Synthetic data enables context-aware bioacoustic sound event detection

Benjamin Hoffman[1], *David Robinson*[1], *Marius Miron*[1], *Vittorio Baglione*[2], *Daniela Canestrari*[2], *Damian Elias*[3], *Eva Trapote*[4], *Felix Effenberger*[1], *Maddie Cusimano*[1], *Masato Hagiwara*[1], *Olivier Pietquin*[1]

[1]Earth Species Project, USA [2] Universidad de León, Spain
[3] University of California Berkeley, USA [4] Universidad de Valladolid, Spain

***Abstract*—We propose a methodology for training foundation models that enhances their in-context learning capabilities within the domain of bioacoustic signal processing. We use synthetically generated training data, introducing a domain-randomization-based pipeline that constructs diverse acoustic scenes with temporally strong labels. We generate over 8.8 thousand hours of strongly-labeled audio and train a query-by-example, transformer-based model to perform few-shot bioacoustic sound event detection. Our second contribution is a public benchmark of 13 diverse few-shot bioacoustics tasks. Our model outperforms previously published methods, and improves relative to other training-free methods by 64%. We demonstrate that this is due to increase in model size and data scale, as well as algorithmic improvements. We make our trained model available via an API, to provide ecologists and ethologists with a training-free tool for bioacoustic sound event detection.**

1. INTRODUCTION

Foundation models can learn new tasks at inference time from a few labeled examples—a process known as few-shot or in-context learning (ICL) [1]. This is attractive for application-driven ML domains—like bioacoustics, ecology, and conservation—where domain experts often lack ML experience and large labeled datasets [2], [3]. Despite growing interest in adapting foundation models for these fields, data scarcity limits the tasks they can be trained to perform [3].

An example of this situation occurs in few-shot bioacoustic sound event detection (FSBSED), which attempts to provide flexible modeling for the diversity of problems that arise in bioacoustics. In this task, formalized in [4], a model receives a *support set*: an audio recording with onset and offset annotations for the first few events of interest. The model must predict onsets and offsets of these events in the *query set*, which is the remainder of the recording.

Temporally fine-scale detection is crucial for many applications in animal behavior and ecology [5], but the time and expertise needed to annotate bioacoustic events has resulted in a lack of data available for training models capable of FSBSED. Prior efforts for FSBSED largely rely on a single 22-hour training dataset described in [4], leading to lightweight models tailored to small-scale data.

In this work, we investigate simultaneously scaling model parameters and training data volume, for a FSBSED model tailored for ICL (Figure 1). To overcome limited annotated data, we turn to synthetic data, transforming raw audio into strongly labeled scenes via custom preprocessing and augmentations that introduce domain randomization. Because our focus is ICL, we use a transformer-based few-shot model that attends to support and query audio jointly—common in ICL but rare in FSBSED. We call our method DRASDIC: *Domain Randomization for Animal Sound Detection In-Context*.

To evaluate performance, we introduce a new 13-dataset FSBSED benchmark, FASD13 (*Fewshot Animal Sound Detection-13*). DRASDIC achieves a 64% average improvement over prior methods that also do not use gradient updates at inference. Ablations show improvements are due to simultaneously scaling model size, training data, and improving the few-shot mechanism. We release DRASDIC weights, inference API, and FASD13 benchmark.[1]

[1]Available at www.github.com/earthspecies/drasdic_api.

