MAST: Label-Efficient, Robust, and Generalizable Sound Detection for Biodiversity Monitoring via Masked Audio Pretraining and Self-Training

Tianyi Xu[1,2] **Daniel L. Pimentel-Alarcón**[1]
Zuzana Buřivalová[1] **Claudia Solís-Lemus**[1*]
[1]University of Wisconsin–Madison
[2]Oregon State University

Abstract

Passive acoustic monitoring can measure biodiversity at larger scales, but time–frequency annotation of animal vocalizations is expensive, site-specific, and difficult to sustain at scale. We present a label-efficient framework for binary detection of animal sounds that combines masked audio pretraining with a lightweight detector on mel spectrograms, then further improves robustness through iterative self-training on unlabeled audio. We first pretrain a ViT-based encoder on unlabeled recordings via masked reconstruction and transfer the encoder to a detection backbone. To better separate animal sounds from confounding background, we add a box-level contrastive loss that pulls matched event regions together while pushing noisy negatives apart. We then apply a two-stage pseudo-labeling curriculum to exploit large unlabeled pools without additional annotation. We train and evaluate models separately on two ecologically distinct domains: tropical rainforest soundscapes (Indonesia) and bird vocalizations in Mediterranean habitats (Spain). On both domains, masked audio pretraining and contrastive learning consistently improve time–frequency detection under temporal and cross-site distribution shift, and self-training yields further gains in out-of-distribution performance. On the rainforest domain, MAST with self-training achieves +0.22 mAP and +0.24 F1 over the AudioMAE baseline under cross-site shift. On the bird domain, self-training achieves +0.12 mAP and +0.10 F1 over the AudioMAE baseline under cross-site shift. Overall, our results show that MAST can effectively extend self-supervised audio representations from clip-level tasks to robust box-level localization across diverse bioacoustic settings, providing a practical path for biodiversity monitoring with limited labels. Code is available at https://github.com/tianyi0216/MAST.

1 Introduction

Passive acoustic monitoring (PAM) offers a scalable, non-invasive way to measure biodiversity at landscape scales [60, 71, 30, 8]. Autonomous recorders can operate for months, capturing rich soundscapes that contain vocalizations from birds, mammals, insects, amphibians, as well as environmental and anthropogenic sounds [38, 55]. Turning these recordings into actionable ecological information typically follows two broad directions: one can use acoustic indices that summarize the properties of the soundscape, or species-specific recognition models that predict whether a known target sound occurs in a recording segment. However, the former does not identify which species produced the sounds, while the latter is typically limited to predefined classes of acoustically known

*Corresponding author: Claudia Solís-Lemus. Correspondence: xut2@oregonstate.edu, solislemus@wisc.edu.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

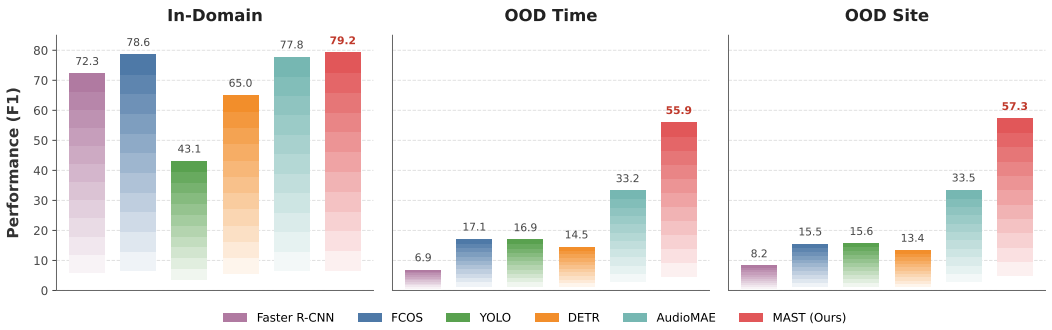


Figure 1: **Performance comparison on rainforest domain.** MAST achieves the best overall performance, with especially large improvements under temporal and cross-site distribution shift, showing stronger robustness than fully supervised baselines and generic SSL transfer.

species. In hyperdiverse and acoustically understudied soundscapes such as tropical rainforests [62], species identities are often unknown and species-level labels are scarce. We therefore first localize animal sounds in time and frequency, then support identification through retrieval or expert review. The main bottleneck is therefore not data collection, but annotation: extracting ecological signals from raw audio requires experts to mark time–frequency regions of animal sounds, which is expensive and difficult to scale [41]. In addition, real-world soundscapes are highly non-stationary, with background conditions shifting across day/night cycles, weather, habitat structure, sensor placement, and species composition [33, 66]. As a result, models trained on a single site or day often generalize poorly [33].

A common strategy is to treat spectrograms as images and apply supervised detectors such as Faster R-CNN, YOLO, or FCOS [49, 19]. While effective in fully supervised settings, these models typically require many annotations and can overfit to local recording conditions, leading to degraded performance under temporal or spatial shift [66].

To address these challenges, we propose MAST, a label-efficient framework for robust sound detection under limited supervision. We study binary detection of "any animal sound" with limited expert labels. In the rainforest domain, labels come from one fully annotated day at a single site, and evaluation covers unseen days and sites in the same region. Figure 2 summarizes the motivation for MAST: rainforest soundscapes exhibit substantial temporal and cross-site distribution shift, while standard supervised detectors trained on narrow label coverage often generalize poorly. MAST addresses this by decoupling representation learning from detection. First, we adopt the ViT-based masked audio encoder architecture of AudioMAE [21] and pretrain it on large unlabeled rainforest audio via masked spectrogram reconstruction. We then transfer the pretrained encoder to a lightweight detector with audio-aware adapters and an FPN [34] neck to recover locality and multi-scale structure. To improve separation between true events and confounding background, we add a box-level contrastive loss over region features. Finally, we exploit large unlabeled pools through iterative self-training with pseudo-labels and labeled-data refinement.

Time–frequency detection is ecologically more informative than clip-level tagging: it reveals which sounds overlap in time and frequency, how vocal activity is distributed across the spectrum, and can help determine whether detected events correspond to known or potentially novel taxa. This information is essential for biodiversity assessment but lost by coarser representations. With models trained and evaluated separately in each domain, MAST improves robustness to temporal and site shift on both tropical rainforest soundscapes and Mediterranean bird vocalizations (Figure 1). Our contributions are as follows:

- We introduce a practical pipeline for binary time–frequency sound detection that combines masked audio pretraining, adaptation, and lightweight detection under limited labels.
- We design a region-level contrastive loss that improves event/background separability and strengthens generalization under distribution shift.
- We show that an iterative self-training curriculum effectively leverages unlabeled data and improves detection in out-of-distribution settings without additional expert annotation.

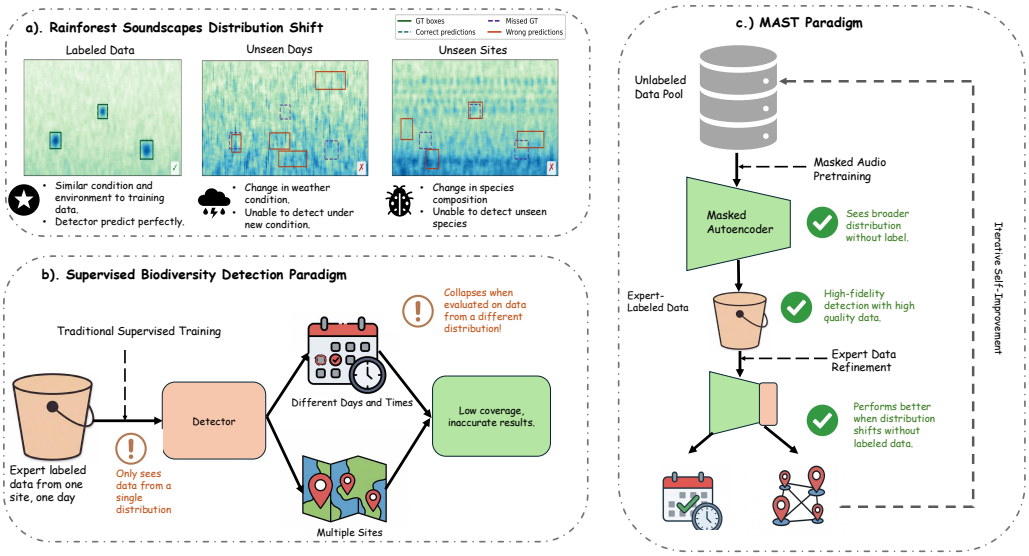


Figure 2: **Motivation of MAST.** (a) Rainforest soundscapes exhibit substantial temporal and cross-site distribution shift, making detection under unseen conditions difficult. (b) Standard supervised biodiversity detectors trained on limited labels from a narrow time/site range often generalize poorly when deployed to new days and sites. (c) These challenges motivate a label-efficient and shift-robust framework that can leverage both limited expert labels and abundant unlabeled soundscapes.

- We evaluate on two distinct bioacoustic domains, tropical rainforest soundscapes and European bird vocalizations, and provide an end-to-end recipe for converting large unlabeled recordings into robust detectors for biodiversity monitoring.

2 Related Work

Passive acoustic monitoring and supervised sound detection. PAM enables long-duration, large-scale biodiversity recording [59, 60, 13], but downstream analysis remains bottlenecked by the need for manual annotation or by acoustic indices that need ground truthing [26, 7, 6]. Classical template-matching detectors scale poorly to diverse soundscapes [22, 64, 25, 9, 3, 26]. Learned systems such as BirdNET [23] and Voxaboxen [35] operate at clip level or along the time axis only, without full time–frequency detection. Casting the problem as object detection on spectrograms [49, 24, 68, 19] enables box-level localization but typically requires substantial supervision and remains brittle under site/day shift [58, 19, 14]. Zhu and Sato [74] use a self-supervised audio spectrogram transformer for time–frequency box detection. Complementary resources include time–frequency annotations in WABAD [48] and Merlin L48 [61], recording-level species labels in iNatSounds [11], and the BioDCASE challenge [5].

Self-supervised audio representations. SSL has advanced audio representation learning via masked modeling, contrastive learning, and predictive objectives [1, 20, 21, 12, 44, 43]. Recent bioacoustic foundation models such as Perch 2.0 [67], Bird-MAE [51], NatureLM-audio [54], and AVEX [39] transfer effectively to low-label monitoring tasks [40, 72]. Perch 2.0 also exposes spatial feature maps before pooling. AudioSAM [2] applies masked audio pretraining to temporal vocalization detection and species classification. Existing benchmarks cover tagging, classification, detection, and retrieval [65, 15, 50, 18]. Extending pretrained representations to dense box-level localization introduces challenges in boundary fidelity, foreground–background imbalance, and stability under non-stationary noise. Our work studies how pretrained features can be adapted for robust time–frequency detection in biodiversity soundscapes.

Weak supervision and unlabeled soundscape learning. Prior SED research has explored weak supervision, semi-supervised learning, and pseudo-labeling [57, 28, 69, 46, 47, 27]. However, most of this literature focuses on temporal event activity rather than explicit time–frequency box detection, and stable learning from pseudo labels remains difficult under distribution shift [57, 35]. Our method

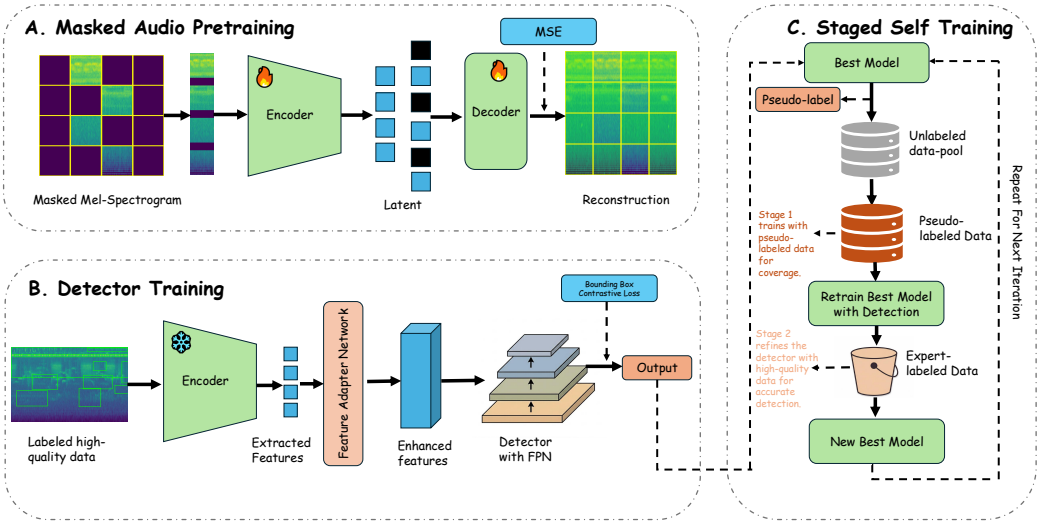


Figure 3: **Overview of the MAST pipeline.** (A) Masked-audio pretraining learns general acoustic representations from unlabeled rainforest spectrograms via masked reconstruction. (B) Detector training transfers the pretrained encoder to a detection model with an audio-aware adapter and lightweight FPN, together with a box-level contrastive objective for event separation. (C) Staged self-training improves robustness under distribution shift by first generating pseudo-labels on unlabeled audio and then retraining with exploration on pseudo-labeled data followed by refinement on expert-labeled data.

differs in both objective and setting: we use confidence-weighted pseudo-boxes within a staged self-training pipeline, where exploration on diverse pseudo-labeled soundscapes expands coverage under shift, followed by expert-guided refinement that re-anchors precision.

3 Methods

Figure 3 provides an overview of MAST, which combines masked-audio pretraining, label-efficient detector adaptation, and self-training for robust sound detection under distribution shift.

3.1 Problem Setup

Given an audio waveform $x(t)$, we compute a log-mel spectrogram $S \in \mathbb{R}^{F\times T}$. Each clip may contain multiple annotated time–frequency boxes $B = \{(b_i, y_i)\}_{i=1}^{N}$, where each box $b_i = [t_i^{(1)}, t_i^{(2)}, f_i^{(1)}, f_i^{(2)}]$. Although the dataset includes sonotype labels y_i, our primary task is binary detection ("any animal sound"), so all annotated events are treated as a single foreground class. We consider a low-label regime with limited expert-labeled data $\mathcal{D}_\ell$ and abundant unlabeled data $\mathcal{D}_u$, and aim to achieve robust detection under temporal and spatial shift.

3.2 Tasks and Objectives

Overview. Our pipeline has three stages: (i) self-supervised representation learning on unlabeled soundscapes, (ii) label-efficient supervised time–frequency detection on expert boxes, and (iii) staged self-training to exploit additional unlabeled audio. See Fig. 3 for an overview of MAST.

