Enhancing WiFi Fingerprinting Localization Through a Co-teaching Approach using Crowdsourced Sequential RSS and IMU Data

Zhendong Xu, Baoqi Huang, *Member, IEEE,* Bing Jia, *Member, IEEE* and Guoqiang Mao, *Fellow, IEEE*

***Abstract*—Crowdsourcing dramatically benefits WiFi fingerprinting localization in reducing the costs of collecting received signal strength (RSS) data during offline site survey, and has gained much attention in the literature. This paper proposes a deep learning based indoor positioning system (IPS), termed SeqIPS, to sufficiently exploit the available information in the crowdsourced sequential RSS data and inertial measurement unit (IMU) data. However, there exist the following three challenges: the relatively large label noises of crowdsourced RSS data, the unavailability of the labels of crowdsourced IMU data and the incorporation of the knowledge in IMU data into the localization model using RSS measurement as inputs during online localization. To this end, a co-teaching network is developed to effectively extract spatio-temporal features from sequential RSS data and meanwhile alleviate the influence of label noises. Also, a novel loss function involving IMU data is defined to impose spatial penalties, so as to further refine the localization model. Moreover, a domain adaptation module is included to effectively label the crowdsourced IMU data. Extensive experiments are conducted in a real scenario and show that, SeqIPS can achieve an average localization error of 3.37 m, outperforming both traditional methods and recent deep learning based methods by 18.4% at least. In summary, the novelty of SeqIPS is that extra crowdsourced IMU data is exploited to refine the localization model during offline training, while only sequential RSS data is required as inputs during online localization, such that the system accuracy, simplicity and costs are reasonably balanced.**

***Index Terms*—WiFi fingerprint-based localization, deep learning, generative adversarial network (GAN), co-teaching, crowdsourcing**

I. Introduction

Location-based service (LBS) plays an increasingly important role in our daily life due to its wide-ranging applications, such as navigation, monitoring, and target tracking [1]. In general, global positioning system (GPS) can provide satisfactory localization performance in outdoor environments, but becomes powerless in indoor environments due to a lack of line-of-sight (LoS) propagation [2]. Therefore, various technologies, like WiFi [3], [4], Bluetooth [5], [6], pedestrian dead reckoning (PDR) [7], [8], vision [9], ultra-wideband [10]–[12], etc., have been exploited to develop accurate indoor positioning systems (IPSs) in the past two decades. In pace with the widespread deployment and availability of WiFi infrastructures, WiFi-based fingerprint localization [13]–[16] has become one of the most promising solutions in indoor environments. It consists of two phases [2], i.e., the offline training phase and the online localization phase. In the offline phase, a site survey is conducted to collect sufficient received signal strength (RSS) data at each predefined reference point, and then a radio map is constructed and stored on a server; in the online phase, real-time RSS data collected by mobile devices is matched with the existing radio map to return the most probable locations.

In the previous research, the major works focused on the following three pivotal problems in fingerprint-based IPSs. Firstly, the applicability of fingerprint-based IPSs is severely restricted due to the labor-intensive and time-consuming site survey. As such, crowdsourcing-based approaches [17]–[20] have been widely adopted to collect fingerprints by exploiting the daily activities of participants. However, an obvious disadvantage of such methods is that it is hard to accurately label fingerprint data. Secondly, the robustness of fingerprint-based IPSs is generally degraded by the instability and inherent noises of wireless signals. Hence, an early study [21] tried to optimize localization results by exploiting temporal diversity and spatial dependency of RSS data, but only achieved limited localization accuracy; afterwards, studies based on deep learning methods [22], [23] attempted to learn the spatio-temporal relationships in sequential RSS data, which cannot combat crowdsourced data with severe label noises. Thirdly, the performance of fingerprint-based IPSs is quite limited. To push the limit of localization accuracy, different fusion methods were proposed to integrate additional information like IMU data [24]–[26] and visual images [27], thus producing evidently better localization accuracy than the pure fingerprint-based ones. However, these methods not only increase computation and deployment costs, but also lower the ubiquity of IPSs.

However, existing studies are still confronted with the following issues at present: 1) precisely labelling crowdsourced RSS or IMU data is quite challenging; 2) with noisy crowdsourced fingerprints, it is hard to produce accurate localization

This work is supported by the National Natural Science Foundation of China (Grant No. 62262046, 41871363 and U21A20446), the Major Program of Natural Science Foundation of Inner Mongolia A. R. of China (Grant No. 2021ZD13), the Science & Technology Plan Project of Inner Mongolia A. R. of China (Grant No. 2022YFSJ0027 and 2021GG0163), and the University Youth Science and Technology Talent Development Project (Innovation Group Development Plan) of Inner Mongolia A. R. of China (Grant No. NMGIRT2318). (Corresponding author: Baoqi Huang.)

Z. Xu, B. Huang, and B. Jia are with the Engineering Research Center of Ecological Big Data, Ministry of Education and Inner Mongolia Key Laboratory of Wireless Networking and Mobile Computing, College of Computer Science, Inner Mongolia University, Hohhot 010021, China (e-mail: xuzhendong@mail.imu.edu.cn; cshbq@imu.edu.cn; jiabing@imu.edu.cn).

G. Mao is with the Research Institute of Smart Transportation, Xidian University, Xi'an, China (e-mail: gqmao@xidian.edu.cn).

models or radio maps; 3) fusion methods impose extra processing costs, making fingerprint-based IPSs more complicated and less ubiquitous.

In order to overcome the aforementioned issues, this paper presents a novel fingerprint-based IPS, termed SeqIPS, which learns an accurate localization model by exploiting crowdsourced sequential RSS data and IMU data. Specifically, the core localization module of SeqIPS is a co-teaching network composed of two identical bi-directional long short-term memory (BiLSTM) structures. Its superiority lies in that, for one thing, the BiLSTM structures can adequately extract forward and backward spatio-temporal features from crowdsourced sequential RSS data, thus effectively mitigating the impacts of inherent noises and instability of wireless signals on localization accuracy; for another, the co-teaching network can effectively combat severe label noises caused by crowdsourcing to ensure the accuracy of the localization model. In particular, since the knowledge contained in training samples is not fully utilized in the learning of the co-teaching network (i.e., some training samples do not participate in training), a fine-tuning operation is included to further refine the localization model by using the other samples. Moreover, considering the fact that IMU information can effectively capture the relative location changes of mobile users, a novel loss function that formulates IMU data as spatial penalties is proposed to constrain the spatial relationships between location predictions to further improve the accuracy of the localization model. Therein, since it is difficult to label crowdsourced IMU data, a domain adaptation module based on a generative adversarial network (GAN) is developed to adapt a source domain model constructed by using a public dataset [28] to the current domain for effectively labelling the crowdsourced IMU data. In comparison with existing approaches, SeqIPS owns the following advantages: 1) SeqIPS is able to integrate extra crowdsourced IMU data to refine the localization model during offline training, but becomes a pure fingerprint-based IPS in online localization in the sense that only RSS data is requested for localization, so as to guarantee its simplicity and ubiquity; 2) SeqIPS is able to effectively combat severe location label noises from crowdsourced fingerprints, thus remarkably improving its accuracy.

To validate the effectiveness and performance of SeqIPS, extensive experiments are conducted in a large real scenario of nearly $1000\ m^2$, and a thorough comparison with other methods indicates that, SeqIPS is able to achieve an average localization error of $3.37\ m$, which outperforms the typical RADAR [29] and Horus [30] IPSs by 34.8% and 30.4%, respectively, as well as the popular deep learning based methods by 18.4% at least.

Table I gives the list of notations and explanations. The remainder of this paper is organized as follows. Section II reviews the literature on fingerprint-based localization. Section III provides an overview of the proposed SeqIPS. Section IV introduces how to label crowdsourced IMU data through the domain adaptation module. Section V proposes how to construct an accurate localization model based on a co-teaching network. In Section VI, experimental results and performance evaluation are reported. We conclude this paper and shed the lights on our future works in Section VII.

TABLE I
LIST OF NOTATIONS AND EXPLANATIONS.

Notation	Explanation
$\boldsymbol{r}^s/\boldsymbol{r}^t$	Sequential IMU data from the source/target domain
m_s/m_t	Number of sequential IMU data $\boldsymbol{r}^s/\boldsymbol{r}^t$
$\Delta\boldsymbol{u}^s$	Labels of $\boldsymbol{r}^s$
n	Length of sequential IMU data
$\boldsymbol{Z}$	Sequential RSS data
$\boldsymbol{L}$	Location labels of $\boldsymbol{Z}$
$\boldsymbol{Z}^+$	Online real-time sequential RSS data
l	Length of sequential RSS data
d_1/d_2	Dimension of label of IMU/RSS data
$\boldsymbol{R}$	Reshaped $\boldsymbol{r}^t$ for corresponding to $\boldsymbol{Z}$
$\Delta\boldsymbol{U}$	Predicted labels of $\boldsymbol{R}$
$\Delta\boldsymbol{x}, \Delta\boldsymbol{y}$	Changes of 2D coordinates transformed by $\Delta\boldsymbol{U}$
$\boldsymbol{\varepsilon}$	Gaussian noise injected into $\boldsymbol{Z}$, satisfying $\mathcal{N}\left(0, \sigma_z^2\right)$
ρ	Maximum ratio of large-noise sequential RSS data
T	Current epoch when training the co-teaching network
N	The epoch when the function value of $\Gamma\left(T\right)$ reaches $1-\rho$
δ	Threshold of distance errors produced by the distances between the predicted locations by the co-teaching network and the corresponding distances predicted by IMU information
Z_ρ	The large-noise sequential RSS data with ratio of ρ
h_ρ	Number of large-noise sequential RSS data Z_ρ
$\boldsymbol{L}^\rho_{Net_1}/\boldsymbol{L}^\rho_{Net_2}$	Locations of Z_ρ predicted by the Network1/Network2 in the co-teaching network

II. RELATED WORK

In this section, we shall briefly review the literature on the most related works in the following.

