

Meta-RFF: Meta-Task Adaptive based Few-Shot Open-Set Incremental Learning for RF Fingerprint Recognition

Taotao Li, Zhenyu Wen*, Senior Member, IEEE, Chendong Jin, Jie Su, Junhao Li, Zhen Hong*, Member, IEEE, Xiaoqin Zhang, Senior Member, IEEE, Shibo He, Senior Member, IEEE

Abstract—In recent years, deep learning (DL) techniques have been extensively utilized for specific emitter identification through the extraction of RF fingerprints. A significant challenge that DL models face in real-world scenarios is the continuous emergence of new wireless devices, such as unknown drones that suddenly appear in the sky. In these situations, the radio monitoring system must be capable of detecting these unknown devices (open-set recognition) and incrementally updating the DL model’s knowledge using only a few captured samples. This requirement presents two main challenges: 1) Incremental updates from few-shot samples can lead to catastrophic forgetting and overfitting in DL models; 2) Constructing reliable open-set thresholds for new devices with few-shot samples is difficult.

To address these challenges, we propose a novel few-shot open-set incremental learning (FSOSIL) framework through meta-learning for RF fingerprint recognition, named *Meta-RFF*. The core idea of *Meta-RFF* is to simulate few-shot incremental learning and open-set recognition scenarios by constructing numerous pseudo-FSOSIL tasks and meta-training them. To enhance the open-set recognition capability, we design RF feature augmentation, open loss, and adaptive open-set thresholding modules. The algorithm’s effectiveness is validated on the large-scale aircraft recognition dataset (ADS-B), showing an improvement in closed-set accuracy and open-set AUROC of the new class by approximately 10-20% compared to other algorithms with 1-shot. We also demonstrate the algorithm’s effectiveness in a real-world test bed.

Index Terms—Deep Learning, RF Fingerprints, Open-Set Recognition, Few-Shot Incremental Learning.

This work was supported in part by the National Key Research and Development Program Project of China [grant no. 2024YFC3306902], the National Natural Science Foundation of China [grant no. U24A20242, 62472387], the Zhejiang Provincial Natural Science Foundation [grant no. LDT23F02024F02], the Zhejiang Provincial Science Fund for Distinguished Young Scholars under Grant LR24F020004, the Zhejiang Provincial Natural Science Foundation of Major Program (Youth Original Project) under Grant LDQ24F020001 and China Postdoctoral Science Foundation under Grant 2025M771517.

T. Li, Z. Wen, C. Jin, J. Su and Z. Hong are with the Institute of Cyberspace Security, and the College of Information Engineering, Zhejiang University of Technology, Hangzhou, 310023, Zhejiang, China (e-mail: 2111903074@zjut.edu.cn; zhenyuwen@zjut.edu.cn; 211124030118@zjut.edu.cn, jiesu@zjut.edu.cn; zhong1983@zjut.edu.cn).

J. Li is with Cyberspace Security Institute, Guangzhou University, Guangzhou, 510006, Guangdong, China (e-mail: lijh@e.gzhu.edu.cn).

X. Zhang is with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, 310023, Zhejiang, China (e-mail: zhangxiaoqinnan@gmail.com).

S. He is with the State Key Laboratory of Industrial Control Technology, College of Control Science and Engineering, Zhejiang University, Hangzhou 310027, China (e-mail: s18he@iipc.zju.edu.cn).

The initial version of this paper was accepted by The 18-th International Conference on Wireless Artificial Intelligence Computing Systems and Applications (WASA 2024) and received the Best Paper Award.

Corresponding author: Zhenyu Wen, Zhen Hong.

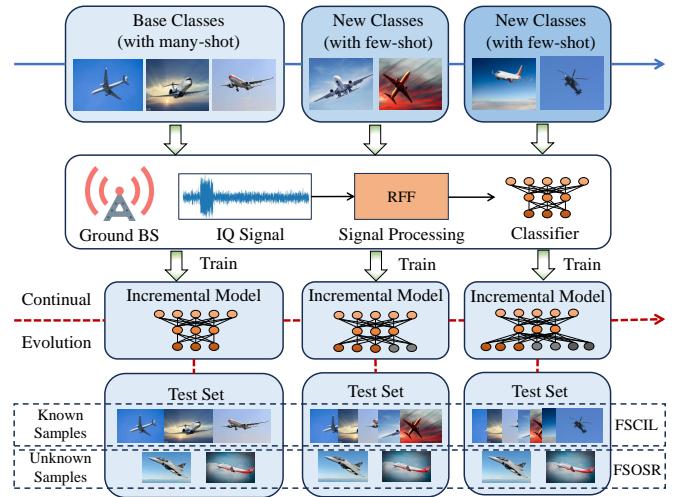


Fig. 1. Aircraft classifier system deployed in ground BS not only needs to accurately recognize known aircraft (base classes) but also needs to recognize unknown aircraft (new classes). This classifier is equipped with continuous few-shot class incremental learning (FSCIL) capability to achieve co-recognition of new and old classes. At the same time, the few-shot OSR capability (FSOSR) also needs to be improved in the continuous increment.

I. INTRODUCTION

IN recent years, the openness of wireless networks has significantly enhanced the convenience of communication between wireless devices [1]. However, this openness has made it challenging to ensure data confidentiality, thereby exposing wireless systems to potential security threats [2]. For instance, in automatic identification systems (AIS), the maritime mobile service identity (MMSI) can be easily spoofed, posing significant security risks to maritime traffic [3], [4]. Similarly, in automatic dependent surveillance-broadcast (ADS-B) systems, aircraft identification information is susceptible to imitation and tampering, leading to potential security hazards [5]. Fortunately, the physical characteristics of terminal devices offer a unique RF fingerprint for specific emitter identification (SEI) [6]–[8]. RF fingerprinting leverages the stable, non-tampering physical properties of hardware components, such as I-Q imbalance [6], loop filter variations [9], and clock jitter [10], to ensure reliable identification. SEI technology is now extensively utilized in both civilian and military domains, including modulation recognition [11]–[13], IoT device authentication [14], [15], spectrum monitoring [16], and aviation management [17], [18].

For achieving automated SEI, deep learning (DL) tech-

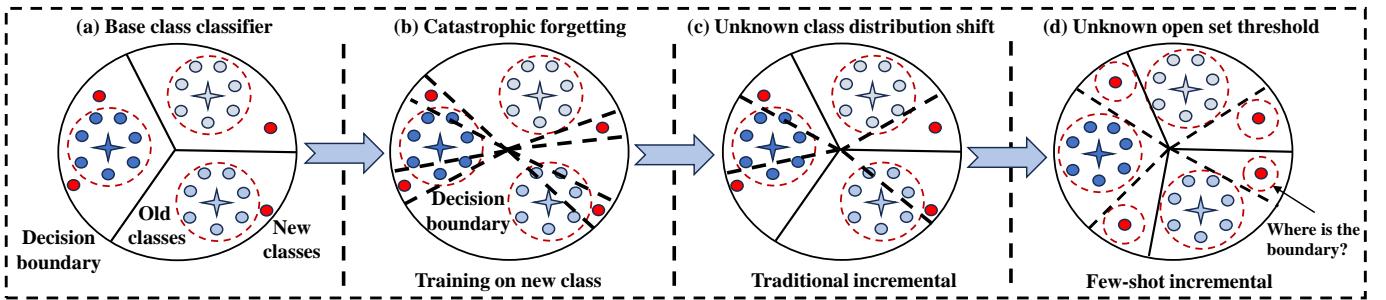


Fig. 2. Existing DL technology bottlenecks: (a)The decision boundary (solid line) of the base class classifier can accurately classify old classes, but not classify the new classes (red); (b)By training the classifier directly on the new class, the model is biased towards fitting the new class (dotted line), forgetting the knowledge of the old class; (c)Uncertain new class distributions are shifted to the vicinity of old classes, resulting in limited classification performance; (d)Difficulty in determining the OSR threshold (red dotted line) for the new class due to its few-shot limitation.

niques have recently been widely used for RF fingerprinting. Utilizing the powerful feature extraction capability of the DL model to achieve intelligent communication device management has important application prospects [19], [20]. However, previous studies have explored a variety of DL-based RF fingerprinting techniques [14], [15], [18], [21]–[24] to enhance the recognition efficiency of wireless devices and do not consider the following deployment challenges.

Autonomous radio monitoring system. Fig. 1 shows that the aircraft identification system deployed ground base station (BS) will utilize the DL model for aircraft classification by extracting RF fingerprints (RFF) of IQ signal. The recognition system must be equipped to meet various practical environmental demands [25], including (1) the capacity for open-set recognition (OSR) to identify sudden appearances of unfamiliar aerial objects [18], (2) the ability to update and recognize new categories using a limited number of samples from these newly emerged classes [26], and (3) the flexibility to continually evolve, accommodating an increasing range of categories and adapting to unforeseen environments in the future [27], [28].

However, building such a generalized evolutionary model that combines few-shot class increment learning (FSCIL), few-shot OSR (FSOSR), and multi-stage continuous increment learning needs to overcome the following challenges.

- **Catastrophic Forgetting and Overfitting.** As shown in Fig. 2a, since the base class classifier cannot correctly classify the new class, we need to add the new knowledge to the base model, and the updated model performs well for both newly added classes and original classes. As the model updates its parameters through the optimizer to fit the new task, these updated parameters are not necessarily applicable to the old task. As shown in Fig. 2b, it inevitably causes the model to forget knowledge [29] and over-fitting effects on the new task with few-shot samples [30], [31].
- **Unknown Class Distribution Shift.** As shown in Fig. 2c, the distribution of unknown classes is difficult to determine, and they may appear in the vicinity of known classes, which inevitably leads to a degradation of the model's classification performance [29], [32].
- **Unknown Open Set Threshold.** As shown in Fig. 2d, existing FSCIL [27], [33]algorithms seem to solve the above challenges, but it is still difficult to determine the

open-set boundary for few-shot samples. The lack of samples makes us unable to see the true class distribution, which hinders the accurate estimation of the open set boundaries.

- **Continuous Incremental Learning Challenges.** The multi-stage continuous incremental process aggravates the above challenges and leads to a drastic decrease in efficiency. In this scenario, the number of new classes continues to increase the density of feature space, which inevitably leads to overlapping decision boundaries and raises the risk of overlapping feature distributions of unknown classes. It will result in a decrease in the model's open-set and OSR capabilities.

To address the above practical challenges, existing algorithms like incremental learning [29], iCARL [32], OSR [34], [35], and FSCIL [27], [33] address only a single level of the problem. Our task mainly involves FSCIL as well as FSOSR, and it can be seen that improving few-shot learning capability is the key to solving the problem. To address the above practical challenges, we have rethought human learning patterns. As we know, a normal 6-year-old child cannot only quickly recognize the unknown but also learns from 1-shot images of new things without forgetting. Human capabilities derive from long-term adaptation to complex environments, i.e., learning how to learn in meta-learning [36]. Motivated by this, we expect the DL model to become a generalized evolution model with both FSCIL, FSOSR, and lifelong learning capabilities like humans through continuous environment simulation. Benefiting from the leapfrog development that meta-learning has brought to the field of few-shot learning in recent years [31], [36], [37], we mainly employ meta-learning to simulate the environment for the multiple tasks defined in this paper and prompt the model to gradually adapt.

In this paper, we propose a meta-learning-based few-shot open-set incremental learning (FSOSIL) for RF fingerprint recognition (*Meta-RFF*) framework. To address the problems of the RFF recognition system, we first propose a signal augmentation scheme for few-shot samples. Second, based on the meta-learning idea, we define the FSOSIL meta-tasks and sampled a large number of pseudo-tasks from the training set to realize environment simulation. Further, we utilize the meta-learning technique for multi-task training to make the neural network adaptive in such an environment. Finally, to solve the FSOSR problem, we incorporate open loss in the meta-

task and propose an open-set threshold-free mechanism. This mechanism is adaptable to the continuous incremental process and automatically generates open-set thresholds for few-shot classes. Our contributions can be summarized as follows:

- To the best of our knowledge, we are the first to formulate an FSOSIL framework for RFF recognition and realize a generalized RFF continuous learning real system.
- We define the FSOSIL meta-tasks and design a multi-task training mechanism based on a prototype network.
- We propose a soft orthogonalization loss and an open loss to calibrate prototype points and OSR automatically.
- To solve the problem of auto-tuning the FSOSR algorithm in continuous increments, we propose an open-set threshold-free mechanism based on reciprocal point synthesis.

