

Article

Calibrated Open-Set Prototypes for Cross-Domain Radio-Frequency Emitter Recognition

Mingdi Li ^{1,2,*} , Yanbin Li ¹, Qi Feng ¹ and Xingjian Li ³ 

¹ The 54th Research Institute of China Electronics Technology Group Corporation (CETC54), Shijiazhuang 050081, China; 13313030889@163.com (Y.L.); fq_learning@outlook.com (Q.F.)

² School of Information and Software Engineering, University of Electronic Science and Technology of China (UESTC), Chengdu 611731, China

³ National Key Laboratory of Advanced Communication Networks, 54th Research Institute of China Electronics Technology Group Corporation (CETC54), Shijiazhuang 050081, China; 18200276858@163.com

* Correspondence: 202310000210@std.uestc.edu.cn

Abstract

Radio-frequency (RF) emitter recognition is a physical-layer tool for wireless security, but recognition accuracy often degrades when the acquisition domain changes. A deployed receiver must also reject unseen emitters and later register novel identities from only a few labeled examples. We address these requirements with a compact prototype framework that uses one shared encoder and two decision-layer modules. Balanced open-set calibration (BOC) maps prototype-score statistics to a calibrated known-class probability and selects a validation-based rejection threshold. Support-aware prototype aggregation (SPA) constructs few-shot novel-class prototypes using centrality weighting, support trimming, and medoid blending. On a curated POWDER cross-day base-station protocol, the full system improves open-set macro-F1 from 55.23% to 69.47%. It also improves 10-shot generalized strict accuracy from 42.33% to 65.54%, relative to a supervised prototype baseline. On a curated Hovering UAV protocol, it improves closed-set accuracy from 33.14% to 59.86%. These gains are obtained without enlarging the deployed neural encoder, supporting RF recognition with calibrated rejection, support-aware registration, and a compact neural footprint.

Keywords: radio-frequency fingerprinting; open-set recognition; few-shot registration; domain adaptation; prototype learning; calibration; wireless security

1. Introduction

Radio-frequency (RF) fingerprinting identifies wireless emitters from hardware-induced signal distortions that supplement protocol-level identifiers [1]. Recent survey work shows a clear shift from handcrafted descriptors to learned representations [2], while broader reviews of RF learning highlight the need to evaluate cross-condition robustness [3]. Related signal-processing advances appear in burst demodulation [4], asynchronous grant-free acquisition [5], and preamble-free blind sequence recovery for overloaded sensor uplinks [6]. RF fingerprinting is relevant to physical-layer authentication, unauthorized-device screening, spoofing detection, and low-altitude platform monitoring [7]. Its main deployment challenge is that the observed fingerprint is jointly shaped by the transmitter, the propagation channel, and the receiver chain. A model trained on one acquisition day, distance, receiver, frequency, or waveform condition can degrade under receiver/channel shifts [8], multi-frequency conditions [9], or crossing signal modes [10].



Academic Editor: Djuradj Budimir

Received: 18 June 2026

Revised: 7 July 2026

Accepted: 11 July 2026

Published: 13 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and

conditions of the [Creative Commons Attribution \(CC BY\) license](https://creativecommons.org/licenses/by/4.0/).

Public over-the-air datasets highlight this deployment gap. Recent benchmarks studying base-station fingerprinting [11], unmanned aerial vehicle (UAV) emissions under motion [12], and receiver-agnostic environments [8] demonstrate the critical need to evaluate cross-condition robustness beyond favorable same-domain closed-set splits.

Several research directions address parts of this problem. Domain adaptation reduces cross-device mismatch [13], and receiver-agnostic learning targets cross-receiver robustness [14]. Collaborative receiver-agnostic learning extends this direction [15], scalable adversarial adaptation provides another route [16], and source-free transfer reduces dependence on labeled target data [17]. Open-set RF identification detects unknown emitters through prototype-extreme-value modeling [18], feature enhancement [19], or lightweight open-set neural designs [20]. Few-shot SEI supports rapid registration from limited labeled samples with metric ensembles [21], attention-based time-frequency models [22], few-shot open-set contrastive learning [23], and cross-domain self-supervised contrastive learning [24]. These directions are usually evaluated separately. A deployed receiver, however, must first identify known emitters, reject unfamiliar emitters, and later register selected novel identities from only a few labeled examples. The decision layer must remain stable before and after class expansion.

This paper develops a compact prototype framework with two lightweight decision-layer modules above a shared encoder. Balanced open-set calibration (BOC) uses prototype-score geometry to stabilize the known/unknown operating point. Support-aware prototype aggregation (SPA) uses centrality weighting, trimming, and medoid blending to reduce the effect of unstable support samples during registration. The emphasis on the decision layer is deliberate. Updating a calibrator and prototype bank in a deployed receiver is more efficient than retraining or enlarging the neural encoder, directly addressing the coupled rejection–registration workflow.

The technical contributions are summarized as follows:

- **Balanced open-set calibration (BOC).** We propose a lightweight calibration module for prototype RF classifiers. BOC converts four prototype-score statistics—top score, score margin, top-score average, and score entropy—into a calibrated known-class probability and selects the known/unknown threshold on validation data. It improves unknown-emitter rejection using a compact decision-layer calibrator.
- **Support-aware prototype aggregation (SPA).** We propose a robust few-shot prototype estimator for registering novel emitters. SPA measures support consistency by centrality, trims atypical support samples, and blends the weighted aggregate with a medoid-like exemplar, reducing the sensitivity of novel-class prototypes to noisy or domain-shifted support examples.
- **A unified decision-layer framework for rejection and registration.** BOC and SPA are integrated above a shared encoder, so the deployed neural backbone remains unchanged. The same prototype bank supports pre-registration open-set rejection and post-registration recognition after adding few-shot novel-class prototypes.

2. Related Work

2.1. RF Fingerprinting Under Deployment Shift

RF fingerprinting relies on hardware-specific distortions that survive protocol-level identifiers. In practice, those distortions are observed through receiver chains, waveform settings, and propagation conditions that can change across deployments. Early physical-layer identification studies established the feasibility of exploiting such imperfections [1]. Robust RF fingerprinting now requires more than high same-domain classification accuracy, because learned representations may still encode acquisition-specific cues [2]. Cross-condition evaluation is central to recent RF-learning benchmarks and reviews [3].

Frequency-hopping SEI studies also show the value of combining feature extraction with deep residual representations when the signal format is nonstationary [25]. Public benchmarks make the deployment gap concrete. POWDER studies over-the-air base-station recognition on a realistic platform [11]. Hovering focuses on same-model UAV transmitters, where motion and channel variation can obscure weak device-specific cues [12]. WiSig highlights receiver and channel agnosticism at a larger scale [8]. Related multi-channel studies show that frequency changes can alter the apparent fingerprint [9], and mixed-mode studies report similar effects across signal modes [10]. Many closed-set studies evaluate the identity classifier after all classes are assumed to be known. The present work evaluates a deployment sequence in which a receiver must reject unregistered emitters before later adding them to the prototype bank.

2.2. Open-Set RF Recognition

Open-set recognition considers the realistic case in which some test identities are absent from training. The open-space-risk formulation [26], probabilistic open-set models [27], and OpenMax [28] provide general foundations. Maximum-softmax is a common out-of-distribution reference score [29]. Energy-based scoring is another widely used baseline [30], and Mahalanobis distance provides a feature-space alternative [31]. These scores are useful references, but a single score can be brittle for prototype RF classifiers under domain shift. For example, the largest prototype score may remain high for an unknown emitter. Covariance-based Mahalanobis estimates can also be unstable when the usable base-identity pool is small. RF-specific studies have introduced prototype extreme-value modeling [18], feature-enhancement strategies [19], and lightweight open-set RF models for edge deployment [20]. Their main emphasis is usually the rejection decision itself. BOC differs in that it calibrates multiple score-space cues from the frozen prototype bank. It avoids replacing the encoder or fitting a separate neural rejector, which is useful when only limited validation data are available.

2.3. Few-Shot Registration and Prototype Learning

Metric learning is well suited to post-deployment class expansion. New classes can be added by storing support embeddings, avoiding full classifier retraining. Matching Networks [32], Prototypical Networks [33], and nearest-class-mean classifiers [34] are representative examples. FaceNet further illustrates the broader value of embedding spaces for class separation [35]. In specific emitter identification, deep metric ensemble learning [21], attention-based few-shot models [22], contrastive masked learning [36], cross-domain contrastive self-supervised learning [24], and few-shot open-set contrastive learning [23] improve low-shot RF recognition. A practical limitation remains when the support set is small and contaminated by channel or receiver mismatch. Uniform averaging assumes that each support example is equally representative. Under deployment shift, atypical supports can pull the novel prototype away from the central region of the class. SPA addresses this decision-layer weakness by weighting central supports, trimming atypical ones, and blending the aggregate with a medoid-like exemplar. It complements encoder-side few-shot methods as a lightweight registration module.