A. Crowdsourcing-based Localization Methods

To reduce the workloads and costs of site survey, crowdsourcing becomes one of the most feasible solutions, but a main weakness is that crowdsourced fingerprints induce severe label noises. Therefore, a great deal of research has been conducted to alleviate the influence of label noises. In [31], PDR was used to infer the locations of crowdsourcing participants given known initial locations of a trajectory, which suffers from the inherent error accumulations. In [32]–[34], crowdsourcing was utilized to collect RSS fingerprints by imposing specific constraints or restrictions, but still incurs significant location label errors. In [35], [36], global optimization methods were proposed to optimize fingerprint-based localization results with the usage of PDR information. In [37], [38], the map-assisted methods were proposed to generate fingerprint maps by using the crowdsourced fingerprints calibrated by additional map information. In [39], an unsupervised radio map learning scheme, termed MapICT, was proposed to build radio maps by using low-quality trajectory data, and achieved acceptable localization performance. In [40], [41], federated learning was applied to reduce the workload of labeled data collection and improve the performance of indoor localization through crowdsourced data, but lacks effective validation in real crowdsourcing scenarios. In a word, existing crowdsourcing-based methods aim to obtain available crowdsourced fingerprints with the aid of extra information, such as landmarks, maps, paths and etc., reducing the ubiquity

of crowdsourcing-based methods, and more importantly, these methods generally ignore the influence of severe location label noises of crowdsourced fingerprints on localization accuracy. Unlike the previous ones, the proposed SeqIPS can effectively alleviate the influence of location label noises from crowdsourced fingerprints when training the localization model, so as to improve localization accuracy.

B. Fusion Localization Methods

Since the performance of fingerprint-based IPSs is quite limited, extensive studies attempted to improve localization accuracy by fusing RSS data with other types of data, e.g., the most common IMU data [24], [26], [42], [43]. In [42], [43], fingerprint-based localization results were optimized by imposing coarse distance constraints obtained by PDR, but ignored orientation information. Moreover, in [24], [26], different methods were developed to fuse RSS and IMU data for location optimizations through the Kalman filter. However, these methods incur more processing costs, and cannot be applied without IMU data. Conversely, SeqIPS can employ extra IMU data to improve the accuracy of the localization model in offline training, while it becomes a pure fingerprint-based IPS in online localization, namely that only RSS data is requested for localization. As a result, SeqIPS maintains high online localization efficiency and system ubiquity.

C. Deep Learning Based Localization Methods

Deep learning has been introduced into fingerprint-based localization for improving localization accuracy [1], [22], [23], [44]–[47]. In the beginning, simple neural networks [44], [45] were only able to achieve limited localization performance, due to their limited ability to extract complex fingerprint features. Afterwards, complex neural networks [1], [22], [46]–[48] were proposed to mine more features from fingerprints, so that the localization accuracy is greatly improved. Moreover, since the RSS data collected during walking has obvious temporal correlations, the recurrent neural network (RNN) [23] was employed to deal with sequential RSS data, which can effectively alleviate the instability of localization results. However, existing deep learning based methods focus on improving localization accuracy based on the assumption that fingerprint data is labelled by accurate location labels, but ignore the situation where fingerprint data attains noisy labels, leading to inaccurate localization models. In contrast, SeqIPS is robust to the fingerprints with noisy location labels due to the effective mitigation of label noises during the training of the localization model.

D. Labeling IMU Data by Domain Adaptation

Recently, data-driven methods have replaced the original integration method to accurately label IMU data. However, due to the high collection costs and labeling costs of generating an IMU dataset, some works attempt to develop domain adaptation methods to label IMU data in a new environment. For instance, a novel framework, termed MotionTransformer, was proposed to extract domain-invariant features of raw IMU sequences from arbitrary domains and transform them into new domains without any paired data, enabling the transfer among different phone placements [49]. Moreover, a framework, termed TinyOdom, was proposed to train and deploy lightweight neural inertial models on ultra-resource-constrained (URC) devices, which validates the transferability of inertial models across entirely different datasets [50]. Different from the existing methods, SeqIPS focuses on learning domain-invariant features across different devices.

III. System Overview

Different from the typical fingerprint-based localization IPSs [29], [51], the proposed SeqIPS follows a deep learning framework, which builds a mapping from RSS to location, as shown in Fig. 1. SeqIPS consists of three modules: *adversarial learning based domain adaptation, co-teaching learning based localization model training and location predictions by the fine-tuned co-teaching model*.

A. Adversarial Learning Based Domain Adaptation

In this module, crowdsourced sequential IMU data is labelled and converted into the changes of location coordinates, thus formulating spatial penalties to refine the localization model during the training of sequential RSS data. The main structure is a domain adaptation network, which is composed of the source and target feature extractors, domain discriminator, movement predictor and movement decomposition.

Specifically, given a set of sequential IMU data $\boldsymbol{r}^s = \left[\boldsymbol{r}_1^s, \boldsymbol{r}_2^s, \cdots, \boldsymbol{r}_{m_s}^s\right]$ in a public dataset [28] with the accurate labels of $\Delta\boldsymbol{u}^s = \left[\Delta\boldsymbol{u}_1^s, \Delta\boldsymbol{u}_2^s, \cdots, \Delta\boldsymbol{u}_{m_s}^s\right]$ (where the superscript s denotes the source domain, m_s is the number of sequential IMU data, $\boldsymbol{r}_i^s = [\mathbf{r}_1^s, \mathbf{r}_2^s, \cdots, \mathbf{r}_n^s]$ is the i-th sequential IMU data with $\mathbf{r}_j^s$ being the j-th IMU sample and n being the length of $\boldsymbol{r}_i^s$, $\Delta\boldsymbol{u}_i^s = [\Delta\boldsymbol{d}_i^s, \Delta\boldsymbol{\theta}_i^s]$ is the label of $\boldsymbol{r}_i^s$ with $\Delta\boldsymbol{d}_i^s = [\Delta d_1^s, \Delta d_2^s, \cdots, \Delta d_n^s]$ and $\Delta\boldsymbol{\theta}_i^s = [\Delta\theta_1^s, \Delta\theta_2^s, \cdots, \Delta\theta_n^s]$ representing the displacement and heading, respectively), the source feature extractor and movement predictor are jointly trained to learn a source domain model; then, the target feature extractor with the inputs of a set of sequential IMU data in a target domain, denoted $\boldsymbol{r}^t = \left[\boldsymbol{r}_1^t, \boldsymbol{r}_2^t, \cdots, \boldsymbol{r}_{m_t}^t\right]$ (where the superscript t denotes the target domain, m_t is the number of sequential IMU data, and $\boldsymbol{r}_i^t = [\mathbf{r}_1^t, \mathbf{r}_2^t, \cdots, \mathbf{r}_n^t]$ is the i-th sequential IMU data with $\mathbf{r}_j^t$ being the j-th IMU sample), steadily combats with the domain discriminator to produce consistent feature representations with the source feature extractor in an identical feature space $\mathbb{R}^C$, so that the target feature extractor combined with the movement predictor can efficiently label the IMU data in the target domain; finally, a movement decomposition operation is conducted to convert the labels of sequential IMU data into the changes of location coordinates, denoted $(\Delta\boldsymbol{x}, \Delta\boldsymbol{y})$, on the two dimensional (2D) reference coordinate system, which is subsequently formulated as spatial penalties to refine the localization model (Subsection III-B).

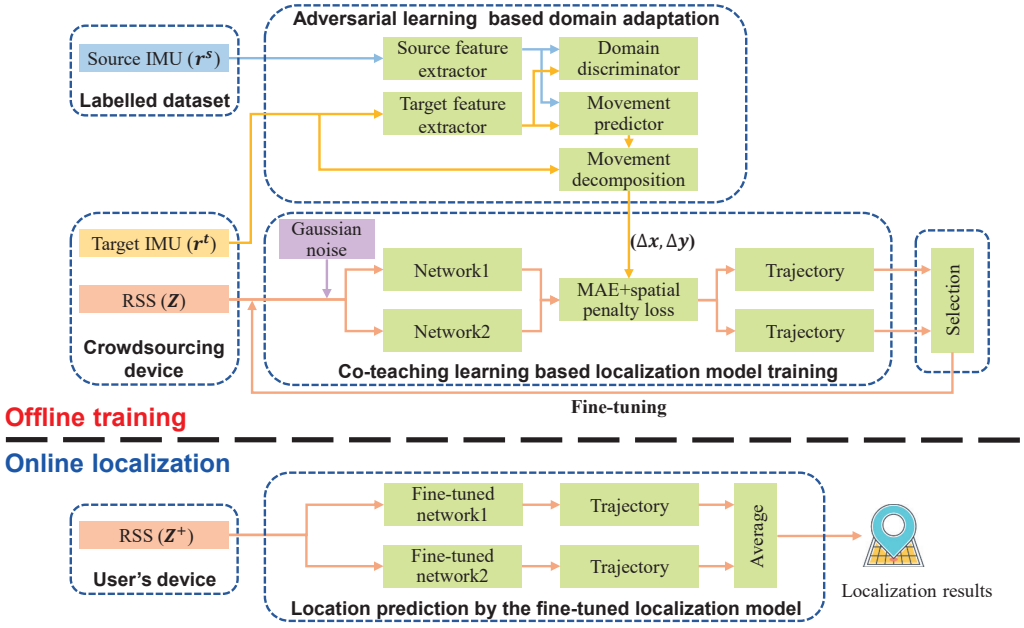


Fig. 1. The system architecture of the proposed SeqIPS. (Note that, the fine-tuned network1 and fine-tuned network2 in the online localization are the trained network1 and network2 in the offline training phase.)

B. Co-teaching Learning Based Localization Model Training

The objective of this module is to learn an accurate localization model from crowdsourced sequential RSS data. To this end, a co-teaching network is developed to adequately extract spatio-temporal features and meanwhile effectively alleviate the influence of severe label noises.

Given a set of sequential RSS data $\boldsymbol{Z} = [\boldsymbol{Z}_1, \boldsymbol{Z}_2, \cdots, \boldsymbol{Z}_h]$ with the noisy location labels of $\boldsymbol{L} = [\boldsymbol{L}_1, \boldsymbol{L}_2, \cdots, \boldsymbol{L}_h]$ (where h is the number of sequential RSS data, $\boldsymbol{Z}_i = [\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_l]$ represents the i-th sequential RSS data with $\boldsymbol{z}_j$ being the j-th RSS sample and l being the length of $\boldsymbol{Z}_i$, $\boldsymbol{L}_i = [\boldsymbol{l}_1, \boldsymbol{l}_2, \cdots, \boldsymbol{l}_l]$ denotes the corresponding 2D location labels of $\boldsymbol{Z}_i$); by regarding $\boldsymbol{Z}$ as inputs, the co-teaching network including a Network1 and a Network2 is initially trained through the sequential RSS data with small-loss labels based on the co-teaching strategy [52]; then, the remaining sequential RSS data with large-loss labels is re-labelled by the existing co-teaching network and filtered through a selection operation, so as to further be used to fine-tune the co-teaching network. Moreover, a novel loss function, which integrates both the mean absolute error (MAE) loss and the spatial penalty loss produced by $(\Delta \boldsymbol{x}, \Delta \boldsymbol{y})$, is designed to refine the co-teaching network. Note that, the trained co-teaching network is known as the localization model. In addition, to mitigate the impacts of measurement noises, we impose appropriate Gaussian noises into RSS data during the training of the co-teaching network.