Organization. In Section II, we review the existing work on RFF recognition, FSCIL, and FSOSR. The basic definition of RFF recognition and FSOSIL tasks are given in Section III. We propose the details of *Meta-RFF* in Section IV. The setups and discussion of experiments are given in Section V. In Section VII, we conclude our paper and describe future work.

II. RELATED WORK

In the following, we briefly discuss the most relevant approaches related to our work.

A. RF-Fingerprinting (RFF)

RF fingerprinting has become an effective solution for authenticating wireless devices. It has been observed that there are certain obvious imperfections in the analog circuits of RF transmitters [14]. RFF is a permanent imperfection that is extracted on the receiving side. Traditional RF fingerprint identification methods use an intuitive understanding of device fingerprints such as I-Q imbalance [6], power amplifier nonlinearity [7], [8], loop filter variations [9], clock jitter [10]. Earlier studies were performed in controlled environments [7], [8], which led to poor robustness and practicality of earlier fingerprint recognition methods. Recently, modern methods for RF fingerprinting employ deep learning algorithms and residual connectivity [14], [15], [18], [21], [22], [38] such as ResNet-50, GoogleNet and AlexNet to mine RFF features. Residual connectivity in deep learning networks can dramatically improve RFF performance. However, the catastrophic forgetting and overfitting problems of deep learning models when performing incremental updates with few-shot samples have led to the gap still existing with the real environment.

B. Few-Shot Class Incremental Learning

Few-Shot Class Incremental Learning (FSCIL) represents a special case of conventional incremental learning, where models are required to adapt to new classes using only a limited number of examples while preserving their ability to recognize previously learned classes. There has been significant research interest in developing FSCIL methods and effectively learning new classes without catastrophic forgetting

or overfitting [39]–[43]. Inspired by the Regularized Lottery Ticket Hypothesis, [44] proposes that a subnet can perform on par with or better than the whole network. The soft subnetwork jointly and partially updates the model weights and adaptive soft masks to minimize catastrophic forgetting and avoid overfitting novel samples. [45] proposes fixing a learnable classifier as a geometric structure for few-shot incremental learning. By using predetermined prototype vectors instead of unconstrained learned vectors, the approach alleviates perplexity and collapse that arise from training with limited samples for newly introduced classes. Later, the Continually Evolved Classifier (CEC) [33] is proposed, employing a graph model to propagate context information between classifiers learned in individual incremental sessions. Further, Zhou [27] et al. proposed a few-shot incremental learning scheme by sampling multi-stage tasks, i.e., LIMIT, which can improve the classification performance under continuous increments relative to CEC. However, related methods have limited effectiveness in solving the OSR problem.

C. Few-Shot Open-Set Recognition

Few-Shot Open-Set Recognition (FSOSR) has emerged as a promising direction in recent deep learning research. Its goal is to efficiently recognize and classify instances from known few-shot classes, given very few labeled samples, while also detecting and rejecting instances from unknown or undisclosed classes. Recently, [46] presents a meta-learning-based solution for FSOSR, introducing an open-set loss in the meta-training process to calibrate a few-shot prototype-based classifier. [47] improves the limitation of negative sampling by imposing a transformation consistency regularization on few-shot samples. Later, [48] proposes the Task-Adaptive Negative class Envision (TANE) framework to avoid the manual threshold selection procedure by dynamically estimating the rejection boundaries concerning few-shot classes. However, the related methods cannot adapt to the problem of OSR under continuous increment. The method of our paper achieves for the first time the harmonization of FSCIL and FSOSR.

III. PROBLEM DEFINITION

In wireless communication systems, the messages of wireless devices will be modulated into a radio signal for wireless transmission. In-ground BS, a received time-series modulation signal $r(t)$ can be illustrated as

$$r(t) = s(t) * h(t) \exp[j2\pi\Delta f t + \Delta\varphi] + wgn(t), \quad (1)$$

where $s(t)$ denotes the modulated signal, $*$ denotes the convolution operation, $h(t)$ denotes the wireless channel response, Δf denotes the frequency offset, $\Delta\varphi$ denotes the phase offset, $wgn(t)$ denotes the white Gaussian noise in wireless environment. To facilitate signal information extraction and signal recovery, in-phase signals $\text{Re}[r(t)]$ and quadrature-phase signals $\text{Im}[r(t)]$ are used to jointly characterize the relevant modulation information, i.e. I-Q data [49]. For calculation purposes, the system will sample the I-Q data and convert it

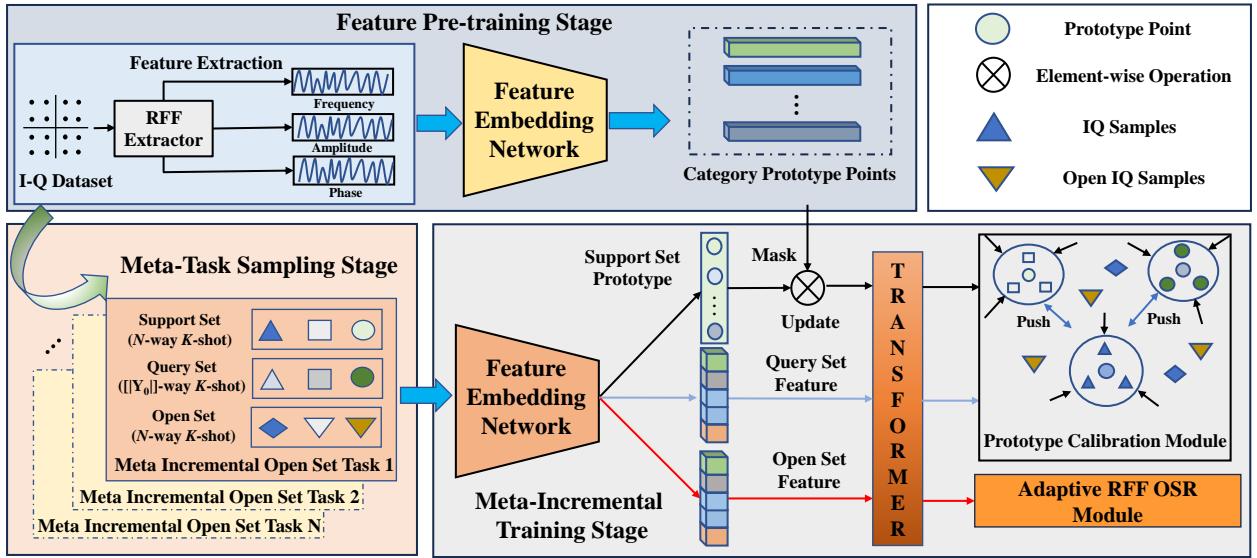


Fig. 3. The workflow of *Meta-RFF* framework. Our framework consists of 3 phases: (1) the feature pre-training stage focuses on extracting features and constructing class prototypes; (2) the meta-task sampling stage constructs pseudo-FSOSIL tasks; and (3) the meta-incremental training stage performs few-shot incremental training and open-set training.

into a discrete complex signal $x_{IQ}(n) = \{x_I(n), x_Q(n)\}$, which can be calculated by

$$\{x_I(n), x_Q(n)\} = \text{sample}\{\text{Re}[r(t)], \text{Im}[r(t)]\}. \quad (2)$$

RFF Recognition. In a traditional DL-based RFF recognition setting, the RF receiver (*Rx*) will get a series of signals $x_{IQ}(n)$ from a set of RF transmitters ($Tx = Tx_1, Tx_2, \dots, Tx_k$). We define the base session (*0-th* session) dataset consisting of k transmitters as $\mathcal{D}^0 = \{x_i, y_i\}_{i=0}^{n_0}$ with sufficient instances, where $x_i = x_{IQ}$ represents the training sample from class $y_i \in \mathbb{Y}_0$, and \mathbb{Y}_0 is the corresponding label space. In RFF recognition, an algorithm will fit a DL model $f_\theta^0(\cdot)$ to minimize the expected risk over instance distribution \mathcal{D}^0 :

$$\arg \min_{\theta} \mathbb{E}_{\{x_i, y_i\} \sim \mathcal{D}^0} [\ell(f_\theta^0(x_i), y_i)] \quad (3)$$

where $\ell(\cdot, \cdot)$ measures the discrepancy between the prediction and the ground-truth label. The model $f_\theta^0(x)$ comprises an embedding function $\psi(\cdot) : x \rightarrow \mathbb{R}^d$ and a linear classifier $W_0 = \{w_i\}_{i=0}^{|\mathbb{Y}_0|}$, i.e., $f_\theta^0(x) = W_0^T \psi(x)$ and \mathbb{R}^d denotes the d -dimensional feature space. Each category y_i corresponds to a weight vector w_i . With parameter optimization by Eq. 3, the DL model $f_\theta^0(\cdot)$ can predict the category of the test sample x in \mathcal{D}^0 .

Few-Shot Open-Set Incremental Learning for RFF Recognition. In an autonomous radio monitoring system, the prevalence of emerging unknown source signals necessitates that DL models be updated quickly for adaptation. Our FSOSIL task setup is first to have the model perform OSR for unknown source signals, i.e., $\mathcal{D}^o = \{(x_i, y_i) | y_i \in \mathbb{Y}_o\}_{i=1}^{n^o}$, where the \mathbb{Y}_o is the label space. If the classes are known, we will perform closed-set classification. Once the result is unknown, manual research and labeling of sample labels are required. Here, to reduce the cost of manual labeling, only a small number of samples are labeled. In this case, the new training sets (i.e., $\{\mathcal{D}^1, \dots, \mathcal{D}^b\}$) often arrive incrementally with limited instances, i.e., $\mathcal{D}^b = \{(x_i, y_i) | y_i \in \{\mathbb{Y}_1, \dots, \mathbb{Y}_b\}\}_{i=1}^{n^b}$. The \mathbb{Y}_b

is the label space of task b , and $\mathbb{Y}_b \cap \mathbb{Y}_o = \emptyset$. Then the n^b and n^o denote the number of samples in \mathcal{D}^b and \mathcal{D}^o , respectively. When facing a new dataset \mathcal{D}^b , a model should learn new classes while maintaining performance on old classes and rejecting unknown classes.

For the FSCIL phase in FSOSIL, it can be formalized as the minimization of the expected risk overall on the base session and new session data:

$$\min_{\theta} \mathbb{E}_{\{x_i, y_i\} \sim \{\mathcal{D}^0, \dots, \mathcal{D}^b\}} [\ell(f_\theta^b(x_i; \mathcal{D}^b, \psi_{b-1}, W_{b-1}), y_i)]. \quad (4)$$

By Eq. 4, the model $f_\theta^{b-1}(\bullet)$ should construct the new model based on the new dataset \mathcal{D}^b and the current model W_{b-1}, ψ_{b-1} . Then in real-world testing, we expect that the newly constructed model $f_\theta^b(\bullet)$ to minimize the loss over all base and new test datasets.

For the FSOSR phase in FSOSIL, it can be formalized as the minimization of the expected risk overall on the known and unknown tested data:

$$\min_{\theta} \mathbb{E}_{\{x_i, y_i\} \sim \{\mathcal{D}^0, \dots, \mathcal{D}^b, \mathcal{D}^o\}} [\ell(f_\theta^b(x_i; \psi_b, W_b), \tilde{y}_i)], \quad (5)$$

where $\tilde{y} \in \{\mathbb{Y}_1, \dots, \mathbb{Y}_b, \mathbb{Y}_o\}_{i=1}^{n^b}$ denotes the open-set prediction label.

IV. METHODOLOGY

Design Idea. Motivated by the success of meta-learning in the field of few-shot recognition, our idea is to leverage a large number of meta-incremental open-set tasks to simulate realistic FSOSIL scenarios. We will simulate a few-shot open-set incremental scenario, fine-tune the model by utilizing the few-shot sample as a support set, and test it on a many-shot query set. Through gradient optimization, the neural network will gradually adapt to the few-shot open-set incremental scenario.

