High Accuracy Fingerprint Localization: A Robust Federated Learning Method

Jiaen Lv and Shaowei Wang
Key Laboratory of Optoelectronic Devices and Systems with Extreme Performances of MOE and
School of Electronic Science and Engineering, Nanjing University, Nanjing 210023, China
Email: DZ21230022@smail.nju.edu.cn, wangsw@nju.edu.cn

2024 IEEE/CIC International Conference on Communications in China (ICCC) | 979-8-3503-7841-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCC62479.2024.10681862

***Abstract*—Deep learning has been widely applied to fingerprint localization due to its powerful feature extraction capability. Traditionally, deep learning-based techniques require centralized privacy data collection, which may not be feasible in realistic application scenarios. To address this, a distributed collaborative framework called federated learning has been introduced. This enables fingerprint localization with data privacy requirements in mobile edge computing, where training occurs at distributed clients without the need for data sharing, and clients only need to upload model parameters to the central server for further processing. Nevertheless, the consequent issues of non-independent and identically distributed (non-IID) data and noise caused by decentralization significantly influence the performance of federated learning in practice. To this end, we propose a robust and fair federated learning framework based on Ditto to handle the fingerprint localization problem, where a regularization term is introduced for the loss function of local model training on clients to address noise-contaminated and non-IID data. Numerical results demonstrate that our proposal outperforms traditional federated learning methods in terms of localization accuracy.**

***Index Terms*—Deep learning, fairness, federated learning, fingerprint localization, mobile edge computing, robustness.**

I. INTRODUCTION

The localization of mobile devices holds paramount importance in multitude applications for mobile edge computing [1], including navigation, mapping, location-based services, and context-aware computing [2]. Currently, the global positioning system (GPS) is widely used for outdoor localization due to its global coverage and relatively high accuracy. Nevertheless, GPS signals can be obstructed by various factors, such as tall structures and dense foliage, leading to inaccurate localization results [3]. Moreover, GPS technology consumes a significant amount of power, which negatively impacts the battery life of mobile devices.

To address this issue, cell tower triangulation methods, such as time-based and angle-based localization methods, have been utilized [4]. In time-based methods, the time of arrival or time difference of arrival of incoming signals is measured to calculate the terminals' location. In angle-based methods, the angle of arrival of the received signals is measured to determine the terminals' location. These methods require the locations of the base station. Whereas, if base station location information is maliciously exploited, it can lead to network attacks or other security issues. Therefore, it is difficult for mobile devices to obtain such information [5].

Fingerprint-based localization technology has emerged as a promising alternative to address the limitations of GPS and other traditional methods [6]. As compared to GPS, fingerprint-based methods offer enhanced power efficiency by harnessing the distinctive characteristics of radio signals in various locations [7]. Unlike time-based and angle-based methods, fingerprint-based methods are independent of base station locations, which provide flexibility in establishing and updating the fingerprint database [8]. Consequently, fingerprint-based localization holds great potential for delivering precise and dependable location information [9].

By automatically extracting and representing features from raw signal data [10], deep learning techniques have revolutionized fingerprint-based localization. The authors in [11] propose an angle-delay channel amplitude matrix fingerprint extraction method based on deep convolutional neural network (CNN). The authors in [12] propose an interpolation-aided fingerprint-based localization system architecture, incorporating a deep autoencoder to handle missing samples and outliers.

Applying deep learning models needs a substantial amount of labeled training data, which requires aggregating small data sets collected from a large number of mobile users. Whereas, collecting a sufficient number of samples for training models is not only costly but also raising significant data privacy concerns, especially when location information is involved. Federated learning (FL) has emerged as a distributed deep learning framework for privacy protection [13], fundamental concept of which is to exchange the weights of clients' model rather than raw data. By enabling model training directly on clients' devices and model aggregating on a central server, FL eliminates the necessity for transmission of massive training samples and centralized storage of location data, thus ensuring the clients privacy and communication efficiency. Consequently, FL provides privacy protection and data security for users. In [14], a novel FL model update method for indoor localization is proposed, which leverages local client reliability through model uncertainty and employs Monte Carlo dropout for computation efficiency. In [15], a distributed hyper-parameter optimization scheme is proposed, which demonstrates near-centralized performance while preserving user privacy in diverse localization services.