C. Location Predictions by the Fine-tuned Localization Model

In the online localization phase, given a set of real-time sequential RSS data collected by user's devices, denoted $\boldsymbol{Z}^+ = \left[\boldsymbol{Z}_1^+, \boldsymbol{Z}_2^+, \cdots, \boldsymbol{Z}_g^+\right]$ with g being the number of sequential RSS data, we feed them into the fine-tuned localization model, i.e., the co-teaching network consisting of the Fine-tuned network1 and the Fine-tuned network2, and the mean values of their outputs are returned as user's final localization results, on account of the fact that the two networks have the same importance in the co-teaching network. *Note that, only the real-time sequential RSS data is used for localization, and there is no need to input IMU data in the online localization phase.* Subsequently, we shall introduce the specific details of the proposed SeqIPS.

IV. Labelling IMU Data Through A Domain Adaptation Module

In this section, a domain adaptation module based on GAN is designed to efficiently label IMU data. Firstly, the backgrounds of the problem are introduced (Subsection IV-A). Then, the implementation of the domain adaptation module is elaborated. Specifically, a source domain model is first constructed using an open IMU dataset with accurate labels (Subsection IV-B); then, an adversarial learning method is presented to build a target domain model, which aims to generate consistent feature representations with the source domain model in the same feature space (Subsection IV-C); finally, how to label IMU data by the target domain model is illustrated (Subsection IV-D).

A. Background

PDR, a typical inertial navigation system (INS), is widely exploited to infer the locations of pedestrians via low-cost micro-electro-mechanical system (MEMS) IMU platforms on smartphones. Traditionally, by assuming that the walking state of a pedestrian maintains a periodic property, PDR detects the user's steps, estimates their stride lengths and headings using real-time IMU data, and further updates the user's

locations [53], [54]. However, IMU data generally incurs high measurement noises and biases in practice, so giving rise to the critical issue of error accumulations. biases and measurement noises have to be precisely modeled, which is a challenging task.

Benefiting from the development of deep learning, data-driven methods have recently become an effective solution, which is able to directly build an accurate regression model from the noisy IMU data to the displacements and headings of pedestrians, thus effectively mitigating the impact of biases and measurement noises on the accuracy of PDR estimation [50], [55], [56]. Among them, the sequential deep learning method shows excellent performance as it can adequately utilize the spatio-temporal features of sequential IMU data. Since building an IMU dataset requires deploying specific delicate devices, and more importantly, these devices are not available for complex indoor environments, a domain adaptation module is proposed by firstly building a source domain model by using the labelled sequential IMU data in an open dataset [28], and then adapting the source domain model to a target domain for predictions. In this way, IMU data can be accurately labelled without additional costs. The proposed domain adaptation module is elaborated in the subsequent subsections.

B. Constructing the Source Domain Model

Since sequential IMU data is obviously temporally related, a three-layer BiLSTM feature extractor $\mathrm{M}_{\mathrm{F}}^{s}$ is developed to extract latent features. Given the sequential IMU data $\boldsymbol{r}^{s}$ as inputs; let $\boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{F}}^{s}}$ represent the parameters of $\mathrm{M}_{\mathrm{F}}^{s}$, the latent feature $\boldsymbol{X} \in \mathbb{R}^{C}$ can be obtained by

$$\boldsymbol{X} = \mathrm{M}_{\mathrm{F}}^{s}\left(\boldsymbol{r}^{s}; \boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{F}}^{s}}\right), \tag{1}$$

where $\mathbb{R}^{C}$ is the feature space of $\boldsymbol{X}$ with the dimensions being C. Then, a movement predictor comprising of a one-layer fully connected (FC) network, denoted M_{P}, is connected to predict the labels of $\boldsymbol{r}^{s}$ according to the latent features $\boldsymbol{X}$, i.e.,

$$\Delta\hat{\boldsymbol{u}}^{s} = \mathrm{M}_{\mathrm{P}}\left(\boldsymbol{X}; \boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{P}}}\right), \tag{2}$$

where $\boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{P}}}$ denotes the parameters of M_{P}, and $\Delta\hat{\boldsymbol{u}}^{s}$ represents the predicted labels. In order to optimize the training of the source domain model, the loss function of the movement predictor is defined as MAE, which is written as

$$L_{P} = \frac{1}{m_{s}} \sum_{i=1}^{m_{s}} \sum_{j=1}^{n} \sum_{k=1}^{d_{1}} |\Delta\boldsymbol{u}_{i,j,k}^{s} - \Delta\hat{\boldsymbol{u}}_{i,j,k}^{s}|, \tag{3}$$

where $|\cdot|$ represents the absolute value operation, and d_1=2 denotes the dimensions of labels (the labels consist of displacement and heading components).

In the training phase, by minimizing L_P, the feature extractor $\mathrm{M}_{\mathrm{F}}^{s}$ and movement predictor M_{P} are continuously refined, thus efficiently labelling the IMU data in the source domain.

C. Domain Adaptation by an Adversarial Learning

In order to adapt the source domain model to a target domain, we design a domain adaptation module based on the adversarial discriminant domain adaptation [57] that can learn the domain invariant representations of sequential IMU data in different domains, that is to say, a target feature extractor, denoted $\mathrm{M}_{\mathrm{F}}^{t}$, can generate consistent feature representations with $\mathrm{M}_{\mathrm{F}}^{s}$ in a feature space $\mathbb{R}^{C}$.

First, the target feature extractor $\mathrm{M}_{\mathrm{F}}^{t}$, which has the same network structure and different parameters with $\mathrm{M}_{\mathrm{F}}^{s}$, is employed to extract the latent features $\boldsymbol{Y}$ of sequential IMU data $\boldsymbol{r}^{t}$ from the target domain, i.e.,

$$\boldsymbol{Y} = \mathrm{M}_{\mathrm{F}}^{t}\left(\boldsymbol{r}^{t}; \boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{F}}^{t}}\right), \tag{4}$$

where $\boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{F}}^{t}}$ denotes the parameters of $\mathrm{M}_{\mathrm{F}}^{t}$.

Then, based on the latent features $\boldsymbol{X}$ and $\boldsymbol{Y}$, a domain discriminator M_{D} consisting of a two-layer FC network is constructed to differentiate their domains, i.e.,

$$\hat{\boldsymbol{o}} = \mathrm{M}_{\mathrm{D}}\left([\boldsymbol{X}, \boldsymbol{Y}]; \boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{D}}}\right), \tag{5}$$

where $\hat{\boldsymbol{o}} = [\hat{\boldsymbol{o}}^{s}, \hat{\boldsymbol{o}}^{t}]$ is the predicted domain distribution and $\boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{D}}}$ represents the parameters of M_{D}. The cross-entropy L_D of domain distribution $\hat{\boldsymbol{o}}$ and the true domain label $\boldsymbol{o} = [\boldsymbol{o}^{s}, \boldsymbol{o}^{t}]$ is defined as the loss function of the domain discriminator, namely

$$\begin{aligned} L_{D} = -\frac{1}{(m_{s}+m_{t})} \sum_{i=1}^{m_{s}+m_{t}} \sum_{j=1}^{n} & \boldsymbol{o}_{i,j} \log \hat{\boldsymbol{o}}_{i,j} \\ & + (1-\boldsymbol{o}_{i,j}) \log\left(1-\hat{\boldsymbol{o}}_{i,j}\right). \end{aligned} \tag{6}$$

Besides, we hope that $\mathrm{M}_{\mathrm{F}}^{t}$ can produce more and more similar feature representations with $\mathrm{M}_{\mathrm{F}}^{s}$, so that the discriminator M_{D} cannot distinguish true and fake. Therefore, $\mathrm{M}_{\mathrm{F}}^{t}$ is also trained by $\boldsymbol{r}^{t}$ with the inverted domain labels $\boldsymbol{o}^{t*}$ (i.e., the domain label of the source domain) and differentiated by the domain discriminator M_{D}. Here, the cross-entropy loss of the domain discriminator is defined as

$$L_{D^{*}} = -\frac{1}{m_{t}} \sum_{i=1}^{m_{t}} \sum_{j=1}^{n} \boldsymbol{o}_{i,j}^{t*} \log \hat{\boldsymbol{o}}_{i,j}^{t*} + \left(1-\boldsymbol{o}_{i,j}^{t*}\right) \log\left(1-\hat{\boldsymbol{o}}_{i,j}^{t*}\right), \tag{7}$$

where $\hat{\boldsymbol{o}}^{t*}$ is the domain distribution predicted by M_{D}.

During training, by alternately minimizing L_D and L_{D^*}, the target feature extractor and domain discriminator constantly optimize and improve themselves, and ultimately, $\mathrm{M}_{\mathrm{F}}^{t}$ is able to generate consistent feature representations with $\mathrm{M}_{\mathrm{F}}^{s}$ in the feature space $\mathbb{R}^{C}$.