Self-supervised pretraining. Given an unlabeled log-mel spectrogram chunk $S \sim \mathcal{D}_u$, we pretrain a masked audio encoder E_ϕ by masking a set of time-frequency patches M and reconstructing the masked content with a lightweight decoder. We minimize the masked reconstruction loss

$$\min_{\phi} \mathbb{E}_{S\sim\mathcal{D}_u}\left[\mathcal{L}_{\mathrm{MAE}}(S;\phi)\right], \quad \mathcal{L}_{\mathrm{MAE}} = \frac{1}{|M|}\sum_{(f,t)\in M} \|\hat{S}[f,t] - S[f,t]\|_2^2.$$

Supervised detection on expert labels. For time-frequency detection, we reuse the pretrained encoder as the detector backbone and attach lightweight audio-aware adapters A_ψ and a feature pyramid N_γ. A detection head D_θ predicts box coordinates and confidence scores. We train on the labeled set $\mathcal{D}_\ell$ with a standard detection objective:

$$\mathcal{L}_{\text{det}} = \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{box}}\mathcal{L}_{\text{box}} + \lambda_{\text{iou}}\mathcal{L}_{\text{iou}},$$

where $\mathcal{L}_{\text{cls}}$ is focal loss, $\mathcal{L}_{\text{box}}$ is ℓ_1 regression on box corners, and $\mathcal{L}_{\text{iou}}$ is GIoU loss [53]. To improve robustness under background shift, we add a box-level contrastive term that encourages invariance across perturbations of the same event while separating events from hard background regions. For each ground-truth box we form positives via spatial/temporal jittering, and sample negatives from low-IoU proposals (sampling details in App. D.4). Let z_i be the normalized projected feature pooled from box i, the contrastive loss is

$$\mathcal{L}_{\text{con}} = \frac{1}{N_+}\sum_i \left[-\log \frac{\sum_{p\in P_i}\exp\left(\langle z_i, p\rangle/\tau\right)}{\sum_{p\in P_i}\exp\left(\langle z_i, p\rangle/\tau\right) + \sum_{n\in N}\exp\left(\langle z_i, n\rangle/\tau\right)}\right].$$

The supervised training objective becomes

$$\min_{\psi,\gamma,\theta,\phi_{\text{top}}} \mathbb{E}_{(S,B)\sim\mathcal{D}_\ell}\left[\mathcal{L}_{\text{det}}\left(S,B\right) + \lambda_{\text{con}}\,\mathcal{L}_{\text{con}}\left(S,B\right)\right],$$

where ϕ_{top} denotes the parameters of the top k encoder blocks that are unfrozen during fine-tuning.

Self-training on unlabeled audio. Starting from a seed detector trained on $\mathcal{D}_\ell$, we generate pseudo boxes on $\mathcal{D}_u$ using class-agnostic NMS and a confidence threshold q, yielding $\Pi_q\left(\mathcal{D}_u\right)$. We first train on $\mathcal{D} = \Pi_q(\mathcal{D}_u)$ and then refine on $\mathcal{D} = \mathcal{D}_\ell$:

$$\min_{\psi,\gamma,\theta,\phi_{\text{top}}} \mathbb{E}_{(S,B)\sim\mathcal{D}}\left[w(S)\mathcal{L}_{\text{det}}\left(S,B\right) + \lambda_{\text{con}}\,\mathcal{L}_{\text{con}}\left(S,B\right)\right].$$

Here $w(S)$ is the average pseudo-label confidence in Stage 1 and equals one in Stage 2. To reduce confirmation bias, we keep the backbone largely frozen in early self-training before unfreezing top layers. See App. A for the full two-stage objectives.

3.3 Detector Design

We use the pretrained ViT-B/16 masked audio encoder E_ϕ as the backbone for time-frequency detection. Given a log-mel spectrogram chunk, the encoder outputs a token grid which we reshape into a feature map $E \in \mathbb{R}^{C\times H\times W}$. We then attach a lightweight audio-aware adapter A_ψ, a feature pyramid network N_γ, and a detection head D_θ. For label-efficient transfer, we either freeze E_ϕ or fine-tune only the top k transformer blocks, while training A_ψ, N_γ, and D_θ. Our default head is an anchor-free FCOS predicting classification, centerness, and box regression. Full architectural details, adapter variants, and additional ablations are provided in App. D.3, D.4, and E.5.

Resolution bottleneck and asymmetric adapter. Transferring patch-based ViTs to dense time–frequency detection introduces a *resolution bottleneck*: with patch size p=16 and spectrogram $T\times F = 1024\times128$, the encoder produces a 64×8 token grid with strides s_t=s_f=16. We show (Proposition 1, App. B) that grid quantization can degrade IoU to at most $\delta_t\delta_f/[(\delta_t+s_t)(\delta_f+s_f)]$ in the worst case, and that narrow-band events with $\delta_f \le p$ may fail to reach IoU ≥ 0.5 under worst-case alignment (Corollary 2). This bottleneck is asymmetric: the frequency axis has only 8 tokens, $\sim$1 kHz each, while many vocalizations span <1 kHz in bandwidth.

Our adapter resolves this via *asymmetric learnable upsampling* (u_t=2, u_f=4), producing a 128×32 feature map with effective strides 8×4. The larger frequency factor addresses the more severe bottleneck. Upsampled features are refined by *anisotropic* $k\times1$ and $1\times k$ convolutions that exploit the separable spectro-temporal structure of animal sounds. Empirically, this adapter improves cross-site OOD F1 from 0.348 to 0.447 and achieves the highest mean IoU (0.682) among all variants (Tables 3, 13). See App. B for formal analysis with proofs.

3.4 Self-training on Unlabeled Soundscapes

Starting from a seed detector trained on $\mathcal{D}_\ell$, we generate pseudo boxes on unlabeled audio $\mathcal{D}_u$ by running the detector, applying class-agnostic NMS, and retaining predictions above a confidence threshold q, yielding a pseudo-labeled dataset $\Pi_q(\mathcal{D}_u)$. We then retrain with a two-stage curriculum: Stage 1 trains on $\Pi_q(\mathcal{D}_u)$ to broaden coverage across diverse backgrounds, and Stage 2 fine-tunes on $\mathcal{D}_\ell$ to re-anchor the model to clean expert boxes and calibrate false positives. In both stages we optimize the detection objective with our contrastive term. The main MAST+ST results use two rounds of this cycle. We show (Proposition 5, App. C) that this procedure progressively reduces the distributional gap to the target domain, and that confidence weighting provably reduces effective label noise (Proposition 6). See App. A for formal definitions and App. C for theoretical analysis.

4 Experiment Setup

Datasets and Splits. We train and evaluate models separately on two ecologically distinct domains.

Rainforest domain. We study sound detection on passive acoustic recordings from tropical rainforest in East Kalimantan, Indonesia, from 2017–2019. Our primary labeled dataset consists of 24 hours recorded on July 10, 2018 at one site. Audio is resampled to 16 kHz and segmented into 10.24 s clips; each clip is converted to a 128×1024 log-mel spectrogram (App. D.1). A single expert annotated time–frequency boxes for 131 sonotypes (distinct sound types), consulting additional specialists on uncertain cases. We collapse all sonotypes into a single foreground category and evaluate binary detection. We evaluate in two regimes: in-domain, where we do a 70/15/15 train/val/test split of the labeled site/day, and out-of-distribution (OOD), where recordings come from different days and different sites within the same landscape, introducing covariate shift in background conditions, species activity, and sensor characteristics. A large pool of unlabeled rainforest audio of $\sim$661.7 h from multiple sites is used for masked audio pretraining and self-training. See App. D.2 for detailed statistics.

Bird domain. To assess the framework in a different ecological setting, we additionally evaluate on the BIRDeep dataset [36], which contains 641 recordings of bird vocalizations from 9 autonomous recording sites across 4 habitat types: low shrubland, high shrubland, ecotone, and marshland, in Doñana National Park, Spain. Annotations cover 38 bird species with time–frequency bounding boxes; we again collapse to binary detection. For masked audio pretraining, we combine BIRDeep unlabeled audio with BirdSet XCM [50], a large-scale bird audio collection of $\sim$900K spectrogram chunks. We use seven sites for the in-domain train/val/test split and construct temporal-OOD and site-OOD evaluation splits. See App. D.2 for dataset details.

Baselines. Recent work has applied object detectors to spectrograms for dense time–frequency detection in marine mammals [19], bird song [49, 68], and general soundscapes [14], but these rely on fully supervised detectors trained from scratch. We compare against representative architectures, all adapted to mel-spectrograms and trained on the same labeled set:

- **Faster R-CNN [52]**: A two-stage detector with a ResNet-50 backbone and FPN, adapted to log-mel spectrogram. We use class-agnostic detection with standard region proposal + RoIAlign.
- **FCOS [63]**: An anchor-free one-stage detector with a ResNet-50 + FPN backbone, predicting per-location classification, centerness, and box regression on the spectrogram lattice.
- **YOLO26 [56]**: A real-time one-stage detector. We use the YOLO26x architecture, adapted to log-mel spectrogram inputs.
- **DETR [10]**: A transformer-based end-to-end detector using set prediction with bipartite matching, adapted to spectrogram inputs. This baseline evaluates whether a query-based detection paradigm transfers to detection in low-label soundscapes.
- **AudioMAE [21]**: A transfer-learning baseline that initializes the backbone with a publicly available AudioMAE checkpoint pretrained on AudioSet, then fine-tunes the detector on our labeled split. This isolates the effect of generic large-scale audio pretraining versus in-domain self-supervised pretraining used in our method.

Training Details. All models use the same log-mel spectrogram processing and dataset-wide normalization. We train with AdamW optimizer and a warmup–cosine learning-rate schedule.

Table 1: **In-domain detection performance** on two domains: Rainforest (tropical soundscapes) and BIRDeep (European bird vocalizations [36]). *Higher is better.*

	Rainforest					BIRDeep				
Method	Prec.	Rec.	F1	mAP	mIoU	Prec.	Rec.	F1	mAP	mIoU
Faster R-CNN [52]	0.6832	**0.8390**	0.7226	**0.7715**	0.8590	0.3716	0.2061	0.2651	0.1537	0.6587
FCOS [63]	**0.9708**	0.6601	0.7858	0.6334	0.9290	0.7571	0.8091	**0.7822**	0.6650	0.6849
YOLO26 [56]	0.8659	0.3306	0.4309	0.3475	0.8102	0.7897	0.7737	0.7816	0.6183	0.7181
DETR [10]	0.5905	0.7839	0.6501	0.7422	0.8400	0.3256	0.7525	0.4545	0.3723	0.6024
AudioMAE [21]	0.9454	0.6903	0.7775	0.6736	0.9352	**0.7951**	0.6859	0.7364	0.6025	0.6756
MAST (ours)	0.9535	0.7021	**0.7916**	0.6853	0.9387	0.7824	0.7010	0.7395	0.6011	0.6822
MAST + ST (ours)	0.9548	0.6277	0.7338	0.6277	**0.9457**	0.6709	**0.9061**	0.7709	**0.7783**	**0.7339**

Table 2: **Out-of-distribution robustness** on Rainforest and BIRDeep. Temporal OOD evaluates on held-out dates; the BIRDeep temporal split also includes the held-out site AM4. Cross-site OOD evaluates on sites held out from labeled training. *Higher is better.*

	Rainforest					BIRDeep				
Method	Prec.	Rec.	F1	mAP	mIoU	Prec.	Rec.	F1	mAP	mIoU
Temporal OOD										
Faster R-CNN [52]	0.0400	0.2479	0.0689	0.0168	0.4259	0.3199	0.1020	0.1547	0.0909	0.5559
FCOS [63]	0.5072	0.1029	0.1711	0.1145	0.6429	0.7269	0.2885	0.4130	0.2116	0.6510
YOLO26 [56]	**0.9459**	0.0926	0.1688	0.0909	**0.7603**	0.7559	0.2040	0.3212	0.1520	**0.6767**
DETR [10]	0.1463	0.1800	0.1450	0.1316	0.5237	0.1923	**0.6521**	0.2971	0.3585	0.5528
AudioMAE [21]	0.8178	0.2085	0.3323	0.1636	0.6382	**0.7959**	0.3660	0.5014	0.2983	0.6328
MAST (ours)	0.7810	0.2968	0.4301	0.2577	0.6859	0.7618	0.4808	0.5895	0.3838	0.6083
MAST + ST (ours)	0.7120	**0.4603**	**0.5591**	**0.3340**	0.6292	0.6755	0.6247	**0.6491**	**0.4280**	0.5903
Cross-site OOD										
Faster R-CNN [52]	0.0497	0.2341	0.0820	0.0281	0.4223	0.8824	0.0744	0.1372	0.0909	0.5863
FCOS [63]	0.5848	0.0892	0.1548	0.0909	0.6513	**0.9725**	0.2852	0.4410	0.2592	0.6566
YOLO26 [56]	**0.9910**	0.0849	0.1564	0.0909	**0.7484**	0.8889	0.1934	0.3177	0.1675	**0.7010**
DETR [10]	0.1269	0.1758	0.1338	0.1130	0.5607	0.1334	**0.6621**	0.2220	0.3032	0.5554
AudioMAE [21]	0.8028	0.2113	0.3346	0.1742	0.6677	0.7266	0.4284	0.5390	0.3217	0.6072
MAST (ours)	0.7562	0.3169	0.4466	0.2573	0.6824	0.6476	0.5059	0.5680	0.3803	0.6001
MAST + ST (ours)	0.6978	**0.4838**	**0.5715**	**0.3921**	0.6420	0.6522	0.6438	**0.6434**	**0.4414**	0.5943

For MAST detectors, we use component-wise learning rates (smaller for the backbone, larger for adapters/FPN/head) and unfreeze only the top encoder blocks during fine-tuning. We also apply lightweight spectrogram augmentations. Unless stated otherwise, we match training budgets and augmentation pipelines across methods. See App. D.5, D.6 for details.

Evaluation. We evaluate binary time–frequency detection of "any animal sound," reporting precision, recall, F1, mean average precision (mAP), and mean intersection-over-union (mIoU) at IoU ≥ 0.5. We evaluate in two regimes: in-domain with held-out chunks from the labeled sites and dates, and OOD on different days or sites. Confidence thresholds are tuned on the in-domain validation split. See App. D.7 for full details.

5 Results

In-domain performance. Table 1 reports in-domain results. On rainforest, MAST achieves the best F1 of 0.792, with a mean IoU of 0.939, demonstrating that self-supervised pretraining and audio-aware adaptation do not sacrifice in-domain accuracy despite being designed primarily for robustness. On BIRDeep, FCOS and YOLO26 lead in F1 at 0.782, while MAST reaches 0.740. Self-training closes this gap: MAST+ST reaches 0.771 F1 and surpasses all baselines in mAP at 0.778 vs. 0.665 for FCOS, in recall at 0.906, and in mean IoU at 0.734. Crucially, the remaining in-domain difference inverts under distribution shift, where MAST provides the largest gains (Table 2), highlighting that in-domain metrics alone are insufficient for field deployment.

Out-of-distribution robustness. Table 2 reports OOD results under temporal and cross-site shift. Across both domains, detectors trained from scratch degrade sharply. For example, FCOS drops from 0.786 to 0.155 cross-site F1 on rainforest, a $5\times$ collapse. In contrast, MAST achieves the strongest

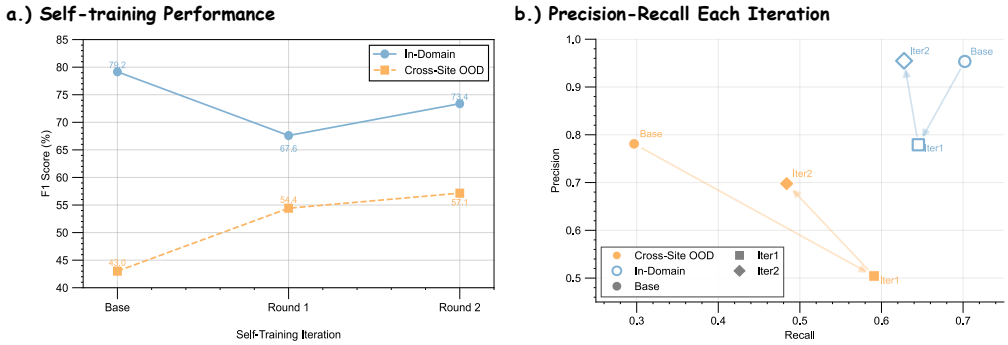


Figure 4: Effect of iterative self-training on F1 for rainforest domain through Round 2. Round 0 is the supervised MAST detector; Rounds 1–2 use the two-stage curriculum. Cross-site OOD F1 rises from 0.4466 to 0.5442 and 0.5715, while in-domain F1 drops from 0.7916 to 0.6760 and partially recovers to 0.7338. The Round 3 extension is reported in Table 14.

Table 3: **Pipeline ablation on cross-site OOD detection on rainforest domain.** Starting from the full MAST model, we remove components cumulatively, so each removal also retains the removals above it. We also report an AudioSet-pretrained variant and the FCOS baseline for reference. *Higher is better*.

