Receiver-Agnostic Radio Frequency Fingerprinting Based on Two-stage Unsupervised Domain Adaptation and Fine-tuning

Jiazhong Bao*, Xin Xie*, Zhaoyi Lu*, Jianan Hong*†, Cunqing Hua*
*School of Cyber Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
†hongjn@sjtu.edu.cn

GLOBECOM 2023 - 2023 IEEE Global Communications Conference | 979-8-3503-1090-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/GLOBECOM54140.2023.10437074

***Abstract*—Radio frequency fingerprint identification (RFFI) has been widely studied as a physical layer security scheme for device identification and authentication in wireless scenarios, such as Internet of Things (IoTs), industrial wireless networks, Internet of Vehicles (IoV), etc.. Typical RFFI approaches train a model at the receiver to extract hardware defects of the transmitter RF front-end using a deep learning-based method and achieve classification. However, few works have taken into account its shortage in multiple-receiver scenarios, where the identification accuracy significantly decreases when migrating a model trained on the known receivers to the new ones, directly. In this paper, we propose a novel cross-receiver RFFI scheme to improve the performance and the generalization of the fingerprinting classification tasks on new receivers. This scheme tackles the shortage by two means: 1) we extract receiver-independent features using global domain adaptation based on adversarial training and relevant subdomain adaptation based on local maximum mean discrepancy (LMMD); 2) The performance is further improved by fine-tuning on few labeled samples when domain adaptation is not effective. The second mechanism brings in significant performance advantage, without a large amount of labeled data on new receivers. Experimental results on public datasets show the outstanding performance of the proposed scheme in cross-receiver scenarios.**

***Index Terms*—RFFI, cross-receiver, domain adaptation, fine-tuning.**

I. Introduction

The widespread use of IoT devices and the increasing reliance on 5G networks to facilitate their communication [1], [2] have made it imperative to ensure the security and privacy of 5G IoT wireless networks. In this regard, the authentication of IoT devices plays a critical role in verifying the legitimacy of the devices. However, traditional cryptographic authentication schemes, although widely used, have several drawbacks when applied to the dynamic and resource-constrained environment of 5G IoT. One of the key issues is their reliance on pre-shared keys or certificates, which poses difficulties in large-scale IoT networks for management and distribution purposes. Moreover, these schemes often lack the flexibility to adapt to the dynamic and heterogeneous nature of IoT environments, where devices exhibit different capabilities and constraints. Finally, the computational overhead required for traditional cryptographic algorithms is not viable for resource-constrained IoT devices.

Radio frequency fingerprint identification (RFFI) has emerged as a promising non-cryptographic technique to solve the above problems. The distinct radio frequency fingerprints (RFFs) possessed by wireless devices are attributed to the imperfections in the hardware components of their transmitter front-end, which are challenging to replicate. These hardware imperfections, such as I-Q imbalance and power amplifier nonlinearity [3], arise due to circuit design and manufacturing, and are present even in devices of the same model and product line [4], [5], leading to small and unique distortions in the actual RF signals at the receiver. Previous studies [6], [7] have shown that hardware features extracted from wireless signals transmitted by IoT devices can be utilized to infer their identities. The deep learning-based RFFI scheme has been widely studied in literature [3], [8]–[10] due to its advantages such as low reliance on prior knowledge, end-to-end learning paradigm, and the ability to learn complex features.

However, the RFFI scheme incurs problems when deployed in highly dynamic networks. When the RF fingerprint model trained on known receivers is directly deployed to a new receiver, the transmitter classification performance of the model will significantly drop. The main reason is that the received signal not only contains the characteristics of the transmitter chain but is also affected by the receiver chain. Although some previous work [11]–[14] has been devoted to extracting receiver-independent RF fingerprints, they have to label datasets on all new receivers, which is not feasible in 5G network authentication scenarios.

In this paper, we attempt to improve the classification performance on the new receiver using unlabeled data on the new receiver, thereby reducing deployment costs. To this end, we propose a novel cross-receiver RFFI scheme. The core idea of this scheme is to learn receiver-independent RF fingerprints using labeled data from known receivers and unlabeled data from new receivers, and to further improve performance on new receivers through fine-tuning when necessary. The contributions of the paper are summarized as follows:

- We approach receiver distortion by treating it as a shift in data distribution and propose a two-stage unsupervised domain adaptation process. Adversarial training is used for global domain adaptation, followed by local maximum mean discrepancy (LMMD) based subdomain adaptation for fine-grained data distribution alignment, ultimately resulting in a receiver-independent RFFI model.
- To enhance the performance on new receivers with significant data distribution differences, we propose a fine-tuning scheme based on few-sample selection strategies. This scheme reduces the number of labeled samples required for the new receivers and maintains the model's

classification performance on previous receivers.