In FL, the existence of non-independent and identically dis-

This work was partially supported by the National Natural Science Foundation of China under Grants 61931023.

tributed (non-IID) and noise-contaminated data across decentralized clients is a common challenge. The former refers to instances where the data distributions across clients vary, leading to differences in the characteristics of local datasets. The latter occurs when clients introduce errors or outliers during data collection or transmission. The variation in data distribution and noise can significantly impact the performance of FL models. For non-IID data, the model trained on one client may not generalize well to others, leading to suboptimal global model performance. The noise-contaminated data further exacerbates this issue by introducing uncertainties and inaccuracies into the learning process, potentially hindering convergence and degrading overall model accuracy. A model trained on non-IID and noisy data may exhibit vulnerabilities to adversarial attacks or fail to achieve equitable localization performance across different clients, resulting in a decrease in robustness and fairness. The studies mentioned above primarily focus on global localization accuracy by testing the averaging model on the overall test set, while often overlooking the crucial aspects of robustness and fairness in the face of noisy and non-IID data. By considering these aspects, our proposed FL-based fingerprint localization method can provide more reliable and accurate results in real-world scenarios.

In this paper, we present a robust and fair FL framework for fingerprint-based localization with the consideration of data privacy. The localization method is decomposed into two parts: the fingerprint database construction and the database matching. For the database construction, we utilize a densely connected convolutional network (DenseNet) with a feature fusion network, which can effectively capture the unique characteristics of fingerprints across different locations. To train the model, we leverage the Ditto-based FL framework, which introduces a regularization term for the loss function of local model training on clients, facilitating the model to handle non-IID and noise-contaminated data and achieve robustness [16]. For the database matching, we provide a two-stage matching method, where the rough matching range is determined based on the information of the serving cell, and the final position matching is achieved by utilizing the information of neighboring cells. Overall, our proposed FL framework achieves high location accuracy while ensuring the robustness and fairness of FL, making it a promising solution for fingerprint-based localization.

II. PROBLEM FORMULATION

An FL based outdoor localization system with B base stations and K participated clients is considered. The received signal strength (RSS) is used as the fingerprint, a measurement of the power present in a received radio signal. And the E-UTRAN cell identifier (ECI) is used to identify cells. As shown in Fig. 1, the RSS measurements and location information reported by clients are exploited to construct the fingerprint database of the target route, which helps realize the localization of the clients.

We assume that the route is divided into P points, and the distance between each point is δ_p. The set of points is denoted by $\mathcal{G} = \{\mathbf{c}_p\}_{p=1}^{P}$, where $\mathbf{c}_p$ is a two-dimensional coordinate. Let $\mathbf{R}_p$ denotes the RSS measurement of B_p cells, including a serving cell and several neighboring cells at a point $\mathbf{c}_p$, given by

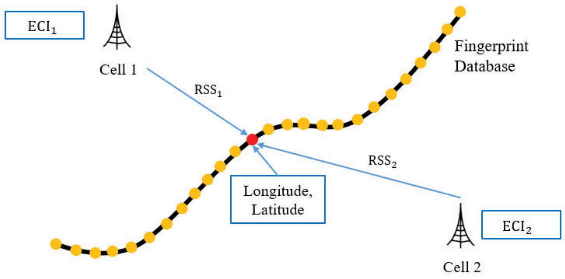

Fig. 1: An illustration of the fingerprint database.

