DOMAIN-BALANCED REPRESENTATION LEARNING AND MULTI-PROTOTYPE MAHALANOBIS INFERENCE FOR CROSS-DOMAIN MOSQUITO CLASSIFICATION

Technical Report

Tianyan Deng[1], *Wanchuan Chen*[1], *Jiahao Du*[2], *Rui Gao*[2], *Yanxiong Li*[2*]

[1] School of Mathematics
[2] School of Electronic and Information Engineering
South China University of Technology, Guangzhou, China

ABSTRACT

BioDCASE 2026 Task 5 requires nine-species mosquito classification under extreme recording-domain shift: laboratory domain D5 contributes 99.4% of training samples, whereas field domains D1, D3, and D4 contribute only 0.6%. Under such imbalance, models easily learn device-specific acoustic signatures rather than species-discriminative wingbeat features, leading to spurious species–domain coupling. We investigate domain–class sampling and mainly address the final system through class-conditional domain entropy (CCDE), class-conditional domain balance (CDB), and inference-only Soft Multi-Prototype Mahalanobis classification. CCDE/CDB improve balanced accuracy on unseen domains (BA_{unseen}) by 0.031, giving 0.3580 with softmax. Soft multi-prototype inference further reaches 0.4258 BA_{unseen} and reduces the domain-shift gap (DSG) to 0.3902.

***Index Terms*—** mosquito bioacoustics, domain generalization, class-conditional alignment, contrastive learning, prototype classification

1. INTRODUCTION

BioDCASE 2026 Task 5 classifies nine mosquito species from flight-tone audio across five recording domains (D1–D5). The core difficulty is not merely nine-way classification, but learning stable species-discriminative acoustic cues under extreme domain imbalance. Laboratory domain D5 provides 212,339 training samples (99.4%), recorded with a single high-quality device in a controlled environment. Field domains D1, D3, and D4 provide only 1,308 samples (0.6%), captured with mobile phones and outdoor recorders under varying noise levels and spectral responses. This imbalance creates a fundamental risk: the model can achieve high seen-domain accuracy by exploiting D5-specific device coloration, background noise profiles, or room acoustics, without learning the wingbeat harmonic structure that generalizes across recording conditions.

Our design goal is therefore to reduce spurious coupling between species identity and recording domain. Building on the Domain-Adversarial Neural Network (DANN) [1], adversarial contrastive learning [2], FourierMix [3], MixStyle [4], and Domain-Robust Bioacoustic Learning (DR-BioL) [5], we investigate sampling and combine class-conditional alignment with multi-prototype inference. Class-conditional domain entropy (CCDE) and class-conditional domain balance (CDB) are the core training contributions, while Soft Multi-Prototype Mahalanobis models multimodal within-species geometry at inference. We denote balanced accuracy on seen and unseen domains by BA_{seen} and BA_{unseen}, respectively, and define the domain-shift gap as $DSG = |BA_{seen} - BA_{unseen}|$.

*Corresponding author: eeyxli@scut.edu.cn

2. MODEL ARCHITECTURE

We use the multitemporal resolution convolutional neural network (MTRCNN) baseline architecture [6], which has 0.22 million parameters. Three parallel branches use dilated kernels of sizes 3, 5, and 7, each with three convolutional stages. Outputs are concatenated into a 192-dimensional representation and projected through a Gaussian error linear unit (GELU)-activated linear layer to a 32-dimensional embedding. Inputs are 64-bin log-mel spectrograms at 8 kHz, globally normalized with per-bin mean and standard deviation computed over the training set. Although large pretrained models and adaptive few-shot audio classifiers offer strong representation capacity [7, 8, 9], our evaluated Pretrained Audio Neural Networks (PANNs) CNN14 model (80 million parameters) and Audio Spectrogram Transformer (AST; 87 million parameters) attain higher BA_{seen} but substantially weaker BA_{unseen}, confirming that overparameterized models overfit to dominant D5 acoustics under extreme domain imbalance.

3. METHODOLOGY

We organize the method by the failure mode each component targets: D5-dominated batches, recording variability, aggregate and class-conditional domain leakage, class imbalance, and rare-species inference instability.