The remainder of the paper is organized as follows. Section II introduces the system model of RF fingerprint and the adverse effects of receiver chain. In Section III, we introduce the proposed cross-receiver RFFI scheme. The performance of the proposed scheme is evaluated in Section IV, and conclusions are drawn in Section V.

II. RF Fingerprinting Problem

We consider the conventional RF fingerprinting problem in a wireless communication system consisting of K transmitters and one receiver. The received baseband signal $y(t)$ can be mathematically given by:

$$y(t) = G(h(t) * F^k(s(t))) + n(t), \tag{1}$$

where $s(t)$ is the baseband signal from the k-th transmitter, $F^k(\cdot)$ represents the front-end hardware components effect, $h(t)$ is the wireless channel impulse response, $G(\cdot)$ denotes the hardware effects of the receiver, $n(t)$ is the additive white Gaussian noise, and $*$ denotes convolution. The received signal $y(t)$ is used to extract the RF fingerprint to identify the transmitter.

The deep learning-based RFFI is modeled as a multi-class classification problem [3], [8]–[10] and consists of two stages: training and inference. In the training phase, the receiver collects signals from K transmitters to construct the training dataset $\mathcal{D}^{train}$:

$$\mathcal{D}^{train} = \{(\mathbf{x_1}, \mathbf{y_1}), (\mathbf{x_2}, \mathbf{y_2}), ..., (\mathbf{x_m}, \mathbf{y_m}), ..., (\mathbf{x_M}, \mathbf{y_M})\}, \tag{2}$$

where $\mathbf{x_m}$ represents the training sample m, $\mathbf{y_m}$ is the corresponding one-hot encoded label and M denotes the number of training samples. Then we can build a neural network f and obtain its parameters Θ by:

$$\Theta = \arg\min_{\Theta} \sum_{(\mathbf{x_m}, \mathbf{y_m}) \in D^{train}} J(f(\mathbf{x_m}; \Theta), \mathbf{y_m}), \tag{3}$$

where $J(\cdot, \cdot)$ is the cross-entropy loss function. In the inference phase, the receiver captures a signal $\mathbf{x}'$ in real time and makes prediction with the well-trained model $f(\cdot; \cdot)$:

$$\hat{\mathbf{y}} = f(\mathbf{x}'; \Theta), \tag{4}$$

$$\hat{l} = \arg\max_{k}(\hat{\mathbf{y}}), \tag{5}$$

where $\hat{l}$ is the predicted transmitter label.

However, the aforementioned scheme encounters problems in practical scenarios. In highly dynamic networks such as the UAV and IoV networks, vehicles or drones frequently switch between different base stations or access points. During the training phase, a large amount of training data with transmitter labels can be collected at one or several access points to obtain a well-performing model. However, during the inference phase, the vehicles or drones may switch to a new (unseen during training) access point, where the receiver effect $G'(\cdot)$ during inference may differ from the receiver effect $G(\cdot)$ during training. This can lead to degraduation of classification accuracy of the previously trained model on the new receiver, as it violates the basic assumption of independent and identically distributed data.

In order to resolve this issue, a straightforward approach is to gather a substantial amount of data with transmitter labels on the new receiver when the drone or vehicle switches to a new access point. This would expand the previous training set and allow for retraining on the expanded dataset. As a result, the updated model can achieve high classification accuracy on all known receivers. To accomplish this, it can be assumed that there is a central node connected to all access points. This central node collects training data from access points, completes model training and sends the trained model to all access points for inference at their respective access points. However, this solution is undesriable, the reasons are twofold: (i) in practical scenarios, labeling data is very expensive; (ii) Furthermore, in order to complete the retraining of the model, the new access point needs to transmit a large amount of data to the central node.

III. Cross-receiver RFFI Method

A. System Overview

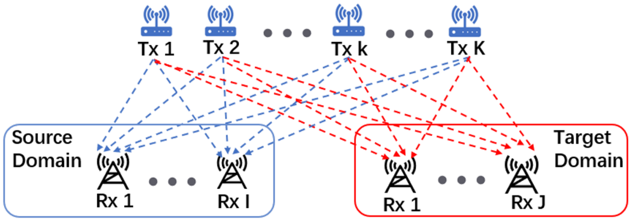


Fig. 1. Overview of cross-receiver scenarios.