2.4. Cross-Domain and Receiver-Agnostic Recognition

Domain-adversarial learning seeks representations that preserve label information while reducing domain-specific cues [37]. RF studies have adapted this idea for cross-device fingerprinting [13] and cross-receiver fingerprinting [14]. Receiver-agnostic methods further target collaborative transfer [15], scalable adversarial adaptation [16], and source-free transfer [17]. These approaches are important for strengthening closed-set transfer, yet a deployed receiver also needs a calibrated unknown-emitter operating point and

a lightweight class-update mechanism. Encoder adaptation can require target-domain data or retraining resources that may be unavailable after deployment. Decision-layer calibration and prototype updating are lighter-weight operations that preserve the deployed neural footprint. Distributed IoT learning surveys provide complementary system-level perspectives on scalable wireless intelligence [38]. Recent physical-layer security designs based on directional energy control offer a parallel view of secure wireless operation [39]. The present study focuses on RF identity decisions in that broader security context.

3. Problem Formulation

Let $\mathcal{X}$ denote the signal space and let $\mathcal{Y}$ denote emitter labels. A labeled sample is represented as

$$(x, y, r), \quad x \in \mathcal{X}, \quad y \in \mathcal{Y}, \quad r \in \mathcal{D}, \quad (1)$$

where x is a complex-baseband segment with in-phase and quadrature components, y is the emitter identity, and r is a domain indicator. In the evaluated protocols, r denotes an acquisition condition such as day, distance, receiver condition, or another split-defining collection state. In this study, r is an acquisition-condition tag used to construct source/validation/test partitions and to define the domain-adversarial loss in Equation (20).

During training, only a set of base classes is observed,

$$\mathcal{Y}_b = \{1, 2, \dots, C_b\}. \quad (2)$$

At test time, signals may come from either known emitters $\mathcal{Y}_b$ or previously unseen emitters $\mathcal{Y}_u$,

$$\mathcal{Y}_b \cap \mathcal{Y}_u = \emptyset. \quad (3)$$

Domain shift occurs when the conditional signal distribution changes between source and target conditions, for example

$$P_{\text{tr}}(x \mid y, r = r_s) \neq P_{\text{te}}(x \mid y, r = r_t), \quad r_s \neq r_t. \quad (4)$$

Thus, the same emitter can occupy different regions of the embedding space under different acquisition conditions.

We consider three coupled tasks.

Closed-set identification.

For $y \in \mathcal{Y}_b$, the model predicts one of the known classes:

$$f_{\text{closed}} : \mathcal{X} \rightarrow \mathcal{Y}_b. \quad (5)$$

Open-set rejection.

Before registration, queries may come from either $\mathcal{Y}_b$ or $\mathcal{Y}_u$. The model must either assign a known class or reject the signal:

$$f_{\text{open}} : \mathcal{X} \rightarrow \mathcal{Y}_b \cup \{\text{unknown}\}. \quad (6)$$

Few-shot registration.

For a novel class $c \in \mathcal{Y}_u$, a small support set is provided,

$$\mathcal{S}_c = \{(x_i, c)\}_{i=1}^k, \quad k \ll |\mathcal{D}_{\text{tr}}|. \quad (7)$$

The system then registers c and performs generalized recognition over the augmented class set,

$$f_{\text{reg}} : \mathcal{X} \rightarrow \mathcal{Y}_b \cup \mathcal{Y}_{\text{reg}}, \quad \mathcal{Y}_{\text{reg}} \subseteq \mathcal{Y}_u. \quad (8)$$

The workflow is sequential: closed-set training builds a base prototype bank, BOC rejects unregistered emitters before a support set is available, and SPA constructs cached prototypes once selected novel emitters are registered.

4. Materials and Methods

4.1. Overall Architecture

Figure 1 summarizes the proposed framework. Stage I trains a compact one-dimensional encoder. The projector and auxiliary classifier heads are used only during training. Stage II freezes the encoder and performs prototype-based inference. BOC is used before registration, when the system must reject emitters that have not yet been registered. SPA is used at registration time, when a small labeled support set for a novel emitter is accepted. The resulting novel-class prototype is cached and is recomputed only if the support set is updated or the encoder is replaced. The same frozen encoder supports pre-registration open-set rejection and post-registration recognition over an augmented prototype bank.

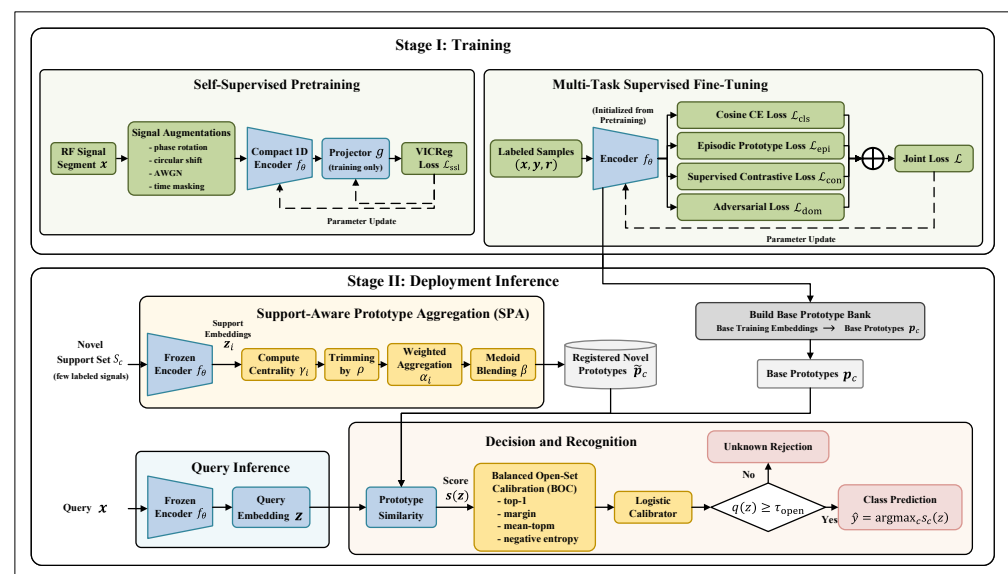


Figure 1. Framework overview. Green blocks denote training losses or training data flow, blue blocks denote encoder/query embedding operations, yellow blocks denote decision-layer calibration or support aggregation, gray blocks denote stored prototypes, and red blocks denote final decisions. Only the compact encoder, prototype bank, BOC calibrator, and cached SPA prototypes are used during deployment inference.

Given a signal segment $x \in \mathbb{R}^{2 \times L}$, the encoder $f_{\theta}(\cdot)$ maps the in-phase and quadrature channels to an embedding

$$z = f_{\theta}(x), \quad z \in \mathbb{R}^{d_e}. \quad (9)$$

The encoder consists of four one-dimensional convolutional blocks with progressively increasing channel widths (32, 64, 96, and 128), decreasing kernel sizes (9, 7, 5, and 3), and respective strides of 4, 4, 4, and 2. These blocks are followed by adaptive average pooling and a linear projection into a 128-dimensional embedding space. The backbone is intentionally compact, which helps isolate the effect of the decision-layer modules from the effect of model scale.

For each base class $c \in \mathcal{Y}_b$, let $\mathcal{Z}_c$ denote its training embeddings. A class prototype is

$$\mathbf{p}_c = \frac{1}{|\mathcal{Z}_c|} \sum_{\mathbf{z}_i \in \mathcal{Z}_c} \mathbf{z}_i, \quad (10)$$

or a small set of prototypes $\{\mathbf{p}_c^{(m)}\}_{m=1}^{M_c}$ when multiple modes are retained. For a query embedding $\mathbf{z}$, cosine similarity to class c is

$$s_c(\mathbf{z}) = \max_m \frac{\mathbf{z}^\top \mathbf{p}_c^{(m)}}{\|\mathbf{z}\|_2 \|\mathbf{p}_c^{(m)}\|_2}. \quad (11)$$

Closed-set prediction over an active prototype bank $\mathcal{P}$ is

$$\hat{y}_{\text{closed}} = \arg \max_{c \in \mathcal{P}} s_c(\mathbf{z}). \quad (12)$$

4.2. Training Objective

The encoder is trained in two stages. The first stage applies self-supervised pretraining using two stochastic views $t_1(x)$ and $t_2(x)$. The RF augmentations are random phase rotation, circular shift, additive white Gaussian noise, and time masking. The self-supervised loss uses Variance-Invariance-Covariance Regularization (VICReg) [40]. Specifically, the invariance term enforces similarity between two augmented views of the same signal, the variance term prevents embedding dimensional collapse, and the covariance term penalizes redundant projected dimensions. These properties are useful before supervised fine-tuning because labeled RF data are limited and domain-shifted.

