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

Danyang Li[1*], Jingao Xu[1*], Zheng Yang[1†], Yumeng Lu[1], Qian Zhang[1], Xinglin Zhang[2],
[*] Co-primary author [†]Corresponding author
[1]School of Software and BNRist, Tsinghua University
[2]South China University of Technology, China
{xujingao13, lidanyang1919, hmilyyz, lym.mongo, qzhangqz123, zhxlinse}@gmail.com

Abstract*—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 delightful 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 paper, 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 8 different devices demonstrate that *iToLoc* achieves remarkable performance with an accuracy of 1.92m 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.

I. 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, such as WiFi [1]–[6], RFID [7], [8], acoustic signals [9], visual images [10], [11], *etc.* have been proposed in the past decade. 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 [12]–[15]. This approach generally has a training stage, in which RSS fingerprints with location labels are collected by manual site survey or leveraging crowdsourcing scheme [12], [14] to automatic 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 [16]; XiaoTianCai and HUAWEI develop smartwatches integrated with such modules to locate and protect children [17]; 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

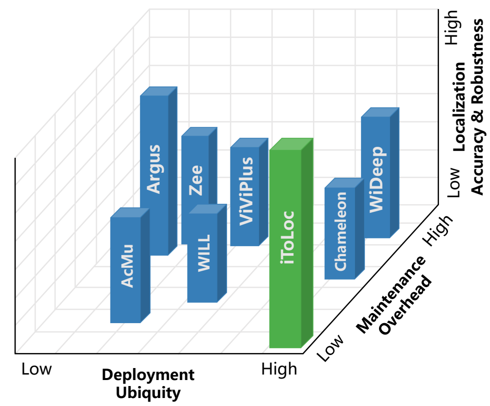
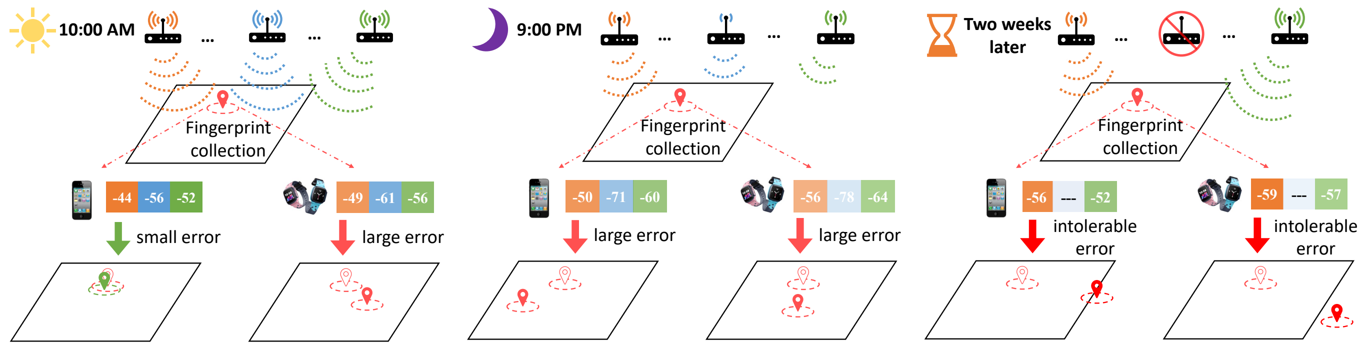


Fig. 1. A comparison of state-of-the-art works.

has not yet stepped in the prime time for wide deployment. We evaluate the performance of the RSS fingerprint-based localization system in real business environments across 7 months, and finally find that the primary hurdles are *signal variation*, *device heterogeneity* and *database deterioration* as illustrated in Fig. 2. On one hand, diverse RSS will be encountered at different time by different devices, which will lead to query fingerprint's mismatch against the database. Typically, location errors sometimes increase up to ten meters. 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 [2], [18], and further employ deep neural networks for classification [19]. And 2) additional information: inertial sensing [20], image matching [21], and even physical layer Channel State Information (CSI) [22] 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 [23], aiming to ease the maintenance cost.

Albeit inspiring, previous works make a sacrifice to overcome the above challenges and thus face severe limitations.



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

Fig. 2. Illustration of three key reasons that lead to frequent large localization bias: *signal variation*, *device heterogeneity*, and *database deterioration*. (a) and (b) show that RSS values is 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 AP under even identical wireless conditions. (c) Because of environmental changes, an initial fingerprint database may gradually deteriorate, leading to inaccurate location estimations.

