2023 IEEE 20th International Conference on Mobile Ad Hoc and Smart Systems (MASS) | 979-8-3503-2433-4/23/$31.00 ©2023 IEEE | DOI: 10.1109/MASS58611.2023.00053

More Accuracy Less Fingerprints: Wi-Fi Indoor Localization via Generative Adversarial Networks

1st Jiayang Cui
College of Intelligence and Computing
Tianjin University
Tianjin, China
jycui@tju.edu.cn

2nd Yuefan Tong
College of Intelligence and Computing
Tianjin University
Tianjin, China
tttong@tju.edu.cn

3rd Zenghua Zhao*
College of Intelligence and Computing
Tianjin University
Tianjin, China
zenghua@tju.edu.cn

4th Jiankun Wang
College of Intelligence and Computing
Tianjin University
Tianjin, China
jiankunwang@tju.edu.cn

5th Mengling Ou
College of Intelligence and Computing
Tianjin University
Tianjin, China
oumengling@tju.edu.cn

6th Bin Wu
College of Intelligence and Computing
Tianjin University
Tianjin, China
binw@tju.edu.cn

***Abstract*—High-cost site survey is one of the bottlenecks of Wi-Fi fingerprinting indoor localization. Rather than leverage inertial sensors or floorplan, we propose LocGAN to generate virtual fingerprints (VFPs) with only a small number of labeled (with locations) and yet a large number of unlabeled ones. To this end, LocGAN is a semi-supervised deep generative model consisting of TriReg, encoder, generator, and discriminator. As a tri-net based regressor, TriReg provides pseudo-labels for unlabeled fingerprints. Under a generative adversarial network (GAN) framework, LocGAN is able to learn underlying distributions of fingerprints from both labeled and unlabeled ones thus generating high-accuracy VFPs. We also design several effective training strategies to further improve its performance. To evaluate LocGAN, we prototype a Wi-Fi indoor localization system based on it. Extensive experiments are carried out in real-world scenarios with areas over $8,200\,\mathrm{m}^2$. The experiment results demonstrate that compared with the state-of-the-art counterparts, LocGAN achieves more accuracy with less labeled fingerprints, reducing the cost of site survey significantly.**

***Index Terms*—Indoor localization, Wi-Fi fingerprinting, semi-supervised deep learning, deep generative model, generative adversarial network.**

I. Introduction

Wi-Fi fingerprinting indoor localization has attracted many researchers' attention in past decades due to pervasive deployments of access points (APs) and ubiquities of Wi-Fi enabled mobile devices (e.g., smartphones and smartwatches) [1]. Although fruitful works have been produced in academia [2]–[5], few have been deployed in real-world large-scale venues (e.g., shopping malls and airports) [6]. One primary reason is high cost of the site survey which constructs a fingerprint map by collecting fingerprints (Wi-Fi received signal strength indicator (RSSI) from APs) at reference points (RPs). Site survey usually needs professional work by trained workers [1]. Moreover, high-accuracy localization usually depends on high-density of RPs [7]. In this sense, conventional Wi-Fi fingerprinting localization systems achieve high accuracy at a high cost thus hindering their large-scale deployments.

To reduce the cost of site survey, many works construct fingerprint maps by crowdsourcing [8]–[13]. However, the crowdsourced fingerprints are collected without location information. In order to infer their locations, Inertial Measurement Unit (IMU) sensor readings are leveraged to produce users' trajectories [9]. Although IMU sensors are equipped with off-the-shelf smartphones, it is complicated to estimate walking distance and heading direction due to their low accuracy [14]. Moreover, the estimation error significantly impacts on the accuracy of localization. In addition, floorplans are also used to provide constraints for location inference [11]. However, floorplans are always unavailable in practice.

In contrast, we explore to trade high-cost site survey off low-cost crowdsourcing without losing localization accuracy. The basic idea is utilizing only a small number of fingerprints with location tags (labeled fingerprints) collected at sparse reference points, and yet a large number of crowdsourced ones (unlabeled fingerprints). To this end, we generate virtual fingerprints (VFPs) by a deep generative model (DGM) under a generative adversarial network (GAN) framework. DGMs have been successfully applied in many areas, including image generation and classification [15], [16], speech generation and recognition [17], to name a few. However, existing DGMs cannot be applied to generate VFPs directly, since most of them are supervised and focus on classification tasks. On one hand, a supervised model requires a lot of labeled data for training and cannot handle unlabeled ones. On the other hand, since locations are continuous rather than discrete, a classification model will impair the localization accuracy.

In this paper, we propose LocGAN, a semi-supervised deep generative model for Wi-Fi localization. LocGAN consists of TriReg, encoder, generator, and discriminator. To make use of unlabeled fingerprints, the regressor TriReg provides pseudo-labels for them. Cascaded to TriReg, the encoder and generator

Zenghua Zhao is the corresponding author. This work was supported by National Natural Science Foundation of China (NSFC) under Grant 61972283.

2155-6814/23/$31.00 ©2023 IEEE
DOI 10.1109/MASS58611.2023.00053

is able to learn underlying distributions of fingerprints from both labeled and unlabeled ones. Moreover, the generator and discriminator form a conditional GAN. The generator generates fake fingerprints to deceive the discriminator, while the discriminator tries to distinguish them from real ones. By adversarial training, the generator is driven to learn an effective representation of both labeled and unlabeled fingerprints. TriReg is also encouraged to improve accuracy of pseudo-labels, which in turn improves the performance of generator. Therefore, the generator is able to generate high-accuracy VFPs at given locations. In addition, we design several effective training strategies to further improve the performance.

To verify the effectiveness of LocGAN, we prototype a Wi-Fi localization system which uses the well-trained generator to generate VFPs. The abundant VFPs along with the limited number of labeled fingerprints compose the Wi-Fi fingerprint map. Protecting localization accuracy from impact of localization algorithm, a simple K-Nearest Neighbors (KNN) algorithm is used to estimate locations. Extensive experiments are carried out in real-world deployments, including an office building and a large-scale shopping mall, with a total area over 8,200 m^2. The performance of LocGAN is compared with that of well-known DGMs: CVAE [18], CVAE-GAN [16], and ACGAN [19], as well as that of representative Wi-Fi fingerprinting systems: DeepPrint [20], WiDeep [21], Modellet [7], and RADAR [22]. The results show that LocGAN achieves acceptable accuracy with extremely sparse RPs. Specifically, LocGAN yields 2.2 m median localization accuracy, with only 5 RPs in the office building, and 5.0 m median accuracy with RPs 60 m apart from each other in the shopping mall. At the same time, with a comparable accuracy, the number of RPs required by LocGAN decreases by $61.2\% - 84.4\%$ compared with counterparts, thus reducing fingerprint mapping cost significantly.