D. Labelling IMU Data in the Target Domain

Since $\mathrm{M}_{\mathrm{F}}^{t}$ can generate consistent feature representations with $\mathrm{M}_{\mathrm{F}}^{s}$ in the feature space $\mathbb{R}^{C}$, the labels of sequential IMU data in the target domain are able to be efficiently labelled by cooperating $\mathrm{M}_{\mathrm{F}}^{t}$ with M_{P}. Hence, we can predict the labels of $\boldsymbol{r}^{t}$ by

$$\Delta\hat{\boldsymbol{u}} = \mathrm{M}_{\mathrm{P}}\left(\mathrm{M}_{\mathrm{F}}^{t}\left(\boldsymbol{r}^{t}; \boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{F}}^{t}}\right); \boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{P}}}\right), \tag{8}$$

where $\Delta\hat{\boldsymbol{u}} = \left[\Delta\hat{\boldsymbol{d}}, \Delta\hat{\boldsymbol{\theta}}\right]$ with $\Delta\hat{\boldsymbol{d}}$ and $\Delta\hat{\boldsymbol{\theta}}$ being the predicted displacement and heading, respectively. In the proposed SeqIPS, the objective of labelling sequential IMU data is

to formulate spatial penalties to refine the training of sequential RSS data. Hence, the sequential IMU data $\boldsymbol{R} = [\boldsymbol{R}_1, \boldsymbol{R}_2, \cdots, \boldsymbol{R}_h]$ corresponding to the sequential RSS data $\boldsymbol{Z}$ (as shown in Fig. 2, $\boldsymbol{Z}_i = [\boldsymbol{z}_i, \boldsymbol{z}_{i+1}, \cdots, \boldsymbol{z}_{i+l-1}]$ is a sequential RSS data with l being its lengths and an arbitrary element $\boldsymbol{z}_j$ being the RSS sample at j-th location, and the $\boldsymbol{R}_i = [\boldsymbol{r}_i, \boldsymbol{r}_{i+1}, \cdots, \boldsymbol{r}_{i+l-2}]$ is the corresponding IMU data with an arbitrary element $\boldsymbol{r}_j$ being the sequential IMU data between $\boldsymbol{z}_j$ and $\boldsymbol{z}_{j+1}$) is actually labelled by Eq. (8). The predicted labels are defined as $\boldsymbol{\Delta\hat{U}} = \left[\Delta\hat{\boldsymbol{U}}_1, \Delta\hat{\boldsymbol{U}}_2, \cdots, \Delta\hat{\boldsymbol{U}}_h\right]$ with $\Delta\hat{\boldsymbol{U}}_i = [\Delta\hat{\boldsymbol{u}}_i, \Delta\hat{\boldsymbol{u}}_{i+1}, \cdots, \Delta\hat{\boldsymbol{u}}_{i+l-2}]$ ($\Delta\hat{\boldsymbol{u}}_i$ is the labels of $\boldsymbol{r}_i$ in $\boldsymbol{R}_i$); the displacement component and the heading component of $\Delta\hat{\boldsymbol{U}}_i$ are defined as $\Delta\hat{\boldsymbol{D}}_i$ and $\Delta\hat{\boldsymbol{\Theta}}_i$, respectively. Note that, $\boldsymbol{R}$ is reshaped to adapt the input format of $\mathrm{M}_{\mathrm{F}}^t$ before prediction, and $\boldsymbol{\Delta\hat{U}}$ maintains the previous format of $\boldsymbol{R}$ by reshaping after prediction.

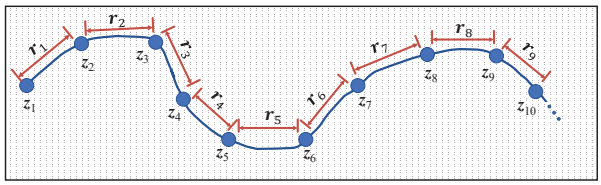


Fig. 2. The permutations of sequential RSS and IMU data on a pedestrian trajectory. Where $\boldsymbol{r}_i$ is a sequential IMU data, and $\boldsymbol{z}_i$ is an RSS sample.

Then, a movement decomposition operation is conducted to convert $\boldsymbol{\Delta\hat{U}}$ into the changes of location coordinates, i.e., $(\Delta\boldsymbol{x}, \Delta\boldsymbol{y})$, on the 2D reference coordinate system. For an arbitrary $\Delta\hat{\boldsymbol{U}}_i \in \boldsymbol{\Delta\hat{U}}$ with $i = 1, 2, \cdots, h$, we can compute the changes of location coordinates by

$$\Delta\boldsymbol{x}_i = \Delta\hat{\boldsymbol{D}}_i \cos\left(\alpha + \Delta\hat{\boldsymbol{\Theta}}_i\right), \tag{9}$$

$$\Delta\boldsymbol{y}_i = \Delta\hat{\boldsymbol{D}}_i \sin\left(\alpha + \Delta\hat{\boldsymbol{\Theta}}_i\right), \tag{10}$$

where α denotes the initial absolute orientation, which is computed by the initial several IMU samples in the first sequential IMU data $\boldsymbol{r}_i$ of $\boldsymbol{R}_i$ according to the heading estimation method [58]. Finally, the results of $\Delta\boldsymbol{x} = [\Delta\boldsymbol{x}_1, \Delta\boldsymbol{x}_2, \cdots, \Delta\boldsymbol{x}_h]$ with $\Delta\boldsymbol{x}_i = [\Delta x_1, \Delta x_2, \cdots, \Delta x_{l-1}]$ and $\Delta\boldsymbol{y} = [\Delta\boldsymbol{y}_1, \Delta\boldsymbol{y}_2, \cdots, \Delta\boldsymbol{y}_h]$ with $\Delta\boldsymbol{y}_i = [\Delta y_1, \Delta y_2, \cdots, \Delta y_{l-1}]$ are converted into spatial penalties to refine the training of sequential RSS data in the following section (Section V). It is noticeable that $\boldsymbol{r}^s$ and $\boldsymbol{r}^t$ are raw IMU data from smartphones, i.e., they are not processed.

V. Constructing an Accurate Localization Model by the Co-teaching Network

In this section, an accurate localization model is constructed by using crowdsourced sequential RSS data with relatively large label noises. Firstly, the background information is introduced (Subsection V-A). Then, the implementation of the proposed localization model is elaborated. Specifically, we first illustrate how to learn an initial localization model by the co-teaching network through exploiting the samples with small label noises (Subsection V-B), then present a novel loss function to refine the co-teaching network (Subsection V-C), and further propose how to fine-tune the localization model using the remaining samples with large label noises (Subsection V-D).

A. Background

Nowadays, fingerprint-based localization is mainly implemented in two ways. One way is to construct a radio map using the fingerprints collected at each reference point [30]. Another way is to learn an accurate localization model mapping from RSS data to location labels based on deep learning methods [23]. However, the former generally results in unstable localization results due to the instability and inherent noises of wireless signals, while the latter always relies on reliable fingerprints, i.e., fingerprints with accurate location labels, to generate a localization model, which will definitely be degraded by crowdsourced fingerprints due to the severe location label noises.

To this end, the main objective of this section is to construct a localization model using sequential crowdsourced data, which is expected to achieve high localization accuracy by adopting sequence-based learning, noise mitigation and model enhancement. Sequence-based learning can adequately exploit the spatio-temporal features in sequential RSS data by the BiLSTM structure, thus boosting the stability and accuracy of localization results. Noise mitigation emphasizes the usage of sequential RSS data with small label noises by the co-teaching network in the model training, thus effectively alleviating the influences of label noises on localization model accuracy. Model enhancement refines the spatial relationships between location predictions by imposing spatial penalties using IMU data, thus further improving the accuracy of the localization model. We illustrate the detailed implementations of the proposed localization model in the subsequent subsections.

B. Constructing an Initial Localization Model

As is known to all, the labels of crowdsourced RSS data suffer from severe noises, which result in an inaccurate localization model, thus degrading localization accuracy. In this instance, we hope that a localization model can be constructed by using the sequential RSS data with the location labels as accurate as possible (i.e., the sequential RSS data with small label noises), thus ensuring the accuracy of the localization model. Fortunately, the co-teaching network [52] conforms to our idea since it can obtain the samples with small label noises by selecting small-loss samples during training. Therefore, a co-teaching network is developed to learn latent spatio-temporal features of sequential RSS data and meanwhile alleviate the influence of label noises.

In order to mitigate the impacts of measurement noises, we impose Gaussian noises onto sequential RSS data in the input layer of the co-teaching network. Given a set of sequential RSS data $\boldsymbol{Z}$, the actual input $\boldsymbol{Z}^*$ of the co-teaching network is obtained by

$$\boldsymbol{Z}^* = \boldsymbol{Z} + \boldsymbol{\varepsilon}, \tag{11}$$

where $\boldsymbol{\varepsilon}$ denotes the Gaussian noise and satisfies $\mathcal{N}\left(0, \sigma_z^2\right)$. Then, considering the abundant spatio-temporal information

involved in sequential RSS data, the BiLSTM structure is adopted to construct the co-teaching network to extract spatio-temporal features. Therein, the co-teaching network consists of two identical networks, i.e., $\mathrm{M}_{\mathrm{Net}_1}$ and $\mathrm{M}_{\mathrm{Net}_2}$, each of which contains a feature extractor with a four-layer BiLSTM network and a location predictor with a two-layer FC network. Here, the co-teaching network with the input of $\boldsymbol{Z}^*$ is defined to learn spatio-temporal features by

$$\hat{\boldsymbol{L}}^{Net_1} = \mathrm{M}_{\mathrm{Net}_1}\left(\boldsymbol{Z}^*; \boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{Net}_1}}\right), \quad (12)$$

$$\hat{\boldsymbol{L}}^{Net_2} = \mathrm{M}_{\mathrm{Net}_2}\left(\boldsymbol{Z}^*; \boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{Net}_2}}\right), \quad (13)$$

where $\hat{\boldsymbol{L}}^{Net_1} = \left[\hat{\boldsymbol{L}}_1^{Net_1}, \hat{\boldsymbol{L}}_2^{Net_1}, \cdots, \hat{\boldsymbol{L}}_h^{Net_1}\right]$ and $\hat{\boldsymbol{L}}^{Net_2} = \left[\hat{\boldsymbol{L}}_1^{Net_2}, \hat{\boldsymbol{L}}_2^{Net_2}, \cdots, \hat{\boldsymbol{L}}_h^{Net_2}\right]$ represent the location outputs of $\mathrm{M}_{\mathrm{Net}_1}$ and $\mathrm{M}_{\mathrm{Net}_2}$, and $\boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{Net}_1}}$ and $\boldsymbol{\Omega}_{\mathrm{M}_{\mathrm{Net}_2}}$ are the parameters of $\mathrm{M}_{\mathrm{Net}_1}$ and $\mathrm{M}_{\mathrm{Net}_2}$, respectively. Note that, since the co-teaching network follows a multiple-input multiple-output (MIMO) mode, $\mathrm{M}_{\mathrm{Net}_1}$ and $\mathrm{M}_{\mathrm{Net}_2}$ will output a set of consecutive locations when a sequential RSS data is inputted.

During training, to combat severe label noises, the learning of the co-teaching network adheres to such a strategy: in each epoch, the two networks ($\mathrm{M}_{\mathrm{Net}_1}$ and $\mathrm{M}_{\mathrm{Net}_2}$) select small-loss samples as useful knowledge in forward propagation (for the purpose of selecting the samples with small label noises) and teach such samples to their peer network for updating parameters in back propagation. Moreover, to control the ratio of small-loss samples, a ratio function $\Gamma(T)$ is defined as

$$\Gamma(T) = 1 - \min\{\frac{T}{N}\rho, \rho\}, \quad (14)$$

where T denotes the current epoch, ρ (named noise ratio) represents the maximum ratio of large-loss samples, and N represents the epoch when the ratio of small-loss samples reaches $1-\rho$. It can be found that, with the increase of T, the ratio of small-loss samples $\Gamma(T)$ gradually reduces, and reaches $1-\rho$ when $T \geq N$. The main reason lies in that the neural network first fits relatively clean samples, and then fits noisy samples progressively, thus leading to overfitting to some extent [52]. Hence, it is necessary to filter out the samples with large label noises before overfitting, so as to maintain the accuracy of the model.

