for indoor localization in iot-sensor systems. *Sensors* (2019).

[35] Michał Nowicki and Jan Wietrzykowski. 2017. Low-effort place recognition with WiFi fingerprints using deep learning. In *Proceedings of the International Conference Automation.* Springer.

[36] Sinno Jialin Pan, Vincent Wenchen Zheng, Qiang Yang, and Derek Hao Hu. 2008. Transfer learning for WiFi-based indoor localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 6.

[37] Anshul Rai, Krishna Kant Chintalapudi, Venkata N. Padmanabhan, and Rijurekha Sen. 2012. Zee: Zero-effort crowdsourcing for indoor localization. In *Proceedings of the ACM Annual International Conference on Mobile Computing and Networking (MOBICOM'12).* 293–304.

[38] Kuniaki Saito, Yoshitaka Ushiku, and Tatsuya Harada. 2017. Asymmetric tri-training for unsupervised domain adaptation. Retrieved from https://arXiv:1702.08400

[39] Longfei Shangguan, Zheng Yang, Alex X. Liu, Zimu Zhou, and Yunhao Liu. 2016. STPP: Spatial-temporal phase profiling-based method for relative RFID tag localization. *IEEE/ACM Trans. Netw.* 25, 1 (2016), 596–609.

[40] Guobin Shen, Zhuo Chen, Peichao Zhang, Thomas Moscibroda, and Yongguang Zhang. 2013. Walkie-Markie: Indoor pathway mapping made easy. In *Proceedings of the USENIX Symposium on Networked Systems Design and Implementation (NSDI'13).* 85–98.

[41] Yuanchao Shu, Yinghua Huang, Jiaqi Zhang, Philippe Coué, Peng Cheng, Jiming Chen, and Kang G. Shin. 2015. Gradient-based fingerprinting for indoor localization and tracking. *IEEE Trans. Industr. Electr.* 63, 4 (2015), 2424–2433.

[42] Xudong Song, Xiaochen Fan, Chaocan Xiang, Qianwen Ye, Leyu Liu, Zumin Wang, Xiangjian He, Ning Yang, and Gengfa Fang. 2019. A novel convolutional neural network based indoor localization framework with WiFi fingerprinting. *IEEE Access* 7 (2019), 110698–110709.