First, the *localization accuracy and cross-device robustness remain low* in practice because of severe environment dynamics and device heterogeneity [24]. 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 [12], [23]. 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, the rationale behind the requirement is impractical [21], [25]. It is also required to deploy extra infrastructures, including surveillance cameras, ultrasonic beacons, or specific WiFi devices to collect CSI. Fig. 1 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 delightful deployment ubiquity.

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 ***T**rain **o**nce, update automatically, **Loc**ate 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 delightful localization performance, 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) [26], [27], 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 seem 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* [28], [29] 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 1.92m and a 95th percentile accuracy of 3.4m, 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 delightful deployment ubiquity, *iToLoc* takes an essential step towards 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 8 different types of devices (including 2 smartwatches) in real environments for 7 months. Encouraging results demonstrate that *iToLoc* makes a great progress towards fortifying WiFi fingerprint-based localization to an entirely practical service for wide deployment.

The rest of paper is organized as follows. We present an overview in Section II and introduce *adversarial learning based robust localization* in Section III. *Co-training based reliable model update* is provided in Section IV, followed by implementation experiments in Section V. We review related works in Section VI and conclude this paper in Section VII.

II. System Overview

As illustrated in Fig. 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 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 III). 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 IV). 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.

III. Adversarial Learning Based Robust Localization

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

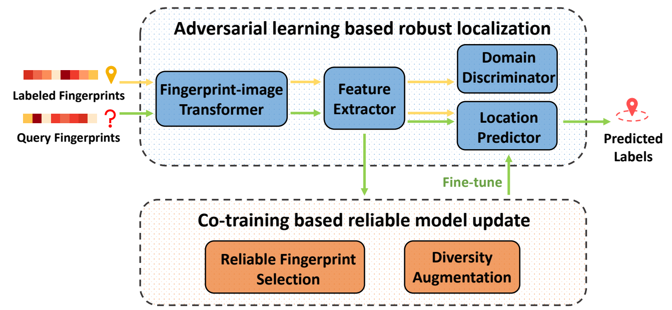


Fig. 3. System overview

Towards this goal, the 1-D RSS fingerprint vectors are first transformed into 2-D 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 [30]. 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.

A. Fingerprint-image Transformer

Suppose that each user receives the RSS values from neighboring N APs, namely, the data $\boldsymbol{f} = \{f_1, f_2, \cdots, f_N\}$ is a 1-D 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 1-D fingerprint vector as the input, just like recent CNN based localization systems [31], [32]. However, the adversarial network hardly achieves delightful performance. 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 [27], [30]. 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) [33], [34]. To this end, the original 1-D 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.

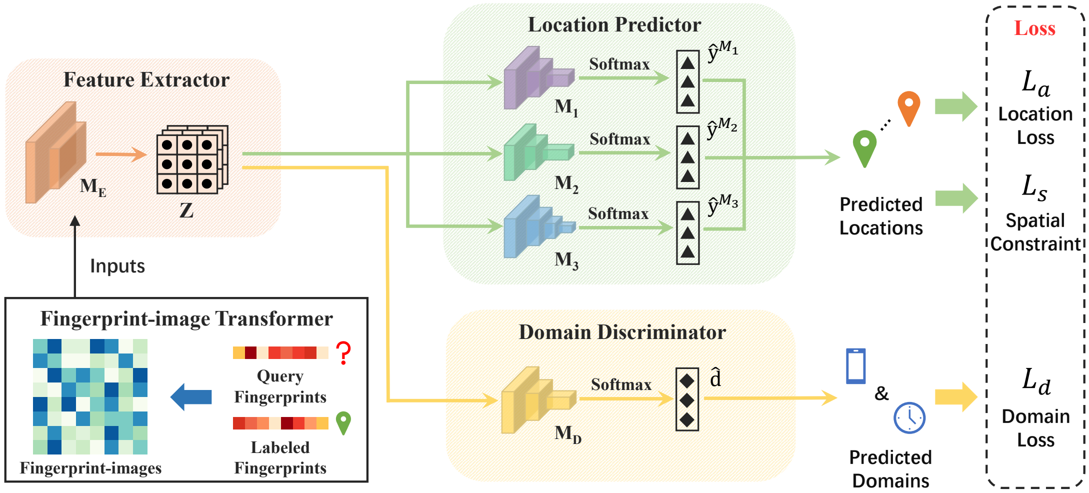


Fig. 4. Overview of Domain Adversarial Learning Based Model

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

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

where $f^*_{j,k} = (f_j - f_k) * \frac{1}{f_k}$. Compared with the 1-D fingerprint vector, each element in $\mathbf{x}$ is relevant to neighboring elements. In the meantime, all original RSS values can be recalculated from $\mathbf{x}$ without losing information. Effectiveness of the proposed fingerprint-image transformer will be evaluated in Section V-C.