3.1. Domain–class joint sampling

A standard class-balanced sampler still produces batches dominated by D5, limiting cross-domain comparisons for adversarial and contrastive objectives. We assign sample i, with class y_i and domain d_i, the weight

$$w_i = \frac{1}{N(y_i, d_i)\,|\mathcal{D}(y_i)|}, \tag{1}$$

where w_i is the sampling weight, $N(y_i, d_i)$ counts the corresponding class–domain group, and $\mathcal{D}(y_i)$ is the set of domains containing that class. A D4 *Aedes aegypti* sample receives about 3,300 times the weight of a D5 counterpart. This sampler was used in one submitted configuration to expose rare field combinations. We do not attribute a standalone quantitative gain to it.

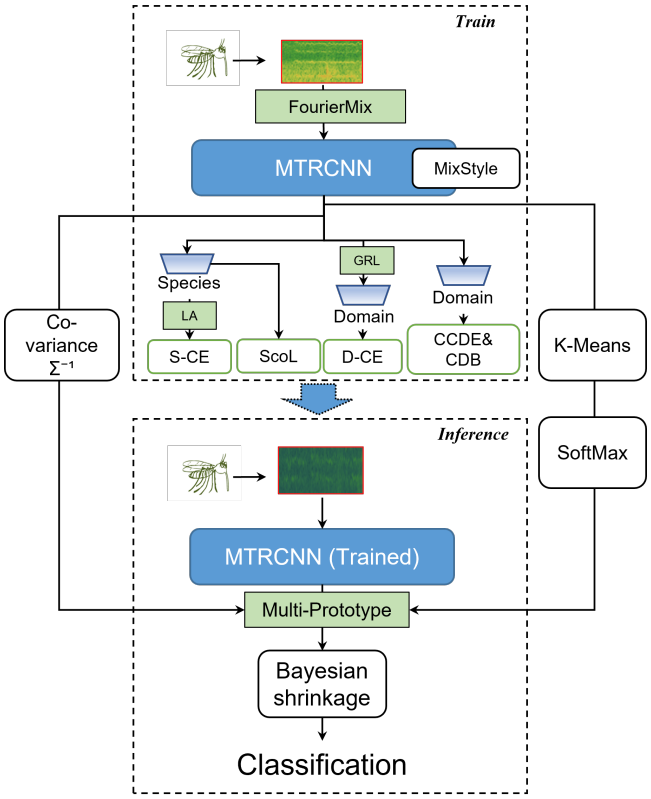

Figure 1: Training pipeline and inference-only prototype classification.

3.2. Spectral style augmentation

FourierMix. Adapted from the Fourier-based framework of Xu et al. [3], we empirically use the Fourier amplitude spectrum of a log-mel spectrogram as a proxy for domain style, while keeping the original phase to better preserve sample-specific spectral structure. For samples from different domains,

$$\begin{aligned} \widehat{A}(x_i) &= (1-\lambda)A(x_i) + \lambda A(x_j), \\ \widehat{x}_i &= \mathcal{F}^{-1}\left(\widehat{A}(x_i)e^{-\mathrm{j}P(x_i)}\right), \end{aligned} \tag{2}$$

where $A(x)$ and $P(x)$ are the Fourier amplitude and phase, $\lambda \sim \mathcal{U}(0,1)$ is the mixing coefficient, and $\mathcal{F}^{-1}$ is the inverse Fourier transform. Full-spectrum mixing is applied with probability 0.5, selecting x_j only from a domain different from that of x_i. This provides an empirical perturbation of recording-related spectral style while retaining the source sample's phase.

MixStyle. MixStyle [4] is inserted after the first convolutional stage. With probability 0.5, it mixes channel-wise means and standard deviations using Beta(0.1, 0.1) coefficients, providing implicit domain style perturbation.

3.3. Global domain-adversarial alignment

The Domain-Adversarial Neural Network (DANN) learns a feature extractor G_f, species classifier G_y, and domain discriminator G_d [1]. A gradient reversal layer (GRL) is the identity in the forward pass and multiplies the encoder gradient by $-\lambda_d$ during backpropagation, yielding the saddle-point objective

$$\min_{\theta_f,\theta_y}\left(\max_{\theta_d} \mathcal{L}_{\text{species}} - \lambda_d \mathcal{L}_{\text{domain}}\right). \tag{3}$$

Here, θ_f, θ_y, and θ_d are the parameters of G_f, G_y, and G_d; $\mathcal{L}_{\text{species}}$ and $\mathcal{L}_{\text{domain}}$ are species and domain cross-entropy losses; and λ_d is the adversarial strength. Following the domain-divergence view of Ben-David et al. [10], we reduce domain-discriminative information in $z = G_f(x)$. We use a sigmoid warmup to a maximum $\lambda_d = 0.3$ to avoid early collapse.