C. MAE and Spatial Penalty Loss

In order to more accurately learn the spatial features of sequential RSS data, we propose a novel loss function that not only minimizes the common MAE between the predicted and true location labels, but also minimizes the spatial penalties formulated by the predicted ($\Delta\boldsymbol{x}$, $\Delta\boldsymbol{y}$) in Section IV.

For an arbitrary network $\mathrm{M}_{\mathrm{Net}_\mathrm{b}}$ with $b \in \{1, 2\}$, given the predicted location labels $\hat{\boldsymbol{L}}^{Net_b}$ and the true location labels $\boldsymbol{L}$, the MAE loss function is defined as

$$L_M = \frac{1}{h}\sum_{i=1}^{h}\sum_{j=1}^{l}\sum_{k=1}^{d_2}|\hat{\boldsymbol{L}}_{i,j,k}^{Net_b} - \boldsymbol{L}_{i,j,k}|, \quad (15)$$

where $d_2 = 2$ represents the dimension of the location label.

Moreover, ($\Delta\boldsymbol{x}$, $\Delta\boldsymbol{y}$) is formulated as spatial penalties to refine the co-teaching network, that is to say, the coordinate distance between two adjacent locations is constrained by ($\Delta\boldsymbol{x}$, $\Delta\boldsymbol{y}$). The spatial penalty loss function is defined as

$$L_S = \frac{1}{h}\sum_{i=1}^{h}\sum_{j=1}^{l-1}\sum_{k=1}^{d_2}\Xi(\boldsymbol{\Delta\eta}_{i,j,k} > \delta) * \log\left(\boldsymbol{\Delta\eta}_{i,j,k} - \delta + 1\right), \quad (16)$$

where δ is a distance threshold, and $\boldsymbol{\Delta\eta}_i = |\hat{\boldsymbol{L}}_{i,2:l}^{Net_b} - \hat{\boldsymbol{L}}_{i,1:l-1}^{Net_b} - \left[\ \Delta\boldsymbol{x}_i, \Delta\boldsymbol{y}_i\ \right]|$ denotes the distance error produced by the distance between the predicted locations by co-teaching network and the distance predicted by IMU information, i.e., ($\Delta\boldsymbol{x}$, $\Delta\boldsymbol{y}$). Ξ is a sign function with the value being 1 when the condition $\boldsymbol{\Delta\eta}_{i,j,k} > \delta$ is true (producing additional penalty), otherwise being 0 when the condition $\boldsymbol{\Delta\eta}_{i,j,k} > \delta$ is false (without additional penalty). Finally, by combining L_M and L_S, the total loss function of the co-teaching network is defined as

$$L_{co} = \lambda_1 L_M + \lambda_2 L_S, \quad (17)$$

where λ_1 and λ_2 are the hyper-parameters.

In the training phase, by treating the loss L_{co} as an optimizing criterion, the co-teaching network is iteratively trained.

D. Localization Model Fine-tuning

According to the afore-mentioned co-teaching network learning, It can be found that, when the training reaches stable, only a part of sequential RSS data with the ratio of $1-\rho$ is employed to optimize the parameters of the co-teaching network, but the other RSS data with the ratio of ρ does not take some effects in practice. Hence, to adequately exploit the knowledge in sequential RSS data, a fine-tuning procedure is conducted to further refine the co-teaching network using the other RSS data with the ratio of ρ.

Since these RSS data theoretically have large label noises after filtering by the co-teaching network. Therefore, we re-label them by the existing co-teaching network, and then a selection operation is conducted to select suitable data for fine-tuning. Specifically, let $\boldsymbol{Z}^\rho = \left[\boldsymbol{Z}_1, \boldsymbol{Z}_2, \cdots, \boldsymbol{Z}_{h_\rho}\right]$ denote those sequential RSS data with $h_\rho = \lceil h \times \rho \rceil$ being the number of sequential RSS data ($\lceil\cdot\rceil$ represents the ceiling function); the two existing networks $\mathrm{M}_{\mathrm{Net}_1}$ and $\mathrm{M}_{\mathrm{Net}_2}$ are respectively used to label the locations of $\boldsymbol{Z}^\rho$, which are defined as

$$\boldsymbol{L}_{Net_1}^\rho = \left[\boldsymbol{L}_1^{Net_1}, \boldsymbol{L}_2^{Net_1}, \cdots, \boldsymbol{L}_{h_\rho}^{Net_1}\right], \quad (18)$$

$$\boldsymbol{L}_{Net_2}^\rho = \left[\boldsymbol{L}_1^{Net_2}, \boldsymbol{L}_2^{Net_2}, \cdots, \boldsymbol{L}_{h_\rho}^{Net_2}\right]. \quad (19)$$

Then, a sample selection operation is conducted to exclude a small number of unreliable samples. For an arbitrary sequential RSS data $\boldsymbol{Z}_i$ in $\boldsymbol{Z}^\rho$ with $i = 1, 2, \cdots, h_\rho$, we compute the average Euclidean distance between $\boldsymbol{L}_i^{Net_1}$ and $\boldsymbol{L}_i^{Net_2}$, and if the distance is fairly distant, i.e., it is more than the distance threshold empirically set as 1.2 m, $\boldsymbol{Z}_i$ is discarded. After that, the remaining RSS data with the location labels being the mean value of $\boldsymbol{L}_{Net_1}^\rho$ and $\boldsymbol{L}_{Net_2}^\rho$ is fed into the existing co-teaching network for fine-tuning, so as to further enhance the prediction performance of the co-teaching network.

To sum up, the training of the localization model can be summarized as follows: the co-teaching network first selects

Algorithm 1 Training of Localization Model

Input: $\boldsymbol{Z}$, $\boldsymbol{r}^s$, $\boldsymbol{r}^t$;
Output: A localization model.

// Label IMU data by domain adaptation
Construct a source domain model $\mathrm{M_P}\left(\mathrm{M_F^s}\left(\boldsymbol{r}^s;\boldsymbol{\Omega}_{\mathrm{M_F^s}}\right);\boldsymbol{\Omega}_{\mathrm{M_P}}\right)$ by $\boldsymbol{r}^s$;
Generate a target domain model $\mathrm{M_P}\left(\mathrm{M_F^t}\left(\boldsymbol{r}^t;\boldsymbol{\Omega}_{\mathrm{M_F^t}}\right);\boldsymbol{\Omega}_{\mathrm{M_P}}\right)$ by domain adaptation using $\boldsymbol{r}^t$;
Compute the changes of location coordinates $(\Delta\boldsymbol{x},\Delta\boldsymbol{y})$ by the target domain model and movement decomposition;

// Learn a localization model by co-teaching network
Learn initial localization models $\mathrm{M_{Net_1}}$ and $\mathrm{M_{Net_2}}$ using $\boldsymbol{Z}$ and $(\Delta\boldsymbol{x},\Delta\boldsymbol{y})$;
Re-label the remaining RSS data $\boldsymbol{Z}^\rho$ by using $\mathrm{M_{Net_1}}$ and $\mathrm{M_{Net_2}}$;
Fine-tune the localization model $\mathrm{M_{Net_1}}$ and $\mathrm{M_{Net_2}}$ using the remaining RSS data $\boldsymbol{Z}^\rho$ and the corresponding $(\Delta\boldsymbol{x},\Delta\boldsymbol{y})$;
return The fine-tuned localization model.

RSS data with small label noises to construct an initial localization model; then, the other RSS data with large label noises is re-labelled for further fine-tuning the model. The procedure of learning a localization model is summarized in Algorithm 1.

VI. EXPERIMENTAL EVALUATION

In this section, extensive experiments are conducted to thoroughly evaluate the performance of the proposed SeqIPS.

A. Datasets

Two datasets, i.e., "Library" which includes sequential RSS data with noisy location labels and unlabelled IMU data and "OxIOD" which contains labelled IMU data [28], are used in the experiments. We introduce them as follows.

1) ***Library dataset***: sequential RSS and IMU data are collected in a large open space with a total area of nearly $1000\ m^2$, i.e., the Reading Room on the third floor of a library building, which includes a number of bookracks with a height of around 2 m, desks and chairs, as illustrated in Fig. 3. It is noticeable that more than 100 distinct access points (APs) are detected in the target space, and most of them only provide occasional and extremely weak RSS measurements; hence, 22 APs with strong and active RSS measurements are selected for use.

In the experiments, data collection was conducted in a crowdsourcing manner, and lasted nine rounds. Specifically, regarding each round, the training data was collected by a student with a smartphone (i.e., HUAWEI P7) held in front of his chest to arbitrarily traverse the target space; additionally, the student was also asked to walk along the predefined trajectories (with a length between 20 m and 30 m) at a constant speed to produce testing data. Ultimately, the total training

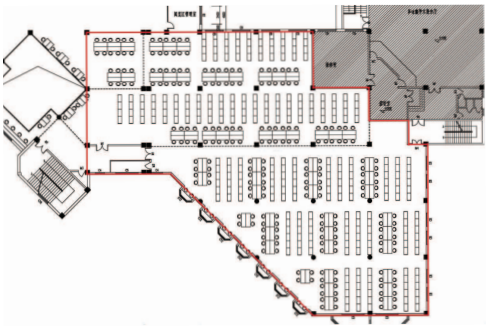


Fig. 3. The floor plan of experimental areas

data contains 8499 RSS samples and numerous IMU samples, and the total testing data contains 1812 RSS samples. Therein, the training RSS samples were rapidly labelled through an existing coarse-grained fingerprint-based IPS constructed by the Gaussian process regression (GPR) using only a few fingerprints (Note that, the training RSS samples are also able to be labelled by other methods, e.g., PDR.). Obviously, these training RSS samples incur severe label noises. Accordingly, the testing RSS samples were accurately labelled by evenly assigning locations on the predefined trajectories due to the constant speed. In addition, the scanning frequencies of RSS data and IMU data in collecting these data were set as 1 Hz and 100 Hz, respectively.