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 2-D fingerprint image $\mathbf{x}$, the domain label vector $\mathbf{d}$, and the ground truth location $\mathbf{y}$.

B. Feature Extractor

In the feature extractor, we employ a two-layer CNN module M_E to extract latent features. Let Θ_{M_E} be the set of parameters of M_E. Given the input data $\mathbf{x}$, we can obtain the latent feature $\boldsymbol{Z}$ as follows:

$$\boldsymbol{Z} = M_E(\mathbf{x}; \Theta_{M_E}) \tag{2}$$

C. 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 $\boldsymbol{Z}_i$:

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

where Θ_{M_k} are the parameters of M_k to be learned. In order 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:

$$\hat{\mathbf{y}}_i^{M_k} = \text{Softmax}\left(\boldsymbol{W}_v^{M_k} \boldsymbol{V}_i^{M_k} + \boldsymbol{b}_v^{M_k}\right); k = 1, 2, 3 \tag{4}$$

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:

$$L_a = -\frac{1}{|\mathbf{X}|} \sum_{i=1}^{|\mathbf{X}|} \sum_{k=1}^{3} \sum_{c=1}^{C} \mathbf{y}_{ic} \log\left(\hat{\mathbf{y}}_{ic}^{M_k}\right) \tag{5}$$

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.

D. Domain Discriminator

In our adversarial network, the domain is defined as *a pair of device and time*. The rationale behind making this definition are two-folds: on 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 [35].

In order 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 $\boldsymbol{Z}_i$:

$$\boldsymbol{U}_i = M_D\left(\boldsymbol{Z}_i; \Theta_{M_D}\right) \tag{6}$$

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

$$\hat{\mathbf{d}}_i = \mathrm{Softmax}\left(\mathbf{W}_u \mathbf{U}_i + \mathbf{b}_u\right) \tag{7}$$

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

$$L_d = -\frac{1}{|\mathbf{X}|} \sum_{i=1}^{|\mathbf{X}|} \sum_{j=1}^{|\mathcal{D}|} \mathbf{d}_{ij} \log\left(\hat{\mathbf{d}}_{ij}\right) \tag{8}$$

where $\mathbf{d}_i$ is the one-hot vector of true domain labels. The goal of the domain discriminator is to minimize L_d so as 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 Eq. 5 and Eq. 8, we can obtain the loss function as follows:

$$L = L_a - \lambda L_d \ (\lambda > 0) \tag{9}$$

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

E. 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:

$$L_s = \frac{1}{|\mathbf{X}|} \sum_{i=1}^{|\mathbf{X}|} \sum_{c=1}^{C} w_{y_i c} \hat{\mathbf{y}}_{ic} \tag{10}$$

where $w_{y_i c}$ is the weight representing the physical distance between the c-th location and ground truth $\mathbf{y}_i$. 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 extra consumption.

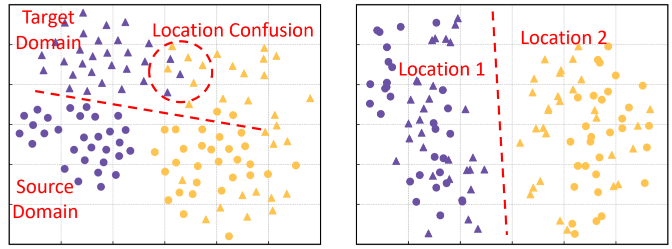
F. Objective and Training

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

$$L = L_a + \gamma L_s - \lambda L_d \ (\gamma, \lambda > 0) \tag{11}$$

Following the training process of DANN [26], 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 Eq. 2 by baseline CNN frameworks and *iToLoc*, respectively. Finally, we plot the learned representations of these samples on a 2-D space with a manifold learning algorithm t-SNE [36]. In Fig. 5, we use



(a) (b)

Fig. 5. 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.

orange and purple colors to represent different locations where the fingerprints are collected, circle and triangle markers to represent source domain (time and device labels are known) and target domain (unknown time and device), respectively. As shown in Fig. 5a, 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 Fig. 5b, 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.

IV. Co-Training Based Reliable Model Update

The training process of model update is shown in Fig. 6. 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 IV-A. 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 IV-B. With the above techniques for the model update, *iToLoc* can adapt to the unpredictable environmental dynamic and provide long-term services.