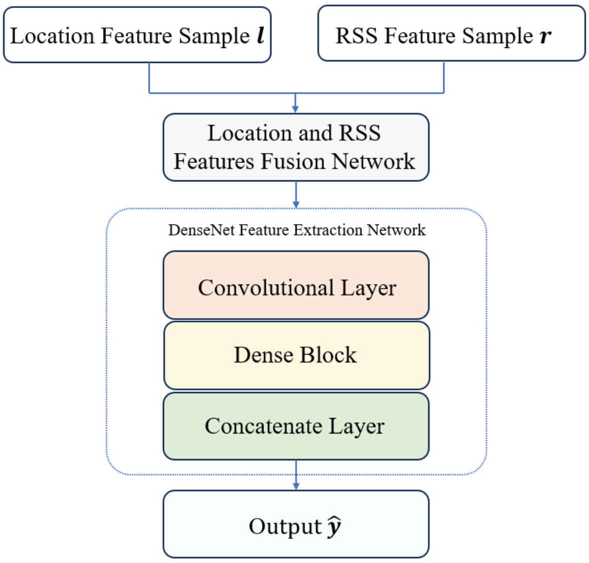

Fig. 2: DenseNet based RSS estimation network.

$$\mathbf{R}_p = \begin{bmatrix} ECI_1 & \cdots & ECI_b & \cdots & ECI_{B_p} \\ RSS_1 & \cdots & RSS_b & \cdots & RSS_{B_p} \end{bmatrix}^{\mathrm{T}}, \quad (1)$$

where ECI_b denotes the ECI of a cell b and RSS_b denotes the corresponding RSS value. Specifically, ECI_1 and RSS_1 denote the ECI and RSS values of the serving cell at a point, and others denote the neighboring cells. Let $\mathcal{G}_N = \{\mathbf{c}_n\}_{n=1}^{N}$ be the set of N non-measured points and $\mathcal{G}_M = \{\mathbf{c}_m\}_{m=1}^{M}$ be the set of M measured points. $\mathbf{R}_m$ denotes the RSS measurement at point $\mathbf{c}_m$ and $\hat{\mathbf{R}}_n$ denotes RSS value at non-measured point $\mathbf{c}_n$.

The construction of fingerprint database involves estimating the RSS values at all non-measured points with RSS values and location information of the measured points. Based on the fingerprint database, a client can be located by matching its RSS measurement $\mathbf{R}'$ with the fingerprint database to accurately determine the location.

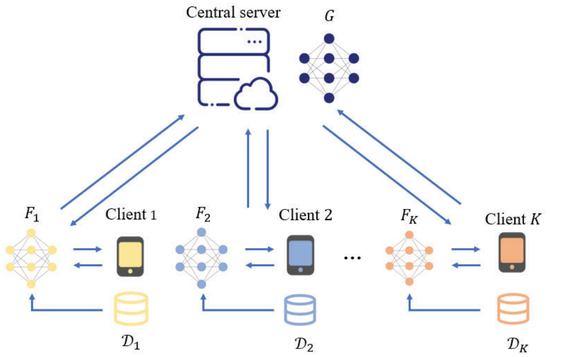

Fig. 3: FL-based fingerprint database construction framework.

III. DENSENET BASED FINGERPRINT DATABASE CONSTRUCTION SCHEME

The process of building the fingerprint database entails the estimation of RSS values at unmeasured points, a task reliant on the analysis of spatial correlation within the RSS values. Deep learning, renowned for its proficiency in capturing spatial relationship across features, proves to be an effective methodology for addressing the challenge of estimating RSS values at unmeasured points.

Subsequently, the DenseNet is employed to exploit the relationship across RSS values at distinct locations [17]. DenseNet's efficiency in feature extraction and modeling intricate spatial dependencies makes it an apt choice for this endeavor. For a non-measured point, we use the RSS measurements and location information of neighboring N_c points. We have N_s samples for training with the ith original sample consisting of a feature sample $\mathbf{x}_i$ and a corresponding label $\mathbf{y}_i$. The feature sample $\mathbf{x}_i$ consists of a location feature sample $\mathbf{l}_i$ and an RSS feature sample $\mathbf{r}_i$. $\mathbf{l}_i \in \mathbb{R}^{N_c \times 2}$ denotes the coordinates of the neighboring N_c points and the non-measured points. $\mathbf{r}_i \in \mathbb{R}^{N_c \times B_p}$ denotes the RSS of B_p cells from N_c points. The label $\mathbf{y}_i$ denotes the RSS value of the non-measured point. The input data for the model includes the coordinates of unmeasured points, the coordinates of measured points, and the corresponding RSS values. The output data comprises the RSS values for the unmeasured points.