As shown in Fig 1. we consider IoT scenarios such as the UAV and IoV networks. Due to the mobility of the devices, multiple access points are deployed to provide network services, and devices frequently switch between different access points. Authentication of the transmitter is achieved through the radio frequency fingerprint of the transmitter at the access points (receivers). However, the accuracy of authentication will degraduate severely if the model trained on the source receiver is directly adopted by the new receiver as disucssed in previous section.

As a solution, we propose a novel RFF learning scheme based on adversarial training and subdomain adaptation, and the optional few-sample fine-tuning scheme. Adversarial training and subdomain adaptation is applied in situations where it is impossible to label which transmitter the samples collected by the new receiver come from or the labeling cost is high, and requires new receivers to provide unlabeled samples to extract receiver-invariant features as radio frequency fingerprints. Adversarial training is implemented through deep adversarial neural network(DANN) [15], which mainly focuses on learning the global domain shift, without considering the relationship between subdomains within the same category but different domains, resulting in unsatisfactory performance. Therefore, we utilize local maximum mean difference (LMMD) to align the relevant subdomain distributions of domain-specific layer activations in different domains after adversarial training. Nevertheless, when the source and target domains have a significant difference, the performance of adversarial training and subdomain adaptation is inadequate.

Few-sample fine-tuning is optional, which can be employed when further improvement is required. The parameters of the trained neural network are slightly updated using a small number of target domain samples which are selected by uncertain sampling methods. In order to ensure the performance on the source domain, a small number of samples on the source domain are also used for fine-tuning.

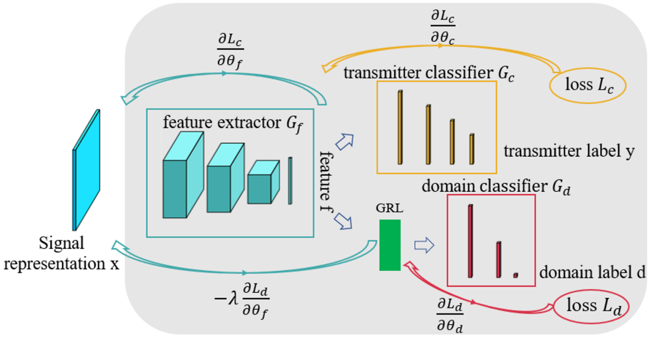

$\frac{\partial L_c}{\partial \theta_f}$
$\frac{\partial L_c}{\partial \theta_c}$
transmitter classifier G_c
loss L_c
feature extractor G_f
feature f
transmitter label y
domain classifier G_d
GRL
Signal
representation x
domain label d
$-\lambda \frac{\partial L_d}{\partial \theta_f}$
$\frac{\partial L_d}{\partial \theta_d}$
loss L_d

Fig. 2. The adversarial-based model architecture.

B. Adversarial Training and Subdomain Adaptation on Unlabeled Target Domain Samples

We have the source and target datasets after signal collection and pre-processing. In the cross-receiver scenario, we are given a source domain of N_s labeled samples as

$$\mathcal{D}_s = \begin{Bmatrix} (\mathbf{x}^s_{1,1}, \mathbf{y}^s_{1,1}) & \cdots & (\mathbf{x}^s_{1,I}, \mathbf{y}^s_{1,I}) \\ \vdots & \ddots & \vdots \\ (\mathbf{x}^s_{N_s,1}, \mathbf{y}^s_{N_s,1}) & \cdots & (\mathbf{x}^s_{N_s,I}, \mathbf{y}^s_{N_s,I}) \end{Bmatrix} \tag{6}$$

and a target domain of N_t labeled samples as

$$\mathcal{D}_t = \{(\mathbf{x}^t_{1,1}, ..., \mathbf{x}^t_{1,J}), ..., (\mathbf{x}^t_{N_t,1}, ..., \mathbf{x}^t_{N_t,J})\}, \tag{7}$$

where $\mathbf{x}^s_{n,i}$ represents the sample n from the i-th source receiver, $\mathbf{y}^s_{n,i} \in \mathbb{R}^K$ denotes a one-hot vector indicating the transmitter label (K is the number of transmitters), and I is the count of source receivers. $\mathbf{x}^t_{n,j}$ represents the sample n from the j-th target receiver and J is the count of target receivers. The objective of the RFFI scheme utilizing adversarial training and subdomain adaptation is to enable the model to achieve high performance on all target receivers by only using $\mathcal{D}_s$ and $\mathcal{D}_t$. The proposed scheme consists of two steps: global domain adaptation by adversarial training and relevant subdomain adaptation by LMMD.