For a minibatch $\{x_i\}_{i=1}^B$, let

$$\mathbf{u}_i^{(1)} = g(f_\theta(t_1(x_i))), \quad \mathbf{u}_i^{(2)} = g(f_\theta(t_2(x_i))), \quad (13)$$

and let $\mathbf{U}^{(1)}, \mathbf{U}^{(2)} \in \mathbb{R}^{B \times d_p}$ collect the projected views row-wise. The self-supervised objective is

$$\begin{aligned} \mathcal{L}_{\text{ssl}} = & \lambda_{\text{sim}} \frac{1}{B} \sum_{i=1}^B \|\mathbf{u}_i^{(1)} - \mathbf{u}_i^{(2)}\|_2^2 \\ & + \lambda_{\text{var}} (\mathcal{V}(\mathbf{U}^{(1)}) + \mathcal{V}(\mathbf{U}^{(2)})) \\ & + \lambda_{\text{cov}} (\mathcal{C}(\mathbf{U}^{(1)}) + \mathcal{C}(\mathbf{U}^{(2)})), \end{aligned} \quad (14)$$

where λ_{sim} , λ_{var} , and λ_{cov} are dimensionless weights. For clarity, the VICReg regularizers are

$$\mathcal{V}(\mathbf{U}) = \frac{1}{d_p} \sum_{\ell=1}^{d_p} \max \left(0, 1 - \sqrt{\text{Var}(\mathbf{U}_{:\ell})} + \epsilon \right), \quad (15)$$

$$\mathcal{C}(\mathbf{U}) = \frac{1}{d_p} \sum_{\ell \neq m} \text{Cov}(\mathbf{U})_{\ell m}^2, \quad (16)$$

with a small numerical constant ϵ .

The second stage performs supervised optimization with

$$\mathcal{L} = \mathcal{L}_{\text{cls}} + \lambda_{\text{epi}} \mathcal{L}_{\text{epi}} + \lambda_{\text{con}} \mathcal{L}_{\text{con}} + \lambda_{\text{dom}} \mathcal{L}_{\text{dom}}, \quad (17)$$

where $\mathcal{L}_{\text{cls}}$ is cosine-classification cross-entropy, $\mathcal{L}_{\text{epi}}$ is a batch-internal episodic prototype loss, $\mathcal{L}_{\text{con}}$ is supervised contrastive learning [41], and $\mathcal{L}_{\text{dom}}$ is domain-adversarial cross-entropy [37]. The cross-entropy term keeps base-class discrimination, the episodic term exposes the encoder to support/query prototype geometry, the contrastive term improves

local class compactness, and the domain-adversarial term uses r to reduce domain-specific information in z . These weights are selected via validation to balance task performance, as individual terms do not monotonically improve all metrics.

Concretely,

$$\mathcal{L}_{\text{cls}} = -\frac{1}{B} \sum_{i=1}^B \log \frac{\exp(\cos(z_i, w_{y_i}) / \tau_{\text{cls}})}{\sum_{j=1}^{C_b} \exp(\cos(z_i, w_j) / \tau_{\text{cls}})}, \quad (18)$$

$$\mathcal{L}_{\text{con}} = -\frac{1}{|I|} \sum_{i \in I} \frac{1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(z_i^\top z_p / \tau_{\text{con}})}{\sum_{a \in I \setminus \{i\}} \exp(z_i^\top z_a / \tau_{\text{con}})}, \quad (19)$$

and

$$\mathcal{L}_{\text{dom}} = -\frac{1}{B} \sum_{i=1}^B \log p_\psi(r_i | \text{GRL}(z_i)), \quad (20)$$

where B is the batch size, $I = \{1, \dots, B\}$ indexes the batch, $P(i) = \{p \in I \setminus \{i\} : y_p = y_i\}$ denotes the positives of anchor i , $\{w_j\}_{j=1}^{C_b}$ are the trainable cosine-classifier weight vectors used only during supervised training, τ_{cls} and τ_{con} are dimensionless temperature parameters, $p_\psi(\cdot)$ is the domain classifier, and GRL denotes the gradient-reversal layer. The episodic loss randomly splits each minibatch class-wise into support and query subsets, forms prototypes from the support subset, and applies prototype-space cross-entropy on the query subset. At inference, class prototypes are computed from encoder embeddings, while the auxiliary classifier weights are discarded. Table 1 summarizes the main symbols and fixed hyperparameters used by the method.

Table 1. Key symbols, hyperparameters, and units. The score temperatures are dimensionless because they scale cosine similarities.

Symbol	Meaning	Unit	Role
r	Domain indicator such as day, distance, or receiver condition	categorical	Used for split construction and $\mathcal{L}_{\text{dom}}$
λ_{sim}	VICReg invariance weight	dimensionless	Controls view consistency
λ_{var}	VICReg variance weight	dimensionless	Prevents dimensional collapse
λ_{cov}	VICReg covariance weight	dimensionless	Penalizes redundant dimensions
λ_{epi}	Episodic prototype-loss weight	dimensionless	Encourages support/query geometry
λ_{con}	Supervised contrastive-loss weight	dimensionless	Encourages class compactness
λ_{dom}	Domain-adversarial loss weight	dimensionless	Reduces domain-specific cues
τ_{cls}	Cosine-classifier temperature	dimensionless	Scales logits in Equation (18)
τ_{con}	Contrastive temperature	dimensionless	Scales pair similarities in Equation (19)
T	BOC score temperature	dimensionless	Scales prototype scores before entropy
ρ	SPA trim fraction	dimensionless	Removes atypical supports
β	SPA medoid blending weight	dimensionless	Balances aggregate and medoid
τ_s	SPA centrality temperature	dimensionless	Controls support weights

4.3. Balanced Open-Set Calibration

A closed-set prototype classifier can yield overconfident predictions for unknown emitters. BOC calibrates the binary known/unknown decision with summary statistics of the base-class prototype scores. For a query embedding z , let $s(z) = [s_1(z), \dots, s_{C_b}(z)]$ denote the base-class scores, and let $m = \min(3, C_b)$. We construct

$$\phi(z) = [\text{top-1}(s), \text{margin}(s), \text{mean-top}_m(s), -H(\text{softmax}(s/T))]. \quad (21)$$

Here, top-1 is the maximum score, margin is the gap between the largest and second-largest scores, mean-top _{m} averages the m largest scores, $H(\cdot)$ is entropy, and T is a dimensionless

score temperature used only in this transformation. The four statistics were selected because they capture complementary score-space evidence: absolute prototype affinity, nearest-competitor ambiguity, local neighborhood support, and distributional concentration. We use a class-balanced logistic regressor because the validation sets are small, the parameters are interpretable, and the deployed state is limited to one linear calibrator.

The calibrator maps the feature vector to a known-class probability:

$$q(\mathbf{z}) = \sigma(\mathbf{a}^\top \phi(\mathbf{z}) + b), \quad (22)$$

where $\mathbf{a}$ and b are calibrator parameters. The rejection threshold is selected on the validation split by maximizing binary known/unknown macro-F1:

$$\tau_{\text{open}} = \arg \max_{\tau} \text{macro-F1}(\hat{y}_{\tau}^{\text{K/U}}(\mathbf{z}), y^{\text{K/U}}), \quad (23)$$

where $\hat{y}_{\tau}^{\text{K/U}}(\mathbf{z}) = \text{known}$ if $q(\mathbf{z}) \geq \tau$ and unknown otherwise. The calibrator is fitted only on the validation split and does not use test labels.

4.4. Support-Aware Prototype Aggregation

In few-shot registration, simple support averaging can be unstable. The support set may be small, noisy, or shifted. For a novel class c with support embeddings $\mathcal{E}_c = \{\mathbf{z}_1, \dots, \mathbf{z}_k\}$, we first normalize the supports,

$$\bar{\mathbf{z}}_i = \frac{\mathbf{z}_i}{\|\mathbf{z}_i\|_2}, \quad i = 1, \dots, k, \quad (24)$$

and define support centrality as

$$\gamma_i = \frac{1}{k} \sum_{j=1}^k \bar{\mathbf{z}}_i^\top \bar{\mathbf{z}}_j. \quad (25)$$

Only the most central fraction of the supports is retained:

$$\mathcal{K}_c = \text{TopK}(\{\gamma_i\}_{i=1}^k, \lceil (1 - \rho)k \rceil), \quad (26)$$

where ρ is the trim fraction and $\text{TopK}(\cdot)$ returns the index set of the largest centrality scores. Retained supports are weighted by

$$\alpha_i = \frac{\exp(\gamma_i / \tau_s)}{\sum_{j \in \mathcal{K}_c} \exp(\gamma_j / \tau_s)}, \quad i \in \mathcal{K}_c, \quad (27)$$

and the weighted prototype is blended with the most central support, which acts as a medoid-like exemplar [42]:

$$\tilde{\mathbf{p}}_c = \frac{(1 - \beta) \sum_{i \in \mathcal{K}_c} \alpha_i \bar{\mathbf{z}}_i + \beta \bar{\mathbf{z}}_{m_c}}{\|(1 - \beta) \sum_{i \in \mathcal{K}_c} \alpha_i \bar{\mathbf{z}}_i + \beta \bar{\mathbf{z}}_{m_c}\|_2}, \quad (28)$$

where $m_c = \arg \max_{i \in \{1, \dots, k\}} \gamma_i$. The hyperparameters ρ , β , and τ_s were selected once on validation episodes and then fixed across the two protocols. Smaller ρ makes SPA close to weighted averaging, larger ρ discards more supports, larger β moves the prototype toward the medoid exemplar, and smaller τ_s concentrates the support weights on the most central samples. The constructed prototype is cached at registration time and is rebuilt only when the support set changes. Algorithm 1 summarizes how the same prototype bank is used in closed-set, open-set, and registered inference modes.