A. Reliable Fingerprint Selection

The pseudo-labels of the newly labeled fingerprints may be incorrect, and these incorrect pseudo-labels will degenerate the performance. Therefore, we must carefully select confident and stable fingerprints. Here, confident prediction means that the average maximum posterior probability of the two modules

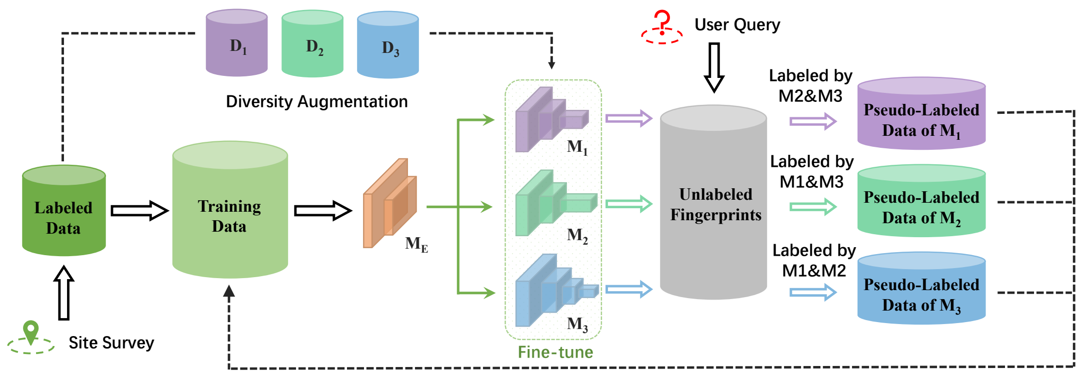


Fig. 6. Training process of model update

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 [29] 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 $k > \frac{K}{3}$, 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.

B. 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. In order to maintain the diversity, we use *output smearing* 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 c-th class otherwise $\mathbf{y}_{ic} = 0$. In output smearing, we add noise into every component of $\mathbf{y}_i$.

$$\mathbf{y}'_{ic} = \mathbf{y}_{ic} + \mathrm{ReLU}\left(z_{ic} \times std\right) \tag{12}$$

where z_{ic} is sampled independently from the standard normal distribution, std is the standard deviation, ReLU is a function to ensure $\mathbf{y}'_{ic}$ non-negative. And normalize $\mathbf{y}'_{ic}$ as follows:

$$\mathbf{y}'_i = \left(\mathbf{y}'_{i1}, \mathbf{y}'_{i2}, \ldots, \mathbf{y}'_{ic}\right) / \sum_{c=1}^{C} \mathbf{y}'_{ic} \tag{13}$$

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.

C. Rationale Behind Reliable Model Update

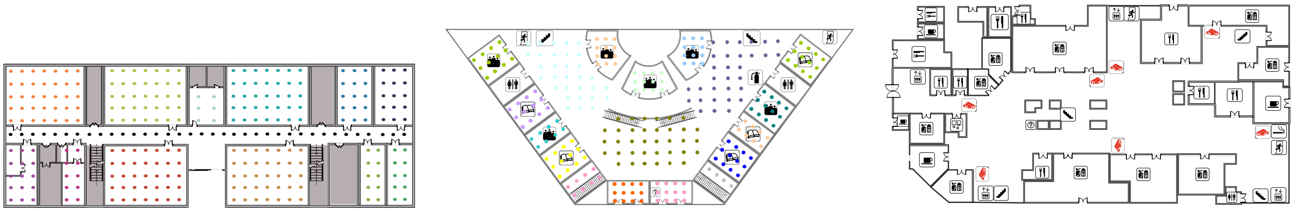
In this subsection, we explain the rationale behind the proposed strategy for the reliable model update. First, the fundamental of the strategy is the *disagreement based semi-supervised learning* whose basic idea is to train multiple learners for the same task but exploit the disagreements during the learning process. Many theoretical studies have explained why unlabeled data can improve learning performance and efficiently fine-tune models based on such strategy [37]–[40]. 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. 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.

V. Implementation and Evaluation

A. 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.

1) Experimental Scenarios & Datasets: We carry out experiments in three different buildings with various floor layouts and diverse wireless environments. Fig. 7 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.



(a) Office building (b) Classroom building (c) Shopping mall

Fig. 7. Experimental areas

TABLE I
DATA COLLECTION IN DIFFERENT SCENARIOS

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

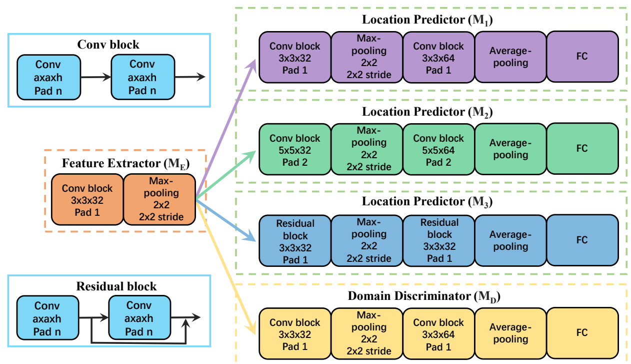


Fig. 8. The architecture of *iToLoc*

The data collection details are summarized in Table I. 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 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.