Variant	Precision	Recall	F1	mAP	Mean IoU
MAST (full)	0.7562	0.3169	**0.4466**	**0.2573**	**0.6824**
– Contrastive loss	0.6536	0.3327	0.4410	0.2434	0.6555
– Audio-aware adapter	0.6380	0.2397	0.3484	0.2237	0.6607
Replace with AudioSet pretraining	**0.8028**	0.2113	0.3346	0.1742	0.6677
– All pretraining (from scratch)	0.3607	**0.3666**	0.3636	0.1840	0.5588
FCOS baseline	0.5848	0.0892	0.1548	0.0909	0.6513

overall F1 and mAP. On the rainforest domain, MAST raises cross-site F1 from 0.335 to 0.447 and temporal F1 from 0.332 to 0.430 over the AudioMAE baseline. On BIRDeep, MAST improves temporal F1 from 0.501 to 0.590 and cross-site F1 from 0.539 to 0.568 over AudioMAE, confirming that domain-matched pretraining is more transferable than generic AudioSet initialization. Bird-MAE also improves over AudioMAE, while MAST achieves higher F1 in all four OOD settings (App. E.1). Notably, most baselines exhibit a sharp precision–recall imbalance under shift: FCOS and YOLO26 retain high precision but collapse in recall, while DETR and Faster R-CNN maintain recall at the cost of precision. MAST is the only method that sustains both, yielding the highest F1 across conditions. The consistent pattern across both ecologically distinct domains, with modest in-domain gaps but substantial OOD improvements, confirms that MAST's primary benefit is robustness to distribution shift, the key deployment challenge in real-world biodiversity monitoring. Self-training further boosts rainforest OOD F1 to 0.559 temporal and 0.572 cross-site, and BIRDeep OOD F1 to 0.649 temporal and 0.643 cross-site.

Effect of self-training. Figure 4 shows that rainforest cross-site OOD F1 increases from 0.447 at Round 0 to 0.544 after one self-training round and to 0.572 after two rounds, a +0.13 absolute gain, while in-domain F1 dips after Round 1 and partially recovers after Round 2. The in-domain dip is expected: pseudo-label training broadens the decision boundary to cover unseen acoustic conditions, temporarily trading in-domain precision for OOD coverage, while the Stage 2 expert refinement partially corrects this drift. The first round provides the largest jump, suggesting that pseudo-labeling mainly helps by exposing the detector to diverse background conditions absent from the labeled site/day. On BIRDeep, self-training similarly improves OOD F1 from 0.568 to 0.643 cross-site and from 0.590 to 0.649 temporal, with especially large mAP gains from 0.380 to 0.441 cross-site. An additional third round raises rainforest cross-site F1 to 0.5813, a further gain of 0.0098 over Round 2 (App. E.3). These results indicate diminishing returns, with the first one to two rounds contributing most of the observed improvement.

Table 4: **Pretraining × adapter ablation on rainforest cross-site OOD.** We vary pretraining and the audio-aware adapter independently before self-training. *Higher is better.*

Pretraining	Adapter	F1	mAP
Domain-matched (MAST)	Yes	**0.4466**	**0.2573**
From scratch	Yes	0.3902	0.2370
Domain-matched	No	0.3484	0.2237
From scratch	No	0.3636	0.1840

Table 5: **Ablation of the self-training curriculum on cross-site OOD detection on rainforest domain.** We compare the MAST model against Stage 1-only, Stage 2-only, a one-stage union of expert-labeled and pseudo-labeled data, and the full two-stage curriculum. *Higher is better.*

Variant	Precision	Recall	F1	mAP	Mean IoU
Base (no ST)	0.7562	0.3169	0.4466	0.2573	0.6824
Stage 1 only	0.9714	0.1201	0.2138	0.1731	0.7439
Stage 2 only	**0.9971**	0.1046	0.1893	0.1789	**0.7644**
One stage union	0.1572	**0.7341**	0.2590	**0.3894**	0.5525
Stage 1 + Stage 2 (Full)	0.5044	0.5909	**0.5442**	0.3864	0.6067

Ablation studies. Table 3 decomposes MAST's gains under cross-site OOD shift by progressively removing components from the full model before self-training. Removing the contrastive loss drops precision sharply (0.756→0.654) while recall rises slightly, leaving F1 nearly unchanged; the loss thus acts primarily as a confidence calibrator that sharpens the decision boundary, producing higher-quality pseudo-labels for self-training. Removing the adapter further reduces F1 from 0.441 to 0.348, demonstrating the importance of resolving the resolution bottleneck (Sec. 3.3). Replacing domain-matched pretraining with generic AudioSet pretraining yields lower mAP, 0.174 vs. 0.224, confirming the value of in-domain representations. Removing all pretraining entirely still outperforms the CNN baseline in F1, 0.364 vs. 0.155, suggesting that the ViT's self-attention can model long-range spectro-temporal dependencies that aid generalization under background shift, even without pretrained weights. See App. E.5 for additional adapter comparisons.

The cumulative removals in Table 3 do not separate the effects of pretraining and the adapter. Table 4 varies these two factors independently. With the adapter present, pretraining raises F1 from 0.3902 to 0.4466 and mAP from 0.2370 to 0.2573. Without the adapter, pretraining improves mAP but slightly lowers F1, showing that its benefit depends on the detection architecture and metric. Increasing the pretraining pool from 25% to 100% further improves OOD F1 with little change in in-domain F1 (App. E.2).

Table 5 ablates the self-training curriculum. Stage 1 or Stage 2 alone both collapse recall despite high precision, yielding poor F1, demonstrating that naive single-stage pseudo-label training is insufficient. A one-stage union achieves high recall and mAP but precision drops to 0.157, indicating that mixing noisy pseudo supervision with expert labels without calibration leads to degenerate behavior. Only the full two-stage curriculum achieves the best F1 of 0.544 by balancing coverage and precision: Stage 1 broadens coverage under shift, while Stage 2 re-anchors the detector to clean expert annotations. This validates the curriculum design as essential rather than incidental to self-training success.

Retrieval-based event characterization. A practical benefit of MAST's design is that the pretrained encoder naturally provides meaningful box-level embeddings: each detected event can be represented by pooling the corresponding time–frequency region from the feature map. This enables a retrieval-based workflow where detected sounds are matched to annotated examples via nearest-neighbor lookup for zero-shot classification, or grouped by acoustic similarity to discover candidate novel sounds for expert review. On oracle ground-truth boxes, retrieval achieves 72.3% Top-1 accuracy over 131 sonotypes; on high-confidence predicted boxes with IoU ≥ 0.5, Top-1 reaches 96.1%, indicating that detections align with acoustically prototypical events. This allows ecologists to perform species-level analysis directly on detector outputs without retraining a multi-class model. Full results are in App. E.7.

Label budget and sampling strategy. A practical question for bioacousticians deploying a new detector is: *how much annotation effort is needed, and how should it be allocated?* We study this by training the FCOS baseline on varying numbers of labeled chunks under three sampling strategies. Both random and time-balanced sampling substantially outperform sequential labeling, especially

at low budgets where the gap reaches +0.13 F1 at $N=1000$ chunks, because sequential annotation under-represents acoustic conditions outside the labeled time window. Time-balanced sampling performs slightly better than random at most budgets, and as the budget grows all strategies converge. Since sequential labeling is the most natural default, where annotators typically start at the beginning of a recording and label forward in time, these results serve as a practical warning: practitioners should prioritize temporal diversity over volume when annotation resources are limited. Full results and protocol details are in App. E.8.

Stratified performance and failure analysis. We examine self-training gains by SNR, recording site, sonotype rarity, event size, and frequency band (App. E.4). F1 improves in every reported SNR tertile, and recall increases across all reported rarity groups; on rainforest cross-site OOD, rare-sonotype recall rises from 0.225 to 0.410. Recall also improves across rainforest frequency bands, but the smallest third of events remains difficult, with recall increasing from only 0.004 to 0.169. Site-level gains are uneven: all reported rainforest sites and BIRDeep AM4 improve, whereas F1 at the BIRDeep marshland site AM15 falls from 0.444 to 0.417. Thus, gains extend to low-SNR and rare events but are not uniform across sites. False-positive analysis identifies localization errors and duplicate detections as the dominant error categories.

6 Conclusion

We introduced MAST, a label-efficient framework for biodiversity sound detection that combines masked-audio pretraining, audio-aware detection adaptation, and iterative self-training. With models trained separately in two ecologically distinct domains, tropical rainforest soundscapes and Mediterranean bird vocalizations, MAST substantially outperforms supervised baselines and generic AudioSet transfer under both temporal and cross-site distribution shift. In the rainforest domain, these gains are achieved using expert labels from only a single annotated day at one site. Domain-matched self-supervised pretraining is more transferable than large-scale generic pretraining in our evaluations, the audio-aware adapter addresses a critical resolution bottleneck for narrow-band vocalizations, and the two-stage self-training curriculum yields further OOD gains without additional expert annotation. Increasing the unlabeled pretraining pool primarily improves OOD performance, highlighting the value of broader acoustic coverage for deployment beyond the labeled recording conditions. Together, these results demonstrate a scalable path from large unlabeled acoustic archives to robust time–frequency detectors under realistic label-limited conditions, supporting biodiversity assessment tasks currently bottlenecked by manual analysis, such as tracking changes in vocal activity across habitats, flagging acoustically novel events for taxonomic review, and scaling soundscape monitoring to landscape-level surveys.

Limitations and future work. MAST currently performs binary detection, so species identification still requires a separate step. Self-training depends on the seed detector's quality: gains diminish over successive rounds, and performance can decline at individual sites, as observed at AM15 (App. E.4). The 16 kHz front end excludes acoustic content above 8 kHz. Rainforest annotations were produced by one expert without inter-annotator agreement measurements, and the two-second separation rule yields different annotation granularities across sonotypes. Training also requires substantial computation: approximately 74 GPU-hours for pretraining, 22 for detector fine-tuning, and 95 per self-training round on a single A100 (App. D.5). Future work should extend detection to higher frequencies and multiple classes, with annotations from multiple experts and evaluation across more habitats. Ecological use also requires care with location data, since detections could facilitate poaching or disturbance of sensitive species. Precise locations should therefore be withheld or coarsened, and access to sensitive metadata restricted.

Acknowledgments and Disclosure of Funding

This work was partially supported by NSF CAREER Award #2239479 to DPA. This work was partially supported by the Prince Albert II of Monaco Foundation through grant 3386 to ZB. The authors declare no competing interests.

References

[1] Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. *Advances in neural information processing systems*, 33:12449–12460, 2020.

[2] Zachary Baker, Reece S. Shuttleworth, Daniela Rus, Antonio Torralba, Jacob Andreas, and Pratyusha Sharma. AudioSAM: Automated annotation of bioacoustic soundscapes in the wild. In *NeurIPS Workshop on AI for Non-Human Animal Communication*, 2025. URL https://openreview.net/forum?id=CyzaJbK0eB.

[3] David J Barker, Christopher Herrera, and Mark O West. Automated detection of 50-khz ultrasonic vocalizations using template matching in xbat. *Journal of neuroscience methods*, 236: 68–75, 2014.

[4] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1): 151–175, 2010.

[5] BioDCASE. BioDCASE challenge 2025. Official challenge website, 2025. URL https://biodcase.github.io/challenge2025/index.

[6] Tom Bradfer-Lawrence, Camille Desjonqueres, Alice Eldridge, Alison Johnston, and Oliver Metcalf. Using acoustic indices in ecology: Guidance on study design, analyses and interpretation. *Methods in Ecology and Evolution*, 14(9):2192–2204, 2023.

[7] Tom Bradfer-Lawrence, Brad Duthie, Carlos Abrahams, Matyáš Adam, Ross J Barnett, Amy Beeston, Jennifer Darby, Benedict Dell, Nick Gardner, Amandine Gasc, et al. The acoustic index user's guide: A practical manual for defining, generating and understanding current and future acoustic indices. *Methods in Ecology and Evolution*, 16(6):1040–1050, 2025.

[8] Zuzana Burivalova, Purnomo, Bambang Wahyudi, Timothy M Boucher, Peter Ellis, Anthony Truskinger, Michael Towsey, Paul Roe, Delon Marthinus, Bronson Griscom, et al. Using soundscapes to investigate homogenization of tropical forest diversity in selectively logged forests. *Journal of Applied Ecology*, 56(11):2493–2504, 2019.

[9] Rachel T Buxton, Megan F McKenna, Mary Clapp, Erik Meyer, Erik Stabenau, Lisa M Angeloni, Kevin Crooks, and George Wittemyer. Efficacy of extracting indices from large-scale acoustic recordings to monitor biodiversity. *Conservation Biology*, 32(5):1174–1184, 2018.

[10] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

[11] Mustafa Chasmai, Alexander Shepard, Subhransu Maji, and Grant Van Horn. The iNaturalist sounds dataset. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-4213. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/ef3713b8d72266e803f9346088fed92d-Abstract-Datasets_and_Benchmarks_Track.html.

[12] Sanyuan Chen, Yu Wu, Chengyi Wang, Shujie Liu, Daniel Tompkins, Zhuo Chen, and Furu Wei. Beats: Audio pre-training with acoustic tokenizers. *arXiv preprint arXiv:2212.09058*, 2022.

[13] Anna F Cord, Kevin Darras, Ryo Ogawa, Luc Barbaro, Charlotte Gerling, Maria Kernecker, Nonka Markova-Nenova, Gabriela Rodriguez-Barrera, Felix Zichner, and Frank Wätzold. Leveraging passive acoustic monitoring for result-based agri-environmental schemes: Opportunities, challenges and next steps. *Biological Conservation*, 305:111042, 2025.

[14] Janek Ebbers, François G Germain, Gordon Wichern, and Jonathan Le Roux. Sound event bounding boxes. *arXiv preprint arXiv:2406.04212*, 2024.

[15] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. Clap learning audio concepts from natural language supervision. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[16] Spencer Frei, Difan Zou, Zixiang Chen, and Quanquan Gu. Self-training converts weak learners to strong learners in mixture models. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 8003–8021. PMLR, 28–30 Mar 2022. URL https://proceedings.mlr.press/v151/frei22a.html.

[17] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 776–780. IEEE, 2017.

[18] Masato Hagiwara, Benjamin Hoffman, Jen-Yu Liu, Maddie Cusimano, Felix Effenberger, and Katie Zacarian. Beans: The benchmark of animal sounds. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023.

[19] Quentin Hamard, Minh-Tan Pham, Dorian Cazau, and Karine Heerah. A deep learning model for detecting and classifying multiple marine mammal species from passive acoustic data. *Ecological Informatics*, 84:102906, 2024.

[20] Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM transactions on audio, speech, and language processing*, 29:3451–3460, 2021.

[21] Po-Yao Huang, Hu Xu, Juncheng Li, Alexei Baevski, Michael Auli, Wojciech Galuba, Florian Metze, and Christoph Feichtenhofer. Masked autoencoders that listen. *Advances in Neural Information Processing Systems*, 35:28708–28720, 2022.

[22] Olaf Jahn, Todor D Ganchev, Marinez I Marques, and Karl-L Schuchmann. Automated sound recognition provides insights into the behavioral ecology of a tropical bird. *PloS one*, 12(1): e0169041, 2017.

[23] Stefan Kahl, Connor M Wood, Maximilian Eibl, and Holger Klinck. Birdnet: A deep learning solution for avian diversity monitoring. *Ecological Informatics*, 61:101236, 2021.

[24] Chieh-Chi Kao, Weiran Wang, Ming Sun, and Chao Wang. R-crnn: Region-based convolutional recurrent neural network for audio event detection. *arXiv preprint arXiv:1808.06627*, 2018.

[25] Jonathan Katz, Sasha D Hafner, and Therese Donovan. Tools for automated acoustic monitoring within the r package monitor. *Bioacoustics*, 25(2):197–210, 2016.