According to these collected data, a sliding time window operation was conducted to divide the training RSS and IMU data into small sequential data with a specific length as the inputs of SeqIPS. Concretely, the IMU data was divided with a window size of 1 second (i.e., 100 records) and a step size of 0.2 second to produce $\boldsymbol{r}^t$ for training the domain adaptation network; the RSS data was divided with a window size of l second (i.e., l records) and a step size of 1 second to produce $\boldsymbol{Z}$ for training the co-teaching network; in particular, to facilitate the fusion of RSS and IMU data, according to the permutations of sequential RSS and IMU data in Fig.2 that a sequential IMU data $\boldsymbol{r}_i$ is always sandwiched between two RSS samples, the IMU data was divided into $\boldsymbol{R}$ that corresponds to $\boldsymbol{Z}$ to predict the changes of location coordinates $(\Delta\boldsymbol{x},\Delta\boldsymbol{y})$, so as to formulate spatial penalties to refine the training of the sequential RSS data.

2) ***OxIOD dataset***: OxIOD is an open IMU dataset with a scanning frequency of 100 Hz collected in a Vicon Room with the size of 5 m $\times$ 4 m [28]. Specifically, a pedestrian carrying a commercial-off-the-shelf smartphone (iPhone 7 plus) in four different ways, i.e., handheld, in a pocket, in a handbag and on a trolley, walked normally inside the room to produce sequential IMU data; meanwhile, an optical motion capture system (Vicon) was deployed in this room to label the ground truths of the IMU data, resulting in a high-precision full pose reference with a location error around 0.01 meters and orientation error around 0.1 degrees. Ultimately, the OxIOD dataset contains 158 sequences of IMU data with the total walking distance and recording time being 42.5 km and 14.72 h, respectively. In our experiments, since the IMU data in the library dataset was collected in a handheld fashion, we only

select the data corresponding to the handheld style in OxIOD for use. In addition, by applying the sliding time window operation on the Library dataset to the OxIOD dataset, the IMU data in OxIOD is divided into the same length sequential IMU data, which is used for learning the domain adaptation model.

B. Experimental Setup

The detailed architecture of the proposed SeqIPS is illustrated in Fig. 4, depicting the network structure of each layer, the sizes of input and output neurons, etc. Therein, the IMU data with 3-axis accelerometer, gyroscope readings and the RSS data from 22 APs were respectively inputted into the domain adaptation network and the co-teaching network in the experiments, so that the corresponding input neurons were set as 6 and 22. The learning rate of the domain adaptation network was invariant with the size of 0.001; for the co-teaching network, the learning rates of the initial training (Subsection V-B) and the fine-tuning training (Subsection V-D) were initially set as 0.01 and 0.001, respectively, and then were adjusted by the StepLR schedule method in PyTorch with the parameters step_size=10 and gamma=0.25 in the follow-up iterations. Adam [59], a first-order gradient-based optimizer, was used to train the proposed networks. The batch size was set as 128. The total epochs of domain adaptation and co-teaching networks were both 50. In particular, the dropout with a rate of 0.5 was used to prevent overfitting. Prior to each epoch, the sequential RSS data was corrupted with additional Gaussian noises to enhance the robustness of the co-teaching network, and the corresponding parameter σ_z was set as 5 dBm. After parameter adjustments, the losses L_M and L_S played equally important roles in the training phase, so that we maintained the parameters λ_1 and λ_2 to be both 0.5. The value of N in the ratio function $\Gamma(T)$ was set as 15 (see Eq. (14)).

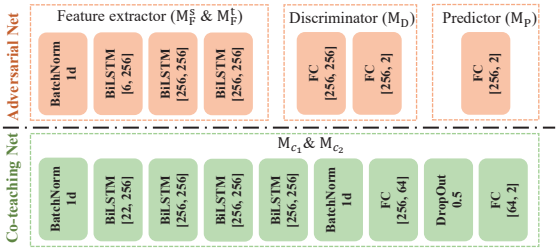


Fig. 4. The detailed architecture of the proposed network.

To verify the effectiveness of the proposed SeqIPS, two typical fingerprint-based localization methods, i.e., RADAR [29] and Horus [30], and several deep learning based localization methods, i.e., CNNLoc [48], LSTM [23], BiLSTM [23], CNN-BiLSTM, WiDeep [46] and iToLoc [47] were implemented for comparison. In particular, since CNNLoc and BiLSTM could achieve accurate localization performance, a combination of them, i.e., CNN-BiLSTM, was developed as a comparison method for performance evaluation. Note that, although the proposed SeqIPS fuses additional IMU data in offline training to improve the accuracy of the localization model, it is still a pure fingerprint localization system in essence. Hence, to be fair, only the typical fingerprint-based localization methods are utilized for comparison.

The pivotal parameters and concrete implementations of these comparison methods are illustrated as follows. RADAR computed the average value of the RSS data at each reference point to generate a radio map. Horus constructed a Gaussian distribution of the RSS data at each reference point as the fingerprint metric. Therein, the grid size of reference points was set as 1 m $\times$ 1 m. Regarding the other six deep learning based methods, CNNLoc, LSTM, BiLSTM and WiDeep followed the parameter settings and network architectures reported in the corresponding literature; CNN-BiLSTM was designed by including a two-layer convolutional neural network (CNN) with the neurons of $[400, 200]$ and a two-layer BiLSTM network with the neurons of $[200, 200]$, where the convolution kernel size and stride of CNN were set as 5 and 1, respectively; the iToloc was implemented by only including the robust localization module, but excluding the model update module, because the environmental changes are not the focus of this work.

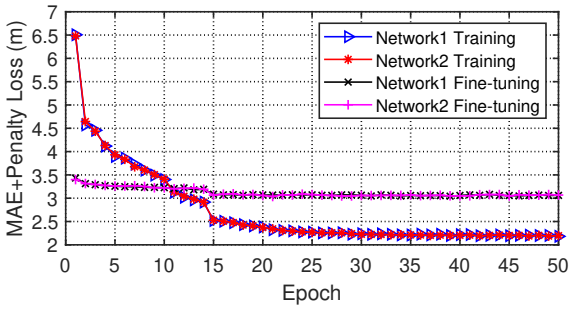
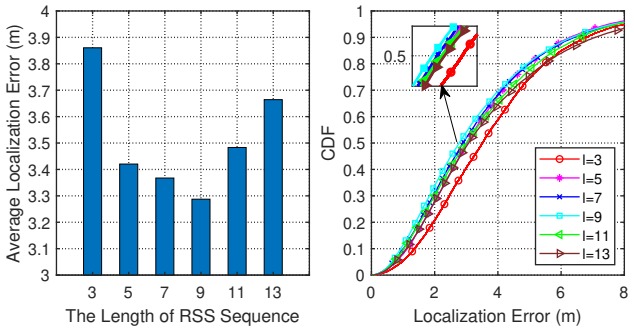


Fig. 5. The loss variation of the localization model with the increase of epoch when training.

The experiments were conducted on a Laptop with an AMD Ryzen 7 4800H CPU and an NVIDIA GeForce RTX 2060 GPU. These deep learning based methods were implemented using the PyTorch framework, and RADAR and Horus were realized in Matlab. In order to avoid the randomness of parameter initialization of these deep learning based methods, we run such methods by ten times and showed their average results in performance evaluation.

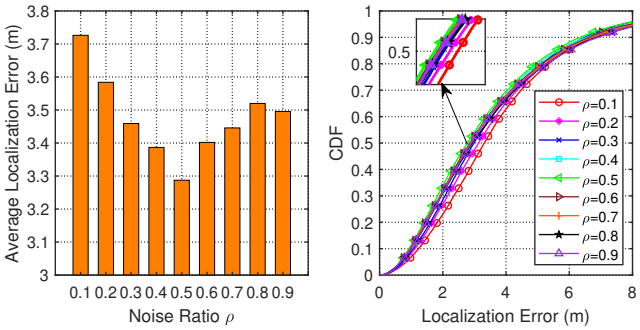
C. Evaluation of Convergence of the Localization Model

In this subsection, the convergence of the localization model, i.e., the co-teaching network in SeqIPS, is evaluated. As shown in Fig. 5, with the increase of epoch, the Network1 and Network2 in the localization model produce similar loss values and gradually converge when the epoch reaches 25 and 15, respectively, in initial network training and network fine-tuning. It can be concluded that the training of the localization model is effective and convergent, and the convergence of each network is consistent. Note that, since the fine-tuning utilizes the remainder of large-noise training samples, the loss value is larger than that of the initial training.



(a) Mean Error (b) Cumulative Error

Fig. 6. Comparison of average localization errors produced by the proposed SeqIPS with different lengths of sequential RSS data.



(a) Mean Error (b) Cumulative Error

Fig. 7. Comparison of the localization errors produced by the proposed SeqIPS with respect to different noise ratios.

D. Impact of the Length of Sequential RSS Data

To explore the impact of the length of sequential RSS data, i.e., l, on localization accuracy, we have l taking values from 3, 5, 7, 9, 11 and 13 for evaluation. The localization errors produced by the proposed SeqIPS with respect to different values of l are shown in Fig. 6. As can be seen, in Fig. 6(a), the average localization error gradually decreases with the increase of l, and the minimum localization error occurs when l is up to 9; afterwards, the localization error gradually increases when l is more than 9. Accordingly, Fig. 6(b) also achieves the best localization performance when $l = 9$. Therefore, we have $l = 9$ in the following experiments.

E. Evaluation of the Noise Ratio ρ

In the SeqIPS, the noise ratio, i.e., ρ, is extremely important for training the co-teaching network, and affects the localization accuracy. Therefore, an experiment is conducted to train the proposed SeqIPS with ρ rising from 0.1 to 0.9 with an interval of 0.1, and the average localization errors are illustrated in Fig. 7. As can be seen, with the increase of ρ, the average localization error in Fig.7(a) reduces gradually, reaches the minimum value less than 3.3 m when $\rho = 0.5$, and rises up when $\rho > 0.5$. It can be concluded that, if ρ is too large, the co-teaching model suffers from the limited representation capability due to insufficient training samples, and conversely, if ρ is too small, the samples with large label noises participate in the training, degrading the co-teaching model accuracy; as a result, the prediction capability of the fine-tuned model is lowered. Likewise, the best localization result is produced in Fig. 7(b) when $\rho = 0.5$. Hence, we set $\rho = 0.5$ in our experiments.

TABLE II
COMPARISON OF LOCALIZATION RESULTS PRODUCED BY THE PROPOSED SEQIPS IN FOUR DIFFERENT MODALITIES.