2) Network Architectures & Parameters: The network architecture of *iToLoc* is illustrated in Fig. 8. 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 M_E, M_1, M_2, M_3 and M_D are trained for 200 epochs. During the model update, in order 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 [41], where t denotes the learning round. We further fine-tune three modules M_1, M_2 and M_3 on the diverse training sets D_1, D_2 and D_3 every 3 rounds after $N = |U|$ to maintain the diversity. As aforementioned, since D_1, D_2 and D_3 are artificially fused with random noise, the confidence threshold σ is decreased by σ_{os}. Specifically, we set $\sigma_0 = 0.96$ and $\sigma_{\text{os}} = 0.1$ in shopping mall dataset; $\sigma_0 = 0.98$ and $\sigma_{\text{os}} = 0.05$ in office and classroom building dataset.

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* [18], *WiDeep* [19], *CNNLoc* [31], and *GIFT* [42]. And we further evaluate the automatic maintainability of *iToLoc* as well as *AcMu* [23], *Chameleon* [43], and *WILL* [20], with model update for long-term (across 7 month).

4) Evaluation Metrics: Similar to existing works, we adopt two methods to evaluate the localization performance: 1) Distance-level localization bias, which is a fine-grained evaluating indicator. The Euler-distance between localization result and ground truth is defined as localization bias. 2) Room (region)-level localization success rate, which is a relatively coarse-grained but intuitive and meaningful indicator. We count the rate that a system locates a user in the correct room or region we segmented. Both two indicators are leveraged in experiments in the office and classroom buildings. And only the localization success rate is used in the shopping mall because of the large fingerprints sampling density and irregular shape of public areas.