3.4. Class-conditional domain alignment: CCDE and CDB

Motivation. A globally confused domain discriminator does not guarantee per-species domain invariance. Individual species can still retain domain-specific subclusters, especially when species and domain are highly correlated. Global DANN alignment is therefore necessary but insufficient; class-conditional alignment is required.

We introduce an auxiliary domain head $q_\phi(d \mid z_i)$ with parameters ϕ, separate from the GRL-equipped discriminator. This head has no GRL and serves as a cooperative regularizer rather than an adversary. Let C_B be the set of species represented in mini-batch B and let $I_c = \{i \in B : y_i = c\}$.

Class-conditional domain entropy (CCDE) maximizes the Shannon entropy of domain predictions within each species:

$$\mathcal{L}_{\text{ccde}} = -\frac{1}{|C_B|}\sum_{c\in C_B}\frac{1}{|I_c|}\sum_{i\in I_c} H\big(q_\phi(d \mid z_i)\big), \tag{4}$$

where $H(\cdot)$ is Shannon entropy and $|\cdot|$ denotes set cardinality. Maximizing entropy pushes the encoder to produce features for which the auxiliary domain head cannot confidently assign a domain, but crucially this constraint is applied per species.

Class-conditional domain balance (CDB) directly matches each species' mean domain prediction to a uniform distribution using Kullback–Leibler (KL) divergence:

$$\begin{aligned} \bar{q}_c &= \frac{1}{|I_c|}\sum_{i\in I_c} q_\phi(d \mid z_i), \\ \mathcal{L}_{\text{cdb}} &= \frac{1}{|C_B|}\sum_{c\in C_B} \mathrm{KL}(\bar{q}_c \,\|\, \mathcal{U}_D)\,. \end{aligned} \tag{5}$$

Here, $\bar{q}_c$ is the average predicted domain distribution for species c, and $\mathcal{U}_D$ is the uniform distribution over the D training domains. CCDE discourages confident domain predictions within each species, whereas CDB penalizes systematic class–domain associations at the aggregate level.

Training details. The auxiliary domain head is trained with standard domain classification cross-entropy without GRL. For CCDE/CDB, gradients are stopped at the auxiliary-head parameters and applied only to the encoder features, preventing these regularizers from directly weakening the head's domain discrimination. CCDE/CDB activate after epoch 5. With $\gamma_{\text{ccde}} = \gamma_{\text{cdb}} = 0.05$, they improve $\text{BA}_{\text{unseen}}$ from 0.3274 to 0.3580 (+0.031) while recovering BA_{seen} from 0.7929 to 0.8412.

3.5. Species cohesion and class imbalance

Species-cohesion contrastive loss (ScoL). Following DR-BioL [5], supervised contrastive learning in the embedding space pulls same-

species embeddings together:

$$\mathcal{L}_{\text{ScoL}} = \frac{1}{|\bar{I}|} \sum_{i \in \bar{I}} -\log \frac{\sum_{p \in P(i)} e^{\text{sim}(z_i, z_p)/\tau_s}}{\sum_{a \in A(i)} e^{\text{sim}(z_i, z_a)/\tau_s}}. \tag{6}$$

Here, $\bar{I}$ contains anchors with at least one positive, $P(i)$ contains same-species positives, $A(i)$ contains all comparison samples except i, sim is cosine similarity, and $\tau_s = 0.01$. The loss weight is 0.5.

Training-time logit adjustment (LA). Following Menon et al. [11], we add $\tau_{\text{train}} \log \pi_c$ to the class-c training logit with $\tau_{\text{train}} = 0.4$, where π_c is the class prior. This adjustment is used only during training; no inference-time LA is used in the final system.