1) Global Domain Adaptation by Adversarial Training: Adversarial-based methods are the most popular unsupervised domain adaptation approach to reduce domain discrepancy between different domains by using an adversarial objective [16]. For the cross-receiver scenario, this method is exploited to guide the neural network to learn receiver-independent features.

As shown in Fig. 2, the proposed model architecture is divided into three parts. The processed signal representation $\mathbf{x}$ first passes through a feature extractor G_f to obtain a D-dimensional feature vector $\mathbf{f} \in \mathbb{R}^D$, i.e., $\mathbf{f} = G_f(\mathbf{x}; \theta_f)$, where θ_f represents the parameters of G_f. We use CNN-based feature extractors because of the good performance shown in previous studies [8]–[10]. Then, the feature $\mathbf{f}$ passes through a transmitter classifier G_c to obtain the transmitter prediction vector $\hat{\mathbf{y}}$, and we denote the parameters of G_c with θ_c. The output of the transmitter classifier, $\hat{\mathbf{y}}$, is a list of probabilities after the softmax activation. We use cross-entropy loss function for the transmitter classifier loss $\mathcal{L}_c$:

$$\mathcal{L}_c = -\sum_{k=1}^{K} y_k log(\hat{y_k}), \tag{8}$$

where $\hat{y_k}$ is the k^{th} element in $\hat{\mathbf{y}}$ and y_k is the corresponding ground truth in $\mathbf{y}$. Note that the corresponding transmitter labels $\mathbf{y}$ are only known for samples from the source domain at training time. Finally, the same feature vector $\mathbf{f}$ passes through a domain classifier G_d with the parameters θ_d to obtain the domain prediction $\hat{d}$. If the sample corresponding to the feature vector $\mathbf{f}$ is from the source domain, then its domain label $d = 0$, otherwise $d = 1$. We use binary cross-entropy loss function for the domain classifier loss $\mathcal{L}_d$:

$$\mathcal{L}_d = -dlog(\hat{d}) - (1-d)log(1-\hat{d}) \tag{9}$$

Here we explain the role of the gradient reversal layer (GRL). As a RFFI solution in the cross-receiver scenario, we need to achieve two goals. The first one is to predict which transmitter the received signal originates from based on the feature vector $\mathbf{f}$ obtained by G_f, which is identical to the goal of G_c. Therefore, we can update the parameters θ_f and θ_c by minimizing the transmitter classifier loss $\mathcal{L}_c$ during training. The second goal is to make the feature vector $\mathbf{f}$ obtained from G_f receiver-independent, i.e., the domain classifier G_d cannot distinguish $\mathbf{f}$ from the source domain or the target domain. The paradox is that updating parameters θ_f requires maximizing $\mathcal{L}_d$, but updating parameter θ_d requires minimizing $\mathcal{L}_d$. To update all parameters during a standard stochastic gradient descent process, the GRL is inserted between the feature extractor G_f and the domain classifier G_d. During forward propagation, GRL acts as the identity transformation. However, during backpropagation, GRL takes the gradient from subsequent layers, multiplies it by $-\lambda$ and passes it to the previous layer [15]. Mathematically, parameters $(\theta_f, \theta_c, \theta_d)$ can be optimized by the stochastic gradient descent within our method:

$$E(\theta_f,\theta_c,\theta_d) = \sum_{(\mathbf{x}^s_{n,i},\mathbf{y}^s_{n,i})\in\mathcal{D}_s} \mathcal{L}_c(G_c(G_f(\mathbf{x}^s_{n,i};\theta_f);\theta_c),\mathbf{y}^s_{n,i})) + \sum_{\mathbf{x}_{n,i}\in\mathcal{D}_s\cap\mathcal{D}_t} \mathcal{L}_d(G_d(R_\lambda(G_f(\mathbf{x}_{n,i};\theta_f));\theta_d),d) \tag{10}$$

$$\theta_f \leftarrow \theta_f - \mu(\frac{\partial\mathcal{L}_c}{\partial\theta_f} - \lambda\frac{\partial\mathcal{L}_d}{\partial\theta_f}) \tag{11}$$

$$\theta_c \leftarrow \theta_c - \mu\frac{\partial\mathcal{L}_c}{\partial\theta_c} \tag{12}$$

$$\theta_d \leftarrow \theta_d - \mu\frac{\partial\mathcal{L}_d}{\partial\theta_d} \tag{13}$$

By adopting the adversarial approach, we can effectively train a feature extractor that extracts transmitter-specific information while being independent of the receiver. During inference, we directly connect the feature extractor to the transmitter classifier to identify transmitters.