As shown in Fig. 2, to employ feature extraction from the input data, we introduce a location and RSS features fusion network, which contains a convolutional layer, a batch normalization layer, and an activation layer [18]. This network exploits the relationship between distance information and RSS values. The DenseNet feature extraction network, which contains a convolutional layer, a dense block, and a concatenate layer will further exploit the spatial correlation of the RSS values across points. This integration empowers the neural network to concurrently capture distance information between target and known locations, alongside RSS information, thereby effectively exploring the spatial correlation among RSS values.

IV. ROBUST AND FAIR FEDERATED LEARNING FRAMEWORK

A. A Federated Learning Framework

Consider a fingerprint database construction system with K clients. The global location and RSS values dataset is given by $\mathcal{D} = \{(\mathbf{c}_p, \mathbf{R}_p))\}_{p=1}^{P}$. To exploit the spatial correlation of RSS values at different points while protecting the clients' private location-specific data, we have adopted an FL framework to train the model [19]. The k-th client needs to train the local model F_k based on the local dataset $\mathcal{D}_k \subset \mathcal{D}$. Then the local model will be uploaded to the center for aggregating the global model G.

As shown in Fig. 3, the FL framework consists of information interaction between the central server and clients with T global rounds. At the t-th round, the FL system includes four steps: global model initialization, local model training, global model aggregation and model fusion.

Global Model Initialization: The central server initiates the global model based on historical fingerprint data. This global model serves as the cornerstone for collaborative model training across decentralized clients. It acts as the starting point for following iterations, during which local models are fine-tuned and integrated to improve performance.

Local Model Training: In this step, each participating client receives a copy of the global model G. Using local dataset with N_k samples, each client independently train their local model F_k and update the model weight ω_k. The objective is to minimize the loss function $L_k(\omega_k)$, which can be formulated as:

$$\min_{\omega_k} \quad L_k(\omega_k) = \frac{1}{N_k} \sum_{i=1}^{N_k} L(y_i, F_k(\mathbf{x}_i); \omega_k) \tag{2}$$

The training involves multiple epochs, enabling the models to adapt to the specific patterns and characteristics within each client's local data. During training, model weights are updated using stochastic gradient descent method.

Global Model Aggregation: The updated local model weight ω_k is then transmitted to the central server. The server aggregates these updates to create an updated global model G with weight ω. This aggregation step can employ techniques like federated averaging (FedAvg) algorithm [13] to combine the local model effectively, which can be given by:

$$\omega = \sum_{k=1}^{K} \frac{|D_k|}{\sum_{k=1}^{K} |D_k|} \omega_k, \tag{3}$$

Model Fusion: The newly generated global model is shared back with the individual clients. This process merges the global knowledge with the client-specific information, enabling the models to benefit from both global trends and local insights.

B. Robustness and Fairness in Federated Learning

FL stands to produce highly accurate models by aggregating knowledge from disparate data sources. Nevertheless, to deploy FL in practice, it is necessary for the resulting systems to be not only accurate, but to also satisfy a number of pragmatic constraints regarding issues such as fairness and robustness.

Specifically, robustness refers to the system's resilience to outliers or adversarial behaviors in data, ensuring that the model can perform consistently across diverse and challenging scenarios. Fairness, on the other hand, underscores the ethical and equitable treatment of data from various sources, preventing biases and discrimination in model predictions. Both robustness and fairness are essential in FL to guarantee the reliability and equity of the collaborative model, especially when dealing with non-IID, noise-contaminated data sources and complex real-world applications. These principles not only enhance the model's performance but also contribute to its inclusive use in various domains, making them fundamental considerations in the FL framework. Nevertheless, simultaneously satisfying these varied constraints can be exceptionally difficult.