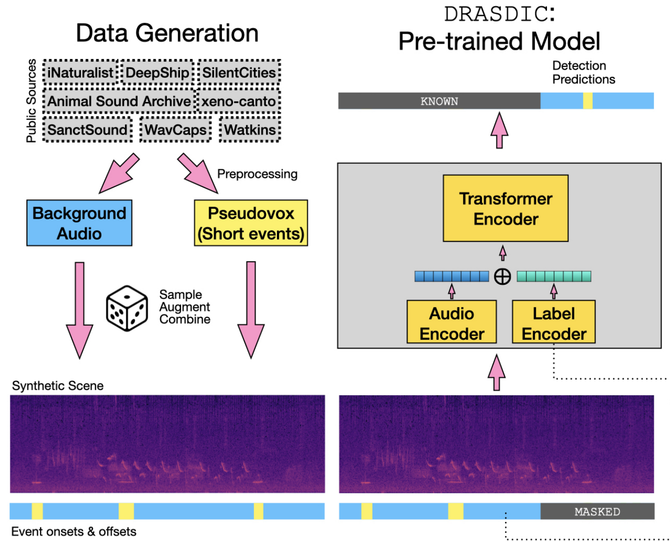

Fig. 1: We introduce a method for generating synthetic acoustic scenes (Left) and a SotA few-shot detection model (Right).

2. RELATED WORK

FSBSED was introduced in [4]. Challenges include sparse vocalizations, diverse target sounds, dynamic environments, and domain generalization [6]. Published methods include prototypical networks [7], representation learning [8], and transductive inference [9]. Prior evaluation of FSBSED systems has centered around the DCASE challenge [6], which provided public training and validation datasets and used a private test set. Subsequent efforts have either used the public validation set for both model selection and model evaluation [6], or skipped model selection [8].

In-context learning (ICL) refers to a model's ability to perform a task specified through demonstrations at inference time [1]. ICL has also been extended to fine-scale tasks in computer vision that somewhat resemble FSBSED, which include sementic segmentation [10], [11] and scene understanding [12]. Similar to our method, [11] employ a simple encoder-based architecture.

Generative vision and audio models have been used to create data for few-shot and low-resource tasks including detection [13] We are not aware of a generative audio model that produces realistic and low-SNR animal sounds, and so instead developed a preprocessing pipeline to isolate potential animal sounds in publicly available data. A similar procedure was developed recently in [14] for low-resource bioacoustic classification. For sound event detection in general audio, synthetic scenes assembled from multiple clips have been used to train models with fixed [15] and open ontologies [16].

3. METHOD

3.1. Data Generation

We propose a two-stage approach to generate scenes (Figure 2). From publicly available unlabeled audio, we derive a set of background

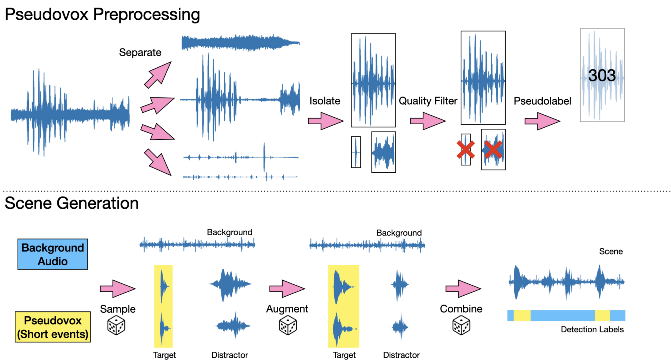

Fig. 2: Summary of training data preprocessing and scene generation.

tracks (5.1e5 tracks, 5540 hours) and a set of short clips containing events dubbed pseudo-vocalizations (*pseudovox*) (5.4e6 events, 577 hours); these are often animal vocalizations but may include non-biological acoustic events. These are pseudo-labeled through clustering, so multiple similar-sounding pseudovox can be sampled together.

In the second stage, performed on-the-fly during training, clips are randomly sampled from these collections, manipulated with data augmentations, and combined into scenes. Generated scenes may not always resemble real audio, due to randomness in the scene generation process. As prior work has shown that domain randomization in synthetic data generation improves transfer to real data [17], we view this as a way of increasing test-time robustness of our method.