2) Relevant Subdomain Adaptation by LMMD: The above scheme mainly focuses on the alignment of the global distribution, without considering the relationship between two subdomains within the same category. In order to make full use of the fine-grained information of the transmitter category, we further use local maximum mean discrepancy(LMMD) as a regularization term to perform subdomain adaptation on the feature extractor G_f after adversarial training.

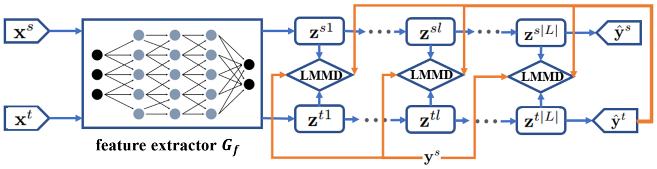


Fig. 3. The LMMD-based model architecture.

The subdomain adaptation network f_{LMMD} based on LMMD is shown in Fig. 3. We reduce the discrepancy between the subdomain distributions of the activations in layers L by minimizing LMMD. The l-th activations $\mathbf{z}^{sl}$ and $\mathbf{z}^{tl}$ ($l \in L = \{1, 2, ..., |L|\}$), the ground-truth transmitter label $\mathbf{y}^s$, and the predicted transmitter label $\hat{\mathbf{y}}^t$ are required to calculate LMMD. The LMMD regularization of layer l can be given by:

$$\begin{aligned} L_{LMMD,l} &= \frac{1}{K}\sum_{k=1}^{K}\left\|\sum \omega_i^{s,k}\psi(\mathbf{z}_i^{sl}) - \sum \omega_j^{t,k}\psi(\mathbf{z}_j^{tl})\right\|^2_{\mathcal{H}} \\ &= \frac{1}{K}\sum_{k=1}^{K}\left[\sum_{i=1}^{N_s}\sum_{j=1}^{N_s}\omega_i^{s,k}\omega_j^{s,k}k(\mathbf{z}_i^{sl},\mathbf{z}_j^{sl}) \right. \\ &\quad + \sum_{i=1}^{N_t}\sum_{j=1}^{N_t}\omega_i^{t,k}\omega_j^{t,k}k(\mathbf{z}_i^{tl},\mathbf{z}_j^{tl}) \\ &\quad \left. - 2\sum_{i=1}^{N_s}\sum_{j=1}^{N_t}\omega_i^{s,k}\omega_j^{t,k}k(\mathbf{z}_i^{sl},\mathbf{z}_j^{tl})\right], \end{aligned} \tag{14}$$

where $\mathbf{z}_i^{sl}$ is the l-th layer activation of $\mathbf{x}^s_{n,i} \in \mathcal{D}_s$, $\mathbf{z}_j^{tl}$ is the l-th layer activation of $\mathbf{x}^t_{n,j} \in \mathcal{D}_t$, $\psi(\cdot)$ represents a feature map to map the original samples to Reproducing Kernel Hilbert Space (RKHS) with a characteristic kernel k. $\omega_i^{s,k}$ and $\omega_j^{t,k}$ denote the weight of $\mathbf{z}_i^{sl}$ and $\mathbf{z}_j^{tl}$ belonging to transmitter k. For the sample $\mathbf{x}_i$, ω_i^k can be computed as

$$\omega_i^k = \frac{y_i^k}{\sum_{(\mathbf{x}_j,\mathbf{y}_j)\in\mathcal{D}} y_j^k}, \tag{15}$$

where y_i^k represents the k-th element of $\mathbf{y}_i \in \mathbb{R}^K$. If $\mathbf{x}_i$ belongs to the source domain, $\mathbf{y}_i$ is the corresponding label in one-hot vector form. If $\mathbf{x}_i$ belongs to the target domain, since its label is unknown, $\mathbf{y}_i$ is the predicted probability vector of the neural network, which can also be written as $\hat{\mathbf{y}}_i$. Then we can get the parameters Θ of f_{LMMD} by:

$$\Theta = \arg\min_{\Theta}\frac{1}{Ns}\sum_{i=1}^{Ns}J(f_{LMMD}(\mathbf{x}_i^s),\mathbf{y}_i^s) + \lambda\sum_{l\in L}L_{LMMD,l}, \tag{16}$$

where $J(\cdot,\cdot)$ is the cross-entropy loss function which measures the classification performance of the K transmitters, and $\lambda > 0$ is the trade-off parameter.