[26] Arik Kershenbaum, Çağlar Akçay, Lakshmi Babu-Saheer, Alex Barnhill, Paul Best, Jules Cauzinille, Dena Clink, Angela Dassow, Emmanuel Dufourq, Jonathan Growcott, et al. Automatic detection for bioacoustic research: a practical guide from and for biologists and computer scientists. *Biological Reviews*, 100(2):620–646, 2025.

[27] Ji Won Kim, Sang Won Son, Yoonah Song, Hong Kook Kim, Il Hoon Song, and Jeong Eun Lim. Semi-supervsied learning-based sound event detection using freuqency dynamic convolution with large kernel attention for dcase challenge 2023 task 4. *arXiv preprint arXiv:2306.06461*, 2023.

[28] Qiuqiang Kong, Yong Xu, Wenwu Wang, and Mark D Plumbley. Sound event detection of weakly labelled data with cnn-transformer and automatic threshold optimization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2450–2460, 2020.

[29] Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In *International conference on machine learning*, pages 5468–5479. PMLR, 2020.

[30] Marco Lassandro, Liam Bolitho, and David Newell. A systematic review of passive acoustic monitoring for anuran conservation. *Bioacoustics*, 34(5):579–628, 2025.

[31] Xiang Li, Wenhai Wang, Lijun Wu, Shuo Chen, Xiaolin Hu, Jun Li, Jinhui Tang, and Jian Yang. Generalized focal loss: Learning qualified and distributed bounding boxes for dense object detection. *Advances in neural information processing systems*, 33:21002–21012, 2020.

[32] Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *European conference on computer vision*, pages 280–296. Springer, 2022.

[33] Jinhua Liang, Ines Nolasco, Burooj Ghani, Huy Phan, Emmanouil Benetos, and Dan Stowell. Mind the domain gap: a systematic analysis on bioacoustic sound event detection. In *2024 32nd European Signal Processing Conference (EUSIPCO)*, pages 1257–1261. IEEE, 2024.

[34] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[35] Louis Mahon, Benjamin Hoffman, Logan James, Maddie Cusimano, Masato Hagiwara, Sarah C Woolley, Felix Effenberger, Sara Keen, Jen-Yu Liu, and Olivier Pietquin. Robust detection of overlapping bioacoustic sound events. *arXiv preprint arXiv:2503.02389*, 2025.

[36] Alba Márquez-Rodríguez, Miguel Ángel Mohedano-Munoz, Manuel J Marín-Jiménez, Eduardo Santamaría-García, Giulia Bastianelli, Pedro Jordano, and Irene Mendoza. A bird song detector for improving bird identification through deep learning: A case study from doñana. *Ecological Informatics*, 90:103254, 2025.

[37] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction, 2020. URL https://arxiv.org/abs/1802.03426.

[38] Nathan D Merchant, Kurt M Fristrup, Mark P Johnson, Peter L Tyack, Matthew J Witt, Philippe Blondel, and Susan E Parks. Measuring acoustic habitats. *Methods in Ecology and Evolution*, 6 (3):257–265, 2015.

[39] Marius Miron, David Robinson, Milad Alizadeh, Ellen Gilsenan-McMahon, Gagan Narula, Emmanuel Chemla, Maddie Cusimano, Felix Effenberger, Masato Hagiwara, Benjamin Hoffman, Sara Keen, Diane Kim, Jane Lawton, Jen-Yu Liu, Aza Raskin, Olivier Pietquin, and Matthieu Geist. AVEX: What matters for animal vocalization encoding. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=MFuM9KAEYc.

[40] Ilyass Moummad, Nicolas Farrugia, and Romain Serizel. Self-supervised learning for few-shot bird sound classification. In *2024 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*, pages 600–604. IEEE, 2024.

[41] Thomas Napier, Euijoon Ahn, Slade Allen-Ankins, Lin Schwarzkopf, and Ickjai Lee. Advancements in preprocessing, detection and classification techniques for ecoacoustic data: a comprehensive review for large-scale passive acoustic monitoring. *Expert Systems with Applications*, 252:124220, 2024.

[42] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

[43] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. Byol for audio: Exploring pre-trained general-purpose audio representations. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 31:137–151, 2022.

[44] Daisuke Niizumi, Daiki Takeuchi, Yasunori Ohishi, Noboru Harada, and Kunio Kashino. Masked spectrogram modeling using masked autoencoders for learning general-purpose audio representation. In *HEAR: Holistic Evaluation of Audio Representations*, pages 1–24. PMLR, 2022.

[45] Daniel S Park, William Chan, Yu Zhang, Chung-Cheng Chiu, Barret Zoph, Ekin D Cubuk, and Quoc V Le. Specaugment: A simple data augmentation method for automatic speech recognition. *arXiv preprint arXiv:1904.08779*, 2019.

[46] Sangwook Park, Ashwin Bellur, David K Han, and Mounya Elhilali. Self-training for sound event detection in audio mixtures. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 341–345. IEEE, 2021.

[47] Sangwook Park, David K Han, and Mounya Elhilali. Cross-referencing self-training network for sound event detection in audio mixtures. *IEEE transactions on multimedia*, 25:4573–4585, 2022.

[48] Cristian Pérez-Granados, Jon Morant, Kevin F. A. Darras, Oscar H. Marín-Gómez, Irene Mendoza, Miguel A. Muñoz-Mohedano, Eduardo Santamaría-García, et al. WABAD: A world annotated bird acoustic dataset for passive acoustic monitoring. *Ecology*, 107(2):e70317, 2026. doi: 10.1002/ecy.70317. URL https://doi.org/10.1002/ecy.70317.

[49] Phuong Pham, Juncheng Li, Joseph Szurley, and Samarjit Das. Eventness: Object detection on spectrograms for temporal localization of audio events. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2491–2495. IEEE, 2018.

[50] Lukas Rauch, Raphael Schwinger, Moritz Wirth, René Heinrich, Denis Huseljic, Marek Herde, Jonas Lange, Stefan Kahl, Bernhard Sick, Sven Tomforde, et al. Birdset: A large-scale dataset for audio classification in avian bioacoustics. *arXiv preprint arXiv:2403.10380*, 2024.

[51] Lukas Rauch, René Heinrich, Ilyass Moummad, Alexis Joly, Bernhard Sick, and Christoph Scholz. Can masked autoencoders also listen to birds? *arXiv preprint arXiv:2504.12880*, 2025.

[52] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1137–1149, 2016.

[53] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666, 2019.

[54] David Robinson, Marius Miron, Masato Hagiwara, Benno Weck, Sara Keen, Milad Alizadeh, Gagan Narula, Matthieu Geist, and Olivier Pietquin. Naturelm-audio: An audio-language foundation model for bioacoustics. *arXiv preprint arXiv:2411.07186*, 2024.

[55] Samuel RP-J Ross, Darren P O'Connell, Jessica L Deichmann, Camille Desjonquères, Amandine Gasc, Jennifer N Phillips, Sarab S Sethi, Connor M Wood, and Zuzana Burivalova. Passive acoustic monitoring provides a fresh perspective on fundamental ecological questions. *Functional Ecology*, 37(4):959–975, 2023.

[56] Ranjan Sapkota, Rahul Harsha Cheppally, Ajay Sharda, and Manoj Karkee. Yolo26: key architectural enhancements and performance benchmarking for real-time object detection. *arXiv preprint arXiv:2509.25164*, 2025.

[57] Romain Serizel, Nicolas Turpault, Hamid Eghbal-Zadeh, and Ankit Parag Shah. Large-scale weakly labeled semi-supervised sound event detection in domestic environments. *arXiv preprint arXiv:1807.10501*, 2018.

[58] Dan Stowell. Computational bioacoustics with deep learning: a review and roadmap. *PeerJ*, 10: e13152, 2022.

[59] Jérôme Sueur and Almo Farina. Ecoacoustics: the ecological investigation and interpretation of environmental sound. *Biosemiotics*, 8(3):493–502, 2015.

[60] Larissa Sayuri Moreira Sugai, Thiago Sanna Freire Silva, José Wagner Ribeiro Jr, and Diego Llusia. Terrestrial passive acoustic monitoring: review and perspectives. *BioScience*, 69(1): 15–25, 2019.

[61] Aaron Sun, Subhransu Maji, and Grant Van Horn. Merlin L48 spectrogram dataset. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-2251. URL https://proceedings.neurips.cc/paper_files/paper/2025/hash/60ea0211b38a3ccd7a241f523dc7cf63-Abstract-Datasets_and_Benchmarks_Track.html.

[62] Yuren Sun, Tatiana Midori Maeda, Claudia Solís-Lemus, Daniel Pimentel-Alarcón, and Zuzana Buřivalová. Classification of animal sounds in a hyperdiverse rainforest using convolutional neural networks with data augmentation. *Ecological Indicators*, 145:109621, 2022.

[63] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9627–9636, 2019.

[64] Michael Towsey, Birgit Planitz, Alfredo Nantes, Jason Wimmer, and Paul Roe. A toolbox for animal call recognition. *Bioacoustics*, 21(2):107–125, 2012.

[65] Joseph Turian, Jordie Shier, Humair Raj Khan, Bhiksha Raj, Björn W Schuller, Christian J Steinmetz, Colin Malloy, George Tzanetakis, Gissel Velarde, Kirk McNally, et al. Hear: Holistic evaluation of audio representations. In *NeurIPS 2021 Competitions and Demonstrations Track*, pages 125–145. PMLR, 2022.

[66] Bart Van Merriënboer, Jenny Hamer, Vincent Dumoulin, Eleni Triantafillou, and Tom Denton. Birds, bats and beyond: Evaluating generalization in bioacoustics models. *Frontiers in Bird Science*, 3:1369756, 2024.

[67] Bart van Merriënboer, Vincent Dumoulin, Jenny Hamer, Lauren Harrell, Andrea Burns, and Tom Denton. Perch 2.0: The bittern lesson for bioacoustics. *arXiv preprint arXiv:2508.04665*, 2025.

[68] Satvik Venkatesh, David Moffat, and Eduardo Reck Miranda. You only hear once: a yolo-like algorithm for audio segmentation and sound event detection. *Applied Sciences*, 12(7):3293, 2022.

[69] Yun Wang, Juncheng Li, and Florian Metze. A comparison of five multiple instance learning pooling functions for sound event detection with weak labeling. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 31–35. IEEE, 2019.

[70] Colin Wei, Kendrick Shen, Yining Chen, and Tengyu Ma. Theoretical analysis of self-training with deep networks on unlabeled data. *arXiv preprint arXiv:2010.03622*, 2020.

[71] Connor M Wood, Jacob Socolar, Stefan Kahl, M Zachariah Peery, Philip Chaon, Kevin Kelly, Robert A Koch, Sarah C Sawyer, and Holger Klinck. A scalable and transferable approach to combining emerging conservation technologies to identify biodiversity change after large disturbances. *Journal of Applied Ecology*, 61(4):797–808, 2024.

[72] Tianyi Xu, Xuan Ouyang, Binwei Yao, Shoua Xiong, Sara Misurelli, Maichou Lor, and Junjie Hu. Sita: Learning speaker-invariant and tone-aware speech representations for low-resource tonal languages. *arXiv preprint arXiv:2601.09050*, 2026.

[73] Shifeng Zhang, Cheng Chi, Yongqiang Yao, Zhen Lei, and Stan Z Li. Bridging the gap between anchor-based and anchor-free detection via adaptive training sample selection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9759–9768, 2020.

[74] Zhi Zhu and Yoshinao Sato. Sound event detection using time-frequency bounding boxes with a self-supervised audio spectrogram transformer. In *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, pages 150–154, 2025. URL https://dcase.community/documents/workshop2025/proceedings/DCASE2025Workshop_Zhu_54.pdf.

A Self-training Operator and Two-Stage Curriculum

Given an unlabeled chunk S, a detector produces a set of scored candidate boxes

$$\widehat{\mathcal{B}}(S) = \left\{ \left(\hat{b}_j, s_j \right) \right\}_{j=1}^{\widehat{N}}, \quad s_j \in [0,1].$$

We apply class-agnostic NMS with overlap threshold η, and enforce a validity predicate $v(\hat{b})$, such as minimum time span, minimum frequency span, and boundary constraints. With a confidence policy $q(\cdot)$, the pseudo-label operator is

$$\Pi_q(S) = \left\{ \hat{b} \in \mathrm{NMS}_\eta(\widehat{\mathcal{B}}(S)) : s(\hat{b}) \geq q(S) \wedge v(\hat{b}) \right\}.$$

Applying Π_q to the unlabeled pool yields a pseudo-labeled dataset

$$\Pi_q\left(\mathcal{D}_u\right) = \left\{ (S, \tilde{B}) : S \in \mathcal{D}_u, \tilde{B} = \Pi_q(S) \right\}.$$

In our experiments we set q on the labeled validation split to balance precision and coverage. Remaining noise is suppressed by confidence weighting during training (see Stage 1 below).

We then train student detectors using two stages:

Stage 1: pseudo pretraining. We train on $\Pi_q\left(\mathcal{D}_u\right)$ using the detection loss with the box-level contrastive loss. To reduce the impact of noisy pseudo labels, we weight each pseudo-labeled example by its average pseudo confidence:

$$\bar{g}(S) = \frac{1}{|\tilde{B}|} \sum_{\hat{b} \in \tilde{B}} s(\hat{b}),$$

and optimize

$$\mathcal{L}_{\text{stage1}} = \mathbb{E}_{(S,\tilde{B}) \sim \Pi_q(\mathcal{D}_u)} \left[\bar{g}(S) \mathcal{L}_{\text{det}}\left(S, \tilde{B}\right) + \lambda_{\text{con}} \mathcal{L}_{\text{con}}\left(S, \tilde{B}\right) \right]$$

Intuitively, Stage 1 increases recall and robustness by exposing the detector to diverse acoustic conditions that are absent from $\mathcal{D}_\ell$, while confidence weighting reduces confirmation bias from uncertain predictions.

Stage 2: expert refinement. We initialize from the Stage 1 student and fine-tune on the expert-labeled set:

$$\mathcal{L}_{\text{stage2}} = \mathbb{E}_{(S,B) \sim \mathcal{D}_\ell} \left[\mathcal{L}_{\text{det}}\left(S, B\right) + \lambda_{\text{con}} \mathcal{L}_{\text{con}}\left(S, B\right) \right].$$

Stage 2 re-anchors training to high-quality annotations, mitigating confirmation bias from pseudo labels and improving calibration under domain shift.

We repeat the cycle of pseudo-labeling $\rightarrow$ Stage 1 $\rightarrow$ Stage 2 for two rounds in the main experiments, and evaluate an additional third round on rainforest cross-site OOD in App. E.3.

B Theoretical Analysis of Adapter Design

We formalize well-known intuitions about the resolution bottleneck induced by patch-based vision transformers [32, 34], derive the optimality of asymmetric upsampling for our setting, and analyze the efficiency of anisotropic convolutions.

B.1 Resolution Bottleneck

With patch size p and spectrogram dimensions $T \times F$, the encoder produces a feature map with strides $s_t = p$ and $s_f = p$, yielding a token grid of size $(T/p) \times (F/p)$. A detector operating on this grid must express box boundaries at the granularity of these strides. Prior work has noted that coarse feature-map strides hurt small-object localization [34, 32], motivating multi-scale feature pyramids. Here we make this intuition precise for the spectrogram setting and show that quantization can severely degrade localization quality, particularly along the frequency axis.