3.1.1. Preprocessing: To construct our pseudovox set, we used public recordings from iNaturalist, Animal Sound Archive, xeno-canto[2], as well as Watkins [18], and WavCaps [19]. To remove background noise, we separated each recording into four stems using BirdMixIT [20]. For each stem, we isolated potential pseudovox: segments where the amplitude envelope exceeded 25% of the recording's maximum, indicating a possible acoustic event. Many segments still lacked a clear acoustic event, so we performed a quality filtering step. We manually annotated a subset of segments for vocalization presence, then trained a binary linear classifier on the final layer BirdNET [21] activations for each segment. We applied this quality filter; passing segments became the final pseudovox set. This procedure resulted in $M = 5.4e6$ pseudovox. Based on performance on a held-out test set, we estimated that 98% of these pseudovox contained a clear acoustic event. To obtain pseudolabels for the pseudovox, we applied k-means clustering to their BirdNET activations. We did this for $k \in K = \{\lfloor M/128 \rfloor, \lfloor M/64 \rfloor, \lfloor M/32 \rfloor, \lfloor M/16 \rfloor, \lfloor M/8 \rfloor\}$, to obtain different levels of cluster homogeneity. We inspected a random sample of 100 clusters, rating clusters as high- or low-quality based on acoustic homogeneity. Of these, 99 were deemed high quality. For background audio, we took the raw audio above, along with audio from SilentCities [22], DeepShip [23], and SanctSound [24].

3.1.2. Scene Generation: Scene generation consists of three parts: sampling audio clips, manipulating them with data augmentations, and combining them to form a scene. In Section 5.4, we investigate how the randomness in this process influences model performance.

We first sample two background tracks which are overlaid on each other. We choose a clustering level $k \in K$ and two clusters c_T, c_D from the clusters of level k. We sample a random number of target pseudovox from c_T, and a random number of distractor pseudovox from c_D. We apply reverb (drawn from [25]), resampling, time flipping, and amplitude augmentations to pseudovox, and resampling augmentations to background tracks. We paste pseudovox into the background track, one-by-one, with a random time gap between pseudovox. We maintain a binary annotation mask for the scene. This mask is initialized with zeros, and changed to ones where target pseudovox are added. Distractor pseudovox do not change the mask; they join whatever sounds are already present in the background tracks. To generate one training example, two scenes (support and query) are generated, drawing pseudovox from the same c_T, c_D for both. With some probability, the background tracks of the query are chosen to be different than those of the support.

[2]www.inaturalist.org/, www.museumfuernaturkunde.berlin/en/research/animal-sound-archive, www.xeno-canto.org, respectively

3.2. Model

Using our synthetic scenes, we train our model DRASDIC. During training the model is given annotated support audio and unannotated query audio, and must predict detection labels for the query audio.

3.2.1. Architecture: Noting that encoder-only architectures have been used successfully for fine-scale ICL problems in computer vision [11], we adopt a simple but highly parametrized BERT-like architecture which applies attention to support and query simultaneously. This is preceded by a CNN spectrogram encoder.

Support and query audio are resampled to 16 kHz, concatenated, and converted to a log mel-spectrogram (256 mels, hop size 160). The CNN encoder is a 2-d convolutional block and two 2-d residual blocks (ker=7, 3, 3, respectively; hidden size 64), with vertical mean pooling (ker=2) after each. Frequency and hidden dimensions are flattened and mean-pooled to a final 50 Hz frame rate. The binary support label mask is max-pooled to 50 Hz, passed to a per-frame label embedding, and added to the encoded audio. This label-enriched representation enters a transformer encoder (hidden size 768, 12 heads, 12 blocks) with rotary position encoding [26]. A final linear layer maps each frame to detection logits.

3.2.2. Training: DRASDIC was randomly initialized and trained with per-frame binary cross-entropy loss on the query labels, using AdamW [27] with $(\beta_0, \beta_1) = (0.9, 0.999)$ and weight decay 0.01. We used support-query pairs of total duration $\mathrm{dur}_s + \mathrm{dur}_q$ seconds. Based on initial experiments, we set $\mathrm{dur}_s = 30$ and $\mathrm{dur}_q = 10$.

Model, data generation, and training hyperparameters were chosen through random search. As our model selection criterion, we used average performance on the validation datasets from [6]. We applied curriculum learning to gradually increase task difficulty during training. This linearly decays the minimum pseudovox signal-to-noise ratio (SNR) from 0 dB to a minimum of -20 dB for an initial $5e4$ steps. The learning rate is linearly increased for $1e4$ steps to a maximum of $2e{-}5$, and then decayed to 0 after $1e5$ steps (cosine schedule) using batch size of 8. Parameters governing data generation are provided in the GitHub repository.