Fig. 3 shows the pipeline of *Meta-RFF* framework for solving the FSOSIL task. Specifically, the proposed *Meta-RFF*

framework can be separated into three stages: **1) Feature Pre-training stage, 2) Meta-Task sampling stage, and 3) Meta-Incremental training stage.** In the feature pre-training stage, the feature embedding networks and classifier weights (i.e., prototype points) are obtained using base session data. Subsequently, we will construct few-shot incremental recognition scenarios and perform task sampling. In the training phase, a prototype network is utilized for few-shot learning, and a Transformer is used to calibrate the distribution of new and old class prototype points in an orthogonality prototype space. Finally, a dynamic threshold-selecting mechanism is employed that calculates the reciprocal point of the rectified prototype points, generating an optimal threshold for OSR. The Table I shows the important symbol notation in the next paragraph.

A. Feature Pre-training Stage

Prior incremental learning approaches [32], [33], [50], [51] have demonstrated that fine-tuning the network with new class data from a subsequent session can result in catastrophic forgetting [29], wherein previously acquired knowledge is lost and overfitting transpires with the introduction of new data. Recent advances in incremental learning [50], [51] suggest that decoupling the feature embedding network from the classifier can largely reduce the effect of catastrophic forgetting. Thus, we follow the previous work and additionally employ the classic prototype network [31] (widely used under the few-shot learning scenario) to alleviate the catastrophic forgetting and overfitting problems under the few-shot sample condition.

Feature Extraction. To improve the recognition efficiency of I-Q data with few-shot, we need extract more signal modal information. In wireless communication, IQ signals are usually defined by using amplitude, frequency, and phase [12]. Therefore, we follow the setting of Zheng et al. [11] and mainly extract the instantaneous amplitude $A(n)$, instantaneous phase $\varphi(n)$, and instantaneous frequency $F(n)$ information of the signal. The relevant calculations are shown below

$$\begin{aligned} A(n) &= \sqrt{x_I(n)^2 + x_Q(n)^2}, \\ \varphi(n) &\propto \arctan(x_Q(n)/x_I(n)), \\ F(n) &= \varphi(n) - \varphi(n-1), n = 1, 2, \dots, N-1, \\ \mathcal{F}(n) &= F(n) - \frac{1}{N} \sum_{n=1}^N F(n), \end{aligned} \quad (6)$$

where $\mathcal{F}(n)$ denotes the centered instantaneous frequency, and the details of $\varphi(n)$ refer to [11]. For easier representation, we redefine the sample symbols as

$$x(n) = \text{concat}\{x_{IQ}(n), A(n), \varphi(n), \mathcal{F}(n)\}. \quad (7)$$

Model Pre-training. Specifically, we first train the feature embedding network $\psi(\bullet)$ and classifier W_0 in the base session. The classifier weights w_i are represented by the average embedding of each new class c_i (i.e., the class prototype or the most representative feature of the class). The class prototype w_j in \mathcal{D}^0 can be calculated by:

$$w_j = \left\| \frac{1}{|\mathcal{D}^0|} \sum_{i=1}^{|\mathcal{D}^0|} I(y_i = j) \psi(x_i) \right\|_2, \quad (8)$$

where $\|\bullet\|_2$ denotes the l_2 normalization and $I(\cdot)$ denotes the indicator function. With the class prototype points w_i , we can

TABLE I
IMPORTANT SYMBOL NOTATIONS

Notations	Description
$\psi(\bullet)$	Feature embedding network
w_i	Classifier weight
\mathcal{D}^0	Base dataset
w_j	Class prototype in \mathcal{D}^0
S_m	Support set
Q_m^s	Query set
Q_m^o	Many-shot query set
Q_m^e	Open set of m -th task
$\mathcal{T}(\cdot)$	Adaptation function to carry out the calibration process
M	Mask matrix
G_θ	Generative network
\mathcal{R}	Reciprocal point
P_i	Prototype point

calculate the class probability p_j for each base session sample as

$$p(y = j|x; \psi_0) = \frac{\exp(\text{sim}(\psi(x), w_j))}{\sum_{j \in Y_0} \exp(\text{sim}(\psi(x), w_j))}, \quad (9)$$

where $\text{sim}(\bullet)$ denotes the cosine similarity function. The Eq. 9 suggests that the similarity of the sample to the class prototype point determines the sample prediction category. Finally, we use the cross-entropy loss function ℓ_{ce} [33] to perform with training on the base session samples:

$$\ell_{ce} = -\frac{1}{|\mathcal{D}^0|} \sum_i \sum_{j=1}^{Y_0} y_{ij} \log(p_{ij}), \quad (10)$$

where y_{ij} denotes that the label of the i -th sample, $\log(p_{ij})$ denotes the logarithmic probability of class j .

During the base session pre-training, a network with robust feature extraction capabilities is obtained by leveraging a substantial amount of available sample data. Then, when a new task arrives, the parameters of the feature embedding network are frozen to prevent knowledge forgetting, and the prototype points of the new class are computed to update the classifier.

B. Multi-phase Meta Task Sampling

The generalization ability of the learned features largely affects the performance of the incremental session [52]. Under the FSOSIL task, this effect is magnified, as the model must generalize to new classes with a limited number while maintaining the ability to reject unknown classes. However, the model does not have direct access to new class data and unknown class data in the incoming incremental sessions, making it challenging to evaluate the generalization ability of learned features for future tasks. Thus, motivated by the idea of meta-learning, we propose to sample a large number of “fake” FSOSIL tasks from the base session data to simulate the procedure of real FSOSIL tasks. The sampled “fake” FSOSIL tasks aim to provide a way for the neural network to learn generalizable embeddings. In the fake task, the model is first fine-tuned using few-shot samples, at which point the parameters of the model have changed. To motivate the model to accurately recognize the old, new, and unknown classes, we create these three data and then have the model test and update them based on the loss. Through multiple iterations of the

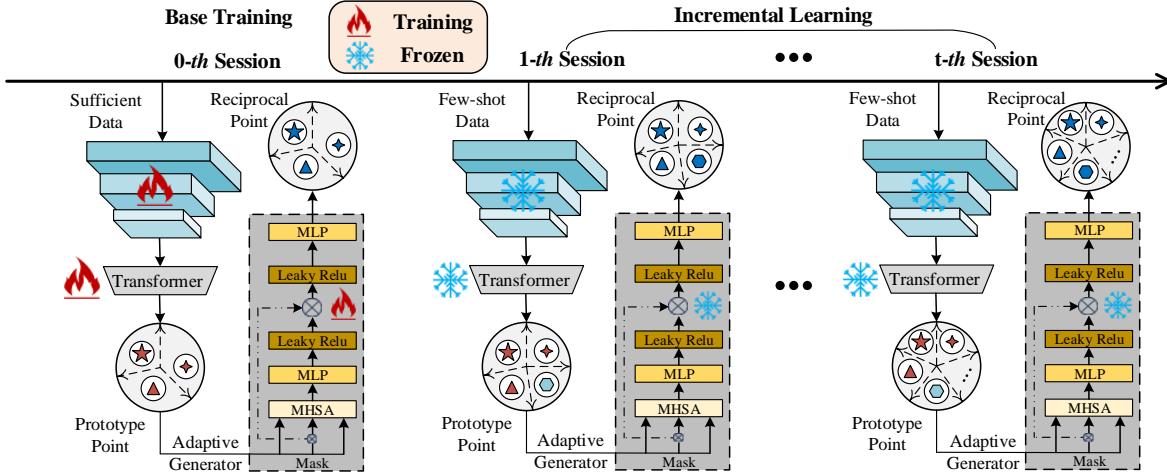


Fig. 4. The workflow of adaptive RFF open set recognition mechanism framework. In each incremental task, our generation module will generate corresponding reciprocal points for the prototype points to be used as decision thresholds for OSR.

meta-task, the model can be gradually adapted to the FSOSIL scenario. Due to the similarity between the test data and the training data distribution, the model obtained by training on the fake task is effectively generalizable to the test data.

To make the “fake” FSOSIL tasks share the same data format as the “real” FSOSIL task, we first divide the base dataset \mathcal{D}^0 into three non-overlapping sets: $T = \{S, Q^s, Q^*, Q^o | C^s, C^*, C^o\}$, where $S = \{x_i, y_i\}_{i=1}^{N_K}$ denotes support set with N -way K -shot and the label is C^s , Q^s denotes the query set, $Q^* = \{x_i, y_i\}_{i=1}^{|Y_0|K}$ denotes the many-shot query set with $|Y_0|$ -way K -shot and the label is C^* , Q^o denotes the open set with N -way K -shot. Also, the label spaces of Q^o and Q^* are non-overlapping, i.e., $C^* \cap C^o = \emptyset$, $C^s \subset C^*$. After the “fake” FSOSIL task sampling, the training objects can be formulated as the minimization of the empirical risk on the m -th task T_m :

$$\min_{\theta} \mathbb{E}_{\{x_i, y_i\} \sim \{Q_m^s, Q_m^o, Q_m^*\}} [\ell((x_i; S_m, \psi_{m-1}, W_{m-1}), y_i)], \quad (11)$$

where S_m, Q_m^s, Q_m^o, Q_m^* denote the support set, query set, many-shot query set, and open set of m -th task, respectively.

C. Meta-Training for FSOSIL

To facilitate the training process of the established “fake” FSOSIL tasks, we propose an optimization strategy that simultaneously trains the network to extract robust features for both incremental recognition and OSR tasks.

Optimization for few-shot incremental recognition. During the incremental session, the classification model will continuously receive new session tasks, which require the network to generalize across both new and old class samples. To emulate the incremental session, we utilize the support set S , and the all class query set Q^* from the base session to represent new class incremental samples, and all class samples, respectively. During the optimization process, the network first generates new prototypes c_i for the few-shot support set in the incoming task using Eq. 9, and then directly employs the pre-trained

old class weights w_i to recognize the old classes. Thus, the classifier weights W_{T_m} for task m will be updated by:

$$W_{T_m} = \begin{cases} c_i, \forall i \in C^s \\ w_i, \forall i \notin C^s, i \in C^* \end{cases}. \quad (12)$$

By jointly constructing new prototype points and optimizing through the similarity-based cross-entropy loss function, we can obtain the optimization function for the incremental session:

$$L_{T_m} = \ell_{ce}(sim(f(Q_m^*), W)). \quad (13)$$

Finally, based on idea of meta-learning, we need to optimize for a large number of “fake” few-shot incremental tasks at the same time. So that the neural network can learn how to adapt to this context. This meta-learning optimization loss can be shown by the following

$$\min_{\theta} \frac{1}{T_m} \sum_{m=1}^{T_m} L_{T_m}(Q_m | S_m, f_{\theta}, W_m). \quad (14)$$

Meta-calibration module. The incremental optimization is based on the many-shot old classes, which is tailored to depict old class features. To calibrate the semantic gap between new and old class prototypes (i.e., updating the relative spatial distribution), we design a transformer-based calibration module capable of extracting inductive bias during meta-training and generalizing to subsequent incremental sessions.

A good calibration module should reflect the contextual relationship between the old and new classes. For example, if the query instance is a ‘tiger’ then the classifier and prototype should be tuned to highlight distinguishing features such as beard and stride length. The calibration process can therefore be seen as performing ‘co-adaptation’ - we need to transform query embedding and classifiers to highlight the discriminative features of specific instances. The transformer [53] with its self-attentive mechanism has been proven to effectively extract discriminative features and incorporate contextual information. Hence, we define the adaptation function $\mathcal{T}(\cdot)$ by using the transformer to carry out the calibration process.

The transformer utilizes a triplet of information (query Q , key K , and value V) and learns through the attention mechanism. First, the query sample $(\psi(x_m) \in \mathbb{R}^{d \times |\mathcal{M}|}, m \in \mathcal{M})$ is

projected linearly using weights W_q , W_k , and W_v respectively. Next, the attention coefficients α_{qk} are computed using Q , K , and softmax functions. Finally, the attention coefficients are applied to weight \mathcal{V} , obtaining the final attention result in $\psi(x_m)'$. This process can be expressed as:

$$\begin{aligned} Q &= W_q^T \psi(x_m), K = W_k^T \psi(x_m), V = W_v^T \psi(x_m), \\ \alpha_{qk} &= \text{softmax}\left(\frac{K^T Q}{\sqrt{d}}\right), \\ \psi(x_m)' &= \psi(x_m) + \sum_m \alpha_{qk} V_m. \end{aligned} \quad (15)$$

In our framework, we drop query samples into $\mathcal{T}(\cdot)$ for adaptive optimization along with prototype points, i.e.,

$$Q = K = V = [w, c]. \quad (16)$$

The calibrated prototype point are then defined as $(\hat{w}, \hat{c}) = \mathcal{T}(w, c)$.