Proposition 1 (Quantization-induced IoU degradation)**.** *Let $b^* = [t_1, f_1, t_2, f_2]$ be a ground-truth box with temporal extent $\delta_t = t_2 - t_1$ and frequency extent $\delta_f = f_2 - f_1$. Let $\hat{b}$ denote the tightest enclosing box whose corners lie on a grid with strides s_t and s_f. Then there exists a placement of b^* relative to the grid such that:*

$$\mathrm{IoU}(b^*, \hat{b}) \leq \frac{\delta_t \cdot \delta_f}{(\delta_t + s_t)(\delta_f + s_f)}.$$

Proof. We construct a worst-case placement as follows. Choose $t_1 = (m{+}1)s_t - \epsilon$ for integer m and small $\epsilon > 0$, so $\lfloor t_1/s_t \rfloor = m$. Write $\delta_t = k_t s_t + r_t$ with $0 \leq r_t < s_t$. If $r_t = 0$, then $t_2 = (m{+}k_t{+}1)s_t - \epsilon$, so $\lceil t_2/s_t \rceil = m + k_t + 1$ and $\hat{\delta}_t = (k_t{+}1)s_t = \delta_t + s_t$. If $r_t > 0$ (choosing $\epsilon < r_t$), then $t_2 = (m{+}k_t{+}1)s_t + r_t - \epsilon$ with $r_t - \epsilon \in (0, s_t)$, so $\lceil t_2/s_t \rceil = m + k_t + 2$ and $\hat{\delta}_t = (k_t{+}2)s_t = \delta_t + 2s_t - r_t > \delta_t + s_t$. In both cases $\hat{\delta}_t \geq \delta_t + s_t$. The same construction applies in frequency. Since $b^* \subseteq \hat{b}$, $\mathrm{IoU}(b^*, \hat{b}) = \delta_t \delta_f / (\hat{\delta}_t \hat{\delta}_f) \leq \delta_t \delta_f / [(\delta_t + s_t)(\delta_f + s_f)]$. □

Corollary 2 (Narrow-band events can fall below detection threshold)**.** *Standard evaluation counts a prediction as a true positive only if* $\mathrm{IoU} \geq 0.5$*. With encoder grid strides $s_t = s_f = p = 16$ and a narrow-band vocalization with $\delta_f \leq p$ (frequency extent ≤ 1 token), the IoU bound from Proposition 1 is monotonically increasing in δ_f, so it is maximized at $\delta_f = p$, giving:*

$$\mathrm{IoU}(b^*, \hat{b}) \leq \frac{\delta_t \cdot p}{(\delta_t + p)(p + p)} = \frac{\delta_t}{2(\delta_t + p)} < \frac{1}{2},$$

for all finite δ_t. By Proposition 1, there exists a placement achieving this bound, hence ***there exist placements where the tightest enclosing grid-aligned box fails to reach IoU*** ≥ 0.5 *for any narrow-band event, regardless of its temporal extent.*

In our setting, each token spans $\Delta t_{\mathrm{tok}} = p \cdot h = 160\,\mathrm{ms}$ in time, where $h = 10\,\mathrm{ms}$ is the STFT hop, and $\Delta f_{\mathrm{tok}} = p \cdot \Delta f_{\mathrm{mel}} \approx 1\,\mathrm{kHz}$ in frequency. Many animal vocalizations such as bird harmonics, typically with 200–500 Hz bandwidth, and insect stridulations, typically with $< 1\,\mathrm{kHz}$, occupy at most one token in frequency while spanning many tokens in time, falling precisely into the regime of Corollary 2.

B.2 Asymmetric Upsampling: Theoretical Analysis

Our adapter applies learnable upsampling with asymmetric factors u_t in time and u_f in frequency, reducing the effective strides to $s'_t = p/u_t$ and $s'_f = p/u_f$. By the same worst-case analysis as Proposition 1:

$$\mathrm{IoU}(b^*, \hat{b}) \leq \frac{\delta_t \cdot \delta_f}{(\delta_t + p/u_t)\,(\delta_f + p/u_f)}.$$

Proposition 3 (Optimal upsampling allocation)**.** *Given a fixed upsampling budget $u_t \cdot u_f = U$ (total spatial expansion), the IoU bound in Proposition 1 is maximized when the upsampling factors are allocated proportionally to the stride-to-extent ratios:*

$$\frac{u_f}{u_t} = \frac{s_f/\delta_f}{s_t/\delta_t} = \frac{\delta_t}{\delta_f}.$$

That is, the axis with greater relative quantization error should receive more upsampling.

Proof. Since the numerator $\delta_t \delta_f$ is constant, maximizing $\mathrm{IoU}_{\mathrm{ub}}$ is equivalent to minimizing the denominator $D(u_t) = (\delta_t + p/u_t)(\delta_f + p u_t/U)$ after substituting $u_f = U/u_t$. Setting $D'(u_t) = 0$:

$$\frac{-p}{u_t^2}\left(\delta_f + \frac{p u_t}{U}\right) + \left(\delta_t + \frac{p}{u_t}\right)\frac{p}{U} = 0 \quad \Longrightarrow \quad \frac{\delta_f + p u_t/U}{u_t^2} = \frac{\delta_t + p/u_t}{U}.$$

Cross-multiplying and expanding: $U\delta_f + pu_t = u_t^2\delta_t + pu_t$. The pu_t terms cancel, giving $U\delta_f = u_t^2\delta_t$, so $u_t = \sqrt{U\delta_f/\delta_t}$ and $u_f = U/u_t = \sqrt{U\delta_t/\delta_f}$, hence $u_f/u_t = \delta_t/\delta_f$. This is a maximum since $D(u_t) \to \infty$ as $u_t \to 0^+$ or $u_t \to \infty$. □

For typical rainforest vocalizations with $\delta_t \gg \delta_f$ (temporally extended, frequency-narrow), this predicts $u_f > u_t$, which is exactly our design choice ($u_t = 2, u_f = 4$, ratio $u_f/u_t = 2$). With these factors, the adapted feature map has size 128×32 and effective strides of 8×4 pixels. For a narrow-band event with $\delta_f = p = 16$, the IoU upper bound improves from < 0.5 (Corollary 2) to $\delta_t \cdot 16/[(\delta_t + 8) \cdot 20]$, which exceeds 0.5 for $\delta_t \geq 14$ (events longer than ∼140 ms).

B.3 Anisotropic Convolution: Separability Analysis

Proposition 4 (Parameter efficiency of separable filtering)**.** *Let $\mathcal{F}_{\text{iso}} : \mathbb{R}^{C\times H\times W} \to \mathbb{R}^{C\times H\times W}$ be an isotropic convolution with kernel size $k \times k$, and $\mathcal{F}_{\text{sep}} = \mathcal{F}_{1\times k} \circ \mathcal{F}_{k\times 1}$ be the composition of two anisotropic convolutions. Then:*

(i) ***Parameter reduction:*** *$|\Theta_{\text{sep}}| = 2C^2k$ versus $|\Theta_{\text{iso}}| = C^2k^2$, a factor of $k/2$ savings.*

(ii) ***Receptive field preservation:*** *Both achieve effective receptive field $k \times k$ on the input.*

(iii) ***Rank constraint:*** *The effective spatial kernel of $\mathcal{F}_{\text{sep}}$ for each input–output channel pair has rank at most $\min(C, k)$, whereas a general $k \times k$ filter is unconstrained and can achieve full rank k. The composition processes time and frequency axes sequentially, imposing a structural separability that acts as beneficial inductive bias for bioacoustic events with approximately independent temporal and spectral structure.*

Proof. (i) The $k \times 1$ filter has C^2k parameters and the $1 \times k$ filter has C^2k parameters, totaling $2C^2k$. The ratio is $C^2k^2/(2C^2k) = k/2$. (ii) The $k \times 1$ filter sees k rows and 1 column; the subsequent $1 \times k$ filter sees 1 row and k columns of the intermediate map, so each output pixel depends on a $k \times k$ neighborhood. (iii) For a fixed channel pair $(c_{\text{in}}, c_{\text{out}})$, the effective $k \times k$ spatial kernel is $K[i, j] = \sum_{c_m=1}^{C} W_f[c_{\text{out}}, c_m, j] \cdot W_t[c_m, c_{\text{in}}, i]$, a sum of C rank-1 outer products. Hence $\text{rank}(K) \leq \min(C, k)$, whereas a general $k \times k$ filter is unconstrained and can achieve full rank k. More fundamentally, $\mathcal{F}_{\text{sep}}$ processes time and frequency axes sequentially rather than jointly, so it cannot model non-separable cross-axis interactions. For signals where temporal and spectral structure are approximately independent, as in harmonic animal vocalizations with smooth temporal envelopes, this separability constraint matches the signal structure and suppresses spurious cross-axis correlations. □

This analysis explains why the anisotropic adapter achieves the best F1 among adapter variants (Table 13): the separability constraint matches the physical structure of bioacoustic events, providing effective regularization without sacrificing expressiveness for the target signal class.

C Theoretical Analysis of Self-Training

Motivated by domain adaptation theory [4] and recent analyses of self-training under distribution shift [29, 70], we provide theoretical justification for MAST's two-stage self-training curriculum, confidence-weighted pseudo-label training, and the improvement conditions under iterative self-training. The following results adapt established frameworks to our specific setting of two-stage pseudo-label exploration and expert refinement for time–frequency detection.

C.1 Self-Training as Distributionally Robust Optimization

Let P_ℓ denote the labeled source distribution and P_u denote the unlabeled target distribution, which may include temporal and cross-site domain shift. The standard supervised objective minimizes $\mathbb{E}_{(S,B)\sim P_\ell}[\mathcal{L}(f_\theta(S), B)]$, but the deployed detector must perform well under P_u. Following the distributional robustness perspective of Ben-David et al. [4], we characterize our two-stage self-training as approximate DRO that progressively reduces the gap between training and deployment distributions.

Proposition 5 (Self-training reduces distributional gap)**.** *Let $f^{(0)}$ be a detector trained on P_ℓ, and let $\hat{P}_u^{(t)} = \Pi_q(P_u; f^{(t)})$ denote the pseudo-labeled distribution induced by applying detector $f^{(t)}$ to unlabeled data with confidence threshold q. Define the* effective training distribution *at round t of the two-stage curriculum as:*

$$P_{\text{eff}}^{(t)} = (1-\alpha)\hat{P}_u^{(t)} + \alpha\, P_\ell,$$

where $\alpha \in (0,1)$ reflects the relative contribution of Stage 2. Let $d_{\text{TV}}(\cdot,\cdot)$ denote total variation distance. If the pseudo-labeling operator has precision $\pi_q = \Pr[\hat{b} \text{ correct} \mid s(\hat{b}) \geq q]$ and the detector achieves coverage $\rho_q = \Pr_{b^* \sim P_u}[\exists \hat{b} : \text{IoU}(\hat{b}, b^*) \geq 0.5 \wedge s(\hat{b}) \geq q]$, then:

$$d_{\text{TV}}(P_{\text{eff}}^{(t)}, P_u) \leq (1-\alpha)(1-\rho_q\pi_q) + \alpha\, d_{\text{TV}}(P_\ell, P_u).$$

Proof. Since $\hat{P}_u^{(t)}$ and P_u share the same marginal on S (pseudo-labeling only modifies the annotation), their TV distance is determined by label disagreement. The pseudo-labeled distribution agrees with P_u on the fraction $\rho_q\pi_q$ of events that are both covered and correctly labeled. The remaining mass $(1-\rho_q\pi_q)$ may have incorrect or missing labels, contributing at most $(1-\rho_q\pi_q)$ to $d_{\text{TV}}(\hat{P}_u^{(t)}, P_u)$. By convexity of total variation:

$$\begin{aligned} d_{\text{TV}}(P_{\text{eff}}^{(t)}, P_u) &\leq (1-\alpha)\, d_{\text{TV}}(\hat{P}_u^{(t)}, P_u) + \alpha\, d_{\text{TV}}(P_\ell, P_u) \\ &\leq (1-\alpha)(1-\rho_q\pi_q) + \alpha\, d_{\text{TV}}(P_\ell, P_u). \end{aligned}$$

□

This result shows that as pseudo-label quality (π_q) and coverage (ρ_q) improve across self-training rounds, $P_{\text{eff}}^{(t)}$ converges toward P_u, reducing the effective domain gap. The two-stage design is crucial: Stage 1 improves ρ_q, the coverage via diverse pseudo-labeled data, while Stage 2 controls α, the re-anchoring to clean labels prevents drift.

C.2 Confidence Weighting as Importance-Weighted Risk Minimization

In Stage 1, we weight each pseudo-labeled sample by its average pseudo confidence $\bar{g}(S) = \frac{1}{|\tilde{B}|}\sum_{\hat{b}\in\tilde{B}} s(\hat{b})$. Building on the importance-weighted learning framework for noisy labels [42], we observe that this weighting can be interpreted as importance-weighted empirical risk minimization that provably reduces the effective noise rate.

Proposition 6 (Confidence weighting reduces effective label noise)**.** *Let $\epsilon(s)$ denote the label error probability for a pseudo-label with confidence s, and assume $\epsilon(s)$ is monotonically decreasing in s (higher confidence $\Rightarrow$ lower error rate). Given n pseudo-labeled samples $(S_i, \tilde{B}_i)_{i=1}^n$, the confidence-weighted empirical risk*

$$\hat{R}_w(\theta) = \frac{1}{Z}\sum_{i=1}^n \bar{g}(S_i)\, \mathcal{L}(f_\theta(S_i), \tilde{B}_i), \quad Z = \sum_{i=1}^n \bar{g}(S_i),$$

has effective noise rate

$$\bar{\epsilon}_w = \frac{\sum_i \bar{g}(S_i)\, \epsilon(\bar{g}(S_i))}{\sum_i \bar{g}(S_i)} \leq \frac{\sum_i \epsilon(\bar{g}(S_i))}{n} = \bar{\epsilon}_{\text{unif}},$$

where $\bar{\epsilon}_{\text{unif}}$ is the uniform (unweighted) noise rate.

Proof. By the assumption that $\epsilon(s)$ is decreasing, samples with high $\bar{g}$ (high weight) have low ϵ, and vice versa. Setting $a_i = \bar{g}(S_i)$ and $b_i = \epsilon(\bar{g}(S_i))$, the monotonicity of ϵ ensures that a_i and b_i are oppositely ordered (i.e., whenever $a_i \leq a_j$ we have $b_i \geq b_j$), which is exactly the condition required by the Chebyshev sum inequality:

$$n\sum_i a_i b_i \leq \left(\sum_i a_i\right)\left(\sum_i b_i\right).$$

Substituting back and dividing both sides by $n \sum_i \bar{g}(S_i)$:

$$\bar{\epsilon}_w = \frac{\sum_i \bar{g}(S_i)\, \epsilon(\bar{g}(S_i))}{\sum_i \bar{g}(S_i)} \leq \frac{\sum_i \epsilon(\bar{g}(S_i))}{n} = \bar{\epsilon}_{\text{unif}}.$$

□

The inequality is strict whenever the confidence scores are non-degenerate (not all identical), meaning confidence weighting always reduces effective noise compared to uniform training on the same pseudo-labeled data. This provides theoretical grounding for the empirical observation that confidence weighting improves OOD generalization in Stage 1.

C.3 Monotonic Improvement under Iterative Self-Training

Prior work has established convergence guarantees for iterative self-training under various distributional assumptions, including gradual domain shift [29], expansion conditions [70], and mixture-model structure [16]. We adapt these ideas to our two-stage curriculum and identify three sufficient conditions under which successive rounds produce non-degrading OOD performance.

Proposition 7 (Sufficient condition for monotonic OOD improvement)**.** *Let $f^{(t)}$ denote the detector after round t of self-training, and let $R_{\text{ood}}(f) = \mathbb{E}_{P_u}[\mathcal{L}(f(S), B)]$ denote the OOD risk. Assume the loss $\mathcal{L}$ is L-Lipschitz in the model parameters. Suppose:*

- *(i)* ***Improving pseudo-labels:*** *The precision-coverage product satisfies $\rho_q^{(t+1)}\pi_q^{(t+1)} \geq \rho_q^{(t)}\pi_q^{(t)}$, and the resulting OOD risk reduction from Stage 1 is at least $\Delta^{(t)} > 0$;*
- *(ii)* ***Bounded Stage 2 drift:*** *Stage 2 fine-tuning runs for T_2 steps with learning rate η_2 and gradient norm bounded by G, so the parameter drift satisfies $\|f_{\text{stage2}}^{(t+1)} - f_{\text{stage1}}^{(t+1)}\| \leq T_2\eta_2 G =: \delta_2$;*
- *(iii)* ***Bounded Stage 1 forgetting:*** *The gradual unfreezing schedule limits catastrophic forgetting such that $\|f_{\text{stage1}}^{(t+1)} - f^{(t)}\|_\infty \leq \beta$, where $\beta > 0$ is controlled by the backbone learning rate $\gamma\eta_{\text{base}}$ and the number of unfrozen blocks $k(e)$ (Remark 8).*

If $L\delta_2 \leq \Delta^{(t)}$, i.e., Stage 2 parameter drift is small enough that Lipschitz-induced risk increase does not exceed the Stage 1 gain, then $R_{\text{ood}}(f^{(t+1)}) \leq R_{\text{ood}}(f^{(t)})$.