Algorithm 1 Task-specific prototype inference

Require: Query x , base prototypes $\{p_c\}$, cached registered prototypes $\{\tilde{p}_c\}$, mode $M \in \{\text{closed, open, registered}\}$

- 1: Compute embedding $z = f_\theta(x)$ using the frozen encoder
- 2: Initialize active prototype bank with the base prototypes
- 3: **if** $M = \text{registered}$ **then**
- 4: Add cached SPA prototypes $\{\tilde{p}_c\}$ to the active prototype bank
- 5: **end if**
- 6: Compute prototype scores over the active bank using Equation (11)
- 7: **if** $M = \text{open}$ **then**
- 8: Compute $q(z)$ from base-class scores using Equation (22)
- 9: **if** $q(z) < \tau_{\text{open}}$ **then**
- 10: Output unknown
- 11: **else**
- 12: Output the highest-scoring base class
- 13: **end if**
- 14: **else**
- 15: Output the highest-scoring class in the active bank
- 16: **end if**

5. Experimental Setup

5.1. Datasets and Protocols

We use POWDER and Hovering UAV as the two main public over-the-air benchmarks because they expose realistic acquisition variation and allow closed-set, open-set, and few-shot protocols to be constructed. POWDER represents fixed infrastructure transmitters under cross-day and multi-technology collection, whereas Hovering UAV represents the same-model aerial platforms under distance and motion-related channel variation. WiSig is used as a separate large-identity scalability check. The preprocessing pipeline removes segments that fail to meet split completeness, minimum count, or protocol-specific filtering constraints. The final evaluated splits are smaller than the candidate archives.

POWDER cross-day.

POWDER is a public over-the-air benchmark for RF fingerprinting of open radio access network base stations [11]. The curated archive used here contains 8300 candidate signal segments from four emitters under cross-day 4G, 5G, and WiFi collection conditions. After protocol filtering, one emitter (meb) is treated as novel, and the other three are base classes. The final split sizes for the base classes are 1899/633/3025 (train/validation/test), while the novel-class splits contain 633/211/1055 samples, respectively. The final protocol contains 5557 base-class samples and 1899 novel-class samples. After curation, this protocol contains only four usable emitters, so it is used as a controlled benchmark for score calibration and cross-day transfer. The larger WiSig ManyTx evaluation in Section 6.4 is used only to probe identity-scale behavior.

Hovering UAV.

Hovering UAV is a public benchmark for identifying same-model UAV emitters from non-standard transmitter waveforms [12]. The curated archive used here contains 12,362 candidate segments from seven emitters collected at multiple distances. After protocol filtering, two emitters, uav_6 and uav_7, are treated as novel classes. In the Hovering UAV protocol, the final split sizes are 4655/575/600 for base classes and 1862/230/240 for novel classes. These splits correspond to 5830 base-class samples and 2332 novel-class samples.

WiSig ManyTx scalability subset.

WiSig is a public RF fingerprinting dataset designed to evaluate receiver- and channel-agnostic transmitter recognition at a larger scale [8]. We use the compact ManyTx subset only for the large-identity scalability experiment. The protocol selects 80 transmitters satisfying the minimum-count requirement and assigns them to 50 base-known, 15 unknown, and 15 novel transmitter identities. Each transmitter contributes at most 1500 segments, and the per-transmitter sample split follows a 70/15/15 train/validation/test ratio. The WiSig signal length is 256, whereas the POWDER and Hovering UAV experiments use a length of 1024. This setting is reported separately because it changes both the dataset scale and the input length.

The base/novel split is determined by two criteria: every selected base class must have enough samples in the source, validation, and held-out target partitions, and every novel class must have enough validation/test segments for unknown rejection and few-shot support/query sampling. No test-domain labels from novel classes are used to train the encoder. For the main POWDER and Hovering UAV benchmarks, open-set threshold selection uses only the validation split, and validation unknowns do not overlap the test segments. The WiSig validation and test segments are also disjoint. This experiment is reported as a scalability check and is not used as the primary cross-domain claim. Larger identity-disjoint protocols remain necessary for a complete scalability claim.

Table 2 summarizes the two primary comparison protocols: the stricter unseen-distance Hovering UAV stress test and the WiSig scalability check. The overlap column refers only to acquisition conditions, never to sample reuse. Thus, the Hovering UAV main protocol uses disjoint train, validation, and test segments even though all three splits include 6 ft, 9 ft, and 12 ft conditions. The unseen-12ft protocol removes 12 ft from training and validation to test a stronger distance shift, while the main Hovering UAV result remains the primary comparison.

Table 2. Dataset split and evaluation protocol summary. The overlap column refers to acquisition conditions only; all sample sets are disjoint.

Protocol	Dataset	Purpose	Training Domains	Validation Domains	Test Domains	Acq. Condition Overlap	Recommended Use
POWDER main protocol	POWDER	Main open-set evaluation	Day 1 4G; Day 1 5G; Day 1 WiFi	Day 1 4G; Day 1 5G; Day 1 WiFi	Day 2 4G; Day 2 5G; Day 2 WiFi	No	Main cross-day open-set evaluation
Hovering UAV main protocol	Hovering UAV	Main-protocol evaluation aligned with the original manuscript	6 ft; 9 ft; 12 ft	6 ft; 9 ft; 12 ft	6 ft; 9 ft; 12 ft	Yes samples disjoint	Main result comparison
Hovering UAV unseen-12ft protocol	Hovering UAV	Strict unseen-distance stress test	6 ft; 9 ft	6 ft; 9 ft	12 ft	No	Stress-test analysis only
WiSig ManyTx scalability	WiSig	Large-identity scalability check	Mixed receiver/channel segments	Mixed receiver/channel segments	Mixed receiver/channel segments	Yes samples disjoint	Scalability check only

5.2. Implementation Details

All experiments were conducted on a workstation equipped with an NVIDIA RTX 4080 GPU (NVIDIA Corporation, Santa Clara, CA, USA) running Ubuntu 22.04.2 LTS. The implementation used Python 3.12, PyTorch 2.1.1+cu118, and CUDA 11.8.

The final experiments use a shared training budget with early stopping and best-checkpoint restoration. Both the self-supervised and supervised phases are trained for a maximum of 200 epochs, with a minimum of 40 epochs and an early-stopping patience of 25 epochs. The supervised prototype baseline uses only the supervised phase. Enhanced-backbone variants and the proposed framework also use the self-supervised phase. The batch size is 128 for both phases. The supervised and self-supervised learning rates are 8×10^{-4} and 10^{-3} , respectively. Weight decay is 10^{-4} , the embedding dimension is 128, and the signal length is 1024.

For VICReg pretraining, we use the standard coefficients $\lambda_{\text{sim}} = 25$, $\lambda_{\text{var}} = 25$, and $\lambda_{\text{cov}} = 1$ [40]. The supervised loss weights are $\lambda_{\text{epi}} = 0.3$, $\lambda_{\text{con}} = 0.12$, and $\lambda_{\text{dom}} = 0.2$. These weights were selected once using validation performance and then fixed across protocols. The default calibration settings are $T = 0.15$ and logistic-regression inverse regularization strength $C = 2.5$. The default support-aggregation settings are $\tau_s = 0.12$, $\beta = 0.4$, and $\rho = 0.15$. The logistic calibrator uses class-balanced logistic regression with a maximum of 300 iterations. All other fixed hyperparameters are unchanged across protocols.

Main results are reported as mean $\pm$ standard deviation over seven full-pipeline runs with seeds 42–48. One global seed is fixed before both the self-supervised and supervised stages. Deterministic backends are enabled when available. Train/validation/test membership is fixed across seeds. For comparability with the originally archived runs, the main generalized few-shot table keeps the same five support/query episodes per shot. To address statistical reliability, Section 6.4 additionally reports 200-episode few-shot estimates and confidence-interval half-widths for the selected protocols. Specifically, based on the base-class counts, the evaluation parameter m in Equation (21) yields $m = 3$ for both evaluated protocols.

5.3. Evaluation Metrics and Efficiency Measures

Closed-set identification is evaluated by accuracy over base-class queries. Open-set rejection is evaluated by binary known/unknown macro-F1 and unknown-F1. Macro-F1 is used because known and unknown test counts can be imbalanced, and unknown-F1 is reported separately because false acceptance of an unregistered emitter is the main security-relevant error. For unknown-F1, the unknown class is treated as the positive class. In few-shot registration, registered novel classes are treated as known classes, and the open-set gate is disabled; the metric reflects post-registration recognition.