Optimization for few-shot open-set recognition. While the aforementioned approach enables the recognition of seen classes, Eq. 13 fails to effectively constrain the open space, resulting in limited open-set recognition capabilities. To address this issue, we propose simulating the open-set scenario in the meta-task to optimize open-set recognition ability.

In previous approaches to OSR [12], [34], [35], the main idea is to increase the inter-class gap and decrease the intra-class gap, which will effectively reduce the overlap between known and unknown data. However, previous methods focus on a fixed closed-set space, making incremental expansion difficult. For this reason, we hope each new class prototype adapts its open space distribution concerning the old class prototypes. To fairly control the distance between class prototypes, we introduce an orthogonalization loss in the inner product space. In orthogonal space, all prototypes are orthogonalized vectors the cosine similarity is 0, i.e., $\|w_i\|^T \|w_j\| = 0, i \neq j, \forall i, j \in C$. Then, to make the prototype points of the new class orthogonal to the old class, we impose the orthogonalization loss ℓ_{or} in the meta task, i.e.,

$$\begin{aligned} \ell_{or}(\hat{W}) &= M \odot \|\hat{W}\|^T \|\hat{W}\|, \\ M_{ij} &= \begin{cases} 0, & i = j \\ 1, & i \neq j \end{cases}, \end{aligned} \quad (17)$$

where $M \in \mathbb{R}^{|Y_0| \times |Y_0|}$ denotes the mask matrix, \odot denotes the element-wise multiplication, $\|\bullet\|$ denotes the ℓ_2 -norm. In each meta-task, the updated prototype point \hat{W} will be orthogonalized by the meta-calibrator \mathcal{T} , which guarantees a similarity of 0 between different classes of prototype points.

After controlling inter-class distances, we need to further control intra-class distances. We use the idea of clustering samples with their class prototype points to progressively reduce the differences. This process can be implemented by the following equation

$$\begin{aligned} \ell_d(f_\theta(x(n)), \hat{W}) &= \ell_e(f_\theta(x(n)), \hat{W}) - \ell_c(f_\theta(x(n)), \hat{W}), \\ \ell_e(f_\theta(x(n)), \hat{W}) &= \frac{1}{d} \|f_\theta(x(n)) - \hat{W}\|_2^2, \\ \ell_c(f_\theta(x(n)), \hat{W}) &= \|f_\theta(x(n))\|^T \|\hat{W}\|, \end{aligned} \quad (18)$$

where ℓ_e denotes the euclidean distance, ℓ_c denotes the cosine similarity. We combine the Eq. 13, 17, 18 and the final meta-task loss is

$$L_{T_m} = \ell_{ce}(f(Q_m^*, \hat{W})) + \alpha \ell_{or}(\hat{W}) + \beta \ell_d(f_\theta(Q_m^*), \hat{W}), \quad (19)$$

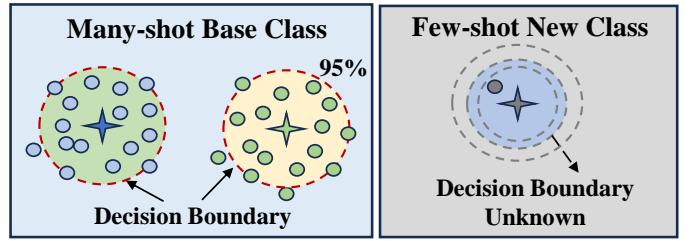


Fig. 5. Differences between many-shot and few-shot OSR. Obviously, with finite samples, it is difficult to estimate the true sample boundary. where α and β denote the hyper-parameters.

D. Adaptive RFF Open Set Recognition Mechanism

In traditional OSR, the most discussed problem is how to determine the open-set decision threshold. This is explored in detail in previous approaches, such as the extreme value theory (EVT) model used by Openmax [54], the three-sigma rule [13], and the probability value of 0.95 by softmax function. Current OSR methods typically require calibrated prediction scores or synthetic negative queries to learn open-set classifiers. They rely on a large amount of data to make a correct estimation of the distribution. However, this isn't easy to achieve if there are only a few shot-labeled instances. For example, as shown in Fig. 5, relying on the prototype points and many-shot samples, we can easily find a reliable open-set boundary threshold, i.e., at the manually defined 95% position. In the few-shot case, due to the lack of samples, it is almost impossible for us to determine the open-set boundary. For this purpose, this paper will propose a threshold-free adaptive OSR mechanism for the few-shot problem.

As mentioned before, the prototype point is a high-level summary of the class features, i.e. "which samples are similar to it". So, is there a point in the defined inner product space that opposes the prototype point? This point indicates the target class samples that are not similar to it, but the other classes are similar to it. Once this point has been computed, we can use this as a threshold for OSR, creating a threshold-free mechanism. Motivated by the notion of Reciprocal point [34], we believe that neural networks can be utilized to automatically synthesize Reciprocal points by powerful meta-learning techniques.

To accommodate the continuous incremental few-shot setting, we added a generative network G_θ and trained it with a large number of meta-tasks to learn how to synthesize Reciprocal point \mathcal{R} . In the previous meta-task setup in T , the many-shot query set Q^* and open set Q^o are divided. For Q^* , we extract its corresponding prototype point P_i of i -th class and generate the Reciprocal point \mathcal{R}_i using the generator G_θ . The generator network structure mainly employs the standard multi-head self-attentional (MHSA) mechanism, which can mine the relationships between prototype points efficiently. We apply the attention block between P_i and W to generate the Reciprocal point \mathcal{R} , i.e.,

$$\begin{aligned} Q' &= G_q^T P_i, K' = G_k^T \hat{W}, V' = G_v^T \hat{W}, \\ \alpha'_{qk} &= \text{softmax}\left(\frac{K'^T Q'}{\sqrt{d}}\right), \\ \mathcal{R}_i &= P_i + \sum_m \alpha'_{qk} V'_{\cdot m}, \end{aligned} \quad (20)$$

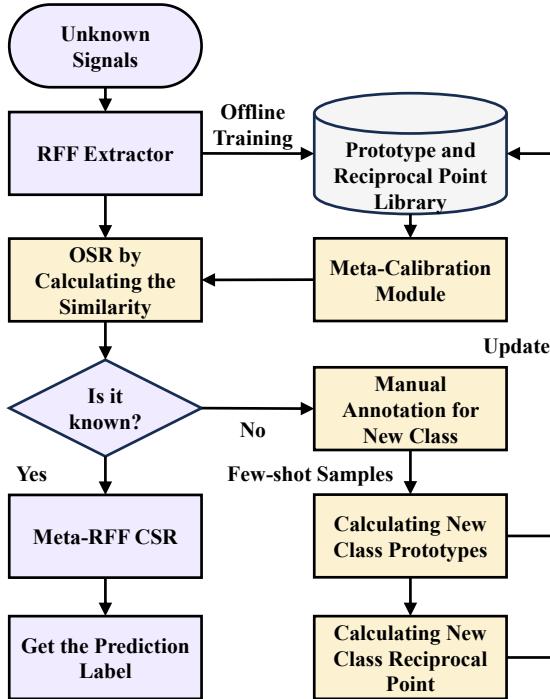


Fig. 6. Test flow of the Meta-RFF algorithm.

where G_q, G_k, G_v denote the parameters of MHSA, \mathcal{R}_i denotes the Reciprocal point of i -th class.

Once we have obtained the Reciprocal point \mathcal{R}_i of the target class, we can optimize it using the binary cross-entropy function l_{bce} :

$$l_{rpi} = \frac{1}{|Q^*|} \sum_{x \in Q^*} l_{bce}(1, sim(f_\theta(x), P_i)) + \frac{1}{|\mathcal{Q}^*|} \sum_{x \in \mathcal{Q}^*} l_{bce}(0, sim(f_\theta(x), \mathcal{R}_i)). \quad (21)$$

Instead, for constructed open set samples Q^o , we expect to possess less similarity to non-target class prototype points P_i and more similarity to Reciprocal points \mathcal{R}_i , which can be denoted by

$$l_{rpo} = \frac{1}{|Q^o|} \sum_{x \in Q^o} l_{bce}(0, sim(f_\theta(x), P_i)) + \frac{1}{|\mathcal{Q}^o|} \sum_{x \in \mathcal{Q}^o} l_{bce}(1, sim(f_\theta(x), \mathcal{R}_i)). \quad (22)$$

Without loss of generality, for task T_m , we have

$$l_{rp}^m = l_{rpi}^m(Q^*) + l_{rpo}^m(Q^o). \quad (23)$$

Finally, we combine Eq. 19 and 23 to obtain the final FSOCIL meta-task loss, i.e.,

$$L_{T_m} = \ell_{ce}(f(Q_m^*, \hat{W})) + \alpha \ell_{or}(\hat{W}) + \beta \ell_d(f_\theta(Q_m^*), \hat{W}) + \gamma \ell_{rp}(Q^*, Q^o), \quad (24)$$

where γ denotes the hyper-parameters.

E. Meta-RFF Working Procedure

Our method will be trained on the base session dataset \mathcal{D}^0 . First, to achieve both few-shot incremental recognition and OSR, we need to sample few-shot support set samples S , query samples Q^* , and open samples Q^o (Line 3) to construct the corresponding task. Second, the support set samples are utilized to update the prototype points and calibrate using the \mathcal{T} (Line 4-5). Further, we will compute the Reciprocal point of

Algorithm 1: Meta-RFF algorithm for FSOSIL Task

Input: Base session dataset \mathcal{D}^0 , pre-trained model $\psi(\cdot)$ and classifier weights W_0 , a randomly initialized Transformer model \mathcal{T} , new few-shot session dataset $\mathcal{D}^1, \dots, \mathcal{D}^b$ and unknown dataset \mathcal{D}^u

Output: The prediction results on session $\{\mathcal{D}^0, \dots, \mathcal{D}^b, \mathcal{D}^u\}$

```

1 for epoch = 1 to Epoch do
2   for j = 1 to n do
3      $\{S_j, Q_j^*, Q_j^o\} \leftarrow$  Sample the j-th support and
      query set for base classes from  $\mathcal{D}^0$  for task  $T_j$ 
      (Sec. IV-B);
4      $\{W_{T_j}\} \leftarrow$  Feed  $S_j$  into  $\psi(\cdot)$  and compute
      few-shot class prototype points  $c_i$  to update
      the classifier  $w_i$ ;
5      $\{\hat{w}, \hat{c}\} \leftarrow$  Feed  $w, c$  into  $\mathcal{T}$  for calibration (Sec.
      IV-C);
6      $\{\mathcal{R}\}$  Calculate the Reciprocal point (see in Sec.
      IV-D);
7      $\{L_{T_j}\} \leftarrow$  Calculate the loss by Eq. 24 ;
8   end
9   Calculate the average loss for all tasks by
10   $L_{T_n} = \frac{1}{n} \sum_{i=1}^n L_{T_j}((x_i; S_j, \psi_{j-1}, W_{j-1}), y_j);$ 
11  Update the parameters of the model  $\psi(\cdot)$  and  $\mathcal{T}$  by
      the SGD optimizer. ;
12 end
13 Freeze parameters  $\psi(\cdot)$ ,  $\mathcal{T}$  and make predictions on
      new session data  $\mathcal{D}^b, \mathcal{D}^u$  to get the labels  $Y_b, Y_u$ ;
14 return Current data labels  $Y_b$  and unknown data labels
       $Y_u$ ;
  
```

the prototype point (Line 6). Finally, we will compute the loss for all pseudo-tasks and perform meta-training (Line 7-10).

In the test phase, as shown in Fig. 6, Our FSOSIL task setup is first to have the model perform open-set recognition. If the classes are known, we will perform closed-set classification. Once the result is unknown, manual research and labeling of sample labels are required. Here, to reduce the cost of manual labeling, only a small number of samples are labeled. Finally, we will be updating the prototype point and Reciprocal point libraries with new classes for few-shot.