4. PUBLIC BENCHMARK

A collection of public FSBSED datasets was previously provided in [4], [6], but were designated as datasets for model training and validation. We complement these with FASD13, a public benchmark curated for model evaluation (Table 1). FASD13 consists of 13 bioacoustics datasets, each of which includes between 2 and 12 audio files. Eleven of these datasets were derived from previous studies; they were chosen for their taxonomic diversity, varied recording conditions, and quality of their annotations. Two (CC and JS) are presented here for the first time. All datasets were developed alongside studies of ecology or animal behavior, and represent a range of realistic problems encountered in bioacoustics data. Details of dataset collection and preprocessing steps are available at the GitHub repository.

Table 1: Details of FASD13. Datasets were chosen for their taxonomic diversity, varied recording conditions, and quality of their annotations. They were manually subsetted (prior to evaluation), to reduce computational overhead. Other (minor) preprocessing steps are described on the project GitHub. Datasets with a † are presented for the first time here. Terrestrial and underwater autonomous passive acoustic monitoring devices are abbreviated T. PAM and U. PAM, respectively.

Dataset	Full Name	N files	Dur (hr)	N events	Recording type	Location	Taxa	Detection target
AS [28]	AnuraSet	12	0.20	162	T. PAM	Brazil	Anura	Species
CC†	Carrion Crow	10	10.00	2200	On-body	Spain	Corvus corone + Clamator glandarius	Species + Life Stage
GS [29]	Gunshot	7	38.33	85	T. PAM	Gabon	Homo sapiens	Production Mechanism
HA [30]	Hawaiian Birds	12	1.10	628	T. PAM	Hawaii, USA	Aves	Species
HG [31]	Hainan Gibbon	9	72.00	483	T. PAM	Hainan, China	Nomascus hainanus	Species
HW [32]	Humpback Whale	10	2.79	1565	U. PAM	North Pacific Ocean	Megaptera novaeangliae	Species
JS†	Jumping Spider	4	0.23	924	Substrate	Laboratory	Habronattus	Sound Type
KD [33]	Katydid	12	2.00	883	T. PAM	Panamá	Tettigoniidae	Species
MS [34], [35]	Marmoset	10	1.67	1369	Laboratory	Laboratory	Callithrix jacchus	Vocalization Type
PM [36]	Powdermill	4	6.42	2032	T. PAM	Pennsylvania, USA	Passeriformes	Species
RG [37]	Ruffed Grouse	2	1.50	34	T. PAM	Pennsylvania, USA	Bonasa umbellus	Species
RS [38]	Rana Sierrae	7	1.87	552	U. PAM	California, USA	Rana sierrae	Species
RW [39]	Right Whale	10	5.00	398	U. PAM	Gulf of St. Lawrence	Eubalaena glacialis	Species

We follow the data format in [4]: Each audio file comes with annotations of the onsets and offsets of *positive* sound events, i.e. sounds coming from a predetermined category (such as a species label or call type). An N-shot detection system is provided with the audio up through the N^{th} positive event, and must predict the onsets and offsets of positive events in the rest of the recording.

5. EXPERIMENTAL EVALUATION

We evaluate models based on their ability to detect events after the $N = 5^{th}$ positive event in each recording of FASD13, using F1@0.3 IoU as described in [4]. We used performance on the validation set from [6] to select a final model to evaluate on FASD13.