In summary, our contributions are two-fold:

- We propose LocGAN, a novel semi-supervised deep generative model under the GAN framework, to generate high-accuracy VFPs at a low cost. Unlike existing works using either site survey or crowdsourcing, LocGAN trades them off by leveraging DGMs, with only a small number of labeled fingerprints and yet a large number of unlabeled ones.
- We prototype LocGAN and carry out extensive experiments in real-world scenarios with a total area over 8,200 m^2. The results demonstrate that LocGAN achieves high localization accuracy with extremely sparse RPs, decreasing the cost of site survey significantly.

The rest of the paper is organized as follows. Section IV reviews the related work. The design of LocGAN is illustrated in Section II. We evaluate the performance of LocGAN in Section III, and then conclude the paper in Section V.

II. Design of LocGAN

The goal of LocGAN is to achieve high localization accuracy yet at a low cost by utilizing a small number of labeled fingerprints and a large number of unlabeled ones, without extra information (e.g., IMU sensor measurements and floorplans). However, it is challenging to generate high accuracy virtual fingerprints with extremely sparse reference points (e.g., 60 m apart from each other). Even though there are numerous unlabeled fingerprints, they are collected randomly by crowdsourcing without location information. To tackle the above challenge, we design LocGAN and present its detail in this section.

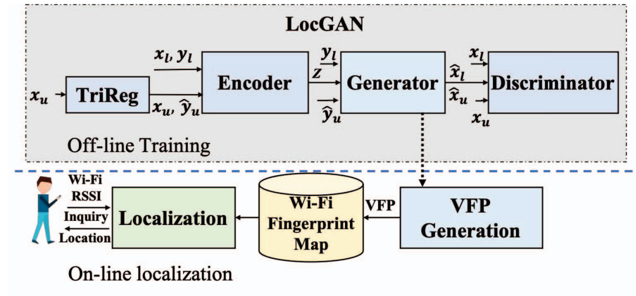


Fig. 1. The framework of the Wi-Fi localization system based on LocGAN.

A. Overview

Let $x = (rssi_1, rssi_2, \ldots, rssi_n), x \in \mathbb{R}^n$, represent Wi-Fi RSSIs received from different APs at location $y \in \mathbb{R}^2$, where n is the total number of APs heard in the area of interest (AoI). Denote x_l as a labeled Wi-Fi fingerprint with location (label) y_l, and x_u the unlabeled fingerprint without location information.

LocGAN is a semi-supervised deep generative model, which generates VFPs at any locations in the AoI with a small number of labeled fingerprints and a large number of unlabeled ones. As illustrated in Fig. 1, it consists of regressor, encoder, generator, and discriminator. After being well-trained off line, the generator generates VFPs at dense locations. The dense VFPs and sparse real FPs (RFPs) constitute the Wi-Fi fingerprint map. During online localization, the inquiry of Wi-Fi signals are input to the localization module, and then the location is fed back to users. We describe each components of LocGAN in detail as follows.

B. TriReg: Tri-Net Based Regressor

To enable generator to learn underlying distributions of unlabeled fingerprints, regressor TriReg provides pseudo-labels for them. To this end, TriReg adopts a semi-supervised model inspired by tri-net [23]. Since tri-net is specified for classification tasks, we design new diverse data augmentation and pseudo-label editing for TriReg. Before diving into the detail, we first introduce its components.

As illustrated in Fig 2, TriReg consists of a shared module R_s and three parallel modules R_1, R_2 and R_3. The shared module and one module $R_i, i \in [1, 3]$ forms a sub-model M_i. For fingerprint x, its pseudo-label $\hat{y}$ is the average of

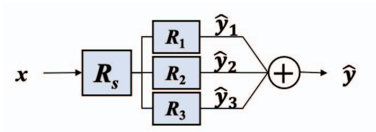


Fig. 2. Diagram of TriReg.

predictions from three sub-models. This process can be written as

$$\hat{y} = R(x) = \frac{1}{3}\sum_{i=1}^{3}\hat{y}_i, \tag{1}$$

where R is the function of TriReg, and $\hat{y}_i$ is the prediction from sub-model M_i. Then $\hat{y}_i$ is expressed as:

$$\hat{y}_i = R_i(R_s(x)). \tag{2}$$

Diverse Data Augmentation. Since the three sub-models M_1, M_2 and M_3 predict pseudo labels independently, it is important to keep them diverse. To do so, their training datasets should be different from each other. However, the number of labeled fingerprints is too limited to provide three diverse datasets. Therefore, the labeled fingerprints have to be augmented. Tri-net augments data by injecting random noises to labels which does not change the ground truth of classification labels (one-hot vectors). However, such augmentation approach cannot be used in TriReg, since it will change the ground truth of regression labels (continuous values).

To address the data augmentation, TriReg injects noises to fingerprints rather than their labels. Specifically, after generator is pre-trained on labeled fingerprints, TriReg uses the generator to generate pseudo-fingerprints at reference points. Even though pseudo-fingerprints are noisy, they share the same distribution with real ones. In this way, labeled fingerprints are augmented with pseudo-fingerprints thus making training datasets of three sub-models diverse. The three sub-models are pre-trained on the diverse training sets.

Pseudo-Label Editing. After pre-training, the three sub-models are trained jointly to improve accuracy of TriReg. During joint training, unlabeled fingerprints are used enabling TriReg to learn underlying distributions of them. Since unlabeled fingerprints do not have ground-truth locations, three sub-models cooperate to edit reliable pseudo-labels for them.

In particular, if any two of sub-models have the same prediction on one unlabeled fingerprint and the prediction is confident and stable, then the two sub-models will teach the third one on this fingerprint. The fingerprint with pseudo-label predicted by the two sub-models is added to the training set of the third. Then the third sub-model is refined with the augmented training set.