C. Fine-tuning Based on Few-sample Selection

When the data distributions of the source domain and target domain differ too much, the performance of the above schemes is still not good enough. Fine-tuning is an viable way to improve classification performance on the target domain. In our cross-receiver scenario, we attempt to achieve the following two goals. The first one is to select as few fine-tuning samples as possible, thereby reducing the cost of sample annotation and communication. The second goal is that the fine-tuned model not only performs well on the target domain, but also maintains the previous good performance on the source domain.

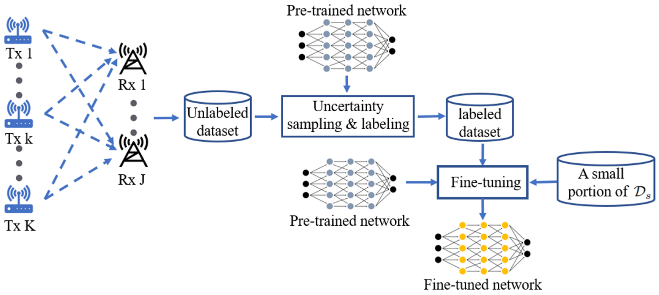


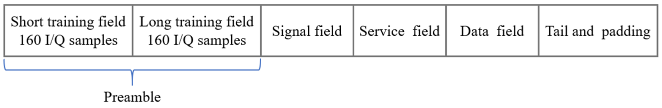
Fig. 4. Fine-tuning of a pre-trained network.

The flow chart of the fine-tuning scheme is shown in Fig. 4. First, unlabeled signals from K transmitters are collected on the receiver in the target domain to construct an unlabeled dataset. Then, the uncertainty sampling methods in the field of active learning [17] are used to sample and label a small number of samples from the unlabeled dataset to build a labeled dataset. The uncertainty sampling methods select and label the samples that are most difficult for the model to distinguish according to the classification results of the samples in the unlabeled dataset by the pre-trained model. Finally, we additionally use a small part of the source domain dataset for the fine-tuning process to avoid the performance of the fine-tuned model on the source domain from greatly decreasing.

IV. Performance Evaluation

A. Dataset Description and Signal Pre-processing

We utilize the WiSig dataset [10] to validate the proposed cross-receiver RFFI scheme. This dataset includes recordings of signals transmitted from a WiFi node to an access point (AP) using a USRP device. The WiFi nodes act as transmitters for fingerprinting, while the USRPs serve as receivers. During a capture from a single WiFi transmitter, data is sent from the transmitter to the AP while all USRP receivers simultaneously record raw IQ samples. We use the subset ManySig [10] because it has the most packets between each pair of transmitter and receiver. ManySig contains data collected by 6 transmitters and 12 receivers over 4 days.



Short training field 160 I/Q samples	Long training field 160 I/Q samples	Signal field	Service field	Data field	Tail and padding

Fig. 5. 802.11a/g packet structure [18].

The collected baseband signal needs to be processed in the following steps: synchronization and preamble extraction, channel equalization, signal normalization and signal representation. Based on previous studies [8], [10], we choose to extract the RF fingerprint from the equalized time-domain preamble of the WiFi signal. We employ power normalization to eliminate differences related to the collection environment such as receive power. Additionally, we preserve the carrier frequency offset (CFO) as we consider it to be a part of the RF fingerprint. As illustrated in Fig. 5, The preamble consists of a short training field and a long training field, with a total of 320 I/Q samples. In order to facilitate subsequent neural network processing, we only select the first 256 samples of the preamble, and the real and imaginary components are reorganized into a matrix resembling an image with dimensions of (256,2). A channel dimension is then added to meet the requirements of Pytorch's Conv2D function.

B. Model Architecture and Parameters

1) Global Domain Adaptation by Adversarial Training: The adversarial-based model is as follows. The feature extractor contains 4 Blocks (a block consists of a convolution layer, a batch normalization layer, a ReLU layer and a maximum pooling layer), and finally output a 128-dimensional feature vector. The transmitter and domain classifiers both comprise a dense layer with 128 neurons activated by the ReLU function, followed by a softmax layer for classification purposes.