5.1. Inference

For DRASDIC, we form predictions by windowing the audio in each recording, making multiple predictions for each window by prompting the model multiple times, and averaging these predictions. In detail, for a fixed dur_q-second window of the query set, we prompt the model $N = 5$ times and average the frame-wise predictions produced by these five prompts. The support set for the i^{th} prompt ($i \in [1, 5]$) is the dur_s-seconds of support audio centered at the i^{th} positive event in the support set (together with the binary detection mask). This procedure is repeated for dur_q-second windows across the entire query set. Frames with predicted detection probability above a fixed threshold of 0.5 become positive detections. These are smoothed: detections separated by a gap of $\min(1, d/2)$ seconds are merged, and then detections lasting less than $\min(1/2, d/2)$ seconds are discarded. Here d is the duration of the shortest event in the support set.

5.2. Comparison methods

We compare DRASDIC with essentially all of the previously published methods we are aware of for FSBSED that contain publicly available implementations. The first, "BEATS+linear" is a simple supervised baseline which consists of a frozen BEATS encoder [40] and a final linear layer. Support audio is windowed (4 seconds, 50% overlap) and the final layer is trained for 100 epochs to predict binary per-frame detection labels (final frame rate: 50 Hz). Training minimizes average per-frame binary cross-entropy loss. The initial learning rate of 0.01 (tuned using the validation set) is decayed to 0 using a cosine schedule. The second "AVES+linear" replaces the BEATS encoder with the pre-trained AVES encoder [41] (BirdAVES Base checkpoint), which was pre-trained on 2570 hours of animal sounds. "Protonet" is the prototypical network from [6], which itself adapts [7]. "Transductive" [9] uses a CNN encoder that is updated using unlabeled audio from the query set. "SCL" applies the supervised contrastive learning method introduced by [8]. "SCL+finetuning", also introduced by [8] extends this by using support audio to fine-tune the encoder that was pre-trained using the SCL method.

For Protonet, SCL, and SCL+Finetuning, we train a version using the training data from [4]. We also train a version using our generated scenes ($5e4$ scenes, each 40 seconds), which represents a $25\times$ increase in data quantity over the data used for training the original models.

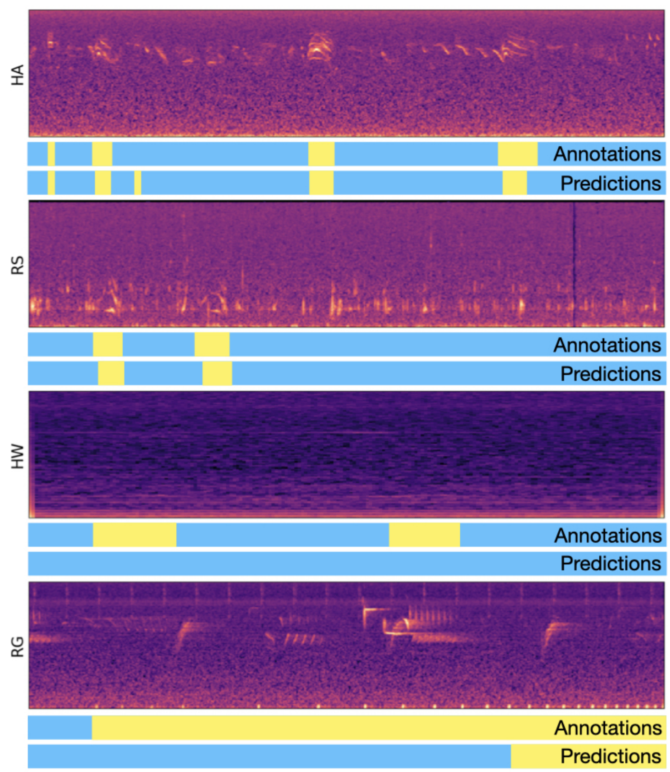

Fig. 3: Qualitative results; events are in yellow. DRASDIC detects target sounds in dynamic environments (top two), but challenges include extremely low SNR (third), and the extended low-frequency drumming displays of ruffed grouse (bottom). Each spectrogram represents 10 seconds of audio.