Different from tri-net, the prediction of TriReg is continuous rather than discrete. Therefore, we have to redefine decision rules of *same prediction*, *confident prediction*, and *stable prediction* for TriReg. We still use dropout technique in pseudo-label editing [23]. Denote M_i, M_j, and $M_k(i \neq j \neq k, i, j, k \in [1, 3])$ as any one of the sub-models. For unlabeled fingerprint x_u, its prediction by sub-model M_i with or without dropout is represented as $\tilde{y}_i$ and $\hat{y}_i$, respectively. Let σ be a small value. Then, for an unlabeled fingerprint x_u, we redefine the decision rules as follows.

- **Same prediction.** M_j and M_k have the same prediction for x_u, when $||\hat{y}_j - \hat{y}_k|| < \sigma$.
- **Confident prediction.** M_j and M_k have confident prediction for x_u, when they have the same prediction and the prediction is different from that of M_i, i.e., $||\hat{y}_j - \hat{y}_k|| < \sigma$, $||\hat{y}_i - \hat{y}_j|| > \sigma$, and $||\hat{y}_i - \hat{y}_k|| > \sigma$.
- **Stable prediction.** M_j and M_k have stable prediction for x_u, when they have same prediction with dropout for Q times, and the prediction is the same with that predicted without dropout, i.e., $||\hat{y}_{j,q} - \hat{y}_{k,q}|| < \sigma$, $||\hat{y}_j - \tilde{y}_{j,q}|| < \sigma$, $||\hat{y}_k - \tilde{y}_{k,q}|| < \sigma, q = 1, ..., Q$, where $\tilde{y}_{j,q}$ is the qth prediction of M_j with dropout.

For x_u, if M_j and M_k have confident and stable prediction, then its pseudo-label $\hat{y}_u$ is the average. That is

$$\hat{y}_u = \frac{1}{2}[\hat{y}_j + \hat{y}_k]. \tag{3}$$

The tuple of $(x_u, \hat{y}_u)$ will be added to the training set of M_i. The augmented training set of M_i is denoted as $\mathcal{PL}_i = \{(x_{pl,i}, y_{pl,i})\}$, $pl \in [1, N_i]$), N_i is the size of $\mathcal{PL}_i$.

Loss Function of TriReg. The loss function of TriReg for labeled fingerprint (x_l, y_l) is calculated as:

$$\mathcal{L}_R^l = ||\hat{y}_l - y_l|| + \frac{1}{3}\sum_{i=1}^{3}||\hat{y}_{l,i} - y_l||, \tag{4}$$

where $\hat{y}_l$ is the pseudo-label of x_l, and $\hat{y}_{l,i}$ is the prediction for x_l from sub-model M_i.

For pseudo-labeled fingerprints $(x_{pl,i}, y_{pl,i}) \in \mathcal{PL}_i(i \in [1, 3])$, the loss function of TriReg is calculated as:

$$\mathcal{L}_R^u = \frac{1}{3}\sum_{i=1}^{3}\mathbb{E}_{(x_{pl,i}, y_{pl,i})\sim\mathcal{PL}_i}||\hat{y}_{pl,i} - y_{pl,i}||. \tag{5}$$

C. Encoder and Generator

The encoder and generator constitute a conditional variational autoencoder (CVAE) [18]. The encoder seeks to represent Wi-Fi fingerprint x with location y in a latent variable space $\mathbb{Z}$. Meanwhile, the generator aims to reconstruct the original x from $\mathbb{Z}$ conditioned on location y. Let $\hat{x}$ be the reconstructed fingerprint, E and G are functions of encoder and generator. Then the process can be represented as:

$$\begin{aligned} z &= E(x, y), z \in \mathbb{Z}, \\ \hat{x} &= G(z, y) = G(E(x, y), y). \end{aligned} \tag{6}$$

Formally, generator generates virtual Wi-Fi fingerprints $\hat{x}$ conditioned on location y from distribution $p_\theta(x|z, y)$, where z is a latent variable drawn from prior distribution $p(z)$. Usually, the posterior distribution $p_\theta(z|x, y)$ is intractable. Fortunately, the parameters of G can be estimated efficiently in the framework of SGVB (Stochastic Gradient Variational Bayes) [24]. In particular, the posterior distribution can be approximated by a distribution $q_\phi(z|x, y)$. It is possible to use a high-capacity model for $q_\phi(z|x,y)$ to well match $p_\theta(z|x,y)$ [25]. As a result, the intractable $p_\theta(z|x,y)$ becomes tractable since we can just use $q_\phi(z|x,y)$ to approximate it. $q_\phi(z|x,y)$ is thus known as the encoder E.

Loss Function of Encoder and Generator. For a fingerprint x with location y, the loss function of this part $\mathcal{L}_{EG}$ is written as:

$$\begin{aligned}\mathcal{L}_{EG} =& KL[q_\phi(z|x,y)||p(z)] \\ &- \mathbb{E}_{q_\phi(z|x,y)}[\log p_\theta(x|z,y)],\end{aligned} \tag{7}$$

where $KL(\cdot)$ is Kullback-Leibler divergence [26].

Note that with the help of TriReg, the input fingerprint x can be labeled x_l with label y_l or unlabeled x_u with pseudo-label $\hat{y}_u$. On one hand, the number of labeled fingerprints is very limited, and the RPs are sparsely distributed in the AoI. With only labeled fingerprints, it is difficult for generator to learn the underlying distributions of Wi-Fi RSSIs in that area. On the other hand, unlabeled fingerprints provide more information due to their large number and full-area coverage. Therefore, learning from both labeled and unlabeled fingerprints, the generator is able to generate high-accuracy VFPs.

D. Generator and Discriminator

Noisy pseudo-labeled fingerprints will degrade the performance of the generator. Meanwhile, even though utilizing three sub-models, the performance of TriReg is still limited by the small number of labeled fingerprints. Therefore, LocGAN adopts the framework of GAN [15]. The basic idea is that by adversarial training the generator is able to generate VFPs with high accuracy so that the discriminator cannot distinguish them from real ones. At the same time, TriReg is driven to produce pseudo-labels with little noises for the generator.

To this end, the generator and discriminator form a conditional GAN [27]. Different from a regular GAN which uses random variables as input, the generator is cascaded after an encoder to improve training efficiency. Moreover, TriReg is also able to cascade on this line and benefits from the adversarial training.