V. EXPERIMENTS AND RESULTS

A. Experiment Setup

Experimental Environment. The *Meta-RFF* algorithm is implemented by using Pytorch 1.10.0 and executed on a computer running Unbantu 18.04.6 LTS, with Intel(R) Core(TM) i9-10900K CPU@3.70 GHz and 2 NVIDIA GeForce RTX3090 GPUs. In addition, we mainly use Adam's [55] optimizer for the optimization of CNN parameters and depict the relevant parameters for our *Meta-RFF*. Then we use 1D ResNet-18 as the backbone network both for our algorithms and other baseline algorithms.

Dataset. We use real-world ADS-B signals to verify FSOCIL methods for wireless device identification. ADS-B signals are

TABLE II

COMPARISON WITH THE STATE-OF-THE-ART ON ADS-B DATASET. “*” DENOTES THE USE OF A FINE-TUNING OPERATION IN THE ALGORITHM, “†” DENOTES THE USE OF CATSTLE IN THE ALGORITHM, FR DENOTES THE PERFORMANCE FORGETTING RATE, AND I_t DENOTES THE INTRANSIGENCE.

Methods	Test set	Accuracy in each Session with 1-shot (%) ↑										FR↓	I_t (avg.)	
		0	1	2	3	4	5	6	7	8	9			
New Task 1-shot	Theoretical Boundary	-	89.71	87.91	85.64	86.35	83.76	84.08	82.78	82.48	81.10	81.35	-	-
Softmax*	Base	98.76	81.44	9.09	8.75	4.53	0.48	1.21	2.57	1.00	0.57	0.09	-98.67%	-
	Incremental	-	0.00	0.00	0.53	1.35	0.35	0.19	0.20	0.59	1.48	0.54	None	83.92%
ARPL*	Base	97.32	90.90	27.62	47.03	19.08	27.32	19.79	14.08	9.10	5.42	6.09	-91.23%	-
	Incremental	-	0.00	0.00	0.02	0.36	0.24	0.43	0.69	1.05	1.27	0.71	None	84.04%
PEELER†	Base	99.20	93.90	93.82	93.72	93.62	93.53	88.13	88.07	88.02	70.35	70.28	-28.92%	-
	Incremental	-	49.52	42.82	41.90	40.03	38.29	36.20	35.36	34.66	27.87	26.40	-23.12%	42.21%
iCaRL	Base	99.86	91.43	90.19	88.89	87.98	86.57	84.43	83.00	80.34	78.15	76.79	-23.07%	-
	Incremental	-	1.51	11.56	5.16	6.28	5.70	6.02	3.83	4.08	3.78	4.35	None	79.62%
CATSTLE	Base	99.32	98.32	98.25	98.23	98.20	98.18	98.12	98.10	98.08	97.99	97.96	-1.36%	-
	Incremental	-	55.38	50.30	46.23	44.30	42.35	42.73	42.01	41.27	40.40	40.66	-14.72%	39.94%
CEC	Base	98.76	97.04	96.94	96.90	96.85	96.77	96.56	96.52	96.45	96.38	96.34	-2.42%	-
	Incremental	-	58.33	49.77	44.83	44.10	42.95	43.26	42.35	41.98	41.84	42.42	-15.91%	39.33%
LIMIT	Base	99.96	98.63	98.15	97.61	97.17	96.84	96.52	96.12	95.88	95.49	95.23	-4.73%	-
	Incremental	-	70.08	67.73	63.82	59.04	57.58	57.95	55.35	54.60	52.92	52.63	-17.45%	25.34%
Meta-RFF (1-shot)	Base	99.86	99.82	99.79	99.78	99.77	99.76	99.76	99.70	99.68	99.67	99.64	-0.22%	-
	Incremental	-	80.17	79.17	78.20	78.21	76.42	76.62	76.98	77.70	76.92	77.03	-3.14%	6.7%
Meta-RFF (5-shot)	Base	99.86	99.83	99.76	99.75	99.73	99.72	99.70	99.65	99.60	99.55	99.54	-0.32%	-
	Incremental	-	94.65	92.89	91.10	90.88	89.88	89.95	91.54	91.49	91.22	91.57	-3.08%	-

transmitted by commercial aircraft to periodically broadcast their route information to air traffic control (ATC) centers in plain text [17]. These signals are easy to receive and decode but are subject to identity spoofing attacks. We use the first 1024 complex samples. This data set is publicly available at [56]. For the FSOCIL task setting, we extracted IQ samples for a total of 893 aircraft in the dataset. Aircraft categories ranging from 0-237 possess many shots, which can be used as the base session dataset. 200-437 range has relatively few samples, which can be used as the few-shot incremental session. The samples in the range of 437-893 are scarce and not easy to train, and this paper serves as an open unknown class. Then, to verify the availability of the relevant algorithms in a realistic environment, we develop a real-time RF signal recognition system based on USRP B210 as described in VI.

Models’ Setup and Comparison Baselines. To ensure the fairness of the comparison, we use a uniform network architecture for all baseline methods. As shown in Fig. 4 and 3, we use 1D ResNet-18 as the backbone network both for our algorithms and other baseline algorithms. In addition, we use standard transformer as a calibration module and adaptive generator. To evaluate the effectiveness of our proposed *Meta-RFF* on the FSOSIL task, we first compare it to current state-of-the-art FSOSR algorithms, *e.g.*,

- Softmax [57]: Supervised learning is used for training and testing.
- ARPL [34]: Open-set recognition by automatic learning of Reciprocal points.
- PEELER [46]: Using meta-learning to train open-set samples for FSOSR task.

Besides, we also compare to current state-of-the-art FSCIL algorithms,

- iCaRL [32]: Classical incremental learning algorithms that preserve old data features through distillation.
- CASTLE [50]: A few-shot incremental learning method based on decoupling.

- CEC [33]: The method mainly updates few-shot samples of prototype point distributions in a continuous incremental phase via a graph network.
- LIMIT [27]: The method mainly utilizes the Transformer calibration to update the distribution gap between the sample and prototype points for the FSCIL task.

For those FSOSR approaches, we utilize the decoupling mechanism from CASTLE to enable the incremental session, with a superscript “*”. We use identical training splits for every method for a fair comparison. For those FSCIL approaches, we follow the training mechanism reported in the relevant papers and evaluate the open-set recognition performance directly.

Metrics. To quantitatively evaluate the FSOSIL task performance, we use Accuracy (Acc) as a metrics

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}, \quad (25)$$

where TP, TN, FP, FN denote the true-positives, true-negatives, false-positives, and false-negatives, respectively. To measure the effectiveness of continuous learning, we also defined forgetting rate (FR) and intransigence (I_t),

$$FR = \frac{1}{t-1} \sum_{k=1}^{t-1} (Acc_k^{\text{initial}} - Acc_k^{\text{after } t}) \quad (26)$$

$$I_t = Acc_t^{\text{upper-bound}} - Acc_t^{\text{task } t}$$

where Acc_k^{initial} denote the old task performance, $Acc_k^{\text{after } t}$ denotes the performance after task t , $Acc_t^{\text{upper-bound}}$ denotes the upper bound on the theoretical performance of task t , $Acc_t^{\text{task } t}$ denotes the current performance of task t . In addition, for open-set recognition, referring to the evaluation index in [12], [34], we use AUROC as a metric.

B. Benchmark Comparison

Comparison of Closed-Set Performance among Baselines. As shown in Table II, we compare the accuracy of closed-set data with mainstream algorithms. We divided the test dataset

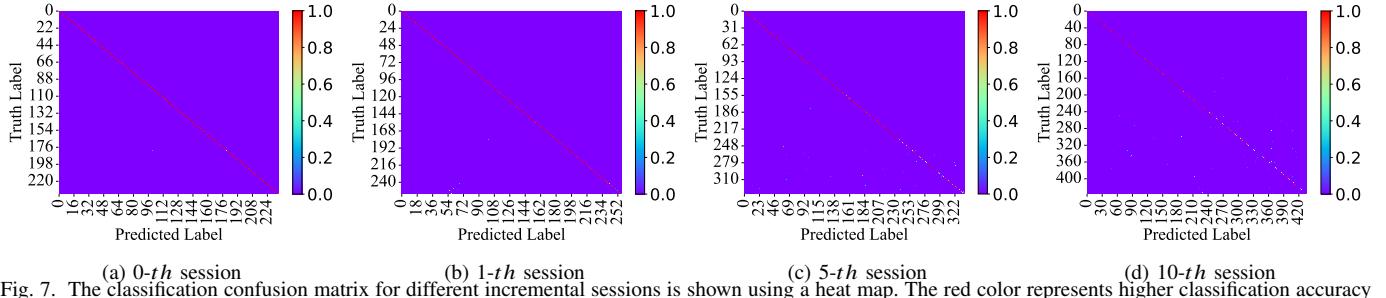


Fig. 7. The classification confusion matrix for different incremental sessions is shown using a heat map. The red color represents higher classification accuracy and the blue color represents low accuracy.

TABLE III

COMPARISON OF FEW-SHOT INCREMENTAL OPEN-SET RECOGNITION PERFORMANCE AND STATE-OF-THE-ART ALGORITHMS. “*” DENOTES THE USE OF A FINE-TUNING OPERATION IN THE ALGORITHM, “†” DENOTES THE USE OF CATSTLE IN THE ALGORITHM.

Methods	FSOSR	FSCIL	AUROC in each session with 1-shot ↑										
			0	1	2	3	4	5	6	7	8	9	10
Softmax*	-	-	0.9546	0.8697	0.6543	0.6116	0.5398	0.5426	0.5881	0.5375	0.4555	0.5569	0.5475
ARPL*	✓	-	0.9564	0.9054	0.7164	0.7593	0.6765	0.6698	0.6848	0.6413	0.6392	0.6057	0.5795
PEELER†	✓	-	0.9414	0.9341	0.9278	0.9229	0.9185	0.9141	0.8918	0.8878	0.8837	0.8330	0.8287
iCaRL	-	✓	0.9090	0.8490	0.8250	0.7940	0.7802	0.7642	0.7422	0.7321	0.7221	0.7071	0.7011
CATSTLE	-	✓	0.7328	0.7308	0.7488	0.7497	0.7629	0.7653	0.7730	0.7807	0.7816	0.7767	0.7652
CEC	-	✓	0.7319	0.7265	0.7940	0.8010	0.8060	0.8073	0.8098	0.8166	0.8165	0.7990	0.7842
LIMIT	-	✓	0.9787	0.9666	0.9534	0.9409	0.9298	0.9174	0.9122	0.9015	0.8984	0.8892	0.8834
Meta-RFF	✓	✓	0.9965	0.9921	0.9867	0.9803	0.9755	0.9715	0.9672	0.9645	0.9622	0.9561	0.9529

TABLE IV

ACCURACY COMPARISON OF OPEN SET RECOGNITION WITH STATE-OF-THE-ART ALGORITHMS. “†” DENOTES THE USE OF CATSTLE IN THE ALGORITHM.