For generalized few-shot evaluation, registered novel-class prototypes are concatenated with the base-class prototypes. Let $\hat{\text{Pr}}[\cdot]$ denote the empirical probability over evaluation queries. We define

$$\begin{aligned} A_{\text{base,known}} &= \hat{\text{Pr}}[\hat{y} \in \mathcal{Y}_b \mid y \in \mathcal{Y}_b], \\ A_{\text{base,strict}} &= \hat{\text{Pr}}[\hat{y} = y \mid y \in \mathcal{Y}_b], \end{aligned} \quad (29)$$

and

$$A_{\text{novel,gen}} = \hat{\text{Pr}}[\hat{y} = y \mid y \in \mathcal{Y}_u], \quad (30)$$

where a base-class prediction on a novel query is counted as an error. $A_{\text{base,known}}$ measures whether base queries remain on the base side of the augmented classifier after novel

prototypes are added, regardless of which base identity is selected. $A_{\text{base,strict}}$ additionally requires the exact base identity and penalizes base-to-base identity errors. We then report

$$\begin{aligned} A_{\text{gen}} &= \frac{A_{\text{base,known}} + A_{\text{novel,gen}}}{2}, \\ A_{\text{gen-strict}} &= \frac{A_{\text{base,strict}} + A_{\text{novel,gen}}}{2}. \end{aligned} \quad (31)$$

The main tables report closed-set accuracy, open macro-F1, unknown-F1, and 10-shot strict few-shot accuracy because these four metrics cover known-class recognition, unknown rejection, and post-registration identity recognition.

For a metric value a_j from seed j and $n = 7$ seeds, the reported mean and standard deviation are

$$\bar{a} = \frac{1}{n} \sum_{j=1}^n a_j, \quad s_a = \sqrt{\frac{1}{n-1} \sum_{j=1}^n (a_j - \bar{a})^2}. \quad (32)$$

We also report deployed neural parameter count, multiply–accumulate operations (MACs), floating-point operations (FLOPs), measured forward latency, and peak memory. MACs and FLOPs are computed from a PyTorch forward-pass profiler for the deployed encoder and checked against the input length; latency is measured with CUDA event timing after warm-up, and peak memory is read from the CUDA memory counter. Because all shared-backbone baselines use the same deployed encoder, the footprint is reported once for the proposed encoder.

5.4. Baselines

We evaluate two groups of variants: internal variants and shared-backbone adaptations of recent RF methods. For internal comparisons, a unified code base varies only the components under study. The internal variants include the supervised prototype baseline, the enhanced backbone without self-supervised pretraining, the enhanced backbone, and versions that add BOC and SPA separately before combining them in the proposed framework. These variants clarify which gains come from representation strengthening, calibration, and support-aware prototype construction.

For external comparison, we retain five concrete shared-backbone adaptations. The DANN-style RFDA baseline follows domain-adaptation RF studies [13,14]. The receiver-agnostic adversarial baseline follows collaborative receiver-agnostic learning [15] and scalable GAN-RXA adaptation [16]. The other rows are adapted from Proto-EVT open-set recognition [18], open-set feature enhancement [19], and DMEL few-shot metric learning [21]. Each row is implemented in the same code base with the same encoder and training budget. The comparison emphasizes decision and training strategies while minimizing differences in model capacity. The supplementary code also exports standard OOD scores, including maximum similarity, energy-style score concentration, and Mahalanobis distance, under the same frozen encoder. BOC uses maximum-score, margin, neighborhood, and entropy cues in the same score space. Full-covariance Mahalanobis estimates are treated as reference scores because the curated protocols have very small base-class pools.

6. Results and Discussion

6.1. BOC Temperature Sensitivity

Figure 2 shows a sensitivity sweep of the BOC temperature around the fixed settings used in the main tables. The temperature T is dimensionless because it scales cosine-similarity scores before the entropy calculation. The displayed range $T \in [0.10, 0.30]$ was chosen from validation runs in which the entropy feature changed without becoming

saturated. The curves show that T affects known/unknown macro-F1, while logistic regularization is less influential in our runs. The default $T = 0.15$ is used as a shared operating point; protocol-specific validation tuning can be used when larger validation sets are available.

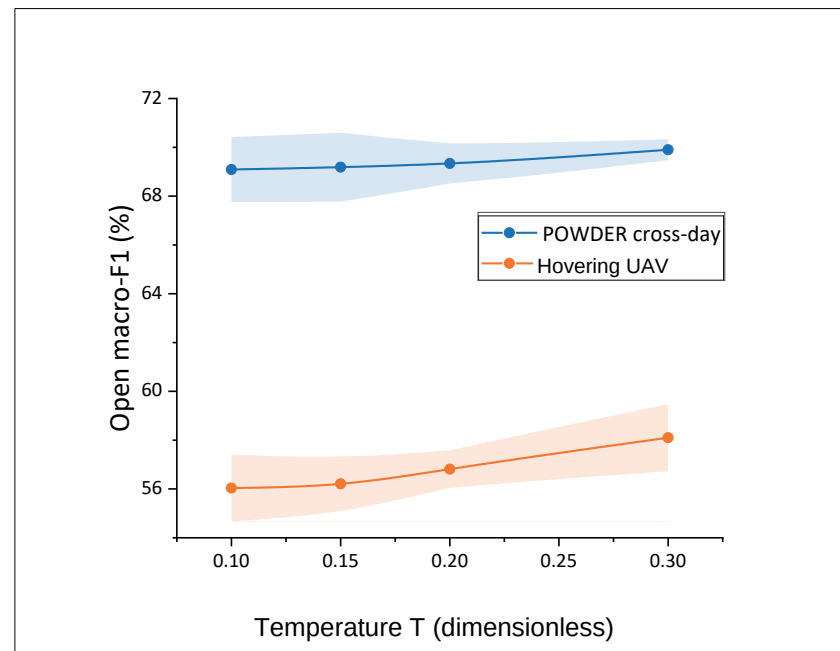


Figure 2. Sensitivity of known/unknown macro-F1 to the BOC temperature T . The temperature is a dimensionless score-scaling parameter. The shaded bands denote the mean $\pm$ one standard deviation over seven runs.

6.2. Internal Baselines and Ablation Study

Table 3 presents the main internal comparison. Closed-set improvement comes mainly from representation strengthening, especially self-supervised pretraining. On POWDER, the enhanced backbone raises closed-set accuracy from 56.43% to 66.52%, and BOC further improves open-set macro-F1 from 65.42% to 69.47% by calibrating the known/unknown boundary. SPA is applied after registration, so it changes closed-set and pre-registration open-set metrics very little; it increases 10-shot generalized strict accuracy from 56.33% to 65.54%.

Table 3. Internal comparison on the two protocols (mean $\pm$ standard deviation, %). Bold values denote the highest mean within each protocol and metric.

Protocol	Variant	Closed Acc.	Open Macro-F1	Unknown F1	Few-Shot Strict@10
POWDER cross-day	Supervised prototype baseline	56.43 $\pm$ 2.96	55.23 $\pm$ 1.81	43.20 $\pm$ 2.10	42.33 $\pm$ 2.50
	Enhanced backbone w/o SSL	60.33 $\pm$ 2.35	56.83 $\pm$ 3.37	45.48 $\pm$ 2.70	47.45 $\pm$ 2.41
	Enhanced backbone	66.52 $\pm$ 3.07	65.42 $\pm$ 1.37	55.61 $\pm$ 2.77	56.33 $\pm$ 1.87
	Enhanced backbone + BOC	66.52 $\pm$ 3.07	69.47 $\pm$ 2.45	58.99 $\pm$ 3.40	56.33 $\pm$ 1.87
	Enhanced backbone + SPA	66.52 $\pm$ 3.07	65.42 $\pm$ 1.37	55.61 $\pm$ 2.77	65.54 $\pm$ 2.86
	Proposed framework	66.52 $\pm$ 3.07	69.47 $\pm$ 2.45	58.99 $\pm$ 3.40	65.54 $\pm$ 2.86
Hovering UAV	Supervised prototype baseline	33.14 $\pm$ 1.17	51.23 $\pm$ 2.79	32.97 $\pm$ 3.17	18.44 $\pm$ 0.40
	Enhanced backbone w/o SSL	34.93 $\pm$ 1.79	53.10 $\pm$ 2.02	36.80 $\pm$ 2.72	18.50 $\pm$ 0.86
	Enhanced backbone	59.86 $\pm$ 2.23	54.46 $\pm$ 1.70	38.09 $\pm$ 2.00	36.76 $\pm$ 2.10
	Enhanced backbone + BOC	59.86 $\pm$ 2.23	55.95 $\pm$ 1.61	38.67 $\pm$ 1.30	36.76 $\pm$ 2.10
	Enhanced backbone + SPA	59.86 $\pm$ 2.23	54.46 $\pm$ 1.70	38.09 $\pm$ 2.00	39.41 $\pm$ 2.44
	Proposed framework	59.86 $\pm$ 2.23	55.95 $\pm$ 1.61	38.67 $\pm$ 1.30	39.41 $\pm$ 2.44

The gain pattern differs for the Hovering UAV protocol. Self-supervised representation strengthening again dominates the closed-set improvement, raising accuracy from 33.14% to 59.86%. SPA increases 10-shot strict accuracy from 36.76% to 39.41%. The Hovering UAV data come from same-model aerial platforms [12]. POWDER uses fixed base-station captures collected across days [11]. We report the two protocols separately and avoid treating the dataset difference as an isolated causal factor. The full system gives the strongest mean values for recognition, rejection, and registration across the reported internal metrics. The reported standard deviations describe across-seed variability. Because not all baseline variants retain paired per-seed records, we avoid a formal paired significance claim for the main tables. Episode-level confidence intervals are provided in the few-shot reliability table where repeated support/query sampling is available.