Specifically, generator generates fingerprints $\hat{x}_l$ at RP y_l, and $\hat{x}_u$ at its pseudo location $\hat{y}_u$. The generated fingerprints $\hat{x} \in \{\hat{x}_l\} \cup \{\hat{x}_u\}$ as well as real fingerprints $x \in \{x_l\} \cup \{x_u\}$ are input to the discriminator. The discriminator tries to distinguish the generated fingerprints from real ones.

Loss Function of Generator and Discriminator. Let G and D be the function of the generator and discriminator, respectively. The adversarial loss function of generator $\mathcal{L}_{GD}$ is expressed as:

$$\mathcal{L}_{GD} = \mathbb{E}_{z\sim p(z),y\sim Q(y)}[D(\hat{x})^2]. \tag{8}$$

The adversarial loss function of discriminator, $\mathcal{L}'_{GD}$ is written as:

$$\begin{aligned}\mathcal{L}'_{GD} =& \mathbb{E}_{x\sim P(x)}[D(x)^2] \\ &+ \mathbb{E}_{z\sim p(z),y\sim Q(y)}[(1-D(\hat{x})^2],\end{aligned} \tag{9}$$

where $\hat{x}$ is generated from latent variable z from prior distribution $p(z)$, conditioned on location y from prior distribution $Q(y)$.

By minimizing the loss functions in an adversarial way, discriminator tries to distinguish VFPs from real ones. In turn, deceiving discriminator generator is driven to generate VFPs very close to real ones. TriReg is also encouraged to produce high-accuracy pseudo-labels for generator.

E. Objective and Training Strategies

Finally, the total loss function of LocGAN is

$$\mathcal{L}_{total} = \lambda_R(\mathcal{L}_R^l + \mathcal{L}_R^u) + \mathcal{L}_{EG} + \lambda_{GD}\mathcal{L}_{GD}, \tag{10}$$

where λ_R, λ_{GD} are weights to adjust different parts in the loss function.

1) Training Strategies: Recall that there are only a small number of labeled fingerprints and yet a large number of unlabeled ones. The large number of unlabeled fingerprints makes generator hard to generate high accuracy fingerprints due to their noisy pseudo-labels produced by TriReg. To reduce the prediction error of TriReg and improve the performance of generator, we design three training strategies: pre-training, TriReg training, and adversarial training.

Pre-Training. The components of a model are pre-trained on limited number of labeled data before joint training. After pre-training, the components are initialized in a supervised way. Moreover, parameters learnt in the pre-training are saved and used in joint training. Compared with random initialization, pre-training enables joint-training to converge more quickly and achieve better performance. Therefore, pre-training is adopted broadly in semi-supervised learning.

In this paper, we pre-train encoder and generator with labeled fingerprints. However, the number of labeled fingerprints is too small to pre-train TriReg due to its requirement of diverse data set. Therefore, we first augment the labeled fingerprint set using diverse data augmentation approach in Sec. II-B. Basically, at reference points the pre-trained generator generates pseudo-fingerprints which are added to the set of labeled fingerprints. TriReg is then pre-trained on the augmented data set.

TriReg Training. After pre-training, TriReg joins adversarial training with other components in LocGAN. Due to its complex training approach, we describe it here.

Avoiding overfitting, apart from labeled ones TriReg is still trained on unlabeled fingerprints. Pseudo-label editing approach (please refer to Sec. II-B) is adopted to select confident and stable pseudo-labels to further augment the training set of each sub-model. Generally, if any two of sub-models have the same prediction on one unlabeled fingerprint and the prediction is confident and stable, then the unlabeled fingerprint and its pseudo-label predicted by the two sub-models will be added to the training set of the third sub-model. Although being confident and stable, the pseudo-labels are still noisy. To control the intensity of the noise, for a batch of data only one sub-model uses the augmented training set while the other two still use the labeled fingerprint set. As batches are iterated during training, each sub-model is trained in turn on augmented training set thus keeping diverse.

Adversarial Training. The adversarial training process is detailed in Algorithm 1. Generally, we joint encoder, generator and TriReg to combat discriminator by alternately updating parameters of them. Let $\theta_E, \theta_G, \theta_R, \theta_D$ be the parameters of encoder, generator, TriReg, and discriminator respectively. We first fix $\theta_E, \theta_G, \theta_R$ and update θ_D with loss function of $\mathcal{L}'_{GD}$ in Eq 8, and then fix θ_D and update $\theta_E, \theta_G, \theta_R$ with the total loss function of $\mathcal{L}_{total}$. After adversarial training, the generator is well-trained and able to generate high-accuracy VFPs.

Algorithm 1 Adversarial training algorithm for LocGAN

Input: Labeled fingerprints dataset $\{X_l, Y_l\}$,
unlabeled fingerprints dataset $\{X_u\}$,
prior distribution $P(y)$
Parameter: λ_R, λ_{GD}, batch size B_l, B_u, training epoch T
Initialize: $\theta_E, \theta_G, \theta_R, \theta_D$
Output: θ_G

```
for epoch = 1 : T do
    for batch = 1 : N_l/B_l do
        Sample B_l labeled fingerprints {x_l, y_l}, sample B_u unlabeled fingerprints {x_u}, let {x_r} = {x_l} ∪ {x_u}.
        Let B = B_l + B_u, sample B labels {y} from P(y), sample B random variables {z} from P(z), generate {x_f} by G(z, y).
        Update θ_D by descending gradient:
        ∇_{θ_D} (1/B) Σ_b L_GD(x_r^(b), x_f^(b))
        Update θ_E, θ_G, θ_R by descending gradient:
        ∇_{θ_E,θ_G,θ_R} (1/B_l) Σ_b L^l_EG(x_l^(b), y_l^(b)) + (1/B_u) Σ_b L^u_EG(x_u^(b), ŷ_u^(b)) + (λ_R/B_l) Σ_b L^l_R(x_l^(b), y_l^(b)) + (λ_R/B_u) Σ_b L^u_R(x_u^(b)) + (λ_GD/B) Σ_b L'_GD(x_f^(b))
    end for
end for
return θ_G
```

F. VFP Generation and Online Localization

After training LocGAN, we use the well-trained generator to generate VFPs at dense locations except reference points in the AoI. Given a location $y_i \in \mathbb{R}^2$, VFPs are generated from random variables $z \in \mathbb{Z}$ with a total number of N_i. The VFP at location y_i is the average of N_i VFPs. Then VFPs and RFPs are combined to construct a high-density Wi-Fi fingerprint map.