5.3. Experiments

We compare model performance on FASD13 (Table 2). Some datasets (JS, KD, MS, PB, PB24) contain events that are above DRASDIC's 8kHz Nyquist frequency, or that are brief relative to the model's 50Hz frame rate. For these, we give the model a 1/2-time version (1/6 for

Table 2: F1 scores @0.3 IoU on FASD13. Methods marked with † were pre-trained using our generated data, rather than the data used in the original publication. Methods marked with * involve no gradient updates at inference time. The second column gives the number of model parameters, and the final column gives the average F1 score across the six validation datasets from [6].

Model	Params	AS	CC	GS	HA	HG	HW	JS	KD	MS	PM	RG	RS	RW	Avg	Val
BEATS+linear	90M	.350	.003	.056	.093	**.242**	.173	.028	.049	.462	.212	**.732**	.007	.316	.209	.358
AVES+linear	90M	.586	.059	.500	.374	.207	.366	.026	**.673**	**.831**	.291	.529	.303	.494	.403	.565
Protonet*	0.7M	.356	.189	.156	.239	.038	.085	.136	.316	.590	.260	.000	.216	.393	.229	.461
Protonet†*	0.7M	.305	.224	.151	.307	.023	.116	.166	.418	.536	.235	.121	.195	.342	.242	.459
Transductive	0.5M	.299	.144	.002	.283	.020	.116	.279	.218	.569	.159	.089	.169	.048	.184	.242
SCL*	7.2M	.516	.333	.025	.438	.010	.255	.281	.263	.402	.237	.049	.219	.509	.272	.514
SCL+finetuning	7.2M	.565	**.341**	.017	.467	.008	.382	**.302**	.381	.476	.327	.042	.285	.275	.298	.525
SCL†*	7.2M	.545	.287	.024	.433	.008	.393	.243	.207	.429	.336	.038	.218	.228	.261	.440
SCL†+finetuning	7.2M	.571	.205	.030	.479	.005	**.453**	.132	.220	.516	.450	.050	.292	.223	.279	.453
DRASDIC †* (ours)	116M	**.645**	.272	**.593**	**.587**	.144	.337	.099	.644	.783	**.474**	.092	**.352**	**.764**	**.445**	**.704**

KD). We give other methods *both* the slowed and full-speed version of the data, and keep the version with the better score.

On FASD13, DRASDIC outperforms all the alternatives on 6 out of 13 datasets. Across datasets DRASDIC has an average F1 score of .042 over the next best model. Compared to other methods that do not require gradient updates at inference time, DRASDIC outperforms the others on 9 of 13 datasets, and has an average F1 score of .173 over the next best model (64% relative improvement). Qualitatively, DRASDIC detected diverse target sounds, even amid others in the same frequency bands (Figure 3, top). Performance is strong across a variety of taxa and conditions. A failure case is for the JS dataset, which consists of jumping spider drumming. Here, the detection targets are specific drum types, and distinguishing between drum types relies partly on the rate of drumming. Our scene generator did not account for this type of information. Other failure cases are in Figure 3, bottom.

For the comparison methods we trained with our generated data, there was no clear performance increase. These methods, which adopt a CNN architecture, employ a different few-shot mechanism than DRASDIC and also have fewer than $1/10$ the trainable parameters. The relative impact of these differences is investigated in Section 5.4.

5.4. Ablation experiments

In our main experiments, we scaled model parameters and data volume, while also adopting a few-shot mechanism that applied attention to support and query audio simultaneously. We conducted experiments to investigate the contributions of these changes, individually (Table 3, top). First, we compared our main model (116.1M parameters), whose transformer encoder has the same structure as BERT Base, to smaller versions based on BERT Small [42] (19.3M parameters) and BERT Tiny (2.5M parameters). Second, we compared our main data generation procedure to one that only generated 220 hours of unique scenes, and one that only generated 22 hours of unique scenes. Additionally, we compared to a version that used the non-synthetic training data (22 hours total) from [4], as well as a version for which 10% of the training examples were from [4] and the other 90% synthetic. Finally, we adjusted our few-shot mechanism to a prototypical network, which prevented attention from being applied to support and query audio simultaneously. For this, we kept the same architecture as our main method but applied a prototypical loss as in [6], [7]. Average performance on FASD13 and on the validation set dropped in all cases, indicating that each of these changes contributed to final model performance. Reducing data scale was especially damaging, likely due to the high number of trainable parameters in our main method.