[43] Zhuo Sun, Yiqiang Chen, Juan Qi, and Junfa Liu. 2008. Adaptive localization through transfer learning in indoor Wi-Fi environment. In *Proceedings of the 7th International Conference on Machine Learning and Applications*. IEEE.
[44] He Wang, Souvik Sen, Ahmed Elgohary, Moustafa Farid, Moustafa Youssef, and Romit Roy Choudhury. 2012. No need to war-drive: Unsupervised indoor localization. In *Proceedings of the ACM Annual International Conference on Mobile Systems, Applications and Services (MobiSys'12)*.
[45] Jue Wang and Dina Katabi. 2013. Dude, where's my card? RFID positioning that works with multipath and non-line of sight. In *Proceedings of the ACM Special Interest Group on Data Communications (SIGCOMM'13)*. 51–62.
[46] Wei Wang and Zhi-Hua Zhou. 2010. A new analysis of co-training. In *Proceedings of the International Conference on Machine Learning (ICML'10)*.
[47] Xuyu Wang, Lingjun Gao, and Shiwen Mao. 2017. BiLoc: Bi-modal deep learning for indoor localization with commodity 5 GHz WiFi. *IEEE Access* 5 (2017), 4209–4220.
[48] Chenshu Wu, Jingao Xu, Zheng Yang, Nicholas D. Lane, and Zuwei Yin. 2017. Gain without pain: Accurate WiFi-based localization using fingerprint spatial gradient. In *Proceedings of the ACM Conference on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT'17)*.
[49] Chenshu Wu, Zheng Yang, and Yunhao Liu. 2014. Smartphones based crowdsourcing for indoor localization. *IEEE Trans. Mobile Comput.* 14, 2 (2014), 444–457.
[50] Chenshu Wu, Zheng Yang, Yunhao Liu, and Wei Xi. 2012. WILL: Wireless indoor localization without site survey. *IEEE Trans. Parallel Distrib. Syst.* 24, 4 (2012), 839–848.
[51] Chenshu Wu, Zheng Yang, Chaowei Xiao, Chaofan Yang, Yunhao Liu, and Mingyan Liu. 2015. Static power of mobile devices: Self-updating radio maps for wireless indoor localization. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM'15)*. IEEE, 2497–2505.
[52] Chenshu Wu, Feng Zhang, Yusen Fan, and K. J. Ray Liu. 2019. RF-based inertial measurement. In *Proceedings of the ACM Special Interest Group on Data Communications (SIGCOMM'19)*.
[53] Chenshu Wu, Feng Zhang, Beibei Wang, and K. J. Ray Liu. 2019. EasiTrack: Decimeter-level indoor tracking with graph-based particle filtering. *IEEE Internet Things J.* 7, 3 (2019), 2397–2411.
[54] Han Xu, Zheng Yang, Zimu Zhou, Longfei Shangguan, Ke Yi, and Yunhao Liu. 2015. Enhancing wifi-based localization with visual clues. In *Proceedings of the ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 963–974.
[55] Jingao Xu, Hao Cao, Danyang Li, Kehong Huang, Chen Qian, Longfei Shangguan, and Zheng Yang. 2020. Edge assisted mobile semantic visual SLAM. In *Proceedings of the IEEE International Conference on Computer Communications (INFOCOM'20)*.
[56] Jingao Xu, Hengjie Chen, Kun Qian, Erqun Dong, Min Sun, Chenshu Wu, Li Zhang, and Zheng Yang. 2019. iVR: Integrated vision and radio localization with zero human effort. In *Proceedings of the ACM Conference on Interactive, Mobile, Wearable and Ubiquitous Technologies (IMWUT'19)*. 1–22.
[57] Jingao Xu, Erqun Dong, Qiang Ma, Chenshu Wu, and Zheng Yang. 2021. Smartphone-based indoor visual navigation with leader-follower mode. *ACM Trans. Sensor Netw.* (2021).
[58] Jingao Xu, Zheng Yang, Hengjie Chen, Yunhao Liu, Xiancun Zhou, Jianbo Li, and Nicholas Lane. 2018. Embracing spatial awareness for reliable wifi-based indoor location systems. In *Proceedings of the IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS'18)*. IEEE, 281–289.
[59] Zheng Yang, Chenshu Wu, and Yunhao Liu. 2012. Locating in fingerprint space: Wireless indoor localization with little human intervention. In *Proceedings of the ACM Annual International Conference on Mobile Computing and Networking (MOBICOM'12)*. 269–280.
[60] Zheng Yang, Chenshu Wu, Zimu Zhou, Xinglin Zhang, Xu Wang, and Yunhao Liu. 2015. Mobility increases localizability: A survey on wireless indoor localization using inertial sensors. *ACM Comput. Surveys* 47, 3 (2015), 1–34.
[61] Qianwen Ye, Xiaochen Fan, Gengfa Fang, Hongxia Bie, Xudong Song, and Rajan Shankaran. 2020. CapsLoc: A robust indoor localization system with WiFi fingerprinting using capsule networks. In *Proceedings of the IEEE International Conference on Communications (ICC'20)*.
[62] Moustafa Youssef and Ashok Agrawala. 2005. The Horus WLAN location determination system. In *Proceedings of the ACM International Conference on Mobile Systems, Applications, and Services (MOBISYS'05)*. 205–218.
[63] Mingmin Zhao, Shichao Yue, Dina Katabi, Tommi S. Jaakkola, and Matt T. Bianchi. 2017. Learning sleep stages from radio signals: A conditional adversarial architecture. In *Proceedings of the International Conference on Machine Learning*.
[64] Zhi-Hua Zhou and Ming Li. 2005. Semi-supervised regression with co-training. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI'05)*, Vol. 5. 908–913.

Received 22 January 2023; revised 5 July 2023; accepted 31 July 2023