In outdoor fingerprint database construction system, RSS value at each point may be affected by different factors in urban environment, such as distance-dependent propagation loss and small-scale fading, leading to the fact that the data collected from different clients are non-IID. Moreover, clients may introduce errors or outliers during data collection or transmission, which makes the data noise-contaminated. These are the main causes leading to a decline in fairness and robustness.

To measure robustness, we assess the mean test performance on benign clients without noise-contaminated data. Specifically, we consider model F_1 to be more robust than F_2 to a specific attack if the mean test performance across the benign clients is higher for model F_1 than F_2 after training with the attack. A model F_1 with weight ω_1 is more fair than F_2 with weight ω_2 if the test performance distribution of F_1 across the network is more uniform than that of F_2, i.e., $\text{std}\{L_k(\omega_1)\}_{k=1}^{K} < \text{std}\{L_k(\omega_2)\}_{k=1}^{K}$, where $L_k(.)$ denotes the test loss on client k, and $\text{std}\{.\}$ denotes the standard deviation. Due to the non-IID data in FL networks, it is possible that the performance of a model varying significantly across the clients. This concern, also known as representation disparity, is a major challenge in FL, as it can potentially result in uneven outcomes for the clients.

To explore the possible fairness and robustness benefits, we introduce a global-regularized FL framework which proposed a regularized objective named Ditto for local model training [16]. A regularization term is incorporated to encourage the personalized models to be close to the optimal global model. The resulting optimization problem for each client k is given by:

$$\min_{\omega_k} \quad h_k\left(\omega_k; \omega\right) := L_k\left(\omega_k\right) + \frac{\lambda}{2}\left\|\omega_k - \omega\right\|^2. \qquad (4)$$

Here the hyperparameter λ controls the interpolation between local and global models. When λ is set to 0, Ditto is reduced to training local models; as λ grows large, it recovers global model objective. Without adversaries, learning a single global model is optimal for generalization. Whereas, in the presence of adversaries, learning globally might introduce corruption, while learning local models may not generalize well due to limited sample size. Ditto with an appropriate value of λ offers a tradeoff between these two extremes: the smaller λ, the more the local model F_k can deviate from the global model G, potentially providing robustness at the expense of generalization. In the heterogeneous case, which can lead to issues of unfairness, a finite λ exists to offer robustness and fairness jointly. The Ditto based FL algorithm is summarized in Algorithm 1.

Algorithm 1 Ditto Based FL for Fingerprint Database Construction

```
for $t = 1 : T$ do
    Server sends $\omega^t$ to $K$ clients;
    for $i = 1 : K$ do
        $\omega_k^t = \omega^t$;
        Update $\omega_k^t$ for $s$ local epochs:
        $\omega_k^{t,s+1} = \omega_k^{t,s} - \eta_l \left(\nabla F_k \left(\omega_k^{t,s}\right) + \lambda \left(\omega_k^{t,s} - \omega^t\right);$
    end for
    Server updates $\omega^{t+1}$ as (4);
end for
return $\{\omega_k\}_{k=1}^{K}$, $\omega$.
```

Algorithm 2 Fingerprint Based location Algorithm

```
Initialize $\alpha^* = 0, S^* = \infty, \mathbf{c}^* = (0,0), \mathcal{C} = \emptyset$;
for $i = 1 : P$ do
    for $j = 1 : B_p$ do
        if $\mathbf{R}_i(j,1) = \mathbf{R}'(1,1)$ then
            add $\mathbf{R}_i$ to the set $\mathcal{C}$;
        end if
    end for
end for
for $(\mathbf{R}, \mathbf{c}) \in \mathcal{C}$ do
    for $m = 1 : B_p$ do
        for $n = 1 : B'_p$ do
            if $\mathbf{R}(m,1) = \mathbf{R}'(n,1)$ then
                $\alpha = \alpha + 1$;
            end if
        end for
    end for
    if $\alpha \geq \alpha^*$ then
        Calculate $S$ as (5);
        $\alpha^* = \alpha$;
        if $S < S^*$ then
            $S^* = S$;
            $\mathbf{c}^* = \mathbf{c}$;
        end if
    end if
end for
return $c^*$
```