6.3. Comparison with Shared-Backbone RF Baselines

Table 4 reports the controlled shared-backbone comparison. On POWDER cross-day, the proposed framework achieves the highest mean on all four reported metrics: 66.52 ± 3.07 closed-set accuracy, 69.47 ± 2.45 open macro-F1, 58.99 ± 3.40 unknown-F1, and 65.54 ± 2.86 few-shot strict accuracy at 10 shots. The largest margins are on rejection quality and generalized few-shot strict accuracy, which match the design focus on rejection calibration and post-registration prototype construction in the same embedding space.

On Hovering UAV, the proposed framework again obtains the highest mean values across the reported metrics, but the absolute values are lower than on POWDER. The two benchmarks differ in transmitter type and acquisition setting: POWDER contains fixed base-station captures collected across days and technologies, while Hovering UAV contains same-model aerial platforms affected by distance- and motion-related variation. These differences provide context for the lower Hovering UAV scores, although the table itself is not intended to isolate the physical cause. The Proto-EVT-inspired row is lower than the proposed framework on both protocols, and BOC gives the best overall known/unknown balance under the shared encoder.

Table 4. Shared-backbone RF baseline comparison (mean $\pm$ standard deviation, %). Bold values denote the highest mean within each protocol and metric.

Protocol	Method	Closed Acc.	Open Macro-F1	Unknown F1	Few-Shot Strict@10
POWDER cross-day	Proposed framework	66.52 ± 3.07	69.47 ± 2.45	58.99 ± 3.40	65.54 ± 2.86
	DANN-style RFDA	55.06 ± 2.74	55.40 ± 5.74	43.16 ± 4.23	39.45 ± 3.21
	Receiver-agnostic adversarial	56.05 ± 3.20	57.11 ± 3.17	46.10 ± 2.35	40.73 ± 3.41
	Proto-EVT-inspired	60.45 ± 3.43	41.25 ± 1.85	45.56 ± 1.00	49.14 ± 3.11
	Open-enhance-inspired	60.86 ± 2.88	57.32 ± 2.89	47.27 ± 2.33	49.87 ± 2.45
	DMEL-inspired	60.90 ± 2.86	56.02 ± 2.73	47.44 ± 1.77	40.88 ± 2.08
Hovering UAV	Proposed framework	59.86 ± 2.23	55.95 ± 1.61	38.67 ± 1.30	39.41 ± 2.44
	DANN-style RFDA	33.19 ± 1.71	51.01 ± 1.99	33.25 ± 3.42	18.59 ± 0.50
	Receiver-agnostic adversarial	34.36 ± 1.91	53.21 ± 2.00	34.78 ± 2.89	18.49 ± 0.50
	Proto-EVT-inspired	33.40 ± 2.20	45.55 ± 2.79	36.05 ± 2.79	17.89 ± 1.10
	Open-enhance-inspired	34.52 ± 2.02	52.50 ± 1.94	36.86 ± 4.01	18.52 ± 0.75
	DMEL-inspired	34.21 ± 1.47	52.44 ± 2.06	37.06 ± 2.73	17.24 ± 0.82

6.4. Extended Evaluation

This subsection reports the extended evaluation in the same order as the tables: cross-protocol summary, WiSig ManyTx scalability, OOD score comparison, loss-term ablation, SPA component ablation, and few-shot episode reliability. These results are presented in

separate tables due to differing row units and metrics; merging them would obscure the distinct evaluations of open-set detection, few-shot aggregation, and episode-count reliability.

Table 5 shows that the main POWDER protocol remains the most favorable setting for calibrated rejection, with 69.47% open-set macro-F1, 58.99% unknown-F1, and 86.73% AUROC. The Hovering UAV main protocol is harder, especially for unknown-emitter detection, where the unknown-F1 is 38.67%. The unseen-12ft setting lowers closed-set accuracy and closed-set macro-F1, indicating that excluding the 12 ft domain during training and validation creates a stronger distance-shift test. This result is discussed separately from the main Hovering UAV protocol.

Table 5. Main-protocol and unseen-domain stress-test results from the extended evaluation runs.

Experiment	Dataset	Closed-Set Accuracy	Closed-Set Macro-F1	Open-Set Macro-F1	Unknown F1	Open-Set Balanced Acc.	AUROC
POWDER main protocol	POWDER	66.52	66.13	69.47	58.99	79.24	86.73
Hovering UAV main protocol	Hovering UAV	59.86	59.19	55.95	38.67	56.96	57.93
Hovering UAV unseen-12ft stress test	Hovering UAV	37.67	37.52	48.45	44.16	55.42	57.18

Tables 6–8 extend the evaluation to the larger WiSig ManyTx subset. Table 6 uses a 50/15/15 base/unknown/novel transmitter split and reports 86.04% closed-set accuracy, 84.16% open-set macro-F1, and 66.85% unknown-F1. This setting is more favorable for closed-set recognition than the curated POWDER and Hovering protocols because the base training pool contains many more identities and samples, but the unknown-F1 still shows that large-identity rejection remains nontrivial.

Table 7 shows that BOC gives the best WiSig open-set macro-F1, balanced accuracy, and unknown-F1 among the direct OOD scores under the same frozen encoder. The margins over maximum-similarity and energy scores are modest, so this result is presented as a large-identity reference. Table 8 compares SPA with mean prototypes at the same 1/5/10-shot support budgets. SPA improves novel accuracy by 0.13, 3.05, and 3.63 percentage points at 1, 5, and 10 shots, respectively. This table focuses on novel-class accuracy because the support-budget curve isolates prototype aggregation for newly registered identities.

Table 9 compares BOC with direct OOD scores in the same frozen encoder/prototype space. On POWDER, BOC improves macro-F1 from 63.70% for Mahalanobis distance to 69.47% and improves AUROC from 78.39% for the energy score to 86.73%. On Hovering UAV, BOC gives the highest macro-F1 and balanced accuracy, while the energy and Mahalanobis scores are slightly higher on unknown-class F1. We use BOC as the default operating point because it gives the best overall balance between known-class retention and unknown rejection in these shared-encoder comparisons.

Table 10 shows that the training terms affect different metrics unevenly. The full objective gives the best 10-shot novel accuracy and 10-shot strict accuracy, whereas the cross-entropy-only and no-SSL settings are substantially lower. Removing the contrastive term slightly increases closed-set accuracy, and removing the domain-adversarial term slightly increases open-set macro-F1. The selected objective is retained because it gives the strongest trade-off across closed-set recognition, rejection, and post-registration accuracy in

this protocol. The strict-accuracy drop without self-supervised pretraining is one reason for retaining the pretraining stage.

Table 6. Large-identity scalability validation on the WiSig ManyTx subset. The Tx split lists base-known/unknown/novel transmitter identities.

Dataset	Tx Split Base/Unk. Novel	Closed Acc.	Open Macro-F1	Unknown F1	1-Shot Novel Acc.	5-Shot Novel Acc.	10-Shot Novel Acc.
WiSig ManyTx	50/15/15	86.04	84.16	66.85	44.50	58.01	61.44

Table 7. WiSig ManyTx unknown-emitter detection baseline comparison under the same frozen encoder/prototype space.

Method	Open-Set Macro-F1	Balanced Accuracy	Unknown F1
Prototype max-sim threshold	84.05	77.34	65.90
Energy score	84.02	77.73	66.44
Mahalanobis distance	83.33	76.93	66.38
BOC	84.16	78.10	66.85

Table 8. WiSig ManyTx novel-class prototype aggregation comparison across support budgets.

Method	Shot	Novel Acc. Mean	Novel Acc. 95% CI
Mean prototype	1	44.37	0.63
SPA	1	44.50	0.61
Mean prototype	5	54.96	0.38
SPA	5	58.01	0.37
Mean prototype	10	57.81	0.36
SPA	10	61.44	0.32

Table 9. Unknown-emitter detection baseline comparison under the same frozen encoder/prototype space.

Dataset	Method	Macro-F1	AUROC	Balanced Accuracy	Known F1	Unknown F1
POWDER	Maximum similarity	57.05	76.56	66.19	63.50	50.59
POWDER	Similarity margin	50.21	54.43	51.94	66.23	34.19
POWDER	Energy score	57.66	78.39	66.69	64.29	51.03
POWDER	Mahalanobis distance	63.70	73.95	68.65	74.37	53.04
POWDER	BOC	69.47	86.73	79.24	79.95	58.99
Hovering UAV	Maximum similarity	54.78	57.74	55.08	72.27	37.28
Hovering UAV	Similarity margin	52.18	54.87	52.17	74.27	30.09
Hovering UAV	Energy score	54.14	58.15	55.58	68.01	40.27
Hovering UAV	Mahalanobis distance	55.81	57.31	56.17	72.85	38.76
Hovering UAV	BOC	55.95	57.93	56.96	73.23	38.67

Table 11 decomposes SPA on POWDER across rotating two-novel-emitter splits. The full aggregation rule is the best row at 1-shot, 5-shot, and 10-shot settings, although the margins over the strongest single-component variants are modest. The split-standard-deviation range remains comparable to the simplified variants, indicating that the gain is spread across the rotating splits.