Methods	Manual Thresholds	Adaptive Thresholds	Average open-set accuracy in each session with 1-shot (%) ↑										
			0	1	2	3	4	5	6	7	8	9	10
PEELER†	✓	-	85.86	84.81	83.91	83.00	82.22	81.51	79.20	78.55	78.09	73.11	72.78
CEC	✓	-	69.79	86.01	86.44	80.41	67.84	56.35	51.44	50.56	50.26	50.03	50.00
LIMIT	✓	-	87.74	87.38	86.71	86.32	85.45	84.33	84.00	83.38	83.71	82.95	82.48
Meta-RFF	-	✓	92.67	90.82	90.44	89.60	88.98	88.02	87.65	85.79	85.52	84.84	84.56

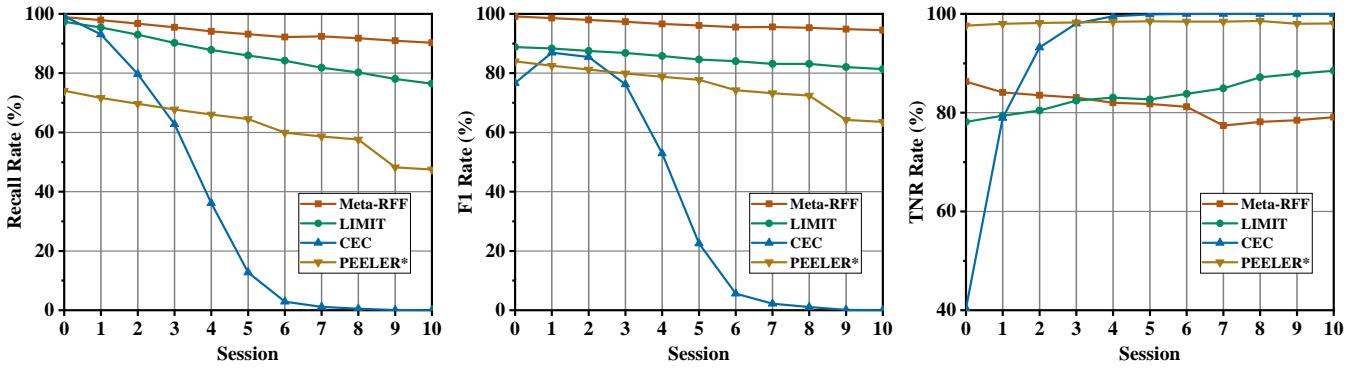


Fig. 8. Performance comparison for open-set recognition.

into base set data and incremental data, where the base set dataset has a total of 237 classes and the incremental data has a total of 200 classes. We set 10 incremental sessions and the number of categories in each session is 20. In traditional supervised learning algorithms (Softmax, ARPL), we can see that the lack of incremental learning capability of the model produces a catastrophic forgetting problem, with test accuracies for the base set and incremental classes of data almost approaching 0. In addition, although the classical incremental learning algorithm iCaRL guarantees a certain base set accuracy, it cannot adapt to few-shot incremental environments and its performance on incremental data is poor. In contrast,

the rest of the algorithms that impose a few-shot incremental mechanism not only solve the catastrophic forgetting problem but also achieve good results on incremental datasets. However, we can see that the proposed *Meta-RFF* achieves state-of-the-art performance on both base and incremental data. With the 1-shot setting, our *Meta-RFF* improves performance over LIMIT by 10% on incremental data and has the lowest performance forgetting rate on the base class and incremental data, 3.14%, 0.22%, respectively. In addition, we evaluated the intransigence rate metrics against the new task, and it can be seen that the other methods have high recalcitrance rates. In contrast, the method in this paper has a low recalcitrance rate.

TABLE V
COMPARISON OF OPEN-SET RECOGNITION EFFICIENCY WITH DIFFERENT TRAINING MODULES.

Prototype	Data Augmentation	Meta Train	Open Loss	AUROC in each session with 1-shot (%) ↑										
				0	1	2	3	4	5	6	7	8	9	10
-	-	-	-	95.46	86.97	65.43	61.16	53.98	54.26	58.81	53.75	45.55	55.69	54.75
✓	-	-	-	98.85	97.60	96.22	95.43	94.43	93.34	92.33	91.29	90.17	87.76	85.76
✓	✓	-	-	98.73	97.75	96.70	96.07	95.21	94.61	93.95	92.26	91.73	90.81	89.79
✓	✓	✓	-	99.03	98.27	97.48	96.84	96.14	95.49	95.13	94.53	94.20	93.68	93.27
✓	✓	✓	✓	99.65	99.21	98.67	98.03	97.55	97.15	96.72	96.45	96.22	95.61	95.29

TABLE VI
COMPARISON OF ACCURACY EFFICIENCY WITH DIFFERENT TRAINING MODULES.

Prototype	Data Augmentation	Meta Train	Open Loss	Accuracy in each session with 1-shot (%) ↑										
				0	1	2	3	4	5	6	7	8	9	10
-	-	-	-	98.76	82.78	9.09	8.75	4.53	0.48	1.21	2.57	1.00	0.57	0.09
✓	-	-	-	98.76	82.38	70.01	63.57	63.21	63.23	58.65	50.03	53.13	52.65	44.34
✓	✓	-	-	99.82	82.78	70.47	64.66	66.69	65.27	59.47	51.72	54.82	53.75	45.74
✓	✓	✓	-	99.82	88.34	86.22	84.55	82.94	81.53	82.05	79.67	78.87	77.57	77.83
✓	✓	✓	✓	99.83	88.34	86.23	84.55	82.95	81.53	82.06	79.66	78.88	77.56	77.83

The lower intransigence rate reflects the greater adaptability of the model to new tasks and demonstrates the degree of reliance on the model's old knowledge. With the 5-shot setting, our algorithm can achieve accuracy beyond 90% for incremental classes. Fig. 7 illustrates the classification confusion matrix for several stages of incremental sessions. We can see a clear red diagonal line and most of the remaining area is blue, indicating a high classification accuracy and low misclassification rate for each category. The above results show that our *Meta-RFF* can significantly improve few-shot incremental performance by imposing RF data augmentation, multi-stage meta-task training, and soft orthogonalization loss.

Comparison of Open-Set Performance among OSR Baselines. Table III shows the AUROC performance for the few-shot incremental open-set recognition task, where higher AUROC means better open-set recognition performance. We can see that our algorithm not only achieves state-of-the-art performance over multiple incremental sessions but also possesses a low-performance decay rate. This result is mainly because we impose open-set loss in multi-stage meta-task training, which reduces the inter-class differences. In addition, we normalize the distance between prototype points using soft orthogonalization loss, which allows new prototype points to be automatically separated from the old ones and adjust the distribution during a continuous incremental process. This mechanism allows our algorithm to adapt to multiple incremental processes, yielding a low AUROC decay rate.

We further compare other metrics for OSR. Table IV shows the open-set recognition accuracy of algorithms. In traditional methods, we usually set 95% as the open-set sample discrimination threshold [34]. It can be seen that the adaptive threshold used by our algorithm can achieve better discriminative performance compared to the manual threshold. As mentioned earlier, due to the few-shot limitation, we cannot utilize traditional methods to obtain a reliable boundary threshold, which leads to the fact that traditional methods are ineffective in this case. In contrast, our algorithm avoids this limitation. Reciprocal points can be generated using only the prototype points of the target class, solving the limitation that traditional methods require samples to estimate decision boundaries. Further, due

to the use of meta-learning techniques, our model can be proficient in the accurate generation of reciprocal points.

Fig. 8 illustrates the results of Recall, F1-score, and True Negative Rate (TNR) for OSR. In Fig. 8a and 8b, it is obvious that our method achieves optimal recognition performance on closed-set compared to existing threshold-based methods. Then in Fig. 8c, we can observe that CEC and PEERLER recognize all the unknown samples as open sets. The reason for this phenomenon is the improper selection of the open-set threshold, i.e., the threshold is selected too harshly, resulting in the accuracy of the known-set samples dropping to 0 in Fig. 8a and 8b. It is observed that the adaptive threshold used in this paper can solve this problem and obtain reliable open-set accuracy while guaranteeing closed-set accuracy.

C. Ablation Study

Effects of different training modules. As shown in Table V, we evaluate the efficiency of OSR for different modules. In multiple incremental sessions, it can be observed that the prototype network can greatly improve the OSR efficiency, which is mainly because the prototype points can narrow down the inter-class distance and reduce the overlapping area between unknown and known classes. Then the operation of data augmentation can further improve the effectiveness of AUROC, which proves that mining relevant signal properties can be helpful for OSR. Multi-stage training using meta-learning will be able to reduce performance decay compared to single-stage training, showing that the use of meta-learning to adapt the network to the FSOSIL task is necessary. Finally, the use of open-set loss further reduces the area of overlap between unknown and known samples, thus improving AUROC. To verify that the improvement in open-set performance may stem from the advancement in closed-set performance, we further evaluated the ablation for few-shot increments, as shown in Table VI. We can see that the few-shot learning ability of the model is greatly improved after the prototype network is imposed, by about 50%. Then the multi-feature data used in this paper can make some contribution to the closed-set accuracy improvement. But these two models fail to solve the problem of continuous incremental accuracy decay well.

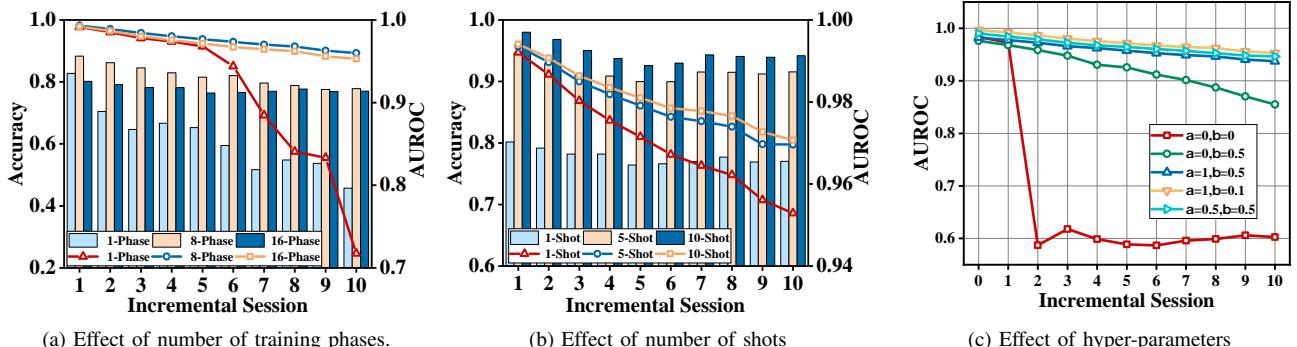


Fig. 9. Accuracy and AUROC performance comparison of different parameters.

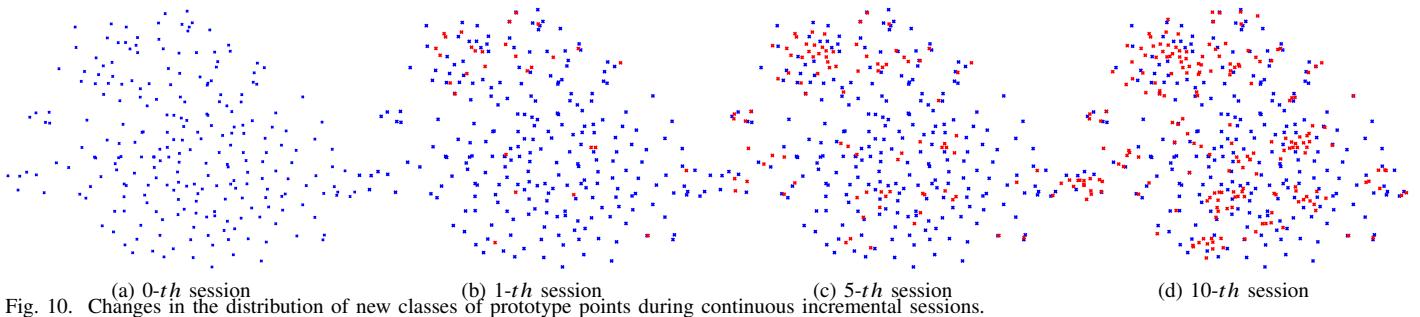


Fig. 10. Changes in the distribution of new classes of prototype points during continuous incremental sessions.