V. TWO-STAGE DATABASE MATCHING ALGORITHM

In this section, we provide a fingerprint-based localization method. As outlined in previous chapters, a database encompassing ECI and the associated RSS measurements of each point is constructed. The location algorithm aims to determine the coordinates $\mathbf{c}^*$ for a client with RSS measurement report $\mathbf{R}'$ based on the fingerprint database $\mathcal{D} = \{(\mathbf{c}_p, \mathbf{R}_p))\}_{p=1}^{P}$. Nevertheless, due to the large number of points in the database, an exhaustive search to find the target point is impractical. To address this, we propose a two-stage location algorithm that utilizes information from the serving cells and neighboring cells.

In the first stage, the algorithm compares the ECI of the serving cell with the entries in $\mathcal{D}$, resulting in a set $\mathcal{C}$ containing potential target points. This step serves as an initial coarse localization and effectively reduces the search space.

In the second stage, each $(\mathbf{R}, \mathbf{c}) \in \mathcal{C}$ is matched with $\mathbf{R}'$. Firstly, the number of cells with the same ECI between $\mathbf{R}$ and $\mathbf{R}'$ is calculated as α and the maximum α is selected as α^*. Among $\mathbf{R}$ with α^*, the sum of RSS difference between $\mathbf{R}$ and $\mathbf{R}'$ is calculated as S, given by

$$S = \sum_{l=1}^{\alpha^*} |\mathbf{R}(l,2) - \mathbf{R}'(l,2)|, \tag{5}$$

and the minimum S is selected as S^*. The element $\mathbf{c}$ corresponding to α^* and S^* is identified as the coordinates matched to $\mathbf{R}'$. The details of this two-stage location algorithm are summarized in Algorithm 2.

The proposed algorithm first utilizes the serving cell of the target point to narrow down the search space. Then, based on the RSS values from the neighboring cells of the target point, the algorithm identifies the point in the database with the closest RSS values, thereby determining the location of the target point.

VI. NUMERICAL RESULTS

The dataset used in the experiments was obtained through on-site measurements and derived from the M176 bus route with a total length of 2km in Shenzhen, Guangdong Province, China. Data collection occurred at intervals of 5 meters, capturing information such as the GPS coordinates, ECI, and RSS values of the current serving cell and neighboring cells at each point. We consider a localization system with $K = 4$ clients, each located at different segments along the route.

We will compare the proposed DenseNet-Ditto method with traditional convolutional neural network (CNN) and FedAvg FL framework [13]. Specifically, we compare the proposed method with the CNN-FedAvg method, the CNN-Ditto method, and the DenseNet-FedAvg method. Firstly, we will evaluate the construction accuracy of the fingerprint database using the root mean square error (RMSE) and standard deviation (SD). Additionally, we will compare the localization accuracy achieved by different algorithms when using the constructed fingerprint database, measured by the mean average error (MAE). The local model at each client is trained with 100 epochs per round, a batch size of 64, and the mean squared error loss function.

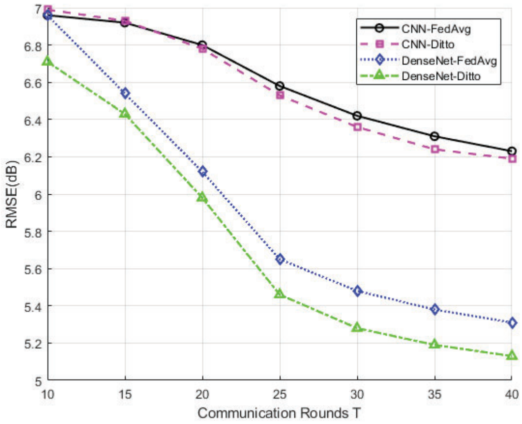

Fig. 4: RMSE against communication rounds.