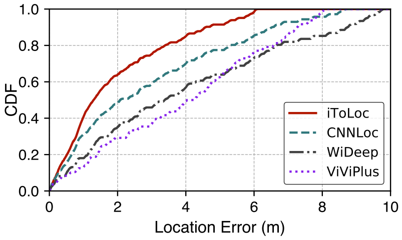


Fig. 9. Accuracy comparison

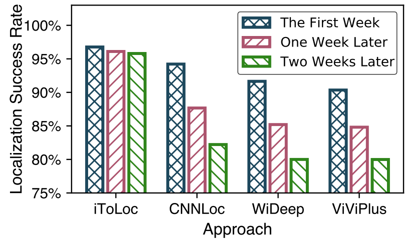


Fig. 10. Comparison in temporal robustness

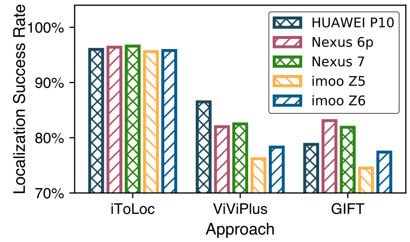


Fig. 11. Comparison in cross-device robustness

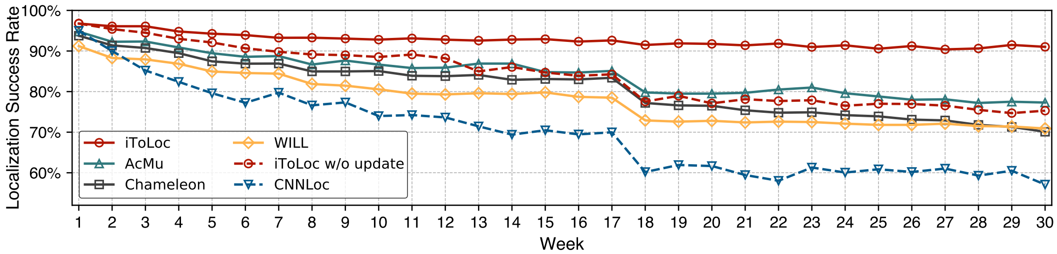


Fig. 12. Comparison in model update

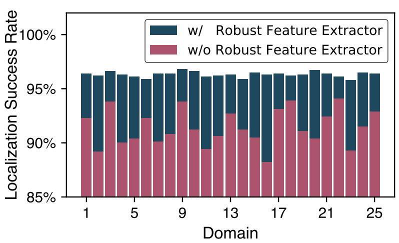


Fig. 13. Accuracy on each target domain

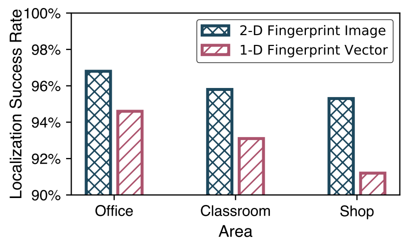


Fig. 14. Performance of 2-D fingerprint image

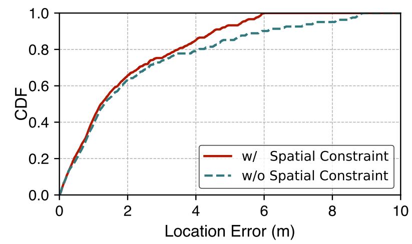


Fig. 15. Performance of spatial constraint

B. Performance Evaluation

1) Overall Performance Comparison: We first evaluate the performance of *iToLoc* without the model update in short-term localization scenarios. Fig. 9 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 1.92m, 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.

2) Performance Comparison with Different Time Interval: We also examine localization robustness in terms of temporal stability. We recollect fingerprints during the different time intervals and use the original fingerprint database without update for localization. Fig. 10 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 to resist the fingerprint temporal instability caused by wireless signal fluctuation.

3) 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 Fig. 11, 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 delightful performance under such a strict condition, demonstrating that *iToLoc* can tolerate device diversity much better than comparative methods.

4) 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. Fig. 12 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 that in the 18th week, due to the redecoration of the shopping mall, the performance of three comparative works reduces sharply. However, there is merely a negligible effect on the performance of the proposed *iToLoc*. 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.

C. Study of Core Components

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. Fig. 13 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.

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 2-D fingerprint image and 1-D fingerprint vector as their inputs, respectively. As shown in Fig. 14, by leveraging the generated image as input, the performance gain is 2.3%, 2.8%, and 4.3% in the office building, classroom building, and shopping mall, respectively. The delightful results demonstrate that the proposed fingerprint-image transformer will further promote *iToLoc* to extract domain-independent robust features.

3) Impact of Spatial Constraint: We further evaluate the spatial constraint, which is proposed to solve the frequent significant localization bias. As shown in Fig. 15, 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 1.86m and 95th percentile accuracy of 5.41m. The results demonstrate that the spatial constraint can effectively reduce large errors and improve overall localization performance.

VI. Related Work

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

Robust wireless fingerprint extractor. To enhance the localization robustness without degrading the delightful ubiquity of fingerprint based approaches. Various subsequent works tend to seek for robust expressive form of RSS fingerprints [13], [44]–[47]. In particular, recent innovations explore the spatial/temporal properties of fingerprints for localization. GIFT [42] defines a metric of binary differential value between RSS observed at two adjacent locations as replacements to the original RSS values as fingerprints; ViVi [2] and ViViPlus [18] 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. Some works are also equipped with deep learning models to enhance fingerprint matching. In [31] and [32], a CNN-based WiFi fingerprinting method was presented to pursue better performance. WiDeep [19] integrates a stacked auto-encoders deep learning model with a probabilistic framework to handle the noise and capture the complex relationship between the WiFi APs signals. Previous approaches [48] and [35] demonstrate that conclude fingerprint collection time and device as domain and leverage domain-adaptive transfer learning based frameworks will further boost the localization accuracy. Inspired by recent works, *iToLoc* leverages adversarial network to extract robust domain-independent representations.

Fingerprint database automatic update. Considering environmental dynamics, pioneer works including LiFs [12], UnLoc [49], WILL [20], AcMu [23], 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 [43], [50]. 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 leverage co-training based framework to automatically update the model, which has been demonstrated more effective to underpin a long-term localization service.

VII. Conclusion

In this paper, we propose *iToLoc*, a deep learning based localization system that achieves all three goals on delightful 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 8 different devices. The results demonstrate its superior performance over previous schemes. Trained once, locate anytime for anyone: *iToLoc* makes a great process towards fortifying WiFi fingerprint-based localization to a fully practical service for wide deployment.

Acknowledgment

This work is supported in part by the NSFC under grant 61832010, 61632008, 61872081, 61972131.

References

[1] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the ACM Mobisys*, 2005, pp. 205–218.