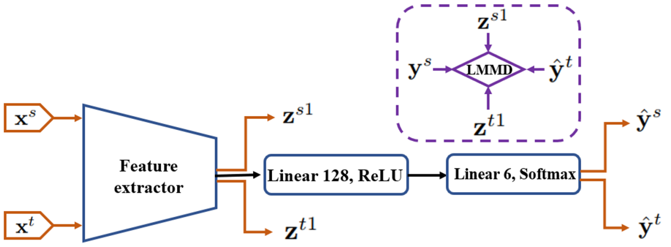


Fig. 6. Model architecture during subdomain adaptation for WiFi

2) Relevant Subdomain Adaptation by LMMD: As shown in Fig. 6, We use LMMD to perform relevant subdomain adaptation based on the feature extractor obtained after previous adversarial training.

C. Experimental results

1) Overall performance: The classification accuracy of the proposed scheme is shown in Fig. 7. The horizontal axis represents the ratio of the number of receivers in the source domain to the number of receivers in the target domain, which we define as R. The smaller the value of R, the greater the difference in data distribution. On the one hand, when the model trained in the source domain is directly deployed to the receiver in the target domain, there is a significant decline in classification accuracy. On the other hand, if the data in the target domain is annotated and incorporated into the training process, The retrained model can achieve exceptional performance. Here, we utilize the classification performance of these two schemes as the lower and upper limits for other schemes respectively. The results demonstrate that our proposed scheme effectively mitigates the loss of transmitter classification performance caused by cross-receiver scenarios. When $R > 2 : 10$, the accuracy achieved is nearly 10% higher than that of the scheme using only source domain

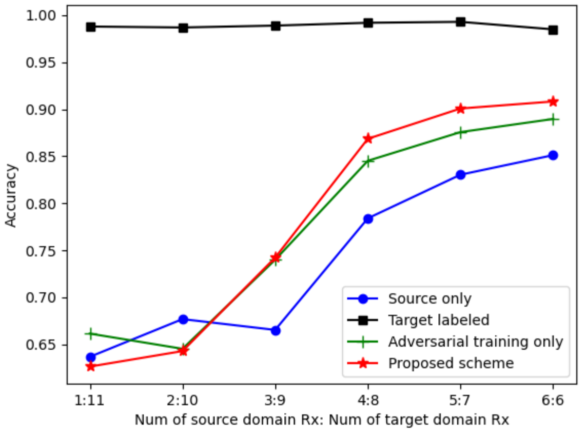


Fig. 7. Performance under different numbers of source domain receivers and target domain receivers.

data. Furthermore, subdomain adaptation based on LMMD after adversarial training is effective. When $R > 3:9$, the accuracy achieved is approximately 1% higher than that of the scheme using only adversarial training. Table I shows the performance of the proposed scheme on each target domain receiver when $R = 6:6$. The model after domain adaptation shows satisfactory accuracy on each receiver, without poor performance on individual receivers.

TABLE I
ACCURACY ON EACH TARGET DOMAIN RECEIVER WHEN R=6:6

	RX1	RX2	RX3	RX4	RX5	RX6
Proposed scheme	0.89	0.94	0.87	0.91	0.92	0.92

2) Benefits of fine-tuning based on few-sample selection: As illustrated in Fig. 7, when the discrepancy in data distribution is excessive ($R < 4:8$), employing adversarial training and subdomain adaptation fails to enhance the transmitter classification performance on the new receiver. In such instances, fine-tuning is necessitated. Fig. 8 depicts the influence of varying small sample selection strategies and the quantity of fine-tuning iterations on the efficacy of the fine-tuned model when $R < 4:8$. The selected labeled sample size remains consistent and constitutes 1/50 of the unlabeled dataset. The results show that fine-tuning based on a small number of samples can effectively improve classification performance on the new receivers. For $R = 1:11$ scenario, the classification accuracy can be improved by nearly 30% utilizing the random sampling strategy and 100 iterations of fine-tuning.

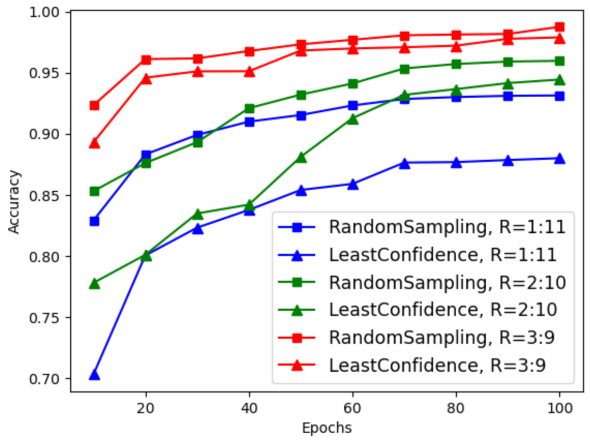


Fig. 8. Overall performance on different uncertainty sampling methods when $R < 4:8$.