Avoiding impacts on the performance of LocGAN, the localization module adopts a simple KNN algorithm. When a user needs a location service, the localization module matches the inquired Wi-Fi RSSI vector against the fingerprint map generated by LocGAN and return the location to the user.

III. Implementation and Evaluation

We implement LocGAN and deploy a Wi-Fi localization system based on it in real-world indoor environments. In this section, we evaluate the performance of LocGAN and the localization accuracy of the system. The results are compared with that of the state-of-the-art DGMs and representative Wi-Fi localization systems.

A. Implementation

The sensing functionality of the system is implemented on Android smartphones. The training, generation, and localization are deployed at a server with CPU Intel Core Processor I7-7700K and GPU GeForce GTX 1080Ti. We implement LocGAN on the deep-learning platform of Pytorch 1.4.0 [28].

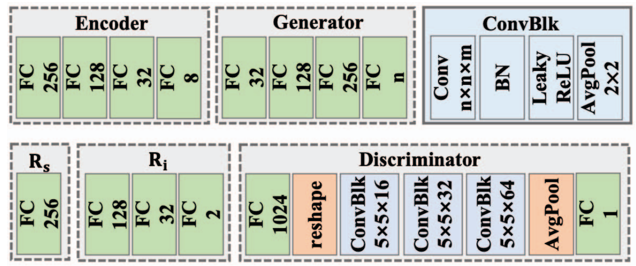


Fig. 3. Architectures of deep neural networks in LocGAN. n in G: the number of APs. m in ConvBlk: the number of convolution kernels. $n \times n$ in ConvBlk: convolution kernel size.

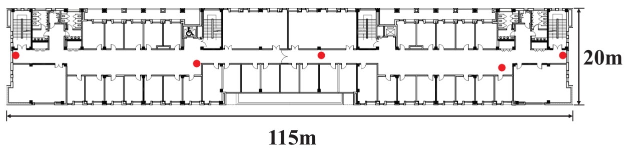
Fig. 3 illustrates the network architectures of LocGAN. Encoder and generator adopt DNNs with four FC (Full-Connection) layers. For TriReg, R_s is a shallow neural network with one FC layer and R_i $(i \in [1, 3])$ are DNNs with three FC layers. Whereas, discriminator uses CNNs (Convolutional Neural Networks) due to their powerful capability in feature extraction [29]. Following the processing of sensor data in [30], we first transform the input Wi-Fi RSSI vector into a vector with a size of 1024 by an FC layer, and then reshape it to a 32×32 matrix, followed by three ConvBlks (CNN Blocks). Each ConvBlk includes a convolutional layer, BN (Batch Normalization) [31], LeakyReLU activation function, and a max-pooling layer. The pre-training of TriReg uses Adadelta [32] as the optimizer which dynamically adjusts the learning rate to speed up the convergence, while the other components use Adam [33] to stabilize the training process.

B. Experiment Setup

We conduct extensive experiments in two typical indoor scenarios: an office building and a shopping mall, as shown in Fig. 4. The total size of tested areas is over $8,200$ m^2. The RPs are very sparse (dots in the figure), with 5 RPs in the office building, and 33 RPs in the mall. They are about 60 m apart from each other along the corridor. Test points are disjoint with RPs. Labeled fingerprints are collected at RPs for 60 s per point at a sampling rate of 5 Hz. Fingerprints at test points are used for testing. Whereas, unlabeled fingerprints are collected by random walking.

Benchmarks. We use four state-of-the-art DGMs and four representative Wi-Fi fingerprinting localization systems as benchmarks.

The four DGMs are SCVAE [20], CVAE [18], AC-GAN [19], and CVAE-GAN [16]. SCVAE is a semi-supervised



(a) Office building

(b) Shopping mall

Fig. 4. Floor plans in experiment scenarios.

CVAE designed for Wi-Fi fingerprint generation. CVAE, CVAE-GAN, and ACGAN are designed for images. Therefore, we modify them fit for Wi-Fi fingerprints. Specifically, we change their neural networks from CNNs to DNNs except for discriminator (if there is), and alter classification to regression.

The four systems are DeepPrint [20], WiDeep [21], Modellet [7], and RADAR [22]. The first two are based on deep-learning models. While deepPrint generates VFPs using SCVAE, WiDeep is based on stacked denoising autoencoders. In addition, Modellet combines model-based and fingerprint-based approaches, generating VFPs to improve localization accuracy. RADAR is a seminal Wi-Fi fingerprinting localization system.

C. Accuracy of VFPs generated by LocGAN

We carry out experiments in the shopping mall to evaluate the accuracy of VFPs generated by LocGAN, and compare the results with that of other DGMs. Root mean square errors (RMSE) between real fingerprints and VFPs at test points are calculated to denote the accuracy.

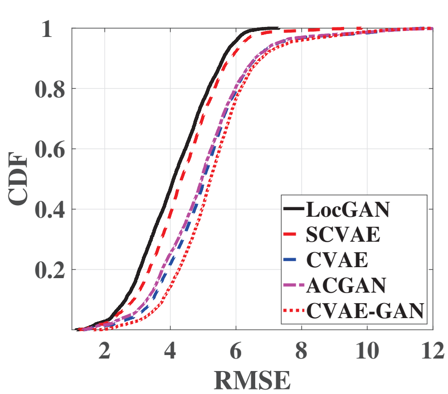


Fig. 5. The error CDF of VFPs.

We plot CDF of RMSE of VFPs in Fig. 5. It shows that LocGAN outperforms other DGMs. This contributes to effective semi-supervised learning techniques adopted by LocGAN.

LocGAN makes full use of lots of unlabeled fingerprints to learn the underlying distribution of Wi-Fi RSSI. Therefore, it generates VFPs with higher accuracy than CVAE and CVAE-GAN which are supervised leveraging only labeled fingerprints. SCVAE is semi-supervised yet without GAN which degrades its performance. Even though CVAE-GAN employs adversarial training, without unlabeled fingerprints its generation accuracy is still comparable to CVAE.

Although ACGAN is also semi-supervised, LocGAN still outperforms it. This is because that ACGAN assumes that training discriminator for classification helps improve capabilities of discriminator and generator. However, this assumption holds when there are abundant labeled data. Therefore, in the scenarios with a small number of labeled fingerprints, LocGAN achieves higher accuracy than ACGAN.