[2] C. Wu, J. Xu, Z. Yang, N. D. Lane, and Z. Yin, "Gain without pain: Accurate wifi-based localization using fingerprint spatial gradient," *Proceedings of the ACM IMWUT*, 2017.
[3] C. Wu, F. Zhang, Y. Fan, and K. J. R. Liu, "Rf-based inertial measurement," in *ACM SIGCOMM*, August 19-24 2019.
[4] C. Wu, F. Zhang, B. Wang, and K. R. Liu, "Easitrack: Decimeter-level indoor tracking with graph-based particle filtering," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 2397–2411, 2019.
[5] Z. Yang, Z. Zhou, and Y. Liu, "From rssi to csi: Indoor localization via channel response," *ACM Computing Surveys (CSUR)*, vol. 46, no. 2, pp. 1–32, 2013.
[6] C. Wu, Z. Yang, and Y. Liu, "Smartphones based crowdsourcing for indoor localization," *IEEE Transactions on Mobile Computing*, vol. 14, no. 2, pp. 444–457, Feb 2015.
[7] J. Wang and D. Katabi, "Dude, where's my card? rfid positioning that works with multipath and non-line of sight," in *ACM SIGCOMM*, 2013, pp. 51–62.
[8] L. Shangguan, Z. Yang, A. X. Liu, Z. Zhou, and Y. Liu, "Stpp: Spatial-temporal phase profiling-based method for relative rfid tag localization," *IEEE/ACM Transactions on Networking*, vol. 25, no. 1, pp. 596–609, 2016.
[9] K. Liu, X. Liu, and X. Li, "Guoguo: Enabling fine-grained indoor localization via smartphone," in *Proceeding of the ACM MobiSys*, 2013, pp. 235–248.
[10] J. Xu, H. Cao, D. Li, K. Huang, C. Qian, L. Shangguan, and Z. Yang, "Edge assisted mobile semantic visual slam," in *Proceedings of the IEEE INFOCOM*, 27-30 April 2020.
[11] E. Dong, J. Xu, C. Wu, Y. Liu, and Z. Yang, "Pair-navi: Peer-to-peer indoor navigation with mobile visual slam," in *Proceedings of the IEEE INFOCOM*, 29 April-2 May 2019.
[12] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: wireless indoor localization with little human intervention," in *Proceedings of the ACM Mobicom*, 2012, pp. 269–280.
[13] G. Shen, Z. Chen, P. Zhang, T. Moscibroda, and Y. Zhang, "Walkie-markie: Indoor pathway mapping made easy," in *USENIX NSDI*, 2013, pp. 85–98.
[14] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: Zero-effort crowdsourcing for indoor localization," in *Proceedings of the ACM Mobicom*, 2012, pp. 293–304.
[15] C. Wu, Z. Yang, and Y. Liu, "Smartphones based crowdsourcing for indoor localization," *IEEE Transactions on Mobile Computing*, vol. 14, no. 2, pp. 444–457, 2014.
[16] D. Lymberopoulos and J. Liu, "The microsoft indoor localization competition: Experiences and lessons learned," *IEEE Signal Processing Magazine*, vol. 34, no. 5, pp. 125–140, 2017.
[17] https://www.imoo.com/.
[18] J. Xu, Z. Yang, H. Chen, Y. Liu, X. Zhou, J. Li, and N. Lane, "Embracing spatial awareness for reliable wifi-based indoor location systems," in *2018 IEEE 15th International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*. IEEE, 2018, pp. 281–289.
[19] M. Abbas, M. Elhamshary, H. Rizk, M. Torki, and M. Youssef, "Wideep: Wifi-based accurate and robust indoor localization system using deep learning," in *2019 IEEE International Conference on Pervasive Computing and Communications (PerCom)*. IEEE, 2019, pp. 1–10.
[20] C. Wu, Z. Yang, Y. Liu, and W. Xi, "Will: Wireless indoor localization without site survey," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 4, pp. 839–848, 2012.
[21] H. Xu, Z. Yang, Z. Zhou, L. Shangguan, K. Yi, and Y. Liu, "Enhancing wifi-based localization with visual clues," in *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2015, pp. 963–974.
[22] X. Chen, H. Li, C. Zhou, X. Liu, D. Wu, and G. Dudek, "Fido: Ubiquitous fine-grained wifi-based localization for unlabelled users via domain adaptation," in *Proceedings of The Web Conference 2020*, 2020, pp. 23–33.
[23] C. Wu, Z. Yang, C. Xiao, C. Yang, Y. Liu, and M. Liu, "Static power of mobile devices: Self-updating radio maps for wireless indoor localization," in *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE, 2015, pp. 2497–2505.
[24] J. Xu, H. Chen, K. Qian, E. Dong, M. Sun, C. Wu, L. Zhang, and Z. Yang, "ivr: Integrated vision and radio localization with zero human effort," *Proceedings of the ACM IMWUT*, vol. 3, no. 3, pp. 1–22, 2019.