Proof sketch. By Proposition 5, improving $\rho_q\pi_q$ (condition i) reduces $d_{\text{TV}}(P_{\text{eff}}^{(t+1)}, P_u)$. Since d_{TV} upper-bounds the $\mathcal{H}\Delta\mathcal{H}$-divergence of Ben-David et al. [4], this tightens their generalization bound $R_u(f) \leq R_s(f) + d_{\mathcal{H}\Delta\mathcal{H}}(P_s, P_u) + \lambda$, yielding a concrete OOD risk reduction $\Delta^{(t)} > 0$ proportional to the TV distance decrease: $R_{\text{ood}}(f_{\text{stage1}}^{(t+1)}) \leq R_{\text{ood}}(f^{(t)}) - \Delta^{(t)}$. Condition (iii) ensures that optimization remains within a β-ball of the previous iterate, so the ERM solution does not escape the basin where the tighter bound applies. For Stage 2, by Lipschitz continuity and condition (ii): $|R_{\text{ood}}(f_{\text{stage2}}^{(t+1)}) - R_{\text{ood}}(f_{\text{stage1}}^{(t+1)})| \leq L\delta_2$. Since $L\delta_2 \leq \Delta^{(t)}$, we have $R_{\text{ood}}(f^{(t+1)}) \leq R_{\text{ood}}(f_{\text{stage1}}^{(t+1)}) + L\delta_2 \leq R_{\text{ood}}(f^{(t)}) - \Delta^{(t)} + L\delta_2 \leq R_{\text{ood}}(f^{(t)})$. □

Empirical verification. Our design enforces these conditions in practice: (i) a better detector at each round produces higher-quality pseudo-labels, verified by increasing pseudo-label F1 across rounds; (ii) Stage 2 uses a low learning rate ($\eta_2 = 5{\times}10^{-5}$, backbone at $0.02\times$) for only $T_2{=}40$ epochs with gradient clipping at $G{=}1.0$, bounding parameter drift δ_2; gradient clipping locally enforces the Lipschitz condition by ensuring $\|\nabla\mathcal{L}\| \leq G$ at every step; additionally, OOD-based model selection provides an operational safeguard against excessive drift; (iii) gradual unfreezing with conservative schedules in Stage 1 (warmup $= 20$, interval $= 10$) and component-specific LR multipliers (backbone at $0.005\times$) bound parameter drift. See Table 10 for the full hyperparameter specification.

C.4 Gradual Unfreezing as Regularized Fine-Tuning

The asymmetry between Stage 1 with conservative unfreezing and Stage 2 with aggressive unfreezing has a natural interpretation through the lens of regularization theory.

Remark 8 (Unfreezing schedule controls effective model complexity)**.** *Consider a ViT encoder with L transformer blocks. At epoch e, let $k(e) \leq L$ blocks be unfrozen with learning rate $\eta_{\text{bb}} = \gamma\eta_{\text{base}}$ where $\gamma \ll 1$. The number of actively trained parameters is $d_{\text{eff}}(e) = d_{\text{head}} + d_{\text{adapter}} + k(e) \cdot d_{\text{block}}$, and the backbone's effective update magnitude scales with $\gamma k(e)$. Under the conservative Stage 1 schedule where $k(e)$ increases slowly with large warmup, the model first learns task-specific head and adapter parameters on noisy pseudo-labels without disturbing pretrained representations. Under the aggressive Stage 2 schedule where $k(e)$ increases rapidly, the full model adapts to clean expert labels.*

This staged complexity control prevents a key failure mode: if the full backbone is unfrozen on noisy pseudo-labeled data, the pretrained representations may overfit to systematic pseudo-label errors, a form of *confirmation bias*. By keeping the backbone frozen during early Stage 1 training, the detector head first calibrates its decision boundary before the backbone features are permitted to drift.

The component-specific learning rates further partition the optimization landscape: backbone blocks receive $0.005\times$ the base rate in Stage 1, to preserve general audio features, versus $0.05\times$ in Stage 2, to allow fine-grained adaptation to the expert-labeled distribution. This $10\times$ ratio between stages reflects the relative trustworthiness of pseudo vs. expert supervision.

D Detailed Experimental Setup

D.1 Data Processing

Audio preprocessing. All raw audio is resampled to 16 kHz mono. We segment each recording into 10.24 s chunks with 50% overlap using 5.12 s hop, yielding maximum coverage while ensuring every segment fits the MAST input size. Chunks shorter than 50% of the target duration are discarded, and those between 50–100% are zero-padded. Before feature extraction, each chunk is mean-centered to remove DC offset.

Spectrogram extraction. We compute log-mel spectrograms using Kaldi-compatible filterbanks via `torchaudio.compliance.kaldi.fbank` with the following parameters: 25 ms Hanning window, 10 ms frame shift, 128 mel bins, HTK-compatible mel scale, no energy feature, and no dither. This produces a 1024×128 (time $\times$ frequency) feature matrix per chunk, with frequency range $[20, 8000]$ Hz. These settings are identical to those used by AudioMAE [21], ensuring compatibility with the pretrained encoder. We apply dataset-wide z-score normalization during training: $(S - \mu)/(\sigma + \epsilon)$, where μ, σ are computed once over $\mathcal{D}_\ell \cup \mathcal{D}_u$ and reused for all training and evaluation. For BIRDeep, normalization statistics are computed separately over the combined BIRDeep + BirdSet XCM pool.

Bounding box coordinate conversion. Expert annotations are provided in (Hz, seconds). We convert to spectrogram coordinates as follows. Time: $t = \lfloor \text{sec}/0.010 \rfloor$ (10 ms frame shift). Frequency: we apply the HTK mel transform $m(\nu) = 2595 \log_{10}(1 + \nu/700)$, distribute 128 bins linearly in mel space, and map each annotation boundary to the nearest bin center. We clip to valid bounds and enforce positive area to ensure $t^{(2)} > t^{(1)}$, $f^{(2)} > f^{(1)}$, with minimum 1 bin and 1 frame. Because the mel filterbank covers 20–8000 Hz, annotations with frequency content entirely above 8 kHz are effectively excluded, and those partially above 8 kHz are clipped to the representable range. This exclusion or frequency clipping affects approximately 12% of the annotated rainforest events. For chunked data, annotation times are adjusted relative to the chunk start; annotations spanning chunk boundaries are included in each overlapping chunk and clipped accordingly, with events shorter than 100 ms after clipping discarded.

Patch tokenization. The encoder uses non-overlapping 16×16 patches, producing a 64×8 token grid (time $\times$ frequency). We reshape tokens to a feature map $E \in \mathbb{R}^{C \times 64 \times 8}$. The detector's FPN operates at multiple pyramid levels with strides $\{s_k\}$; box coordinates are converted to each level by dividing corners by the level stride. During augmentation, boxes are transformed jointly with the spectrogram and clipped to valid bounds.

D.2 Dataset Statistics and OOD Construction

Table 6 summarizes the labeled, unlabeled, and OOD evaluation data for both domains after preprocessing.

Table 6: **Dataset statistics for both domains.** Summary of labeled, unlabeled, and OOD evaluation data after preprocessing into 10.24 s log-mel spectrogram chunks. See Table 7 for BIRDeep site-to-habitat mapping.

	Rainforest			BIRDeep			
Split	**# Chunks**	**# Boxes**	**Avg./Chunk**	**# Chunks**	**# Boxes**	**Avg./Chunk**	**Sites**
Labeled Train	11,726	74,764	6.38	3,558	4,362	1.23	AM1,2,3,8,10,11,16
Labeled Val	2,513	15,598	6.21	762	902	1.18	same
Labeled Test	2,513	16,235	6.46	762	990	1.30	same
Temporal-OOD	286	3,400	11.89	671	1,716	2.56	AM4, AM8
Site-OOD	550	6,492	11.80	1,309	1,613	1.23	AM4, AM15
Unlabeled	481,245	—	—	7,051	—	—	All 9
+ External pretrain	—	—	—	900,000	—	—	—

Rainforest domain. Our recordings were collected from 2017 to 2019 from 15 sites in the Berau and East Kutai Regencies of East Kalimantan, Indonesia. Autonomous mono recorders were mounted 2 m above ground and recorded continuous 30-minute segments at 44.1 kHz with 40 dB gain. Recorders were at least 600 m apart and at least 200 m from roads, trails, and rivers. The labeled dataset contains approximately 24 hours of annotated audio (23.93 h), while the unlabeled pool contains approximately 661.7 hours of recordings from multiple sites in this landscape, corresponding to about 27.6 days of audio and 481,245 spectrogram chunks used for masked audio pretraining. The labeled set contains 16,752 spectrogram chunks and 106,597 annotated time–frequency boxes in total, with 13,170 non-empty chunks (78.6%). Under the in-domain split, the training, validation, and test partitions contain 11,726, 2,513, and 2,513 chunks, respectively, with similar average box density across splits. For out-of-distribution evaluation, we use two annotated settings: a temporal-OOD set from sites 13A and 13B containing 286 chunks and 3,400 boxes, and an unseen-site OOD set containing 550 chunks and 6,492 boxes.

Annotations were created in Raven Pro. A sonotype is a note or sequence of notes forming a distinct acoustic signal; a single species may produce multiple sonotypes. Each continuous vocalization is annotated with one box, while vocalizations separated by more than two seconds receive separate boxes. The 131 sonotypes span birds, mammals, insects, and amphibians. One expert produced the annotations, consulting specialists on uncertain cases; inter-annotator agreement was not measured. Applying the same separation rule across sonotypes yields variable annotation granularity: continuous vocalizations can form long boxes, while intermittent calls form multiple shorter boxes.

Bird domain. The BIRDeep dataset [36] contains 641 recordings from 9 sites across 4 habitat types in Doñana National Park, Spain. See Table 7 for site-to-habitat mapping. Expert annotations provide time–frequency bounding boxes for 38 bird species. We process all audio identically to the rainforest domain. The in-domain splits contain 5,082 spectrogram chunks with 6,254 bounding boxes. We use 7 sites spanning all 4 habitats for a 70/15/15 train/val/test split, corresponding to 3,558/762/762 chunks. For site-OOD evaluation, we hold out AM4 (low shrubland) and AM15 (marshland), held out from labeled training, producing 1,309 chunks with 1,613 boxes. For temporal-OOD evaluation, we hold out two recording dates, producing 671 chunks with 1,716 boxes. There is partial overlap between the two OOD sets as AM4 appears in both; temporal-OOD evaluation at AM4 therefore also includes a site shift. For masked pretraining, we combine unlabeled BIRDeep recordings that have 7,051 chunks with BirdSet XCM [50] that have 900k chunks and compute normalization statistics on this combined pool.

Both domains share the same preprocessing pipeline and evaluation protocol, enabling direct cross-domain comparison.

D.3 Adapter Designs.

We study three adapter variants that bridge the pretrained ViT-B encoder ($C{=}768$, spatial $64{\times}8$) to the FPN-based detection head.

Table 7: **BIRDeep recording sites and habitat types.** All 9 autonomous monitoring sites in Doñana National Park, Spain. Sites AM4 and AM15 are held out for site-OOD evaluation; all others are used for labeled training/validation/test.

Recorder	Place Name	Habitat	Split
AM1	Monteblanco	Low shrubland	Train
AM2	Sabinar	High shrubland	Train
AM3	Ojillo	High shrubland	Train
AM4	Pozo Sta Olalla	Low shrubland	Site-OOD
AM8	Torre Palacio	Ecotone	Train
AM10	Pajarera	Ecotone	Train
AM11	Caño Martinazo	Ecotone	Train
AM15	Cancela Millán	Marshland	Site-OOD
AM16	Juncabalejo	Marshland	Train

(i) **Conv adapter:** Three 3×3 conv layers ($768\to512\to256\to256$) with ReLU, preserving spatial resolution at 64×8. Restores short-range locality lost by global ViT attention but cannot resolve sub-patch events.

(ii) **Anisotropic adapter:** Stem conv ($768\to512$), residual block, separable 3×1 / 1×3 convolutions (Proposition 4), channel reduction ($512\to256$), second residual block, and nearest-neighbor frequency upsampling ($8\to32$). Operates at 64×32.

(iii) **Upsampling adapter** (default): 1×1 channel reduction ($768\to256$), residual block at base resolution, learnable transposed-conv upsampling ($\times2$ time, $\times4$ frequency), separable 3×1 / 1×3 anisotropic refinement at high resolution (Proposition 4), and a final 3×3 residual refinement block. Operates at 128×32 with effective strides 8×4 (Proposition 3).

All adapters output 256 channels and add 2–4 M parameters, keeping $>85\%$ of capacity in the frozen pretrained encoder.

D.4 Architecture Details

Neck and fusion. Starting from the adapter output (P3), we build P4/P5 via stride-2 3×3 convs. We use lateral 1×1 projections and top-down fusion with non-negative learnable weights normalized by their sum, and apply lightweight smoothing blocks (mixing $3 \times 3, 3 \times 1, 1 \times 3$) to preserve narrow-band detail while expanding temporal context.

Detector heads. We use FCOS, predicting per-location classification, centerness, and box regression with center sampling on the spectrogram lattice. We also use ATSS assignment [73], Quality Focal Loss (QFL) [31] for classification, and GIoU [53] for regression.

Contrastive sampling. Positives are formed by jittering each ground-truth box by up to 10% of its time and frequency extents. Negative candidates are sampled as random boxes in the spectrogram and scored by their maximum IoU with any ground-truth box. We sample 32 negatives per image: the first stage seeks hard negatives with IoU between 0.1 and 0.5, targeting 30% of the slots or stopping at a fixed attempt limit; the second stage fills the remaining slots with background negatives with IoU below 0.3. Both groups contribute to the same contrastive denominator. Features pooled over each box's time–frequency region are projected from 256 to 128 dimensions and L2-normalized before computing the contrastive loss.

D.5 Optimization details.

For masked-audio pretraining, we use AdamW with mixed-precision training and gradient clipping. The pretrained MAST encoder is initialized from the checkpoint pretrained on AudioSet [17] and optimized on unlabeled rainforest spectrograms using a masked reconstruction objective. Table 8 summarizes the main pretraining hyperparameters.

For MAST detector fine-tuning, we also use AdamW together with a warmup–cosine learning-rate schedule. The detector is initialized from the best masked-audio pretraining checkpoint and trained for 200 epochs on the labeled spectrogram data. For MAST detectors, we use a smaller effective

Table 8: **Masked Audio pretraining hyperparameters.** Main optimization settings used for masked-audio pretraining on unlabeled rainforest mel-spectrograms.

Hyperparameter	Value
Model	`mae_vit_base_patch16`
Optimizer	AdamW
AdamW betas	$(0.9, 0.95)$
Batch size	64
Training epochs	400
Base learning rate	5×10^{-5}
Minimum learning rate	5×10^{-7}
Warmup epochs	20
Weight decay	0.05
Mask ratio	0.75
Time masking probability	0.6
Frequency masking probability	0.3
Gradient clipping	1.0
Mixed precision	AMP

Table 9: **Detector fine-tuning hyperparameters.** Main optimization settings used for MAST detector training.