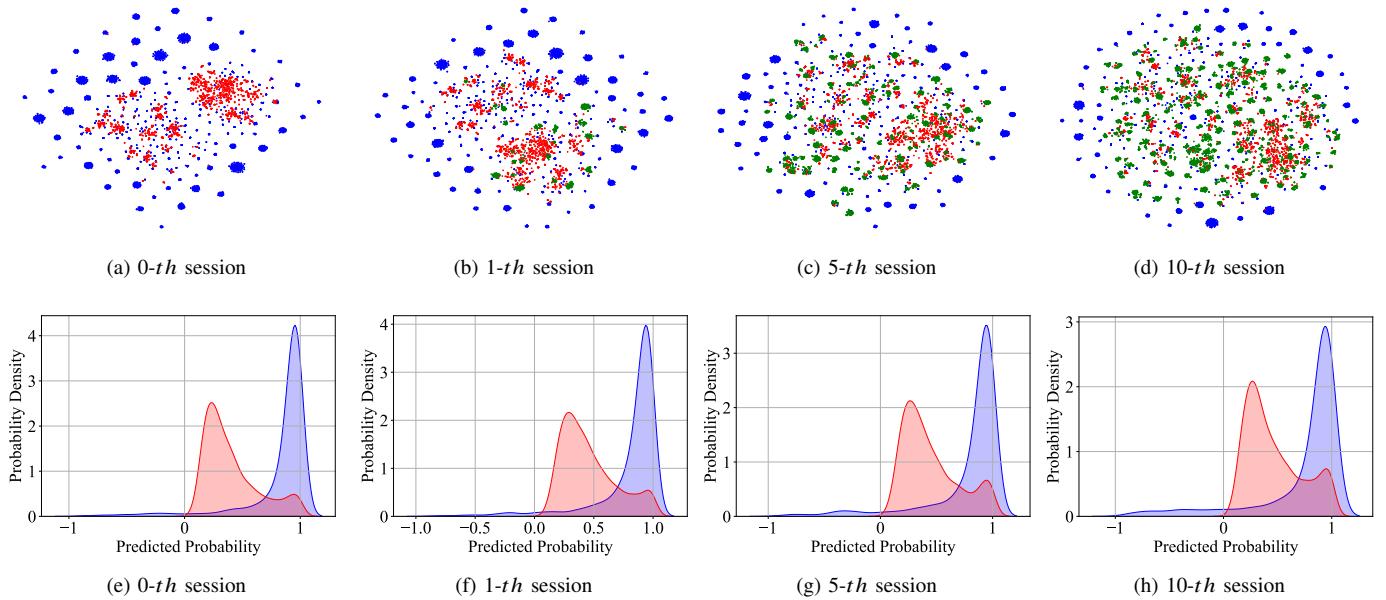


Fig. 11. (a-d) The changes in the distribution of the prototype points over the continuous incremental session. (e-h) The changes in the distribution of probability values of known class samples and unknown samples over the continuous incremental session.

After introducing the meta-training in this paper, the model has higher recognition accuracy during continuous increments, and its decay rate is smaller. Finally, the open-set loss did not produce a large improvement in closed-set accuracy, because it is mainly used to improve open-set performance. The above results illustrate that there is a strong correlation between the improvement of open-set accuracy and closed-set accuracy in Table V and Table VI, and better closed-set performance enables the model to recognize open sets more accurately.

Effects of the number of phases and shots. Fig. 9a illustrates the performance impact of the number of phases on FSCIL as

well as OSR in a continuous incremental session. In this figure, the line graph represents AUROC and the bar graph represents accuracy. We can observe that under 1-phase, the model's classification accuracy and open-set performance decline significantly in incremental sessions, which implies that adding more meta-task phases improves the model's ability. Similarly, in Fig. 9b, increasing the sample size of the incremental class will be able to further improve the model classification ability and open-set recognition ability.

Effect of hyper-parameters. Fig. 9c evaluates the effect of parameters α and β in the model regularization term on OSR,

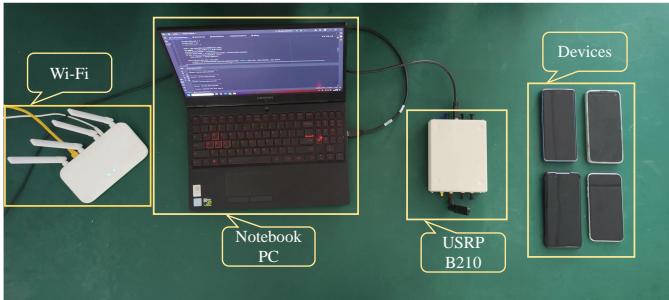


Fig. 12. Real-time RF Fingerprint recognition system, which consists of Notebook PC, USRP B210, WiFi access point, and multiple devices.

where α denotes the prototype orthogonalization loss and β denotes the open loss in Eq. 19. When $\alpha = 0, \beta = 0$, we can observe clearly that the AUROC of OSR performance drops drastically in continuous incremental sessions. The AUROC of the model improves significantly when we apply just $\beta = 0.5$, indicating that reducing the distance between the samples and the prototype points can reduce the area of overlap with the unknown samples. Our model achieved optimal AUROC performance when we parameterized both regularization terms α, β . In particular, modifying the parameter values of α and β produces AUROC performance that is close.

D. Visualization and Interpretability of Meta-RFF

Visualization of feature space by T-SNE. Fig. 10 illustrates the changes in the distribution of the prototype points over the continuous incremental session. First, in Fig. 10a, the base class prototype points (blue) are distributed in the feature space with almost no overlap. In Fig. 10b, 10c, and 10d, we can observe clearly that the new class prototype points (red) have almost no overlap with the base class prototype points, which demonstrates that our model can automatically pair-generate effective few-shot new class prototype points. Then in Fig. 11, we carve out the distribution of features for samples of known (blue) and unknown (red) classes at different incremental sessions. For a more fine-grained representation, we plotted the similarity distribution as shown in Fig. 11e-11h. The blue color indicates the similarity between known class samples and prototype points, and the red color indicates the similarity between unknown class samples and prototype points. We can observe that the probability distribution of most of the known set samples is concentrated around 1, while that of the unknown class samples is concentrated around 0. In addition, we can observe clearly that the open-set recognition performance decays less in continuous incremental sessions.

VI. REAL-WORLD SYSTEM PERFORMANCE ANALYSIS

In this section, we build a test bed and evaluate the performance of our method in real world scenario.

System Setup. We use USRP to obtain WiFi signals from 20 smartphones with the protocol IEEE 802.11n. Fig. 12 shows the data collection test-bed, where the devices (smartphones) is connecting with WiFi router while exchanging data. The USRP captures the WiFi signals in real time. We intercept the raw I/Q of the preamble sequence (i.e., the first 320 complex samples)

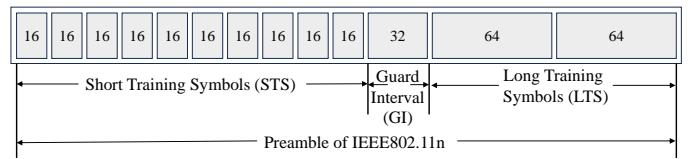


Fig. 13. The preamble of the IEEE 802.11n protocol for WiFi. We use this preamble to perform the classification task.

TABLE VII
CONTINUOUS INCREMENTAL OPEN-SET RECOGNITION PERFORMANCE FOR 20 DEVICES BY USING 1-SHOT FINE-TUNING.

Session	Base Accuracy	Incremental Accuracy	Overall OSR	Open-set Accuracy
0	0.9873	0.9869	0.8988	0.8922
1	0.9803	0.9837	0.8914	0.8807
2	0.9790	0.8960	0.8826	0.8658
3	0.9782	0.7432	0.8847	0.8667
4	0.9787	0.7410	0.9079	0.9034
5	0.9771	0.6223	0.8983	0.8849
6	0.9771	0.5457	0.8909	0.8662
7	0.9766	0.5684	0.9456	0.9909
8	0.9559	0.6178	0.9372	0.9865
9	0.9549	0.5862	0.9271	0.9834

as shown in Fig. 13 of each packet as the RF fingerprinting data, and number the source MAC address of each device as the data label to build the RF fingerprinting dataset.

Evaluation Procedure. We trained our model with the data from 10 devices and tested it in real time on the remaining 10 devices. We set 10 incremental sessions and increment one class at a time. Then, to evaluate the performance of OSR, the remaining classes, except for the base and incremental classes, are also identified as unknown. Once our *Meta-RFF* first identifies the unknown device, then takes 1 new class sample for model fine-tuning and continues to perform the detection.

Results. In Table VII, “base accuracy” and “incremental accuracy” represent the test accuracy for base class samples and incremental samples during a continuous incremental session, respectively. The term “overall OSR” refers to the open-set recognition accuracy for both known and unknown classes, while “open-set accuracy” specifically measures the recognition rate of unknown samples. It is evident that both “base accuracy” and “incremental accuracy” are high, though the incremental class experiences significant decay across multiple increments due to the limitations of few-shot learning. These results indicate that our algorithm effectively mitigates the catastrophic forgetting problem in real-world scenarios and achieves high OSR performance.

Additionally, Fig. 14 illustrates the real-time running latency of the algorithm on a computer equipped with an RTX3060 and an Intel(R) Core(TM) i7-12650H processor. From the received signal, data preprocessing takes approximately 3 ms, while the *Meta-RFF* algorithm requires about 30-35 ms. Although the system’s current latency is relatively high compared to the 40 ms interval between received packets, we plan to employ model distillation and fine-tuning techniques in the future to reduce model parameters and enhance real-time performance.

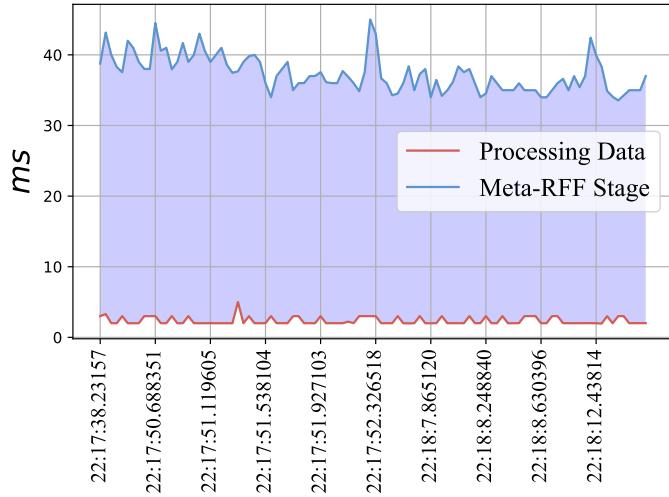


Fig. 14. The runtime delay of the RFF real-time recognition system begins when the packet is received. The horizontal axis indicates the point in time, vertical axis indicates delay time (ms).

VII. CONCLUSION AND FUTURE WORK

In this paper, we propose a continuous evolutionary algorithm for RFF recognition models that implements FSOSIL tasks. Our algorithm incorporates a meta-learning technique that utilizes a large number of FSOSIL tasks constructed and performs meta-training to gradually adapt to the FSOSIL scenarios. In particular, to improve the performance of open-set recognition during a continuous incremental process, we introduce open loss as well as an adaptive open-set threshold generation technique based on reciprocal points. In the future, for the RFF recognition model, we will further construct the signal large model to realize a more general recognition system.