Table 10. Ablation study of training objectives on Hovering UAV.

Training Setting	Best Val. Accuracy	Closed-Set Accuracy	Open-Set Macro-F1	Unknown F1	10-Shot Novel Accuracy	10-Shot Strict Accuracy
Full objective	63.48	59.86	55.95	38.67	59.92	39.41
Without variance–covariance self-supervised pretraining	39.13	34.93	53.10	36.80	53.25	18.50
Without episodic loss	61.39	59.33	54.23	36.85	57.25	33.73
Without contrastive loss	62.78	60.67	52.79	34.18	58.75	34.98
Without domain-adversarial loss	62.96	58.83	56.36	37.69	59.67	33.88
Cross-entropy only	37.91	33.14	51.23	32.97	52.50	18.44

Table 11. Component-wise ablation of support-aware prototype aggregation on POWDER with rotating two-novel-emitter splits.

Method	1-Shot Novel Acc.	5-Shot Novel Acc.	10-Shot Novel Acc.	Mean 95% CI Half-Width	Split Std. Range	Splits
Full support-aware prototype aggregation	69.45	73.95	75.96	2.28/1.85/1.60	5.84–6.02	6
Simple mean prototype	67.70	73.89	75.55	2.31/1.74/1.49	5.91–8.03	6
Medoid blending only	68.67	72.89	75.32	2.32/1.90/1.52	4.91–7.13	6
Outlier trimming only	69.20	73.08	74.88	2.32/1.90/1.58	5.90–6.37	6
Support weighting only	69.40	73.83	75.25	2.35/1.94/1.64	5.29–6.43	6

Table 12 gives the Hovering UAV component-wise SPA result. Compared with the simple mean prototype, the full SPA rule improves novel-class accuracy by 1.30, 2.60, and 2.09 percentage points at 1-shot, 5-shot, and 10-shot, respectively. Among the single-component rows, outlier trimming is highest at 1-shot and 5-shot. The full rule remains best by combining trimming, centrality weighting, and the medoid anchor.

Table 13 quantifies the uncertainty caused by limited few-shot episode sampling. As the episode count increases, the confidence-interval half-width becomes much smaller; for example, the 10-shot half-width decreases from 9.09 at five episodes to 1.64 at 200 episodes. The mean values vary across episode counts because the added support/query episodes include different support and query combinations. The 200-episode rows provide the narrowest intervals in this table.

Table 12. Component-wise ablation of support-aware prototype aggregation on Hovering UAV.

Method	1-Shot Novel Acc.	5-Shot Novel Acc.	10-Shot Novel Acc.	1-Shot 95% CI Half-Width	5/10-Shot 95% CI Half-Width	Episodes
Simple mean prototype	53.30	55.17	57.83	1.77	1.64/1.62	200
Support weighting only	52.57	56.47	56.85	1.93	1.74/1.63	200
Outlier trimming only	53.75	57.28	57.80	1.79	1.78/1.52	200
Medoid blending only	52.35	56.47	57.23	1.80	1.68/1.53	200
Full support-aware prototype aggregation	54.60	57.77	59.92	1.63	1.69/1.64	200

Table 13. Few-shot reliability on Hovering UAV under different episode counts.

Shot	Episodes	Mean	Std.	95% CI Half-Width	Shot	Episodes	Mean	Std.	95% CI Half-Width
1	5	59.00	8.22	7.20	5	5	59.00	12.45	10.91
1	10	59.50	10.39	6.44	5	10	56.00	12.65	7.84
1	20	56.50	11.82	5.18	5	20	54.50	10.75	4.71
1	50	54.40	11.23	3.11	5	50	53.60	12.25	3.40
1	100	55.70	11.76	2.31	5	100	54.55	12.45	2.44
1	200	54.60	11.76	1.63	5	200	57.77	12.19	1.69
10	5	53.00	10.37	9.09	10	10	55.00	10.00	6.20
10	20	56.50	9.33	4.09	10	50	56.80	10.19	2.82
10	100	57.15	10.95	2.15	10	200	59.92	11.83	1.64

6.5. Deployment Considerations

Table 14 reports the deployed footprint used in the controlled comparisons. All compared variants share the same encoder, so the paper focuses on decision-layer behavior with the neural architecture held fixed. The proposed method adds only decision-layer state: a prototype bank, a lightweight logistic calibrator, and cached SPA prototypes.

Table 14. Deployment footprint of the proposed encoder.

Neural Params	MACs	FLOPs	Latency	Peak Memory	Extra State
99,904	1.93×10^6	3.86×10^6	0.1687 ms	22.0 MB	Prototype bank + logistic calibrator + cached SPA prototypes

Table 14 is included only to make the shared-backbone setting reproducible. MACs and FLOPs were computed with a PyTorch forward-pass profiler for the deployed encoder; latency was measured with CUDA event timing on a batch of 128 samples after warm-up. Since all shared-backbone baselines use the same encoder, the result discussion focuses on calibration and support aggregation with neural-network size held fixed.

Two limitations remain. First, the curated POWDER and Hovering UAV protocols contain small usable identity pools, although the WiSig ManyTx experiment adds a larger identity check. Second, few-shot estimates depend on support/query episode sampling; the 200-episode evaluation reduces this uncertainty but still leaves some residual sampling dependence. Future work will use larger identity-disjoint protocols and higher-episode evaluation when larger curated benchmarks are available.

7. Conclusions

We studied RF emitter recognition under domain shift, unknown-emitter rejection, and few-shot post-deployment registration using a compact prototype framework with BOC and SPA. On the curated POWDER cross-day protocol, the proposed framework improves open-set macro-F1 from 55.23% for the supervised prototype baseline to 69.47%, unknown-F1 from 43.20% to 58.99%, and 10-shot generalized strict accuracy from 42.33% to 65.54%. On the curated Hovering UAV protocol, it improves closed-set accuracy from 33.14% to 59.86% and 10-shot generalized strict accuracy from 18.44% to 39.41%. These gains are obtained by changing the decision layer while keeping the deployed encoder size fixed.

The main contribution is a calibrated and support-aware decision layer that connects three deployment stages: base-class recognition, unknown-emitter rejection, and post-registration novel-class recognition. The WiSig ManyTx evaluation checks scalability on a 50/15/15 base/unknown/novel transmitter split, where BOC reaches 84.16% open-set

macro-F1, and SPA improves 10-shot novel accuracy from 57.81% to 61.44% over the mean-prototype rule. The study is still limited by the small number of usable identities in the curated POWDER and Hovering protocols and by the finite few-shot support/query episode sampling. Future work will evaluate larger identity-disjoint datasets and use higher-episode support/query sampling to strengthen statistical reliability. The evaluation also includes direct OOD score checks, such as Mahalanobis and energy scores, under the same frozen encoder.

Author Contributions: Conceptualization, M.L. and Y.L.; methodology, M.L.; software, M.L. and Q.F.; validation, M.L. and Q.F.; formal analysis, M.L.; investigation, M.L.; resources, Y.L. and X.L.; data curation, M.L.; writing—original draft preparation, M.L.; writing—review and editing, Y.L., Q.F. and X.L.; visualization, M.L.; supervision, Y.L. and X.L.; project administration, Y.L.; funding acquisition, Y.L. and X.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Key Laboratory of Advanced Communication Networks, grant number FFX24641X005, and the Postdoctoral Science Foundation of Hebei Province, China, grant number B2023005001.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The public benchmark datasets used in this study are cited in Section 5.1. The curated preprocessing scripts, split files, configuration files, and per-seed result tables are available from the corresponding author upon reasonable request.