Hyperparameter	Value
Backbone initialization	Best masked-audio pretraining checkpoint
Adapter type	Upsampling
Neck	FPN P2-P5
Optimizer	AdamW
Batch size	64
Training epochs	200
Base learning rate	3×10^{-4}
Backbone LR multiplier	0.1
Head LR multiplier	1.0
Weight decay	1×10^{-3}
Gradient clipping	1.0
Learning-rate schedule	Warmup + cosine decay
EMA decay	0.9995
Augmentation probability	0.9
Contrastive weight	0.5
Contrastive temperature	0.1
Contrastive feature dimension	256
Contrastive projection dimension	128
Contrastive negatives per image	32

learning rate for the pretrained backbone and larger learning rates for newly introduced detector components. We train with mixed moderate spectrogram augmentation, exponential moving average, and gradient clipping. Table 9 summarizes the main fine-tuning hyperparameters used for the MAST model.

For staged self-training, we train the model in two phases with different optimization regimes. Stage 1 starts from the best supervised checkpoint and trains on large pseudo-labeled unlabeled data using confidence weighting, moderate augmentation, EMA, contrastive learning, and conservative gradual unfreezing. Its goal is to expand coverage under distribution shift while limiting drift from noisy pseudo supervision. Stage 2 then initializes from the best Stage 1 checkpoint and fine-tunes on clean expert-labeled data with a lower base learning rate, slightly weaker augmentation, and a more aggressive gradual-unfreezing schedule. This second phase re-anchors the detector to high-quality annotations while preserving the broader domain exposure acquired in Stage 1. In both stages, we use AdamW, warmup–cosine scheduling, and gradient clipping. Table 10 summarizes the main hyperparameters.

Table 10: **Staged self-training hyperparameters.** Main optimization settings for Stage 1 pseudo-labeled exploration and Stage 2 expert-labeled refinement.

Hyperparameter	**Stage 1**	**Stage 2**
Training data	Pseudo-labeled unlabeled data	Expert-labeled data
Initialization	Best supervised checkpoint	Best Stage 1 checkpoint
Adapter / neck	Upsampling + FPN P2-P5	Upsampling + FPN P2-P5
Optimizer	AdamW	AdamW
Batch size	64	64
Training epochs	80	40
Base learning rate	1×10^{-4}	5×10^{-5}
Weight decay	1×10^{-3}	1×10^{-3}
Gradient clipping	1.0	1.0
Scheduler	Warmup + cosine decay	Warmup + cosine decay
Warmup epochs	8	4
Backbone LR multiplier	0.1	0.02
Adapter LR multiplier	1.0	1.0
Head LR multiplier	1.0	1.0
Unfreezing warmup	20 epochs	3 epochs
Unfreezing interval	Every 10 epochs	Every 6 epochs
Blocks per stage	3	6
Unfrozen backbone LR multiplier	0.005	0.05
Confidence weighting	Enabled	Not Enabled
Pseudo-label confidence threshold	0.10	–
Data augmentation prob.	0.9	0.8
EMA decay	0.9995	0.9995
Contrastive loss weight	0.3	0.2
Contrastive temperature	0.1	0.1
Contrastive feature / proj. dim	256 / 128	256 / 128

Compute resources. All experiments were conducted on a single NVIDIA A100 GPU (80 GB). Masked-audio pretraining takes approximately 74 GPU-hours, detector fine-tuning approximately 22 GPU-hours per dataset, and each optional self-training round approximately 95 GPU-hours, including pseudo-label generation and both training stages. The pretrained regional encoder can be reused across recording sites, amortizing the pretraining cost.

Pseudo-label confidence threshold. The threshold $q = 0.10$ reflects the low-confidence regime typical of bioacoustic detectors, where true positive scores concentrate well below those seen in natural image detection due to faint, distant, or overlapping animal calls. At this operating point the threshold retains the majority of true positives while filtering out the lowest-scoring false alarms. Remaining label noise is further suppressed by confidence weighting, which down-weights uncertain pseudo-labels in proportion to their average detection score (Section 3, Proposition 6).

D.6 Augmentations

During supervised detector training, we apply moderate spectrogram augmentations with box-consistent transformations. All augmentations operate on 128×1024 log-mel spectrogram chunks and transform the associated time–frequency boxes jointly with the input. After each geometric augmentation, boxes are clipped to valid spectrogram boundaries and filtered if they become too small to remain meaningful training targets.

Our augmentation pipeline includes the following components:

- **Time shift.** We randomly shift the spectrogram along the time axis with zero/min-value padding, and shift all box time coordinates accordingly.
- **Frequency shift.** We randomly shift the spectrogram along the mel-frequency axis and apply the same offset to the box frequency coordinates.
- **Time warp.** We apply mild temporal warping by rescaling the spectrogram along the time axis and mapping box boundaries through the same transformation.

- **Frequency masking.** We mask a random frequency band in the style of SpecAugment [45]. Boxes with heavy overlap with the masked band are removed.
- **Time masking.** We mask a random temporal segment and similarly discard boxes that are largely occluded by the mask.
- **Gaussian noise and volume scaling.** We apply small additive Gaussian noise and moderate global intensity scaling to improve robustness to recording and gain variation.
- **Background mixing.** We optionally mix a spectrogram with background-noise samples drawn from a curated noise bank from our training data containing common environmental and anthropogenic sounds such as rain, thunder, aircraft, chainsaw, and vehicle noise, which improves robustness to realistic acoustic interference.
- **Mixup and CutMix.** We additionally implemented Mixup and CutMix for spectrograms by combining two training examples and merging or clipping their boxes accordingly. These augmentations are designed to simulate overlapping events and partial occlusion.

To avoid destabilizing detection, we use only mild transformation strengths and sample a small random subset of augmentations for each training example. This design preserves biologically meaningful event structure while improving robustness to temporal shifts, frequency variation, and background-condition changes.

D.7 Evaluation Details

We evaluate all models on binary time–frequency detection of "any animal sound" using five metrics: precision, recall, F1 score, mean average precision (mAP), and mean intersection-over-union (mIoU). A prediction is counted as a true positive if it overlaps a ground-truth box with IoU ≥ 0.5. Remaining predictions are false positives and unmatched ground-truth boxes are false negatives. We apply class-agnostic NMS with fixed overlap thresholds. The detection confidence threshold for precision, recall, and F1 is selected on the in-domain validation split and held fixed across all test regimes; the separate threshold $q = 0.10$ is used to generate pseudo-labels during self-training. Metrics are aggregated over all chunks in each test split. OOD split construction is described in Sec. D.2. Unless otherwise stated, all numbers collapse all sonotypes into a single foreground class.

E Additional Experimental Results

E.1 Comparison with Bird-MAE

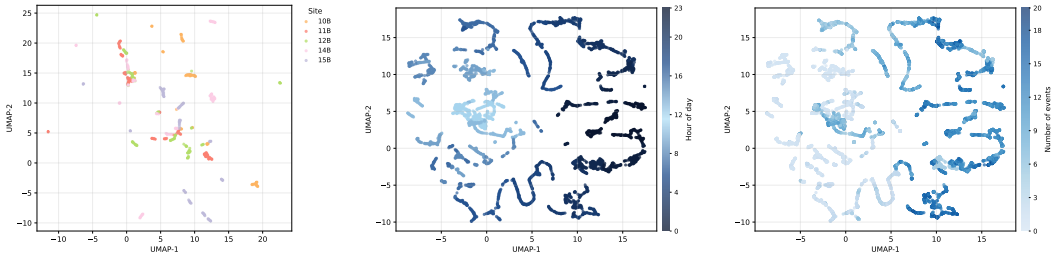
We compare pretrained encoders by keeping the MAST detector and fine-tuning recipe fixed and swapping only the encoder. Bird-MAE [51] shares the ViT-B/16 architecture and can be transferred to the same detector. Table 11 shows that Bird-MAE improves over AudioMAE in all four OOD settings. Domain-matched MAST pretraining yields the highest F1, including on BIRDeep, where Bird-MAE reaches 0.564 temporal and 0.543 cross-site F1 compared with 0.5895 and 0.5680 for MAST.

Table 11: **Pretrained encoder comparison under distribution shift.** We report detection F1 with the same detector recipe and different pretrained encoders, before self-training. *Higher is better*.

	Rainforest		**BIRDeep**	
Encoder	Temporal OOD	Cross-site OOD	Temporal OOD	Cross-site OOD
AudioMAE [21]	0.3323	0.3346	0.5014	0.5390
Bird-MAE [51]	0.399	0.401	0.564	0.543
MAST (ours)	**0.4301**	**0.4466**	**0.5895**	**0.5680**

E.2 Pretraining Data Scale

We pretrain on random subsets containing 25%, 50%, or 100% of the unlabeled rainforest pool, corresponding to approximately 165, 331, and 662 hours, while keeping the detector, labeled split, and fine-tuning budget fixed. As shown in Table 12, cross-site OOD F1 increases from 0.3334



(a) Colored by recording site.

(b) Colored by hour of day.

(c) Colored by annotated event count.

Figure 5: **UMAP visualization of pretrained encoder embeddings before detector fine-tuning on rainforest domain.** Each point is one mel-spectrogram chunk projected from the pretrained representation space into two dimensions. (a) We sample 100 chunks from each of five sites (10B, 11B, 12B, 14B, and 15B). Coloring by recording site reveals spatial clustering, indicating that the encoder captures site-specific acoustic signatures. (b,c) These panels use all chunks from one day at site 13B. (b) Coloring by hour of day shows a diel gradient, reflecting systematic changes in background noise and species activity across the 24-hour cycle. (c) Coloring by annotated event count shows that chunks with more animal-sound activity occupy partially distinct regions from background-dominated chunks, suggesting that masked-audio pretraining captures biologically relevant variation before any supervised training. Together, these patterns provide qualitative evidence of both the usefulness of domain-matched pretraining and the presence of temporal and cross-site distribution shift.

to 0.3881 and 0.4466, and temporal OOD F1 also increases with pretraining data. In-domain F1 changes only slightly, from 0.7841 to 0.7916. Additional unlabeled pretraining data therefore benefits detection primarily under distribution shift in this experiment.

Table 12: **Effect of pretraining data scale on rainforest detection F1.** The detector and fine-tuning budget are fixed across pretraining subsets. Audio durations are approximate. *Higher is better.*

Pretraining data	Hours	In-domain	Temporal OOD	Cross-site OOD
25%	165	0.7841	0.3174	0.3334
50%	331	0.7846	0.4014	0.3881
100% (MAST)	662	**0.7916**	**0.4301**	**0.4466**

E.3 Per-Round Self-Training Results

Table 14 reports per-round self-training results on BIRDeep across in-domain, temporal-OOD, and cross-site-OOD evaluation, together with a third-round extension on rainforest cross-site OOD. On BIRDeep, in-domain mAP rises from 0.601 at Round 0 to 0.778 at Round 2, a +0.18 absolute improvement that surpasses all baselines including FCOS at 0.665. Under cross-site shift, mAP improves from 0.380 to 0.441, and under temporal shift from 0.384 to 0.428. F1 also improves across OOD settings, reaching 0.643 cross-site and 0.649 temporal at Round 2.

Table 13: **Adapter design ablation on cross-site OOD detection on rainforest domain.** All variants use the same MAST detector and training setup; only the adapter module is changed. *Higher is better.*

Adapter Variant	Precision	Recall	F1	mAP	Mean IoU
Conv adapter	0.5928	0.3517	0.4415	**0.3004**	0.6644
Anisotropic adapter	0.6432	**0.3535**	**0.4563**	0.2866	0.6466
Upsampling adapter (Main)	**0.7562**	0.3169	0.4466	0.2573	**0.6824**

Table 14: **Per-round self-training results on BIRDeep and rainforest.** Round 0 is the supervised MAST detector before self-training. The rainforest extension reports cross-site OOD F1 through Round 3. *Higher is better.*

BIRDeep						
Setting	**Round**	**Prec.**	**Rec.**	**F1**	**mAP**	**mIoU**
In-domain	Round 0 (no ST)	**0.7824**	0.7010	0.7395	0.6011	0.6822
	Round 1	0.7054	0.8248	0.7604	0.6870	0.7080
	Round 2	0.6709	**0.9061**	**0.7709**	**0.7783**	**0.7339**
Temporal OOD	Round 0 (no ST)	**0.7618**	0.4808	0.5895	0.3838	**0.6083**
	Round 1	0.7253	0.5780	0.6434	0.3840	0.6020
	Round 2	0.6755	**0.6247**	**0.6491**	**0.4280**	0.5903
Cross-site OOD	Round 0 (no ST)	0.6476	0.5059	0.5680	0.3803	**0.6001**
	Round 1	**0.6893**	0.5914	0.6367	0.3953	0.5963
	Round 2	0.6522	**0.6438**	**0.6434**	**0.4414**	0.5943

Rainforest cross-site OOD				
Metric	**Round 0 (no ST)**	**Round 1**	**Round 2**	**Round 3**
F1	0.4466	0.5442	0.5715	**0.5813**

On rainforest, cross-site F1 increases from 0.4466 at Round 0 to 0.5442, 0.5715, and 0.5813 at Rounds 1–3. The successive gains of 0.0976, 0.0273, and 0.0098 show diminishing returns, with most of the improvement occurring in the first two rounds. Figure 4 displays Rounds 0–2, and the main MAST+ST comparisons use Round 2 for both domains.

E.4 Stratified Performance and Failure Analysis

We compare MAST before and after self-training across acoustic conditions, recording sites, and event properties. We report F1 for SNR and site strata and box-level recall for sonotype rarity, event size, and frequency band. Rarity analysis uses recall because the detector is class-agnostic.

SNR and recording site. Table 15 reports F1 by SNR tertile. All reported strata improve, and the low-SNR tertile receives the largest gain in three of the four splits. Gains vary across settings: low-SNR F1 increases by 0.125 on rainforest cross-site OOD but by only 0.002 on BIRDeep temporal OOD. Table 16 shows improvements at every reported rainforest site and at BIRDeep AM4. AM15, a marshland site with limited data and distinct acoustic conditions, declines from 0.444 to 0.417. This regression shows that aggregate gains can coexist with poorer performance at an individual site.

Table 15: **Detection F1 by SNR tertile.** Each entry reports MAST → MAST+ST. *Higher is better.*

Split	**High SNR**	**Mid SNR**	**Low SNR**
Rainforest cross-site OOD	0.434→**0.553**	0.389→**0.506**	0.403→**0.528**
Rainforest temporal OOD	0.435→**0.530**	0.365→**0.483**	0.380→**0.503**
BIRDeep site OOD	0.525→**0.554**	0.542→**0.584**	0.411→**0.463**
BIRDeep temporal OOD	0.510→**0.556**	0.529→**0.590**	0.501→**0.503**

Sonotype rarity. For each ground-truth box, we count the occurrences of its sonotype within the evaluation split and use this count to divide boxes into three equally sized groups: rare, mid, and common. Recall increases in every reported rarity group (Table 17). On rainforest cross-site OOD, rare-sonotype recall rises from 0.225 to 0.410, showing that self-training also recovers events from the rare tail.

Event size and frequency band. Table 18 reports rainforest cross-site OOD recall after dividing ground-truth boxes into equal-count thirds by area, and separately by frequency band. The smallest third remains difficult after self-training: recall improves from 0.004 to 0.169, compared with 0.847 for the largest third. Recall improves in all three frequency bands, although high-frequency events

Table 16: **Per-site OOD detection F1.** Self-training improves F1 at seven of the eight reported sites, with a decline at BIRDeep AM15. *Higher is better.*

Domain	Site	MAST	MAST+ST
Rainforest	13A	0.402	**0.526**
	13B	0.425	**0.515**
	5A	0.374	**0.489**
	5B	0.436	**0.501**
	6B	0.421	**0.456**
	7A	0.334	**0.404**
BIRDeep	AM4 (low shrubland)	0.506	**0.587**
	AM15 (marshland)	**0.444**	0.417

Table 17: **Box-level recall by sonotype rarity.** Each entry reports MAST → MAST+ST. Groups contain equal numbers of ground-truth boxes ranked by within-split sonotype occurrence count. *Higher is better.*

Rarity	Rainforest cross-site	BIRDeep site	BIRDeep temporal
Rare	0.225→**0.410**	0.559→**0.712**	0.414→**0.629**
Mid	0.373→**0.456**	0.476→**0.621**	0.552→**0.637**
Common	0.354→**0.586**	0.467→**0.528**	0.408→**0.586**

remain less well covered than low-frequency events. These results identify small and high-frequency events as persistent sources of missed detections in this domain.