D. Localization Accuracy

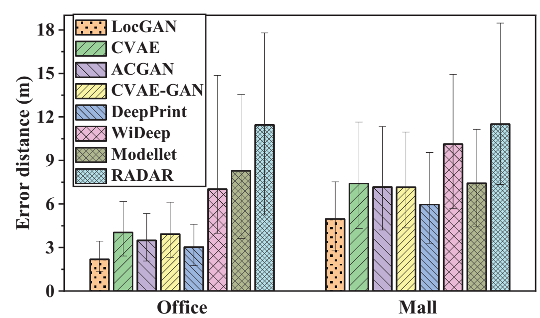


Fig. 6. Localization Accuracy.

To compare the localization accuracy of DGMs, we use a simple KNN algorithm to estimate locations based on fingerprint maps generated by them. The experiment results are shown in Fig. 6. We can see that LocGAN always outperforms other DGMs in both scenarios. In particular, in the office building LocGAN achieves median accuracy of 2.2 m with only 5 RPs, much more accurate than others. Even in the mall with a complex environment, LocGAN still yields median accuracy of 5.0 m with very sparse RPs (60 m apart from each other). Compared with LocGAN, the median error of other DGMs increase by more than 0.8 m and 1.0 m in two scenarios. This is due to the high accuracy of VFPs generated by LocGAN.

As illustrated in Fig. 6, LocGAN outperforms other localization systems too. Compared with RADAR, LocGAN reduces the median error by 9.3 m in the office building, and by 6.6 m in the shopping mall. With extremely sparse RPs, the number of candidate neighbors drops drastically thus degrading the accuracy significantly. Compared with Modellet, LocGAN reduces the median error by 6.1 m and 2.5 m in two scenarios. Although Modellet also generates VFPs, it does not utilize unlabeled data. In addition, there are only 5 RPs in the office building, the number of equations that can be established for Modellet is greatly reduced, so its accuracy is restricted. Compared with WiDeep, LocGAN reduces the median error by $4.8\,\mathrm{m}$ and $5.2\,\mathrm{m}$ in two scenarios. This is because WiDeep relies heavily on labeled fingerprints. Although WiDeep can achieve high accuracy with abundant labeled fingerprints, its accuracy degrades seriously with sparse labeled fingerprints. Although DeepPrint also leverages unlabeled fingerprints, without adversarial training it still suffers from poor accuracy. In short, LocGAN outperforms all the DGMs and Wi-Fi localization systems tested.

E. Effectiveness Analysis of LocGAN Components

To generate high-accuracy VFPs, we design several components to the standard GAN. We then carry out an ablation study to show the effectiveness of each component. In particular, except generator and discriminator, we delete other components one by one, keeping all parameters (e.g., the architecture of neural networks) and data the same as that in LocGAN. The benchmarks are as follows.

- **GD+E+R′.** To show the effectiveness of TriReg, this approach uses a regular regressor instead of TriReg.
- **GD+E.** To show the effectiveness of semi-supervised learning enabled by TriReg, this approach deletes the regressor.
- **GD.** To show the effectiveness of the encoder, this approach only keeps the primary components of a conditional GAN: generator and discriminator.

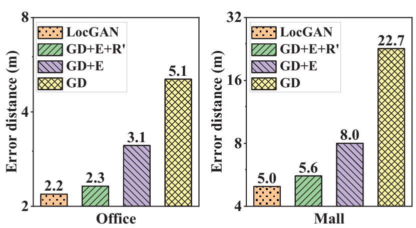


Fig. 7. Ablation study.

We plot median errors in Fig. 7. To show differences in accuracy clearly, the figure is plotted in log scale along the y-axis. In general, LocGAN outperforms all benchmarks in both scenarios.

Effectiveness of TriReg. Compared with GD+E+R′, the median error of LocGAN decreases by $0.1\,\mathrm{m}$ in the office building, and $0.6\,\mathrm{m}$ in the shopping mall. TriReg reduces the uncertainty of pseudo-labels through diverse data augmentation and pseudo-label editing, thereby improving the accuracy of regression and that of the LocGAN.

Effectiveness of Semi-Supervised Learning. Compared with GD+E, GD+E+R′ reduces the median error by $0.8\,\mathrm{m}$ in the office building and $2.4\,\mathrm{m}$ in the shopping mall. This is due to the introduction of the regressor which enables generator to learn from unlabeled fingerprints thus improving the accuracy of VFPs.

Effectiveness of Encoder. Compared with GD, the median error of GD+E decreases by $2.0\,\mathrm{m}$ in the office building and $14.7\,\mathrm{m}$ in the shopping mall. Instead of inputing random variables, with encoder GD+E is able to learn underlying distributions of the fingerprints, which is also in accordance with [16].

This is because it is very inefficient for GD to establish the relationship between fingerprints and locations by only adversarial loss. The introduction of the encoder brings the reconstruction loss, so that the relationship between the fingerprints and the locations can be established more directly and efficiently.

F. Effectiveness of Pre-training

We pre-train generator, encoder and TriReg before adversarial training. Moreover, TriReg is pre-trained with pseudo fingerprints generated by generator avoiding overfitting on the small number of labeled fingerprints. To show the effectiveness of pre-training, we compare it with the following two approaches:

- **noPre.** LocGAN is trained without pre-training.
- **noPseudoPre.** LocGAN is pre-trained yet without pseudo-labeled fingerprints.

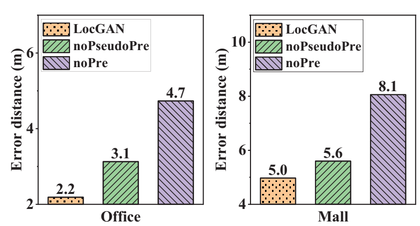


Fig. 8. Localization accuracy with different pre-training strategies.

Results are shown in Fig. 8. Without pre-training, noPre suffers poor localization accuracy in both scenarios. With pre-training, even though only labeled fingerprints are used, noPseudoPre achieves better accuracy than noPre. With pseudo fingerprints in pre-training, LocGAN gives the best performance. The results show that pre-training is effective and benefits the adversarial training followed.