[25] Z. Yang, C. Wu, Z. Zhou, X. Zhang, X. Wang, and Y. Liu, "Mobility increases localizability: A survey on wireless indoor localization using inertial sensors," *ACM Computing Surveys (Csur)*, vol. 47, no. 3, pp. 1–34, 2015.
[26] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2096–2030, 2016.
[27] W. Jiang, C. Miao, F. Ma, S. Yao, Y. Wang, Y. Yuan, H. Xue, C. Song, X. Ma, D. Koutsonikolas *et al.*, "Towards environment independent device free human activity recognition," in *in Proceedings of the ACM Mobicom*, 2018, pp. 289–304.
[28] Z.-H. Zhou and M. Li, "Semi-supervised regression with co-training." in *IJCAI*, vol. 5, 2005, pp. 908–913.
[29] W. Dong-DongChen and Z.-H. WeiGao, "Tri-net for semi-supervised deep learning," in *Proceedings of twenty-seventh international joint conference on artificial intelligence*, 2018, pp. 2014–2020.
[30] M. Zhao, S. Yue, D. Katabi, T. S. Jaakkola, and M. T. Bianchi, "Learning sleep stages from radio signals: A conditional adversarial architecture," in *International Conference on Machine Learning*, 2017.
[31] J.-W. Jang and S.-N. Hong, "Indoor localization with wifi fingerprinting using convolutional neural network," in *Proceeding of the IEEE ICUFN*, 2018.
[32] X. Song, X. Fan, C. Xiang, Q. Ye, L. Liu, Z. Wang, X. He, N. Yang, and G. Fang, "A novel convolutional neural network based indoor localization framework with wifi fingerprinting," *IEEE Access*, vol. 7, pp. 110 698–110 709, 2019.
[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.
[34] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai *et al.*, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.
[35] Z. Sun, Y. Chen, J. Qi, and J. Liu, "Adaptive localization through transfer learning in indoor wi-fi environment," in *Seventh International Conference on Machine Learning and Applications*. IEEE, 2008.
[36] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.
[37] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proceedings of the eleventh annual conference on Computational learning theory*, 1998, pp. 92–100.
[38] M.-F. Balcan, A. Blum, and K. Yang, "Co-training and expansion: Towards bridging theory and practice," in *Advances in neural information processing systems*, 2005, pp. 89–96.
[39] W. Wang and Z.-H. Zhou, "A new analysis of co-training," in *ICML*, 2010.
[40] M.-F. Balcan and A. Blum, "A discriminative model for semi-supervised learning," *Journal of the ACM (JACM)*, vol. 57, no. 3, pp. 1–46, 2010.
[41] K. Saito, Y. Ushiku, and T. Harada, "Asymmetric tri-training for unsupervised domain adaptation," *arXiv preprint arXiv:1702.08400*, 2017.
[42] Y. Shu, Y. Huang, J. Zhang, P. Coué, P. Cheng, J. Chen, and K. G. Shin, "Gradient-based fingerprinting for indoor localization and tracking," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 4, pp. 2424–2433, 2015.
[43] S. He, B. Ji, and S.-H. G. Chan, "Chameleon: Survey-free updating of a fingerprint database for indoor localization," *IEEE Pervasive Computing*, vol. 15, no. 4, pp. 66–75, 2016.
[44] Y. Chen, Q. Yang, J. Yin, and X. Chai, "Power-efficient access-point selection for indoor location estimation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 7, pp. 877–888, 2006.
[45] S. He and S.-H. G. Chan, "Wi-fi fingerprint-based indoor positioning: Recent advances and comparisons," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 466–490, 2015.
[46] L. Li, G. Shen, C. Zhao, T. Moscibroda, J.-H. Lin, and F. Zhao, "Experiencing and handling the diversity in data density and environmental locality in an indoor positioning service," in *Proceedings of the ACM Mobicom*, 2014, pp. 459–470.
[47] P. Mirowski, P. Whiting, H. Steck, R. Palaniappan, M. MacDonald, D. Hartmann, and T. K. Ho, "Probability kernel regression for wifi localisation," *Journal of Location Based Services*, vol. 6, no. 2, pp. 81–100, 2012.
[48] S. J. Pan, V. W. Zheng, Q. Yang, and D. H. Hu, "Transfer learning for wifi-based indoor localization," in *AAAI*, vol. 6, 2008.
[49] H. Wang, S. Sen, A. Elgohary, M. Farid, M. Youssef, and R. R. Choudhury, "No need to war-drive: Unsupervised indoor localization," in *Proceedings of the ACM Mobisys*, 2012.
[50] S. He, W. Lin, and S.-H. G. Chan, "Indoor localization and automatic fingerprint update with altered ap signals," *IEEE Transactions on Mobile Computing*, vol. 16, no. 7, pp. 1897–1910, 2016.