Acknowledgments: The authors thank the providers of the POWDER and Hovering UAV datasets for making public over-the-air RF benchmark data available to the research community.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

BOC	Balanced open-set calibration
FLOPs	Floating-point operations
GRL	Gradient-reversal layer
MACs	Multiply–accumulate operations
OOD	Out-of-distribution
RF	Radio frequency
SPA	Support-aware prototype aggregation
UAV	Unmanned aerial vehicle
VICReg	Variance–Invariance–Covariance Regularization

References

1. Danev, B.; Zanetti, D.; Čapkun, S. On physical-layer identification of wireless devices. *ACM Comput. Surv.* **2012**, *45*, 6. [\[CrossRef\]](#)
2. Jagannath, A.; Jagannath, J.; Kumar, P.S.P.V. A comprehensive survey on radio frequency (RF) fingerprinting: Traditional approaches, deep learning, and open challenges. *Comput. Netw.* **2022**, *219*, 109455. [\[CrossRef\]](#)
3. Xie, L.; Peng, L.; Zhang, J.; Hu, A. Radio frequency fingerprint identification for Internet of Things: A survey. *Secur. Saf.* **2024**, *3*, 2023022. [\[CrossRef\]](#)
4. Li, M.; Fan, W.; Li, Y.; Xie, C.; Duan, Y. End-to-end burst signal demodulation via adaptive masked deep learning framework. *Eng. Appl. Artif. Intell.* **2025**, *162*, 112569. [\[CrossRef\]](#)
5. Li, M.; Li, Y.; Fan, W. Delay-focused preamble-free blind acquisition for asynchronous overloaded grant-free IoT uplinks. *IEEE Internet Things J.* **2026**, early access. [\[CrossRef\]](#)

6. Li, M.; Li, Y.; Feng, Q. Preamble-free blind sequence recovery for asynchronous overloaded sensor uplinks via template-guided polyphase focusing. *IEEE Sens. J.* **2026**, *early access*. [[CrossRef](#)]
7. Alhoraibi, L.; Alghazzawi, D.; Alhebshi, R.; Rabie, O.B.J. Physical layer authentication in wireless networks-based machine learning approaches. *Sensors* **2023**, *23*, 1814. [[CrossRef](#)] [[PubMed](#)]
8. Hanna, S.; Karunaratne, S.; Čabrić, D. WiSig: A large-scale WiFi signal dataset for receiver and channel agnostic RF fingerprinting. *IEEE Access* **2022**, *10*, 22808–22818. [[CrossRef](#)]
9. Gutierrez del Arroyo, J.A.; Borghetti, B.J.; Temple, M.A. Considerations for radio frequency fingerprinting across multiple frequency channels. *Sensors* **2022**, *22*, 2111. [[CrossRef](#)] [[PubMed](#)]
10. Zeng, M.; Yao, Y.; Liu, H.; Hu, Y.; Yang, H. A specific emitter identification system design for crossing signal modes in the air traffic control radar beacon system and wireless devices. *Sensors* **2023**, *23*, 8576. [[CrossRef](#)] [[PubMed](#)]
11. Reus-Muns, G.; Jaisinghani, D.; Sankhe, K.; Chowdhury, K.R. Trust in 5G open RANs through machine learning: RF fingerprinting on the POWDER PAWR platform. In Proceedings of the IEEE Global Communications Conference (GLOBECOM), Taipei, Taiwan, 7–11 December 2020; pp. 1–6. [[CrossRef](#)]
12. Soltani, N.; Reus-Muns, G.; Salehi, B.; Dy, J.; Ioannidis, S.; Chowdhury, K.R. RF fingerprinting unmanned aerial vehicles with non-standard transmitter waveforms. *IEEE Trans. Veh. Technol.* **2020**, *69*, 15518–15531. [[CrossRef](#)]
13. Chen, Z.; Pang, Z.; Hou, W.; Wen, H.; Wen, M.; Zhao, R.; Tang, T. Cross-device radio frequency fingerprinting identification based on domain adaptation. *IEEE Trans. Consum. Electron.* **2024**, *70*, 2391–2400. [[CrossRef](#)]
14. Yang, L.; Li, Q.; Ren, X.; Fang, Y.; Wang, S. Mitigating receiver impact on radio frequency fingerprint identification via domain adaptation. *IEEE Internet Things J.* **2024**, *11*, 24024–24034. [[CrossRef](#)]
15. Shen, G.; Zhang, J.; Marshall, A.; Woods, R.; Cavallaro, J.; Chen, L. Towards receiver-agnostic and collaborative radio frequency fingerprint identification. *IEEE Trans. Mob. Comput.* **2024**, *23*, 7618–7634. [[CrossRef](#)]
16. Zhao, T.; Sarkar, S.; Krijestorac, E.; Čabrić, D. GAN-RXA: A practical scalable solution to receiver-agnostic transmitter fingerprinting. *IEEE Trans. Cogn. Commun. Netw.* **2024**, *10*, 403–416. [[CrossRef](#)]
17. Yang, J.; Zhu, S.; Wen, Z.; Li, Q. Cross-receiver radio frequency fingerprint identification: A source-free adaptation approach. *Sensors* **2025**, *25*, 4451. [[CrossRef](#)] [[PubMed](#)]
18. Wang, C.; Wang, Y.; Zhang, Y.; Xu, H.; Zhang, Z. Open-set specific emitter identification based on prototypical networks and extreme value theory. *Appl. Sci.* **2023**, *13*, 3878. [[CrossRef](#)]
19. Sun, C.; Du, Y.; Qiao, X.; Wu, H.; Zhang, T. Research on the enhancement method of specific emitter open set recognition. *Electronics* **2023**, *12*, 4399. [[CrossRef](#)]
20. Hu, Y.; Fu, Y.; Chen, Y. A lightweight LSTM-based open-set RF fingerprinting identification for edge deployment. *Sci. Rep.* **2025**, *15*, 41568. [[CrossRef](#)] [[PubMed](#)]
21. Wang, Y.; Gui, G.; Lin, Y.; Wu, H.-C.; Yuen, C.; Adachi, F. Few-shot specific emitter identification via deep metric ensemble learning. *IEEE Internet Things J.* **2022**, *9*, 24980–24994. [[CrossRef](#)]
22. Huang, Y.; Hu, A.; Shi, L.; Tian, H.; Fan, J.; Ding, W. Specific emitter identification with few-shot via deep networks based on time and frequency domain with channel, spatial, and self-attention mechanisms. *Electronics* **2025**, *14*, 165. [[CrossRef](#)]
23. Yang, L.; Chai, Z.; Wang, F.T.; Zhao, Z.; Zhou, Y.; Chen, J. Few-shot open-set specific emitter identification: A contrastive learning approach with false negative suppression. *IEEE Commun. Lett.* **2026**, *30*, 902–906. [[CrossRef](#)]
24. Zhang, J.; Liu, Q.; An, S.; He, Y. Cross-domain few-shot specific emitter identification via contrastive self-supervised learning. *IEEE Commun. Lett.* **2025**, *29*, 1564–1568. [[CrossRef](#)]
25. Li, M.; Xie, J.; Yang, H.; Geng, M.; Liu, J. Specific emitter identification of frequency hopping signals based on feature extraction and deep residual network. *IEEE Access* **2022**, *10*, 119084–119094. [[CrossRef](#)]
26. Scheirer, W.J.; Rocha, A.; Sapkota, A.; Boulton, T.E. Toward open set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1757–1772. [[CrossRef](#)] [[PubMed](#)]
27. Scheirer, W.J.; Jain, L.P.; Boulton, T.E. Probability models for open set recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 2317–2324. [[CrossRef](#)] [[PubMed](#)]
28. Bendale, A.; Boulton, T.E. Towards open set deep networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1563–1572. [[CrossRef](#)]
29. Hendrycks, D.; Gimpel, K. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In Proceedings of the International Conference on Learning Representations, Toulon, France, 24–26 April 2017.
30. Liu, W.; Wang, X.; Owens, J.D.; Li, Y. Energy-based out-of-distribution detection. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 21464–21475.
31. Lee, K.; Lee, K.; Lee, H.; Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 7167–7177.
32. Vinyals, O.; Blundell, C.; Lillicrap, T.; Kavukcuoglu, K.; Wierstra, D. Matching networks for one shot learning. *Adv. Neural Inf. Process. Syst.* **2016**, *29*, 3630–3638.
33. Snell, J.; Swersky, K.; Zemel, R. Prototypical networks for few-shot learning. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 4077–4087.

34. Mensink, T.E.J.; Verbeek, J.J.; Perronnin, F.; Csurka, G. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 2624–2637. [[CrossRef](#)] [[PubMed](#)]
35. Schroff, F.; Kalenichenko, D.; Philbin, J. FaceNet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 815–823. [[CrossRef](#)]
36. Li, W.; Wang, J.; Liu, T.; Xu, G. Few-shot specific emitter identification based on a contrastive masked learning framework. *IEEE Commun. Lett.* **2025**, *29*, 408–412. [[CrossRef](#)]
37. Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; Lempitsky, V. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **2016**, *17*, 1–35.
38. AbdelRaouf, H.; Hasnaine, Q.R.; Fouda, M.M.; Fadlullah, Z.M.; Ibrahim, M.I. Paradigm shift toward distributed learning in IoT intelligence: A comprehensive survey of opportunities and challenges. *IEEE Internet Things J.* **2026**, *13*, 20128–20154. . [[CrossRef](#)]
39. Ma, Y.; Ma, R.; Lin, Z.; Miao, C.; Zhang, R.; Long, W.; Wu, W.; Wang, J. Distributed split single-sideband time-modulated arrays for secure communications. *IEEE Internet Things J.* **2026**, *13*, 23862–23875. [[CrossRef](#)]
40. Bardes, A.; Ponce, J.; LeCun, Y. VICReg: Variance-invariance-covariance regularization for self-supervised learning. In Proceedings of the International Conference on Learning Representations, Virtual, 25–29 April 2022.
41. Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; Krishnan, D. Supervised contrastive learning. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 18661–18673.
42. Kaufman, L.; Rousseeuw, P.J. Clustering by means of medoids. In *Statistical Data Analysis Based on the L_1 Norm and Related Methods*; Dodge, Y., Ed.; North-Holland: Amsterdam, The Netherlands, 1987; pp. 405–416.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.