The complete training objective is

$$\begin{aligned} \mathcal{L} = \mathcal{L}_{\text{species}} + \gamma_d \mathcal{L}_{\text{domain}} + \gamma_s \mathcal{L}_{\text{ScoL}} \\ + \mathbf{1}_{e \geq 5} \left(\gamma_{\text{ccde}} \mathcal{L}_{\text{ccde}} + \gamma_{\text{cdb}} \mathcal{L}_{\text{cdb}} \right). \end{aligned} \tag{7}$$

Here, γ_d, γ_s, γ_{ccde}, and γ_{cdb} weight the auxiliary objectives; e is the current epoch.

3.6. Soft Multi-Prototype Mahalanobis inference

This inference-only strategy is applied to a converged checkpoint without modifying training. A single class mean [12] can be dominated by D5 and fail to represent field-domain modes. We therefore cluster the training embeddings of class c into an adaptive number of sub-prototypes:

$$K_c = \min\left(K_{\max}, \left\lfloor \sqrt{\frac{N_c}{N_{\min}}} \right\rfloor \right), \quad \{\mu_{c,k}\}_{k=1}^{K_c} = \text{KMeans}(Z_c, K_c), \tag{8}$$

where N_c is the class count, $N_{\min} = \min_c N_c$, Z_c is the set of class-c training embeddings, and $K_{\max} = 10$. To stabilize small clusters, each centroid is lightly shrunk toward the global training mean:

$$\widehat{\mu}_{c,k} = \frac{n_{c,k}\mu_{c,k} + m\mu_{\text{global}}}{n_{c,k} + m}, \quad m = 2. \tag{9}$$

Here, $n_{c,k}$ is the cluster size and μ_{global} is the global training mean. Embeddings are ℓ_2-normalized before prototype construction and covariance estimation. All classes share a covariance pooled from the training embeddings, avoiding unstable classwise estimates [13]:

$$\Sigma_{\text{reg}} = (1 - \alpha)\Sigma_{\text{sample}} + \alpha I, \quad \alpha = 0.72. \tag{10}$$

For test embedding z, sub-prototype scores and their soft aggregation are

$$\begin{aligned} s_{c,k}(z) &= -(z - \widehat{\mu}_{c,k})^\top \Sigma_{\text{reg}}^{-1} (z - \widehat{\mu}_{c,k}), \\ S_c(z) &= \sum_{k=1}^{K_c} \frac{\exp(s_{c,k}/\tau)}{\sum_{j=1}^{K_c} \exp(s_{c,j}/\tau)} s_{c,k}, \quad \tau = 3. \end{aligned} \tag{11}$$

Soft aggregation interpolates between a hard maximum and an average, reducing the cross-class region expansion caused by taking only the nearest sub-prototype. All prototypes and statistics are computed from labeled training embeddings; evaluation audio is used only for forward scoring.

4. EXPERIMENTAL RESULTS

In addition to the metrics defined above, we denote species-averaged balanced accuracy by $\text{BA}_{\text{species}}$.

Table 1: Test-set comparison: final system versus baselines. Results use seed 42 except the 10-seed baseline.

System	BA_{seen}	$\text{BA}_{\text{unseen}}$	DSG
MTRCNN baseline (10-seed mean)	0.8806	0.1751	0.7055
CNN14 + GRL + FourierMix	0.8635	0.2434	0.6201
v1: MTRCNN + GRL + FourierMix + MixStyle	0.7564	0.3173	0.4391
v2: v1 + ScoL	0.7929	0.3274	0.4655
v3: v2 + CCDE + CDB	0.8412	0.3580	0.4832
v3 + Soft Multi-Proto	**0.8160**	**0.4258**	**0.3902**

Table 2: Stepwise ablation results use seed 42, except the official MTRCNN baseline reported as a 10-seed mean. Each row adds the indicated component.

Configuration	BA_{seen}	$\text{BA}_{\text{unseen}}$	DSG
MTRCNN baseline	0.8806	0.1751	0.7055
+ GRL + FourierMix + MixStyle (v1)	0.7564	0.3173	0.4391
+ ScoL (v2)	0.7929	0.3274	0.4655
+ CCDE only ($\gamma = 0.05$)	0.8363	0.3428	0.4935
+ CDB only ($\gamma = 0.05$)	0.8431	0.3442	0.4989
+ CCDE + CDB ($\gamma = 0.05$, v3)	**0.8412**	**0.3580**	**0.4832**

4.1. Main results

The v3 softmax configuration reaches 0.3580 $\text{BA}_{\text{unseen}}$, a 104% relative gain over the baseline. CCDE/CDB contribute 0.0306 while increasing BA_{seen} from 0.7929 to 0.8412. Soft multi-prototype inference further raises $\text{BA}_{\text{unseen}}$ to 0.4258.