Fig. 4 shows the RMSE against communication rounds, which reflects the performance of the global model for each method. As the number of communication rounds increases, the global model can better integrate data information from different clients, resulting in a gradual decrease in RMSE. The RMSE of the CNN-FedAvg method and the CNN-Ditto method is approximately 0.8dB higher than that of the DenseNet-FedAvg and the DenseNet-Ditto method, indicating that DenseNet with the feature fusion network can effectively explore the relationship between RSS values and location. When comparing the two Ditto-based methods with the two FedAvg-based methods, there is a reduction in RMSE by 0.1-0.2dB, demonstrating that the Ditto method can enhance the performance of the model.

TABLE I: RMSE and SD of the Methods.

Method	CNN-FedAvg	CNN-Ditto	DenseNet-FedAvg	DenseNet-Ditto
RMSE(dB)	6.24	6.18	5.33	5.15
SD(dB)	2.32	2.11	2.23	2.09

TABLE II: RMSE and SD of the Methods with Noisy Data.

Method	CNN-FedAvg	CNN-Ditto	DenseNet-FedAvg	DenseNet-Ditto
RMSE(dB)	7.92	6.79	6.94	5.67
SD(dB)	2.43	2.18	2.45	2.14

TABLE III: MAE of the Methods.

Method	CNN-FedAvg	CNN-Ditto	DenseNet-FedAvg	DenseNet-Ditto
MAE(m)	38.8	36.4	32.1	30.9

Table I shows the precision of fingerprint database construction measured by RMSE and the fairness measured by SD. The communication rounds T between the clients and the server are set to 40. RMSE reflects the average performance of the local models in constructing the fingerprint database, while SD captures the differences in model estimation accuracy among different clients, thus reflecting the fairness of FL. The RMSE of the DenseNet-based method is more than 2 dB lower than that of the CNN-based method, and the RMSE of the Ditto-based methods is 0.6-0.2 dB lower than that of the FedAvg-based methods, indicating that DenseNet and Ditto are helpful in improving the estimation accuracy of the local models. The variability of the estimation results is similar between the CNN-based and DenseNet-based methods, but the Ditto-based methods have a lower SD by 0.1-0.2 dB compared to the FedAvg-based methods, demonstrating that Ditto can enhance the fairness of the FL framework.

Table II shows the estimation precision measured by RMSE, and the fairness measured by SD in the presence of noise-contaminated training data. The communication rounds T between the clients and the server are set to 40. In comparison to Table I, the FedAvg-based methods show an increase of 1.6 dB in RMSE and a 0.1-0.2 dB increase in SD. Whereas, the Ditto-based method demonstrates a smaller increase of only 0.6 dB in RMSE and a 0.1 dB increase in SD. These findings suggest that the Ditto-based method performs better when dealing with noise-contaminated training data, showcasing its stronger robustness in such scenarios.

Table III shows the MAE of the proposed localization algorithm under different methods for constructing the fingerprint database. The database constructed using the DenseNet-Ditto method demonstrates a reduction of approximately 8 meters in MAE compared to the CNN-FedAvg method. This result indicates that the DenseNet network with a feature fusion structure and the Ditto structure can effectively enhance the accuracy of fingerprint-based localization.

VII. CONCLUSION

In this paper, we considered an FL framework for fingerprint-based localization. We addressed the challenges of noise-contaminated and non-IID data in FL-based localization systems by decomposing the method into fingerprint database construction and database matching. We utilized DenseNet with a feature fusion network for constructing the database and leveraged the privacy-preserving Ditto-based FL framework. Our approach ensured robustness and fairness by introducing a regularization term for the loss function of local model training to handle noisy and non-IID data. For the database matching, we proposed a two-stage method based on serving and neighboring cell information. Overall, our work presented a Ditto-based FL framework that addressed robustness, fairness, and privacy preservation in fingerprint-based localization. Numerical results demonstrated that, with higher robustness and fairness of the model, our proposal outperformed traditional FL methods in terms of localization accuracy.