G. Cost Efficiency.

LocGAN achieves comparable localization accuracy with extreme sparse RPs. To show its cost efficiency, we calculate the decreasing rate of the number of RPs used by LocGAN, Modellet, WiDeep, and RADAR at comparable localization accuracy. Suppose n_1 is the number of RPs used by LocGAN, and n_2 the number of RPs used by a counterpart at comparable accuracy. Then the decreasing rate r is calculated by $r = (1 - n_1/n_2) \times 100\%$. From Fig. 9, we can see that compared with other Wi-Fi localization systems, LocGAN decreases the number of RPs by $61.2\% - 84.4\%$ at comparable accuracy.

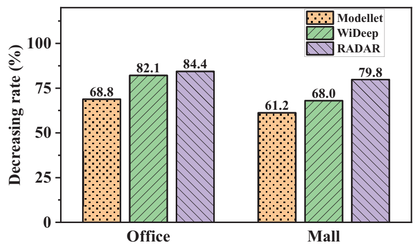


Fig. 9. The decreasing rate of the number of RPs used by LocGAN compared with counterparts at comparable localization accuracy.

The results show that LocGAN achieves high accuracy at a low cost.

IV. Related Work

Conventional Fingerprinting Approaches. In fingerprinting-based localization, a location is estimated based on a fingerprint map through deterministic [22] or probabilistic algorithms [34]. With a high-quality fingerprint map, these approaches can obtain high localization accuracy. However, site survey of the fingerprint map is very time-consuming and labor-intensive. The state-of-the-art Modellet [7] integrates RADAR and EZPerfect [4] to improve localization accuracy in large-scale venues by generating VFPs with EZPerfect. Different from them, LocGAN builds upon recent advances of deep generative models, and explores a novel deep-learning model to construct high-accuracy fingerprint map without site survey.

Crowdsourcing-based Approaches. Fingerprint crowdsourcing has been promoted to relieve the burden of site survey by allowing unprofessional users to participate in fingerprint collection [12], [35]. However, the crowdsourced fingerprints are not annotated with location information. To handle that, LiFS [9] and WILL [36] create a fingerprint space with crowdsourced fingerprints and IMU measurements, mapping fingerprints to real locations. Zee [37] uses indoor floor plan constraints (turns, walls, etc.) to correct user trajectories. Walkie-Marki [11] leverages WiFi-Marks as landmarks to align crowdsourced trajectories into a corridor map. LiPhi [10] leverages transportable Laser-Range Scanners (LRSs or LiDARs) in a user-transparent way to tag Wi-Fi scans. In [8], subarea fingerprints are constructed from crowdsourced RSSI measurements and related to indoor layouts. Despite all these efforts, many crowdsourced approaches still suffer from the low-accuracy of fingerprint annotation in large-scale venues. In contrast, LocGAN trades off between high-cost site survey and low-cost crowdsourced fingerprints, achieving acceptable localization accuracy yet at low cost.

Deep-Learning-based Approaches. Deep Learning is a powerful machine learning paradigm rising recently [38]. Various deep learning models have been investigated to estimate the location of Wi-Fi RSSI received [3], [21], [39]–[42]. They achieve higher accuracy than conventional approaches by leveraging the powerful fitting ability of neural networks. Besides, Gan *et al.* [43] use the LDPL model and ray tracing method to construct a large sample data for weights training. Liu *et al.* [44] propose a Tensor-GAN to generate new fingerprints as training samples, with a regressor estimating locations for radio frequency fingerprints. Unfortunately, these approaches depend on dense labeled fingerprints to achieve high localization accuracy. To tackle this problem, DeepMap [45] employs a deep Gaussian process for fingerprint map construction and location estimation. However, DeepMap does not make use of unlabeled fingerprints to improve construction accuracy. Albeit inspiration, we propose LocGAN, a semi-supervised deep learning model, which leverages lots of unlabeled fingerprints to reduce the cost of site survey.

Deep Generative Model. Our work is close to DGMs. There are two mainstream DGMs, Variational AutoEncoder (VAE) [24] and GAN [15]. VAE derives a lower bound on the marginal likelihood of the model by introducing a latent variable. GAN estimates the data distribution via an adversarial process between a generator and a discriminator. Moreover, by introducing conditions, DGMs can also generate samples with required types [46]. For example, CVAE [18] and CGAN [27] generate structured output representation according to requirements. In situations that lacks or is hard to obtain labeled data, unlabeled data are leveraged to achieve compatible performance by semi-supervised learning. ACGAN [19] is such a semi-supervised DGM. In addition, CVAE-GAN [16] combine a variational auto-encoder with a generative adversarial network to improve performance. Inspired by these appealing works, our LocGAN integrates VAE and GAN in a semi-supervised framework. Different from existing DGMs, the design of LocGAN is specified for Wi-Fi fingerprints, which model VFP generation as a semi-supervised regression task.

V. Conclusion

We propose LocGAN, a semi-supervised deep generative model, to achieve high localization accuracy yet at a low cost. Unlike existing works, LocGAN chooses to trade off between high-cost site survey and low-cost crowdsourcing by leveraging DGMs under the GAN framework. Extensive real-world experiments in large-scale shopping mall and typical office building verify its effectiveness. The localization system based on LocGAN achieves acceptable accuracy with a small number of labeled fingerprints and a large number of unlabeled ones, thus reducing the cost of site survey significantly.

References

[1] 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, Firstquarter 2016.

[2] O. Hashem, M. Youssef, and K. A. Harras, “Winar: Rtt-based sub-meter indoor localization using commercial devices,” in *2020 IEEE International Conference on Pervasive Computing and Communications*, ser. PerCom ’20, 2020, pp. 1–10.

[3] P. Roy and C. Chowdhury, “A survey of machine learning techniques for indoor localization and navigation systems,” *Journal of Intelligent & Robotic Systems*, vol. 101, no. 3, pp. 1–34, 2021.

[4] R. Nandakumar, K. K. Chintalapudi, and V. N. Padmanabhan, "Centaur: Locating devices in an office environment," in *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '12. New York, NY, USA: ACM, 2012, pp. 281–292.

[5] P. Davidson and R. Piché, "A survey of selected indoor positioning methods for smartphones," *IEEE Communications Surveys Tutorials*, vol. 19, no. 2, pp. 1347–1370, Secondquarter 2017.

[6] J. Ni, F. Zhang, J. Xiong, Q. Huang, Z. Chang, J. Ma, B. Xie, P. Wang, G. Bian, X. Li, and C. Liu, "Experience: Pushing indoor localization from laboratory to the wild," in *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, ser. MobiCom '22. New York, NY, USA: Association for Computing Machinery, 2022, pp. 147–157.