4.2. Component-wise ablation

Table 2 demonstrates that each component provides a cumulative gain. The joint CCDE+CDB setting outperforms either loss alone (0.3580 versus 0.3428 and 0.3442), confirming that entropy maximization and distribution balancing provide complementary signals for class-conditional domain disentanglement.

4.3. Multi-prototype inference

Soft multi-prototype inference improves $\text{BA}_{\text{unseen}}$ from 0.3974 to 0.4258 (+0.0284, 7.1%) and reduces DSG by 0.0612 relative to the single-prototype classifier. It also slightly outperforms hard maximum aggregation, supporting soft scoring when sub-prototype and inter-class distances are comparable.

4.4. Per-species and per-domain analysis

Figure 2 shows the largest gains for *albopictus* (+0.146), *aegypti* (+0.078), and *stephensi* (+0.067), indicating that multiple sub-prototypes better cover heterogeneous field modes. Recall decreases for *pipiens* (−0.030) and *dirus* (−0.050), so multi-prototype inference is used only for the compatible seed-42 class-balanced system. After adversarial training, domain accuracy remains about 98% on D5 but near 0% on D1/D3/D4, indicating residual dominant-domain statistics.

For CCDE/CDB weights, $\gamma = 0.01$ under-constrains the representation, while $\gamma = 0.10$ begins to over-constrain it. The selected value $\gamma = 0.05$ gives the best $\text{BA}_{\text{unseen}}$ in our sweep.

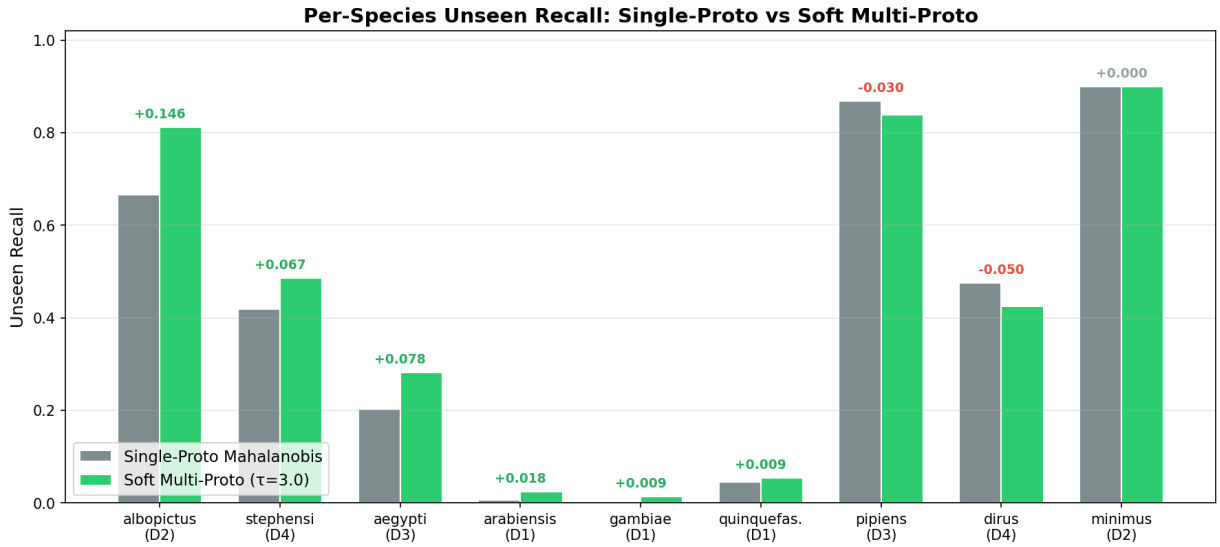

Figure 2: Per-species unseen recall of single-prototype and Soft Multi-Prototype Mahalanobis inference.

Table 3: Inference-only comparison using the same checkpoint.