REFERENCES

- [1] V. Niazmand and Q. Ye, "Joint task offloading, dnn pruning, and computing resource allocation for fault detection with dynamic constraints in industrial iot," *IEEE Transactions on Cognitive Communications and Networking*, 2025.
- [2] D. Jia and Q. J. Ye, "Mobility-adaptive digital twin modeling for post-disaster network traffic prediction," in *2024 IEEE 100th Vehicular Technology Conference (VTC2024-Fall)*. IEEE, 2024, pp. 1–7.
- [3] T. Emmens, C. Amrit, A. Abdi, and M. Ghosh, "The promises and perils of automatic identification system data," *Expert Systems with Applications*, vol. 178, p. 114975, 2021.
- [4] D. Li, J. Qi, S. Hong, P. Deng, and H. Sun, "A class-incremental approach with self-training and prototype augmentation for specific emitter identification," *IEEE Transactions on Information Forensics and Security*, 2023.
- [5] J. Subramani, A. Maria, R. B. Neelakandan, and A. S. Rajasekaran, "Efficient anonymous authentication scheme for automatic dependent surveillance-broadcast system with batch verification," *IET Communications*, vol. 15, no. 9, pp. 1187–1197, 2021.
- [6] Y. Huang *et al.*, "Radio frequency fingerprint extraction of radio emitter based on i/q imbalance," *Procedia computer science*, vol. 107, pp. 472–477, 2017.
- [7] 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.
- [8] A. C. Polak and D. L. Goeckel, "Identification of wireless devices of users who actively fake their rf fingerprints with artificial data distortion," *IEEE Transactions on Wireless Communications*, vol. 14, no. 11, pp. 5889–5899, 2015.
- [9] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008, pp. 116–127.
- [10] S. Jana and S. K. Kasera, "On fast and accurate detection of unauthorized wireless access points using clock skews," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008, pp. 104–115.
- [11] S. Zheng, X. Zhou, L. Zhang, P. Qi, K. Qiu, J. Zhu, and X. Yang, "Toward next-generation signal intelligence: A hybrid knowledge and data-driven deep learning framework for radio signal classification," *IEEE Transactions on Cognitive Communications and Networking*, vol. 9, no. 3, pp. 564–579, 2023.
- [12] T. Li, Z. Wen, Y. Long, Z. Hong, S. Zheng, L. Yu, B. Chen, X. Yang, and L. Shao, "The importance of expert knowledge for automatic modulation open set recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.
- [13] Y. Dong, X. Jiang, H. Zhou, Y. Lin, and Q. Shi, "Sr2cnn: Zero-shot learning for signal recognition," *IEEE Transactions on Signal Processing*, vol. 69, pp. 2316–2329, 2021.
- [14] S. Rajendran and Z. Sun, "Rf impairment model-based iot physical-layer identification for enhanced domain generalization," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 1285–1299, 2022.
- [15] G. Shen, J. Zhang, A. Marshall, and J. R. Cavallaro, "Towards scalable and channel-robust radio frequency fingerprint identification for lora," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 774–787, 2022.
- [16] F. A. Bhatti, M. J. Khan, A. Selim, and F. Paisana, "Shared spectrum monitoring using deep learning," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 4, pp. 1171–1185, 2021.
- [17] Y. Liu, J. Wang, J. Li, S. Niu, and H. Song, "Class-incremental learning for wireless device identification in iot," *IEEE Internet of Things Journal*, vol. 8, no. 23, pp. 17 227–17 235, 2021.
- [18] T. Li, Z. Hong, Q. Cai, L. Yu, Z. Wen, and R. Yang, "Bissiam: Bispectrum siamese network based contrastive learning for uav anomaly detection," *IEEE Transactions on Knowledge and Data Engineering*, 2021.
- [19] X. Li, F. Dong, S. Zhang, W. Guo *et al.*, "A survey on deep learning techniques in wireless signal recognition," *Wireless Communications and Mobile Computing*, vol. 2019, 2019.
- [20] Q. Ye, W. Zhuang, S. Zhang, A.-L. Jin, X. Shen, and X. Li, "Dynamic radio resource slicing for a two-tier heterogeneous wireless network," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 10, pp. 9896–9910, 2018.
- [21] S. Rajendran, Z. Sun, F. Lin, and K. Ren, "Injecting reliable radio frequency fingerprints using metasurface for the internet of things," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 1896–1911, 2020.
- [22] J. Gong, X. Xu, and Y. Lei, "Unsupervised specific emitter identification method using radio-frequency fingerprint embedded infogan," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2898–2913, 2020.
- [23] Y. Lin, Y. Tu, and Z. Dou, "An improved neural network pruning technology for automatic modulation classification in edge devices," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 5, pp. 5703–5706, 2020.
- [24] Y. Lin, Y. Tu, Z. Dou, L. Chen, and S. Mao, "Contour stella image and deep learning for signal recognition in the physical layer," *IEEE Transactions on Cognitive Communications and Networking*, vol. 7, no. 1, pp. 34–46, 2020.
- [25] K. Qu, W. Zhuang, Q. Ye, W. Wu, and X. Shen, "Model-assisted learning for adaptive cooperative perception of connected autonomous vehicles," *IEEE Transactions on Wireless Communications*, vol. 23, no. 8, pp. 8820–8835, 2024.
- [26] H. Li, Y. Tang, D. Lin, Y. Gao, and J. Cao, "A survey of few-shot learning for radio frequency fingerprint identification," in *Artificial Intelligence for Communications and Networks: Third EAI International Conference, AICON 2021, Xining, China, October 23–24, 2021, Proceedings, Part I 3*. Springer, 2021, pp. 433–443.
- [27] D.-W. Zhou, H.-J. Ye, L. Ma, D. Xie, S. Pu, and D.-C. Zhan, "Few-shot class-incremental learning by sampling multi-phase tasks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.
- [28] T. Ya, L. Yun, Z. Haoran, W. Yu, G. Guan, M. Shiwen *et al.*, "Large-scale real-world radio signal recognition with deep learning," *Chinese Journal of Aeronautics*, vol. 35, no. 9, pp. 35–48, 2022.
- [29] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska

- et al.*, “Overcoming catastrophic forgetting in neural networks,” *Proceedings of the national academy of sciences*, vol. 114, no. 13, pp. 3521–3526, 2017.
- [30] S. Ravi and H. Larochelle, “Optimization as a model for few-shot learning,” in *International conference on learning representations*, 2016.
- [31] J. Snell, K. Swersky, and R. Zemel, “Prototypical networks for few-shot learning,” *Advances in neural information processing systems*, vol. 30, 2017.
- [32] S.-A. Rebuffi, A. Kolesnikov, G. Sperl, and C. H. Lampert, “icarl: Incremental classifier and representation learning,” in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 2001–2010.
- [33] C. Zhang, N. Song, G. Lin, Y. Zheng, P. Pan, and Y. Xu, “Few-shot incremental learning with continually evolved classifiers,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12455–12464.
- [34] G. Chen, P. Peng, X. Wang, and Y. Tian, “Adversarial reciprocal points learning for open set recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 8065–8081, 2021.
- [35] H.-M. Yang, X.-Y. Zhang, F. Yin, and C.-L. Liu, “Robust classification with convolutional prototype learning,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3474–3482.
- [36] C. Finn, P. Abbeel, and S. Levine, “Model-agnostic meta-learning for fast adaptation of deep networks,” in *International conference on machine learning*. PMLR, 2017, pp. 1126–1135.
- [37] H. Gharnoun, F. Momenifar, F. Chen, and A. H. Gandomi, “Meta-learning approaches for few-shot learning: A survey of recent advances,” *ACM Computing Surveys*, vol. 56, no. 12, pp. 1–41, 2024.
- [38] Y. Lin, M. Wang, X. Zhou, G. Ding, and S. Mao, “Dynamic spectrum interaction of uav flight formation communication with priority: A deep reinforcement learning approach,” *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 3, pp. 892–903, 2020.
- [39] K. Chen and C.-G. Lee, “Incremental few-shot learning via vector quantization in deep embedded space,” in *International Conference on Learning Representations*, 2021.
- [40] Y. Cui, W. Xiong, M. Tavakolian, and L. Liu, “Semi-supervised few-shot class-incremental learning,” in *2021 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2021, pp. 1239–1243.
- [41] A. Kukleva, H. Kuehne, and B. Schiele, “Generalized and incremental few-shot learning by explicit learning and calibration without forgetting,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 9020–9029.
- [42] X. Tao, X. Hong, X. Chang, S. Dong, X. Wei, and Y. Gong, “Few-shot class-incremental learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12183–12192.
- [43] S. Xie, Y. Li, D. Lin, T. Lay Nwe, and S. Dong, “Meta module generation for fast few-shot incremental learning,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pp. 0–0.
- [44] H. Kang, J. Yoon, S. R. H. Madjid, S. J. Hwang, and C. D. Yoo, “On the soft-subnetwork for few-shot class incremental learning,” *arXiv preprint arXiv:2209.07529*, 2022.
- [45] Y. Yang, H. Yuan, X. Li, Z. Lin, P. Torr, and D. Tao, “Neural collapse inspired feature-classifier alignment for few-shot class incremental learning,” *arXiv preprint arXiv:2302.03004*, 2023.
- [46] B. Liu, H. Kang, H. Li, G. Hua, and N. Vasconcelos, “Few-shot open-set recognition using meta-learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8798–8807.
- [47] M. Jeong, S. Choi, and C. Kim, “Few-shot open-set recognition by transformation consistency,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12566–12575.
- [48] S. Huang, J. Ma, G. Han, and S.-F. Chang, “Task-adaptive negative envision for few-shot open-set recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 7171–7180.
- [49] B. Flowers, R. M. Buehrer, and W. C. Headley, “Evaluating adversarial evasion attacks in the context of wireless communications,” *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 1102–1113, 2020.
- [50] H.-J. Ye, H. Hu, and D.-C. Zhan, “Learning adaptive classifiers synthesis for generalized few-shot learning,” *International Journal of Computer Vision*, vol. 129, pp. 1930–1953, 2021.
- [51] C. Zhang, Y. Cai, G. Lin, and C. Shen, “Deepemd: Few-shot image classification with differentiable earth mover’s distance and structured classifiers,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 12203–12213.
- [52] D.-W. Zhou, F.-Y. Wang, H.-J. Ye, L. Ma, S. Pu, and D.-C. Zhan, “Forward compatible few-shot class-incremental learning,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9046–9056.
- [53] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.
- [54] A. Bendale and T. E. Boult, “Towards open set deep networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1563–1572.
- [55] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [56] Y. Liu, J. Wang, S. Niu, and H. Song, “Ads-b signals records for non-cryptographic identification and incremental learning.” 2021. [Online]. Available: <https://dx.doi.org/10.21227/1bxc-ke87>
- [57] D. Hendrycks and K. Gimpel, “A baseline for detecting misclassified and out-of-distribution examples in neural networks,” *arXiv preprint arXiv:1610.02136*, 2016.



Taotao Li received the Ph.D. degree in control science and engineering from Zhejiang University of Technology, Hangzhou, China, in 2025. Now he is a postdoctoral fellow at the School of Computer Science and Technology, Zhejiang University of Technology. His current research interests include data mining, deep learning, and RF fingerprint recognition.



Career Researchers) in 2020.



Chendong Jin was born in Zhejiang, China. He is currently pursuing a master’s degree in control engineering at the College of Information Engineering, Zhejiang University of Technology, Hangzhou, China. His current research interests include deep learning in signal recognition and generation.



Jie Su received the B.S. degree in computer science and technology from China Jiliang University, Hangzhou, China, in 2017, the M.S. degree (Hons.) in data analytics from the University of Southampton, Southampton, U.K., in 2018, and the Ph.D. degree from Newcastle University, Newcastle upon Tyne, U.K., in 2023. He is currently an Assistant Professor with the Institute of Cyberspace Security and the College of Information Engineering, Zhejiang University of Technology, Hangzhou. His research interests include deep learning, signal processing, and IoT security.



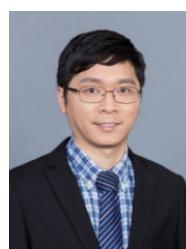
Junhao Li received the B.S. degree from the School of Artificial Intelligence and Big Data, Hefei University, Hefei, China. He is currently pursuing the Master's degree at Guangzhou University, Guangzhou, China. His research interests include AI security and the security of large language models.



Zhen Hong (Member, IEEE) received the B.S. degree from Zhejiang University of Technology, Hangzhou, China, and University of Tasmania, Australia in 2006, respectively, and the Ph.D. degree from the Zhejiang University of Technology Hangzhou, China, in 2012. Now he is a full professor at the Institute of Cyberspace Security, and College of Information Engineering, Zhejiang University of Technology, China. He was a research scholar at CAP Research Group, School of Electrical & Computer Engineering, Georgia Institute of Technology from 2016 to 2018. His research interests include Internet of Things, cyberspace security, and data analytics. He received the first Zhejiang Provincial Young Scientists Title in 2013 and the Zhejiang Provincial New Century 151 Talent Project in 2014. He also received Zhejiang Provincial Science Fund for Distinguished Young Scholars in 2023. He is a member of IEEE and ACM, and a senior member of CCF and CAA, respectively.



Xiaoqin Zhang (Senior Member, IEEE) received the PhD degree in pattern recognition and intelligent system from the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, China, in 2010. He is currently a professor with the Zhejiang University of Technology, China. His research interests include pattern recognition, computer vision, and machine learning. He has published more than 100 papers in international and national journals and international conferences, including IEEE Transactions on Pattern Analysis and Machine Intelligence, International Journal of Computer Vision, IEEE Transactions on Image Processing, ICCV, CVPR, NIPS, IJCAI, AAAI, ACM MM, and others.



Shibo He (Senior Member, IEEE) received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2012. He is a Professor with Zhejiang University. He was an Associate Research Scientist from March 2014 to May 2014, and a Postdoctoral Scholar from May 2012 to February 2014, with Arizona State University, Tempe, AZ, USA. From November 2010 to November 2011, he was a Visiting Scholar with the University of Waterloo, Waterloo, ON, Canada. His research interests include Internet of Things, crowd sensing, big data analysis, etc.