We investigated the impacts of adjusting the randomness governing our scene generation procedure (Table 3, bottom). We perturbed the level of homogeneity of target events in a scene, the typical rate of events, the loudness of events, and whether we apply pitch shifting augmentations. Average performance is stable across some of these perturbations, but decreases when the randomness in event rate and event SNR is decreased. These parameters likely influence the level of diversity present across generated scenes more than the others.

Table 3: Average F1 scores @0.3 IoU on FASD13 and validation [6] datasets. Model ablations appear on top, data ablations on bottom.

Method	Avg (test)	Avg (val)
DRASDIC	.445	.704
BERT Small	.425	.666
BERT Tiny	.323	.521
Reduced data (220 h)	.354	.504
Reduced data (22 h)	.130	.088
Non-synthetic data (22 h)	.135	.165
10% non-synthetic data	.428	.666
Protonet loss	.444	.638
High homogeneity in events	.429	.630
Low homogeneity in events	.444	.606
High events / second	.381	.437
Low events / second	.370	.669
Only high SNR events	.402	.620
Only low SNR events	.438	.682
No pitch/time shifting	.457	.613

Eliminating random pitch shifts resulted in slightly better performance on FASD13. Designing a domain randomization strategy is an optimization problem, which we approached through a model selection criterion. This criterion did not produce the best model on the test set, aligning with the observation [6] that strong domain shifts between few-shot tasks present a challenge for FSBSED model development.

CONCLUSION

To provide a training-free solution for fine-scale bioacoustic sound event detection, we develop a ICL transformer model DRASDIC. We develop a domain-randomization based data-generation pipeline, and train our model on over 8.8 thousand hours of synthetic acoustic scenes. We additionally provide FASD13, a new benchmark for few-shot bioacoustic sound event detection. Our model substantially improves upon previous state-of-the art. We demonstrate that these improvements are due to both our modeling approach and the data scale provided by our scene generation method.

REFERENCES

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan *et al.*, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33, 2020, pp. 1877–1901.

[2] D. Robinson, M. Miron, M. Hagiwara, and O. Pietquin, "NatureLM-audio: An audio-language foundation model for bioacoustics," in *Proceedings of the International Conference on Learning Representations (ICML)*, 2025.

[3] Z. Miao, Y. Zhang, Z. Fabian, A. Hernandez Celis, S. Beery *et al.*, "New frontiers in AI for biodiversity research and conservation with multimodal language models," *Preprint*, 2024. [Online]. Available: https://doi.org/10.32942/X22S6F

[4] I. Nolasco, S. Singh, V. Morfi, V. Lostanlen, A. Strandburg-Peshkin *et al.*, "Learning to detect an animal sound from five examples," *Ecological Informatics*, vol. 77, p. 102258, 2023.

[5] D. Stowell, "Computational bioacoustics with deep learning: a review and roadmap," *PeerJ*, vol. 10, p. e13152, 2022.

[6] J. Liang, I. Nolasco, B. Ghani, H. Phan, E. Benetos, and D. Stowell, "Mind the domain gap: A systematic analysis on bioacoustic sound event detection," in *2024 32nd European Signal Processing Conference (EUSIPCO)*, 2024, pp. 1257–1261.

[7] H. Liu, X. Liu, X. Mei, Q. Kong, W. Wang, and M. D. Plumbley, "Segment-level metric learning for few-shot bioacoustic event detection," *DCASE*, 2022.

[8] I. Moummad, N. Farrugia, and R. Serizel, "Regularized contrastive pre-training for few-shot bioacoustic sound detection," in *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2024, pp. 1436–1440.

[9] D. Yang, H. Wang, Y. Zou, Z. Ye, and W. Wang, "A mutual learning framework for few-shot sound event detection," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 811–815.

[10] A. Bar, Y. Gandelsman, T. Darrell, A. Globerson, and A. Efros, "Visual prompting via image inpainting," *Advances in Neural Information Processing Systems*, vol. 35, pp. 25 005–25 017, 2022.