REFERENCES

[1] S. He *et al.*, "Mobility-driven user-centric ap clustering in mobile edge computing-based ultra-dense networks," *Digital Commun.*, vol. 6, no. 2, pp. 210–216, May 2020.

[2] N. Saeed *et al.*, "A state-of-the-art survey on multidimensional scaling-based localization techniques," *IEEE Commun. Surv. Tutor.*, vol. 21, no. 4, pp. 3565–3583, Jun. 2019.

[3] Q. D. Vo and P. De, "A survey of fingerprint-based outdoor localization," *IEEE Commun. Surv. Tutor.*, vol. 18, no. 1, pp. 491–506, Jun. 2016.

[4] N. BniLam *et al.*, "Loray: Aoa estimation system for long range communication networks," *IEEE Trans. Wirel. Commun.*, vol. 20, no. 3, pp. 2005–2018, Mar. 2021.

[5] N. Baroutis and M. Younis, "Load-conscious maximization of base-station location privacy in wireless sensor networks," *Comput. Netw.*, vol. 124, pp. 126–139, Sep. 2017.

[6] S. Yiu *et al.*, "Wireless RSSI fingerprinting localization," *Signal Process.*, vol. 131, pp. 235–244, Feb. 2017.

[7] Y. Zhang and S. Wang, "K-nearest neighbors gaussian process regression for urban radio map reconstruction," *IEEE Commun. Lett.*, vol. 26, no. 12, pp. 3049–3053, Dec. 2022.

[8] X. Wang *et al.*, "Indoor radio map construction and localization with deep gaussian processes," *IEEE Internet Things J.*, vol. 7, no. 11, pp. 11 238–11 249, Nov. 2020.

[9] Z. Gao *et al.*, "Crisloc: Reconstructable CSI fingerprinting for indoor smartphone localization," *IEEE Internet Things J.*, vol. 8, no. 5, pp. 3422–3437, Mar. 2021.

[10] S. Liu *et al.*, "Hardware impairment estimation in nb-iot: A parallel multitask learning method," *IEEE Internet Things J.*, vol. 10, no. 8, pp. 6859–6869, Apr. 2023.

[11] X. Sun *et al.*, "Fingerprint-based localization for massive mimo-ofdm system with deep convolutional neural networks," *IEEE Trans. Veh. Technol.*, vol. 68, no. 11, pp. 10 846–10 857, Nov. 2019.

[12] J. Purohit *et al.*, "Fingerprinting-based indoor and outdoor localization with LoRa and deep learning," in *Proc. IEEE GLOBECOM'20*, Taipei, Taiwan, Dec. 2020.

[13] D. C. Nguyen *et al.*, "Federated learning for internet of things: A comprehensive survey," *IEEE Commun. Surv. Tutor.*, vol. 23, no. 3, pp. 1622–1658, Apr. 2021.

[14] J. Park *et al.*, "Federated learning for indoor localization via model reliability with dropout," *IEEE Commun. Lett.*, vol. 26, no. 7, pp. 1553–1557, Jul. 2022.

[15] F. Yin *et al.*, "Fedloc: Federated learning framework for data-driven cooperative localization and location data processing," *IEEE Open J. Signal Process.*, vol. 1, pp. 187–215, Nov. 2020.

[16] T. Li *et al.*, "Ditto: Fair and robust federated learning through personalization," in *Proc. ICML'21*, Jul. 2021.

[17] G. Huang *et al.*, "Condensenet: An efficient densenet using learned group convolutions," in *Proc. IEEE CVPR'18*, Salt Lake City, UT, USA, Jun. 2018.

[18] Q.-S. Sun *et al.*, "A new method of feature fusion and its application in image recognition," *Pattern Recognit*, vol. 38, no. 12, pp. 2437–2448, Dec. 2005.

[19] G. Huang *et al.*, "Condensenet: An efficient densenet using learned group convolutions," in *Proc. IEEE CVPR'18*, Salt Lake City, UT, USA, Jun. 2018.