Table 18: **Box-level recall by event size and frequency on rainforest cross-site OOD.** Size groups contain equal numbers of ground-truth boxes ranked by area. *Higher is better.*

Property	Group	MAST	MAST+ST
Box area	Smallest third	0.004	**0.169**
	Middle third	0.177	**0.455**
	Largest third	0.800	**0.847**
Frequency band	Low	0.618	**0.762**
	Mid	0.356	**0.507**
	High	0.262	**0.440**

False-positive composition. Using the IoU ≥ 0.5 matching criterion, we categorize false positives on rainforest cross-site OOD by their relation to the ground truth (Table 19). Localization errors and duplicates account for most false positives in both models. Background detections with no ground-truth overlap account for 2.3% of MAST false positives and 4.3% after self-training. These percentages describe the composition of each model's false positives, rather than the fraction of all predictions or an absolute false-alarm rate.

Table 19: **False-positive composition on rainforest cross-site OOD.** Percentages are relative to all false positives from the corresponding model and sum to 100% within each column.

False-positive type	MAST	MAST+ST
Loose localization of a true event	71.3%	77.1%
Duplicate of a detected event	26.4%	18.6%
Background (no ground-truth overlap)	2.3%	4.3%

E.5 Additional Adapter Design Ablation.

Table 13 compares adapter designs while holding the backbone, training recipe, and detection head fixed. All three adapters enable strong OOD performance, but they emphasize different operating points. The anisotropic adapter achieves the best F1 by slightly improving recall while keeping preci-

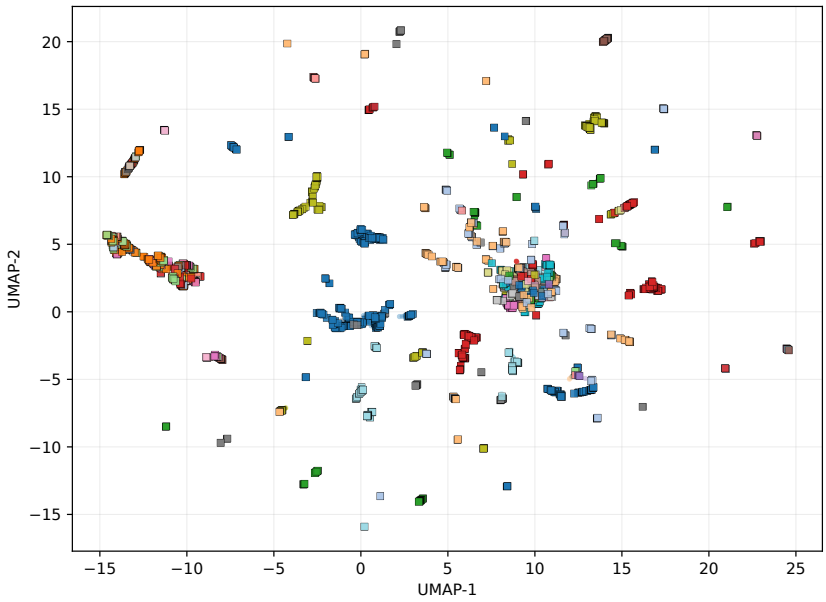


Figure 6: **Box-level embedding space induced by MAST detections on rainforest domain.** Each point corresponds to a ground-truth or predicted time–frequency box, embedded by pooling its corresponding region from the pretrained encoder feature map and projected with UMAP. Boxes with similar acoustic structure form partially coherent clusters, and high-quality predictions tend to lie near ground-truth embeddings of the same or related sonotypes. This suggests a practical downstream use of MAST: detected sound events can be retrieved against annotated examples for tentative zero-shot classification, while acoustically distinct clusters can be surfaced to experts as candidate novel sounds. Colors denote sonotypes: predicted boxes (circles) use nearest-neighbor retrieval labels, while ground-truth boxes (squares) use expert labels.

sion competitive, which aligns with the common structure of animal calls being temporally extended and frequency-concentrated. The convolutional adapter achieves the strongest mAP, suggesting that a simple locality-restoring module can already improve ranking across diverse OOD backgrounds. We use the upsampling adapter in our main result because it yields the highest precision and mean IoU, producing more confident and spatially accurate detections, properties that directly benefit downstream self-training, where high-precision pseudo-labels reduce confirmation bias. Overall, these results show that adapter choice primarily controls the precision–recall and box-quality trade-off, while the combination of in-domain pretraining and the detection stack remains the dominant driver of OOD robustness.

E.6 Representation Analysis of the Pretrained Encoder

To better understand what the pretrained encoder captures before detector fine-tuning, we visualize spectrogram-level embeddings using UMAP [37]. We extract pretrained encoder representations and project them into two dimensions. The site visualization uses 100 chunks from each of five sites (10B, 11B, 12B, 14B, and 15B); the hour-of-day and event-count visualizations use all chunks from one day at site 13B.

Figure 5 provides two qualitative insights. First, the visible gradient with respect to annotated event count suggests that masked-audio pretraining captures variation related to animal-sound activity, even before detector fine-tuning. Second, the organization by site and hour of day indicates that the learned representation also reflects domain-dependent structure associated with habitat, diel cycle, and background acoustics. These patterns qualitatively support both the usefulness of masked-audio pretraining for downstream detection and the presence of temporal and cross-site distribution shift in rainforest soundscapes. We emphasize that UMAP is only a qualitative visualization tool rather than a quantitative measure of separability, but it offers an intuitive view of why both strong pretraining and domain shift aware adaptation are important in this setting.

Table 20: **Retrieval-based sonotype classification from box embeddings on rainforest domain, with 131 sonotypes.** We evaluate nearest-neighbor retrieval against annotated training boxes in the learned embedding space. *Oracle-box classification* uses only ground-truth boxes, while *detected-box retrieval* uses predicted boxes matched to ground truth at IoU ≥ 0.5. Higher is better for Top-1, Top-5; lower is better for mean rank.

Setting	# Queries	# Labels	Top-1	Top-5	Mean Rank
Oracle-box classification (GT boxes)	14,986	131	0.723	0.902	5.76
Detected-box retrieval (matched preds)	8,414	104	0.961	0.982	3.22

E.7 Application: Box-Level Embeddings for Retrieval and Zero-Shot Classification

An additional advantage of MAST box-level localization is that each detected sound event can be embedded and compared directly in a shared representation space. After detection, we extract box-level features from the pretrained encoder by pooling the corresponding time–frequency region from the feature map, yielding one embedding per detected event. This enables a retrieval-style downstream workflow in which detected boxes can be matched to annotated training boxes, grouped by acoustic similarity, or provide detections to experts as candidate novel sounds.

Figure 6 illustrates this application qualitatively. Ground-truth and predicted boxes form partially structured clusters in the learned embedding space, and predictions with high overlap to ground truth tend to lie near embeddings of the same or related sonotypes. To quantify this effect, we also evaluate retrieval-based sonotype classification in two settings: (i) *oracle-box classification*, which uses only ground-truth boxes to test whether the embedding space discriminates sonotypes, and (ii) *detected-box retrieval*, which uses predicted boxes matched to ground truth at IoU ≥ 0.5 and assigns each matched prediction the sonotype of its best-matching ground-truth box. In both cases, we retrieve nearest neighbors from the annotated training set and report Top-1, Top-5, and mean rank.

As shown in Table 20, the learned box embeddings support strong retrieval-based classification. Oracle-box classification achieves 72.3% Top-1 and 90.2% Top-5 accuracy over 131 sonotypes, indicating that the embedding space already captures substantial sonotype structure. Retrieval on matched predicted boxes is even stronger with 96.1% Top-1 and 98.2% Top-5, reflecting that high-quality detections tend to align with acoustically prototypical events. In hyperdiverse and acoustically understudied environments, such a workflow is practically useful: detected sounds can be assigned tentative labels through retrieval against known examples, while acoustically distinct clusters can be prioritized for expert review as potentially novel or rare events.

E.8 Label Budget and Sampling Strategy

We study how label allocation affects performance under a fixed annotation budget. Starting from the full labeled training split, we subsample N chunks for $N \in \{250, 500, 1000, \ldots\}$ to simulate different labeling budgets. For each N, we compare three sampling strategies:

- **Random.** Uniformly sample N chunks from the training split without replacement.
- **Time-balanced.** Group chunks by their hour-of-day and sample approximately $N/24$ chunks per hour, ensuring coverage of both day and night.
- **Sequential.** Take the first N chunks in temporal order, mimicking a naive labeling strategy where annotators start at the beginning and stop once a budget is exhausted.

For each pair we train a FCOS detector from scratch with identical optimization and augmentation settings and evaluate on the same in-domain splits. Figure 7 reports F1 across budgets. At low budgets, the sampling policy is a first-order factor: sequential labeling consistently underperforms, while time-balanced sampling yields the strongest results, achieving F1 of 0.669 vs. 0.539, a +0.130 F1 gain at $N = 1000$. This pattern highlights the impact of temporal distribution shift even within a single site at a single day: labels drawn from one contiguous time range provide limited coverage of acoustic conditions, whereas time-balanced sampling improves robustness by exposing the model to broader diel variation. As the labeling budget increases, the gap between strategies narrows. Practically, this suggests that when annotation budgets are limited, it is better to spread labeling effort across times of day than to annotate one continuous stretch of audio. This finding is especially

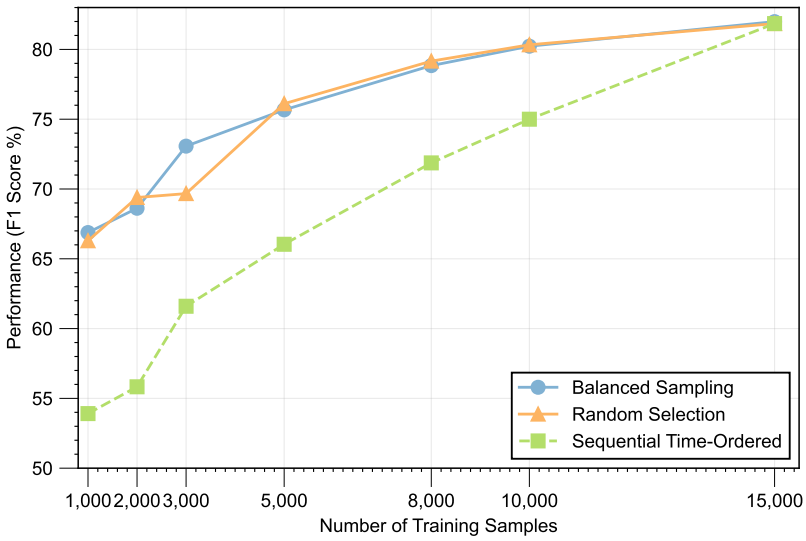


Figure 7: **Label-efficiency study under different sampling strategies on rainforest domain with FCOS**. We vary the number of labeled training chunks (x-axis) and report F1 on the held-out test set (y-axis). Time-balanced sampling across the 24-hour diel cycle consistently outperforms sequential labeling at low budgets, and remains slightly better than random sampling for most budgets. As the label budget increases, the gap narrows and all strategies converge to F1 ≈ 0.82 at $N{=}15000$.

relevant because sequential labeling is the most natural default in practice: annotators typically open a recording and label forward in chronological order until the budget is exhausted, inadvertently concentrating all labels within a narrow temporal window.

E.9 Qualitative Detection Comparison

To provide visual intuition for the quantitative gaps reported in Table 2, we randomly sample two 10.24 s chunks from the cross-site OOD test set and visualize the predictions of every method side by side as in Figure 8.

Several patterns are apparent in these two examples. Faster R-CNN fails to recover the annotated events, producing few or no predictions and yielding near-zero recall. FCOS and DETR exhibit the opposite failure mode: both generate numerous spurious predictions scattered across the spectrogram, resulting in low precision while also missing annotated events. AudioMAE shows partial improvement but still misses many events. In contrast, MAST predictions closely track the ground-truth boxes with markedly fewer false positives, consistent with its substantially higher OOD F1 in Table 2.

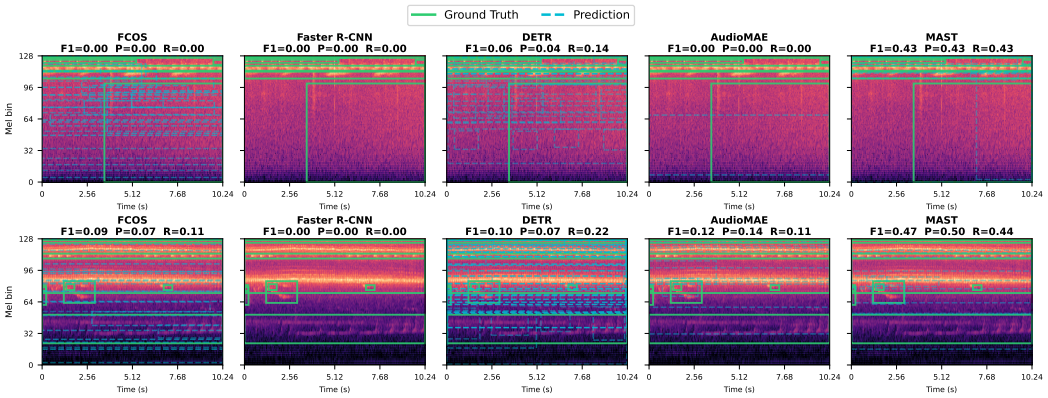


Figure 8: Qualitative detection comparison on two randomly sampled cross-site OOD chunks on rainforest domain. Ground-truth boxes are shown in green solid lines; predicted boxes in blue dashed lines. Baseline detectors produce numerous false positives or miss most events entirely, while MAST recovers more annotated events in these examples with fewer spurious detections.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main detection and robustness claims are supported by Tables 1 and 2; ablations and self-training results appear in Tables 3–5 and Figure 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 6 discusses binary detection, diminishing self-training gains and site-specific regression, frequency coverage, annotation limitations, and compute cost. App. E.4 provides detailed failure analysis.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Assumptions and proofs are provided in App. B and C. Proposition 7 includes a proof sketch with stated assumptions; Remark 8 is explicitly labeled as a remark.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: App. D provides the experimental setup, including hyperparameters in Tables 8, 9, and 10; preprocessing in App. D.1; adapter and architecture details in App. D.3 and D.4; augmentations in App. D.6; and evaluation in App. D.7.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code is publicly available at https://github.com/tianyi0216/MAST. Experimental settings, data preparation, training, and evaluation details are provided in App. D.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training details, hyperparameters, data splits, and preprocessing are provided in App. D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Due to the high computational cost of full pipeline training, results are from single runs without error bars.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: App. D.5 reports a single NVIDIA A100 GPU (80 GB) and approximate costs of 74 GPU-hours for pretraining, 22 for detector fine-tuning, and 95 per self-training round; these costs are also stated in Section 6.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research uses passive wildlife recordings. Section 6 discusses potential poaching and disturbance risks and recommends protecting sensitive location information.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

Justification: Section 6 discusses benefits for biodiversity monitoring and expert review, potential poaching or disturbance of sensitive species, and protection of precise recording locations.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [No]

Justification: Section 6 recommends withholding or coarsening precise locations and restricting access to sensitive metadata. We discuss these measures but do not report implemented release safeguards.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: AudioMAE, BIRDeep, BirdSet, and all baseline methods are cited. The rainforest dataset is collected by co-authors.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The MAST code is released at `https://github.com/tianyi0216/MAST`. Documentation of preprocessing, model architecture, training configurations, and evaluation is provided in App. D.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: No crowdsourcing or human subjects involved.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: No human subjects involved.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not used in the core methodology.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.