Modality	Mean (m)	Median (m)	STD (m)
Without IMU and fine-tuning	3.64	3.02	2.55
Without IMU	3.46	2.89	2.40
Without fine-tuning	3.54	2.91	2.54
With IMU and fine-tuning	**3.29**	**2.78**	**2.24**

F. Effectiveness of IMU Information and Fine-tuning

In this subsection, the effectiveness of IMU information (i.e., the spatial penalty loss) and fine-tuning in the proposed SeqIPS is evaluated. The localization results produced by SeqIPS in four different modalities, i.e., "With IMU and fine-tuning", "Without IMU", "Without fine-tuning" and "Without IMU and fine-tuning" are plotted in Table II. It can be found that, SeqIPS with "With IMU and fine-tuning" modality produces evidently better localization performance with an average localization error (termed "Mean") of 3.29 m, a median localization error (termed "Median") of 2.78 m and a standard deviation (termed "STD") of 2.24 m than other three modalities, that of "Without IMU and fine-tuning" modality derives the worst localization accuracy with the Mean of 3.64 m, the Median of 3.02 m and the STD of 2.55 m; however, "Without IMU" and "Without fine-tuning" modalities achieve the localization performance between "With IMU and fine-tuning" and "Without IMU and fine-tuning". We can conclude that IMU information and fine-tuning are beneficial for improving localization performance to varying degrees.

TABLE III
COMPARISON OF LOCALIZATION ACCURACY PRODUCED BY THE PROPOSED SEQIPS WITH THE DIFFERENT VALUE OF δ.

δ	0.05	0.15	0.25	0.35	0.45
Average Localization Error (m)	3.42	3.42	**3.29**	3.37	3.43

G. Evaluation of Distance Threshold δ

In order to the influence of the distance threshold δ on localization accuracy, we take the value of δ as 0.05, 0.15, 0.25, 0.35, and 0.45 for evaluation. The average localization errors produced by SeqIPS with different δ are listed in Table III. As can be seen, the best localization performance is derived when $\delta = 0.25$. Therefore, we set $\delta = 0.25$ in the subsequent performance comparison.

H. Comparison of Different Methods

1) Comparison of localization performance: First of all, the cumulative distribution function (CDF) of the localization errors produced by different methods is plotted in Fig. 8. As

TABLE IV
COMPARISON OF LOCALIZATION ERRORS PRODUCED BY DIFFERENT METHODS WITH RESPECT TO 10 RUNNINGS

Method	Localization Error (m)										
	NO.1	NO.2	NO.3	NO.4	NO.5	NO.6	NO.7	NO.8	NO.9	NO.10	Average
RADAR [29]	5.17										5.17
Horus [30]	4.84										4.84
CNNLoc [48]	4.94	4.82	4.79	4.61	4.74	4.50	4.77	4.76	4.65	4.58	4.72
LSTM [23]	4.95	4.92	4.93	4.82	4.97	4.73	5.08	4.93	4.96	5.06	4.94
BiLSTM [23]	4.70	4.32	4.37	4.54	4.29	4.39	4.47	4.58	4.70	4.43	4.48
CNN-BiLSTM	4.10	4.19	4.04	4.22	4.13	4.02	4.23	4.08	4.32	3.98	4.13
WiDeep [46]	5.48	5.50	5.47	5.49	5.48	5.50	5.49	5.50	5.50	5.48	5.49
iToLoc [47]	4.52	4.52	4.48	4.49	4.48	4.47	4.54	4.49	4.57	4.52	4.51
SeqIPS	**3.41**	**3.35**	**3.38**	**3.35**	**3.35**	**3.38**	**3.29**	**3.38**	**3.38**	**3.38**	**3.37**

can be seen, the proposed SeqIPS evidently engenders better localization performance than the other eight methods.

Then, the localization errors produced by different methods with respect to 10 runnings are shown in Fig. 10, and the detailed localization errors and the total average values are illustrated in Table IV. (Note that, since the parameters of RADAR and Horus are not stochastic, only once localization result is provided for comparison.) It can be found that, the proposed SeqIPS always produces the best localization performance in each running with a stable localization error around 3.3 m, but other methods incur worse localization accuracy than SeqIPS. Specifically, WiDeep produces the worst localization accuracy of 5.49 m; RADAR and Horus achieve localization accuracy of 5.17 m and 4.84 m, respectively; for the other deep learning based methods, CNNLoc, LSTM, BiLSTM and iToLoc derive the average localization errors fluctuating around 4.7 m, 4.9 m, 4.5 m and 4.5 m respectively, and in particular, CNN-BiLSTM realizes better localization accuracy fluctuating around 4.1 m than the other four methods. To sum up, in comparison with the typical fingerprint-based IPS, i.e., RADAR and Horus, the proposed method, i.e., SeqIPS, is able to improve the average localization accuracy by 34.8% and 30.4%, respectively; that of other deep learning based methods, SeqIPS can improve the localization accuracy by 18.4% at least. Note that, the main reason that WiDeep produces the worst localization performance lies in that the prediction ability of WiDeep is obviously insufficient due to a severe shortage of training data at each reference point.

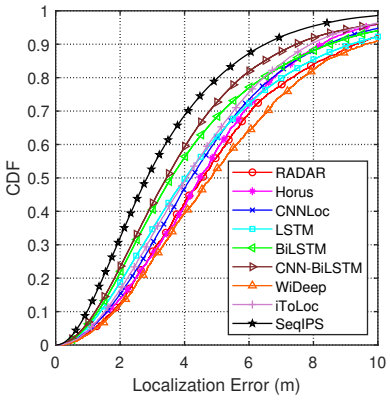


Fig. 8. Comparison of localization errors produced by different methods.

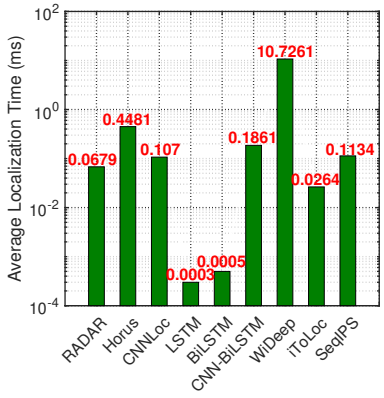


Fig. 9. The comparison of average localization time for different methods

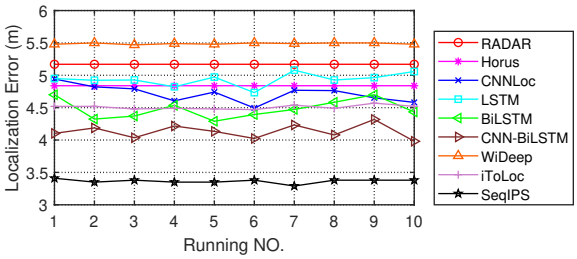


Fig. 10. Comparison of average localization errors produced by different methods with respect to 10 runnings.

2) Comparison of localization efficiency: The average localization time, i.e., the average time between sending one location request and returning its location estimate, with respect to different methods is plotted in Fig. 9. As can be seen, most methods spend an average localization time below 1 ms except the WiDeep method. Specifically, SeqIPS maintains the average localization time at the sub-millisecond level, i.e., 0.1134 ms, which is close to the most typical fingerprint-based IPS; however, LSTM and BiLSTM achieve the highest localization efficiency time with an average localization time below 1 μs due to their simple network structures; in particular, WiDeep spends the most time to return once location result since it requires to input real-time RSS data to many deep learning models (i.e., one model for every AP at each reference point) for online localization.

In summary, the experiments confirm the effectiveness of the proposed SeqIPS, in the sense that SeqIPS dramatically

improves the localization accuracy of fingerprint-based localization, and the comparison among seven different methods validates the superiority and robustness of the proposed SeqIPS.

VII. Conclusion

In this paper, we proposed a deep learning based IPS, termed SeqIPS, that constructs an accurate localization model by exploiting crowdsourced sequential RSS and IMU data. To be specific, a co-teaching network was developed to effectively extract spatio-temporal features from sequential RSS data and meanwhile alleviate the influence of label noises. Meanwhile, a novel loss function was defined to incorporate extra crowdsourced IMU data to impose spatial penalties, so as to further refine the localization model. In addition, a domain adaptation module based on GAN was included to efficiently label crowdsourced IMU data. Extensive experiments were conducted, and a thorough comparison revealed that the proposed SeqIPS outperforms the other popular methods in the literature in terms of both localization accuracy and robustness.

In the future, we plan to improve the quality of crowdsourced fingerprints (i.e., improving the accuracy of location labels) with the help of additional information, e.g., the floor plan of a target space, further explore more appropriate schemes to mitigate the influence of label noises on the model training phase, as well as achieve the large-scale application and deployment of the proposed IPS.

References

[1] K. Li, J. Chen, B. Yu, Z. Shen, C. Li, and S. He, "Supreme: fine-grained radio map reconstruction via spatial-temporal fusion network," in *2020 19th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*. IEEE, 2020, pp. 1–12.

[2] H. Zou, C.-L. Chen, M. Li, J. Yang, Y. Zhou, L. Xie, and C. J. Spanos, "Adversarial learning enabled automatic wifi indoor radio map construction and adaptation with mobile robot," *IEEE Internet of Things Journal*, 2020.

[3] J.-H. Seong and D.-H. Seo, "Selective unsupervised learning-based wi-fi fingerprint system using autoencoder and gan," *IEEE Internet of Things Journal*, vol. 7, no. 3, pp. 1898–1909, 2019.

[4] 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 on interactive, mobile, wearable and ubiquitous technologies*, vol. 1, no. 2, pp. 1–19, 2017.

[5] Z. HajiAkhondi-Meybodi, M. Salimibeni, A. Mohammadi, and K. N. Plataniotis, "Bluetooth low energy and cnn-based angle of arrival localization in presence of rayleigh fading," in *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2021, pp. 7913–7917.

[6] P. Spachos and K. N. Plataniotis, "Ble beacons for indoor positioning at an interactive iot-based smart museum," *IEEE Systems Journal*, 2020.

[7] O. Asraf, F. Shama, and I. Klein, "Pdrnet: A deep-learning pedestrian dead reckoning framework," *IEEE Sensors Journal*, 2021.

[8] B. Huang, G. Qi, X. Yang, L. Zhao, and H. Zou, "Exploiting cyclic features of walking for pedestrian dead reckoning with unconstrained smartphones," in *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, 2016, pp. 374–385.

[9] J. Dong, M. Noreikis, Y. Xiao, and A. Ylä-Jääski, "Vinav: A vision-based indoor navigation system for smartphones," *IEEE Transactions on Mobile Computing*, vol. 18, no. 6, pp. 1461–1475, 2018.

[10] Y. Huang, S. Mazuelas, F. Ge, and Y. Shen, "Indoor localization system with nlos mitigation based on self-training," *IEEE Transactions on Mobile Computing*, 2022.

[11] J. Yang, B. Dong, and J. Wang, "Vuloc: Accurate uwb localization for countless targets without synchronization," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 3, pp. 1–25, 2022.