V. CONCLUSION

In this paper, we propose a novel receiver-independent RFFI scheme which is suitable for roaming authentication scenarios in 5G IoT. We align the data distribution of the source and target domains using global domain adaptation based on adversarial training and subdomain adaptation based on LMMD to extract receiver-independent transmitter features, and further improve model performance through fine-tuning based on few-sample selection when domain adaptation is not effective. Extensive experiments on public datasets have verified the excellent performance of the proposed scheme. In addition, compared to existing studies, our scheme does not require extensive labeling of data on new receivers, resulting in lower deployment costs.

ACKNOWLEDGEMENT

This work was supported by National Key Research and Development Program of China (2022YFB2702302, 2020YFB1807504), and Natural Science Foundation of China (62171278, 62101327, 62202290).

REFERENCES

[1] S. Li, L. Da Xu, and S. Zhao, “5g internet of things: A survey,” *Journal of Industrial Information Integration*, vol. 10, pp. 1–9, 2018.

[2] G. A. Akpakwu, B. J. Silva, G. P. Hancke, and A. M. Abu-Mahfouz, “A survey on 5g networks for the internet of things: Communication technologies and challenges,” *IEEE access*, vol. 6, pp. 3619–3647, 2017.

[3] M. Cekic, S. Gopalakrishnan, and U. Madhow, “Wireless fingerprinting via deep learning: The impact of confounding factors,” in *2021 55th Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2021, pp. 677–684.

[4] A. C. Polak and D. L. Goeckel, “Wireless device identification based on rf oscillator imperfections,” *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 12, pp. 2492–2501, 2015.

[5] A. C. Polak, S. Dolatshahi, and D. L. Goeckel, “Identifying wireless users via transmitter imperfections,” *IEEE Journal on selected areas in communications*, vol. 29, no. 7, pp. 1469–1479, 2011.

[6] Q. Xu, R. Zheng, W. Saad, and Z. Han, “Device fingerprinting in wireless networks: Challenges and opportunities,” *IEEE Communications Surveys & Tutorials*, vol. 18, no. 1, pp. 94–104, 2015.

[7] W. Wang, Z. Sun, S. Piao, B. Zhu, and K. Ren, “Wireless physical-layer identification: Modeling and validation,” *IEEE transactions on information forensics and security*, vol. 11, no. 9, pp. 2091–2106, 2016.

[8] A. Al-Shawabka, F. Restuccia, S. D’Oro, T. Jian, B. C. Rendon, N. Soltani, J. Dy, S. Ioannidis, K. Chowdhury, and T. Melodia, “Exposing the fingerprint: Dissecting the impact of the wireless channel on radio fingerprinting,” in *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*. IEEE, 2020, pp. 646–655.

[9] J. Zhang, R. Woods, M. Sandell, M. Valkama, A. Marshall, and J. Cavallaro, “Radio frequency fingerprint identification for narrowband systems, modelling and classification,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 3974–3987, 2021.

[10] S. Hanna, S. Karunaratne, and D. Cabric, “Wisig: A large-scale wifi signal dataset for receiver and channel agnostic rf fingerprinting,” *IEEE Access*, vol. 10, pp. 22 808–22 818, 2022.

[11] K. Merchant and B. Nousain, “Toward receiver-agnostic rf fingerprint verification,” in *2019 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2019, pp. 1–6.

[12] M. Shi, Y. Huang, and G. Wang, “Carrier leakage estimation method for cross-receiver specific emitter identification,” *IEEE Access*, vol. 9, pp. 26 301–26 312, 2021.

[13] B. He and F. Wang, “Cooperative specific emitter identification via multiple distorted receivers,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 3791–3806, 2020.

[14] S. Andrews, R. M. Gerdes, and M. Li, “Crowdsourced measurements for device fingerprinting,” in *Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks*, 2019, pp. 72–82.

[15] Y. Ganin and V. Lempitsky, “Unsupervised domain adaptation by backpropagation,” in *International conference on machine learning*. PMLR, 2015, pp. 1180–1189.

[16] Y. Zhang, “A survey of unsupervised domain adaptation for visual recognition,” *arXiv preprint arXiv:2112.06745*, 2021.

[17] B. Settles, “Active learning literature survey,” 2009.

[18] E. Perahia and R. Stacey, *Next generation wireless LANs: 802.11 n and 802.11 ac*. Cambridge university press, 2013.