[7] 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 20th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '14. New York, USA: ACM, 2014, pp. 459–470.

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

[9] 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, 2015.

[10] H. Rizk, H. Yamaguchi, M. Youssef, and T. Higashino, "Gain without pain: Enabling fingerprinting-based indoor localization using tracking scanners," in *Proceedings of the 28th International Conference on Advances in Geographic Information Systems*, ser. SIGSPATIAL '20. New York, USA: ACM, 2020, pp. 550–559.

[11] G. Shen, Z. Chen, P. Zhang, T. Moscibroda, and Y. Zhang, "Walkie-markie: Indoor pathway mapping made easy," in *Proceedings of the 10th USENIX Conference on Networked Systems Design and Implementation*, ser. NSDI'13. Berkeley, CA, USA: USENIX Association, 2013, pp. 85–98.

[12] Z. Li, X. Zhao, Z. Zhao, and T. Braun, "Wifi-rita positioning: Enhanced crowdsourcing positioning based on massive noisy user traces," *IEEE Transactions on Wireless Communications*, vol. 20, no. 6, pp. 3785–3799, 2021.

[13] Y. Dong, T. Arslan, Y. Yang, and Y. Ma, "A wifi fingerprint augmentation method for 3-d crowdsourced indoor positioning systems," in *2022 IEEE 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2022, pp. 1–8.

[14] S. Shen, M. Gowda, and R. Roy Choudhury, "Closing the gaps in inertial motion tracking," in *Proceedings of the 24th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '18. New York, USA: ACM, 2018, pp. 429–444.

[15] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NIPS'14*. MIT Press, 2014, pp. 2672–2680.

[16] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua, "Cvae-gan: Fine-grained image generation through asymmetric training," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2764–2773.

[17] M. Rezagholiradeh and M. A. Haidar, "Reg-gan: Semi-supervised learning based on generative adversarial networks for regression," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2018, pp. 2806–2810.

[18] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *NIPS*, 2015, pp. 3483–3491.

[19] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier gans," in *International conference on machine learning*. PMLR, 2017, pp. 2642–2651.

[20] J. Wang, Z. Zhao, J. Cui, Y. Wang, Y. Shi, and B. Wu, "Low-cost wi-fi fingerprinting indoor localization via generative deep learning," in *WASA*. Springer International Publishing, 2021, pp. 53–64.

[21] 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*, ser. PerCom '19, 2019, pp. 1–10.

[22] P. Bahl and V. N. Padmanabhan, "Radar: an in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM '00*, vol. 2, 2000, pp. 775–784.

[23] D. Chen, W. Wang, W. Gao, and Z. Zhou, "Tri-net for semi-supervised deep learning," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, ser. IJCAI '18. AAAI Press, 2018, pp. 2014–2020.

[24] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in *The International Conference on Learning Representations (ICLR)*, 2014.

[25] C. Doersch, "Tutorial on variational autoencoders," 2016, cite arxiv:1606.05908.

[26] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.

[27] M. Mirza and S. Osindero, "Conditional generative adversarial nets," 2014.

[28] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," in *NIPS 2017 Workshop on Autodiff*, 2017.

[29] A. v. d. Oord, N. Kalchbrenner, O. Vinyals, L. Espeholt, A. Graves, and K. Kavukcuoglu, "Conditional image generation with pixelcnn decoders," in *Proceedings of the 30th International Conference on Neural Information Processing Systems*, ser. NIPS'16. Red Hook, NY, USA: Curran Associates Inc., 2016, pp. 4797–4805.

[30] S. Yao, S. Hu, Y. Zhao, A. Zhang, and T. Abdelzaher, "Deepsense: A unified deep learning framework for time-series mobile sensing data processing," in *Proceedings of the 26th International Conference on World Wide Web*, ser. WWW '17. Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee, 2017, pp. 351–360.

[31] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *ICML'15*. JMLR.org, 2015, pp. 448–456.

[32] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012.

[33] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.

[34] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the 3rd International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys '05. New York, NY, USA: ACM, 2005, pp. 205–218.

[35] Z. Li, W. Li, D. Wei, H. Yuan, B. Xu, and H. Shen, "Coordinate mapping method of crowdsourced data for indoor fingerprint localization," in *2022 IEEE 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2022, pp. 1–8.

[36] 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, April 2013.

[37] A. Rai, K. K. Chintalapudi, V. N. Padmanabhan, and R. Sen, "Zee: zero-effort crowdsourcing for indoor localization," in *MobiCom '12*, 2012, pp. 293–304.

[38] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–441, 2015.

[39] X. Song, X. Fan, X. He, and e. a. Xiang, "Cnnloc: Deep-learning based indoor localization with wifi fingerprinting," in *2019 IEEE SmartWorld*. IEEE, 2019, pp. 589–595.

[40] X. Wang, X. Wang, and S. Mao, "Indoor fingerprinting with bimodal csi tensors: A deep residual sharing learning approach," *IEEE Internet of Things Journal*, pp. 1–1, 2020.

[41] ——, "Deep convolutional neural networks for indoor localization with csi images," *IEEE Transactions on Network Science and Engineering*, vol. 7, no. 1, pp. 316–327, 2020.

[42] X. Chen, H. Li, C. Zhou, X. Liu, D. Wu, and G. Dudek, "Fidora: Robust wifi-based indoor localization via unsupervised domain adaptation," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9872–9888, 2022.

[43] X. Gan, B. Yu, L. Huang, and Y. Li, "Deep learning for weights training and indoor positioning using multi-sensor fingerprint," in *2017 International Conference on Indoor Positioning and Indoor Navigation*, ser. IPIN '17, Sep. 2017, pp. 1–7.

[44] X.-Y. Liu and X. Wang, "Real-time indoor localization for smartphones using tensor-generative adversarial nets," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[45] X. Wang, X. Wang, S. Mao, J. Zhang, S. C. Periaswamy, and J. Patton, "Indoor radio map construction and localization with deep gaussian processes," *IEEE Internet of Things Journal*, vol. 7, no. 11, pp. 11 238–11 249, 2020.

[46] T. Lan, X. Wang, Z. Chen, J. Zhu, and S. Zhang, "Fingerprint augment based on super-resolution for wifi fingerprint based indoor localization," *IEEE Sensors Journal*, vol. 22, no. 12, pp. 12 152–12 162, 2022.