[12] B. Guo, W. Zuo, S. Wang, W. Lyu, Z. Hong, Y. Ding, T. He, and D. Zhang, "Wepos: Weak-supervised indoor positioning with unlabeled wifi for on-demand delivery," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 2, pp. 1–25, 2022.

[13] X. Guo, N. R. Elikplim, N. Ansari, L. Li, and L. Wang, "Robust wifi localization by fusing derivative fingerprints of rss and multiple classifiers," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 5, pp. 3177–3186, 2019.

[14] B. Huang, G. Mao, Y. Qin, and Y. Wei, "Pedestrian flow estimation through passive wifi sensing," *IEEE Transactions on Mobile Computing*, 2019.

[15] B. Huang, R. Yang, B. Jia, W. Li, and G. Mao, "A theoretical analysis on sampling size in wifi fingerprint-based localization," *IEEE Transactions on Vehicular Technology*, vol. 70, no. 4, pp. 3599–3608, 2021.

[16] L. Li, X. Guo, Y. Zhang, N. Ansari, and H. Li, "Long short-term indoor positioning system via evolving knowledge transfer," *IEEE Transactions on Wireless Communications*, 2022.

[17] X. Du, X. Liao, M. Liu, and Z. Gao, "Crcloc: A crowdsourcing-based radio map construction method for wifi fingerprinting localization," *IEEE Internet of Things Journal*, 2021.

[18] T. Li, D. Han, Y. Chen, R. Zhang, Y. Zhang, and T. Hedgpeth, "Indoorwaze: A crowdsourcing-based context-aware indoor navigation system," *IEEE Transactions on Wireless Communications*, 2020.

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

[20] X. Tong, K. Liu, X. Tian, L. Fu, and X. Wang, "Fineloc: A fine-grained self-calibrating wireless indoor localization system," *IEEE Transactions on Mobile Computing*, vol. 18, no. 9, pp. 2077–2090, 2018.

[21] S.-P. Kuo and Y.-C. Tseng, "A scrambling method for fingerprint positioning based on temporal diversity and spatial dependency," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, no. 5, pp. 678–684, 2008.

[22] M. Ibrahim, M. Torki, and M. ElNainay, "Cnn based indoor localization using rss time-series," in *2018 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 2018, pp. 01 044–01 049.

[23] M. T. Hoang, B. Yuen, X. Dong, T. Lu, R. Westendorp, and K. Reddy, "Recurrent neural networks for accurate rssi indoor localization," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10 639–10 651, 2019.

[24] W. Liu, J. Li, Z. Deng, X. Fu, and Q. Cheng, "A calibrated-rssi/pdr/map intergrated system based on a novel particle filter for indoor navigation," in *2019 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2019, pp. 1–8.

[25] C. Li, X. Shen, Q. Ge, and W. Chen, "An enhanced transition model for unsupervised localization," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2021.

[26] Y. You and C. Wu, "Hybrid indoor positioning system for pedestrians with swinging arms based on smartphone imu and rssi of ble," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–15, 2021.

[27] 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.

[28] C. Chen, P. Zhao, C. X. Lu, W. Wang, A. Markham, and N. Trigoni, "Oxiod: The dataset for deep inertial odometry," *arXiv preprint arXiv:1809.07491*, 2018.

[29] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *Proceedings IEEE INFOCOM 2000. Conference on Computer Communications. Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (Cat. No. 00CH37064)*, vol. 2. Ieee, 2000, pp. 775–784.

[30] M. Youssef and A. Agrawala, "The horus wlan location determination system," in *Proceedings of the 3rd international conference on Mobile systems, applications, and services*, 2005, pp. 205–218.

[31] W. Zhao, S. Han, R. Q. Hu, W. Meng, and Z. Jia, "Crowdsourcing and multisource fusion-based fingerprint sensing in smartphone localization," *IEEE Sensors Journal*, vol. 18, no. 8, pp. 3236–3247, 2018.

[32] Q. Liang and M. Liu, "An automatic site survey approach for indoor localization using a smartphone," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 191–206, 2019.

[33] Y. Wei and R. Zheng, "Efficient wi-fi fingerprint crowdsourcing for indoor localization," *IEEE Sensors Journal*, 2021.

[34] Y. Zhao, Z. Zhang, T. Feng, W.-C. Wong, and H. K. Garg, "Graphips: Calibration-free and map-free indoor positioning using smartphone crowdsourced data," *IEEE Internet of Things Journal*, vol. 8, no. 1, pp. 393–406, 2020.

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

[36] C. Wu, Z. Yang, and C. Xiao, "Automatic radio map adaptation for indoor localization using smartphones," *IEEE Transactions on Mobile Computing*, vol. 17, no. 3, pp. 517–528, 2017.

[37] 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)*. IEEE, 2022, pp. 1–8.

[38] Y. Yu, W. Shi, R. Chen, L. Chen, S. Bao, and P. Chen, "Map-assisted seamless localization using crowdsourced trajectories data and bi-lstm based quality control criteria," *IEEE Sensors Journal*, vol. 22, no. 16, pp. 16 481–16 491, 2022.

[39] X. Shen, C. Li, W. Chen, and Y. Wang, "Mapict: Unsupervised radio-map learning from imbalanced crowd-sourced trajectories," *IEEE Sensors Journal*, vol. 22, no. 3, pp. 2399–2408, 2021.

[40] W. Li, C. Zhang, and Y. Tanaka, "Pseudo label-driven federated learning-based decentralized indoor localization via mobile crowdsourcing," *IEEE Sensors Journal*, vol. 20, no. 19, pp. 11 556–11 565, 2020.

[41] B. S. Ciftler, A. Albaseer, N. Lasla, and M. Abdallah, "Federated learning for rss fingerprint-based localization: A privacy-preserving crowdsourcing method," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*. IEEE, 2020, pp. 2112–2117.

[42] S. He, S.-H. G. Chan, L. Yu, and N. Liu, "Slac: Calibration-free pedometer-fingerprint fusion for indoor localization," *IEEE Transactions on Mobile Computing*, vol. 17, no. 5, pp. 1176–1189, 2017.

[43] ——, "Maxlifd: Joint maximum likelihood localization fusing fingerprints and mutual distances," *IEEE Transactions on Mobile Computing*, vol. 18, no. 3, pp. 602–617, 2018.

[44] H. Dai, W.-h. Ying, and J. Xu, "Multi-layer neural network for received signal strength-based indoor localisation," *IET Communications*, vol. 10, no. 6, pp. 717–723, 2016.

[45] M. Nowicki and J. Wietrzykowski, "Low-effort place recognition with wifi fingerprints using deep learning," in *International Conference Automation*. Springer, 2017, pp. 575–584.

[46] 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.

[47] D. Li, J. Xu, Z. Yang, Y. Lu, Q. Zhang, and X. Zhang, "Train once, locate anytime for anyone: Adversarial learning based wireless localization," in *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*. IEEE, 2021, pp. 1–10.

[48] J.-W. Jang and S.-N. Hong, "Indoor localization with wifi fingerprinting using convolutional neural network," in *2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE, 2018, pp. 753–758.

[49] C. Chen, Y. Miao, C. X. Lu, L. Xie, P. Blunsom, A. Markham, and N. Trigoni, "Motiontransformer: Transferring neural inertial tracking between domains," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 8009–8016.

[50] S. S. Saha, S. S. Sandha, L. A. Garcia, and M. Srivastava, "Tinyodom: Hardware-aware efficient neural inertial navigation," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 2, pp. 1–32, 2022.

[51] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: wireless indoor localization with little human intervention," in *Proceedings of the 18th annual international conference on Mobile computing and networking*, 2012, pp. 269–280.

[52] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," *arXiv preprint arXiv:1804.06872*, 2018.

[53] R. Harle, "A survey of indoor inertial positioning systems for pedestrians," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1281–1293, 2013.

[54] Y. Shu, K. G. Shin, T. He, and J. Chen, "Last-mile navigation using smartphones," in *Proceedings of the 21st annual international conference on mobile computing and networking*, 2015, pp. 512–524.

[55] M. A. Esfahani, H. Wang, K. Wu, and S. Yuan, "Aboldeepio: A novel deep inertial odometry network for autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 5, pp. 1941–1950, 2019.

[56] S. Sun, D. Melamed, and K. Kitani, "Idol: Inertial deep orientation-estimation and localization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 7, 2021, pp. 6128–6137.

[57] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7167–7176.

[58] X. Yang and B. Huang, "An accurate step detection algorithm using unconstrained smartphones," in *The 27th Chinese Control and Decision Conference (2015 CCDC)*. IEEE, 2015, pp. 5682–5687.

[59] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

Zhendong Xu received the B.E. degree in computer science from Inner Mongolia University, Hohhot, China, in 2017, where he is currently pursuing the Ph.D. degree at the College of Computer Science.

His current research interests include mobile computing and ubiquitous computing.

Baoqi Huang (S'08–M'12) received the Ph.D. degree in information engineering from Australian National University, Canberra, ACT, Australia, in 2012.

From 2013 to 2014, he was a Research Fellow with Nanyang Technological University, Singapore. He is with the College of Computer Science, Inner Mongolia University, Hohhot, China, where he is currently a Professor. His current research interests include wireless networks, sensor networks, mobile computing, and ubiquitous computing.

Dr. Huang was a recipient of the Chinese Government Award for Outstanding Chinese Students Abroad in 2011.

Bing Jia (M'16) received the Ph.D. degree from Jilin Univesity, Changchun, China, in 2013.

She is with the College of Computer Science, Inner Mongolia University, Hohhot, China, where she is currently an Associate Professor. Her current research interests include wireless networks, indoor localization, crowdsourcing, and mobile computing.

Guoqiang Mao (S'98-M'02-SM'08-F'18) is a Distinguished Professor and Dean of the Research Institute of Smart Transportation at Xidian University. Before that he was with University of Technology Sydney and the University of Sydney. He has published over 200 papers in international conferences and journals that have been cited more than 10,000 times. He is an editor of the IEEE Transactions on Intelligent Transportation Systems (since 2018), IEEE Transactions on Wireless Communications (2014-2019), IEEE Transactions on Vehicular Technology (2010-2020) and received "Top Editor" award for outstanding contributions to the IEEE Transactions on Vehicular Technology in 2011, 2014 and 2015. He was a co-chair of IEEE Intelligent Transport Systems Society Technical Committee on Communication Networks. He has served as a chair, co-chair and TPC member in a number of international conferences. He is a Fellow of IET. His research interest includes intelligent transport systems, Internet of Things, wireless localization techniques, wireless sensor networks, and applied graph theory and its applications in telecommunications.