Method	BA_{unseen}	BA_{seen}	DSG
Single-prototype Mahalanobis	0.3974	0.8488	0.4514
Hard max multi-prototype	0.4229	0.8300	0.4071
Soft multi-prototype ($\tau = 3$)	**0.4258**	0.8160	**0.3902**

5. LIMITATIONS

D5-only species such as *A. gambiae* and *A. arabiensis* have no field-domain training examples, leaving an irreducible joint-error term in the adaptation bound [10]. Multi-prototype gains also depend on embedding geometry: dispersed class clusters can expand toward neighboring classes, as observed for some seeds and domain–class-sampled models. We therefore use this inference rule only for the compatible seed-42 class-balanced system.

6. CONCLUSION

We presented a problem-driven framework for cross-domain mosquito classification under extreme laboratory/field imbalance. CCDE/CDB extend adversarial training to class-conditional alignment, improving BA_{unseen} by 0.031. Soft Multi-Prototype Mahalanobis inference models multimodal class geometry and raises BA_{unseen} from 0.3974 for a single prototype to 0.4258 while reducing DSG to 0.3902. Remaining errors point to residual D5 leakage and the need for representative field-domain data.

7. REFERENCES

[1] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016.

Table 4: Key hyperparameters for the final system.

Category	Configuration
Audio	8 kHz, 2.0 s clips, 64 mel bins, 512/80/512 window/hop/fast Fourier transform (FFT) size, global per-bin normalization
Model	MTRCNN (0.22 million parameters), 32-dimensional embedding, dropout 0.2
Optimizer	AdamW, learning rate 0.001, weight decay 0.0001, batch 64, 100 epochs, patience 15, no scheduler
Sampler	Class-balanced for final v3; domain–class joint sampling used in one submitted variant
Augmentation	FourierMix $\alpha = 1.0$, probability 0.5, inter-domain only; MixStyle $p = 0.5$, Beta$(0.1, 0.1)$, depth 1
DANN	GRL max $\lambda = 0.3$ with sigmoid warmup
CCDE/CDB	$\gamma_{ccde} = \gamma_{cdb} = 0.05$, warmup epoch 5
ScoL / train LA	ScoL $\gamma = 0.5$, $\tau = 0.01$; training LA $\tau_{train} = 0.4$
Multi-prototype	$K_{max} = 10$, covariance shrinkage $\alpha = 0.72$, Bayesian prior $m = 2$, soft temperature $\tau = 3$

[2] Y. Si, Y. Li, S. Huang, and B. Liu, "Cross-domain few-shot class-incremental audio classification via adversarial contrastive learning," in *Proceedings of Interspeech*, 2026, to appear.

[3] Q. Xu, R. Zhang, Y. Zhang, Y. Wang, and Q. Tian, "A Fourier-based framework for domain generalization," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 14 378–14 387.

[4] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with MixStyle," in *International Conference on Learning Representations*, 2021.

[5] Y. Hou, Z. Liu, X. Shen, and S. Roberts, "Learning domain-robust bioacoustic representations for mosquito species classi-

fication with contrastive learning and distribution alignment," *arXiv preprint arXiv:2510.00346*, 2025.

[6] Y. Hou, V. Zdravkovic, M. Sinka, Y. Li, W. Wang, M. D. Plumbley, K. Willis, and S. Roberts, "BioDCASE 2026 challenge baseline for cross-domain mosquito species classification," *arXiv preprint arXiv:2603.20118*, 2026.

[7] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "PANNs: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2880–2894, 2020.

[8] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio spectrogram transformer," in *Proceedings of Interspeech*, 2021, pp. 571–575.

[9] Y. Si, Y. Li, J. Tan, G. Chen, Q. Li, and M. Russo, "Fully few-shot class-incremental audio classification with adaptive improvement of stability and plasticity," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 33, pp. 418–433, 2025.

[10] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine Learning*, vol. 79, no. 1–2, pp. 151–175, 2010.

[11] A. K. Menon, S. Jayasumana, A. S. Rawat, H. Jain, A. Veit, and S. Kumar, "Long-tail learning via logit adjustment," in *International Conference on Learning Representations*, 2021.

[12] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 4077–4087.

[13] K. Lee, K. Lee, H. Lee, and J. Shin, "A simple unified framework for detecting out-of-distribution samples and adversarial attacks," in *Advances in Neural Information Processing Systems*, vol. 31, 2018, pp. 7167–7177.