[11] X. Wang, W. Wang, Y. Cao, C. Shen, and T. Huang, "Images speak in images: A generalist painter for in-context visual learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2023, pp. 6830–6839.

[12] I. Balazevic, D. Steiner, N. Parthasarathy, R. Arandjelović, and O. Henaff, "Towards in-context scene understanding," in *Advances in Neural Information Processing Systems*, vol. 36, 2023, pp. 63 758–63 778.

[13] S. Lin, K. Wang, X. Zeng, and R. Zhao, "Explore the power of synthetic data on few-shot object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 638–647.

[14] M. J. Weldy, D. B. Lesmeister, T. Denton, A. Duarte, B. J. Vernasco *et al.*, "Simulated soundscapes and transfer learning boost the performance of acoustic classifiers under data scarcity," *Methods in Ecology and Evolution*, 2025.

[15] R. Serizel, N. Turpault, A. Shah, and J. Salamon, "Sound event detection in synthetic domestic environments," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 86–90.

[16] Y. Wu, C. Tsirigotis, K. Chen, C.-Z. A. Huang, A. Courville *et al.*, "FLAM: Frame-wise language-audio modeling," in *Forty-second International Conference on Machine Learning*, 2025.

[17] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 23–30, 2017.

[18] L. Sayigh, M. A. Daher, J. Allen, H. Gordon, K. Joyce *et al.*, "The Watkins marine mammal sound database: An online, freely accessible resource," *Proceedings of Meetings on Acoustics*, vol. 27, no. 1, p. 040013, 2016.

[19] X. Mei, C. Meng, H. Liu, Q. Kong, T. Ko *et al.*, "WavCaps: A ChatGPT-assisted weakly-labelled audio captioning dataset for audio-language multimodal research," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, pp. 3339–3354, 2024.

[20] T. Denton, S. Wisdom, and J. R. Hershey, "Improving bird classification with unsupervised sound separation," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 636–640.

[21] S. Kahl, C. M. Wood, M. Eibl, and H. Klinck, "BirdNET: A deep learning solution for avian diversity monitoring," *Ecological Informatics*, vol. 61, p. 101236, 2021.

[22] S. Challéat, N. Farrugia, J.-S. P. Froidevaux *et al.*, "A dataset of acoustic measurements from soundscapes collected worldwide during the covid-19 pandemic," *Scientific Data*, vol. 11, p. 928, 2024.

[23] M. Irfan, Z. Jiangbin, S. Ali, M. Iqbal, Z. Masood, and U. Hamid, "Deepship: An underwater acoustic benchmark dataset and a separable convolution based autoencoder for classification," *Expert Systems with Applications*, vol. 183, p. 115270, 2021.

[24] NOAA, "Passive acoustic data collection," 2017.

[25] J. Traer and J. H. McDermott, "Statistics of natural reverberation enable perceptual separation of sound and space," *Proceedings of the National Academy of Sciences*, vol. 113, no. 48, pp. E7856–E7865, 2016.

[26] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu, "Roformer: Enhanced transformer with rotary position embedding," *Neurocomputing*, vol. 568, p. 127063, 2024.

[27] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019.

[28] J. S. Cañas, M. P. Toro-Gómez, L. S. M. Sugai *et al.*, "A dataset for benchmarking neotropical anuran calls identification in passive acoustic monitoring," *Scientific Data*, vol. 10, p. 771, 2023.

[29] N. Yoh, W. Mbamy, B. L. Gottesman, G. Z. Froese, T. Satchivi *et al.*, "Impacts of logging, hunting, and conservation on vocalizing biodiversity in gabon," *Biological Conservation*, vol. 296, p. 110726, 2024.

[30] A. Navine, S. Kahl, A. Tanimoto-Johnson, H. Klinck, and P. Hart, "A collection of fully-annotated soundscape recordings from the island of Hawai'i," 2022. [Online]. Available: https://doi.org/10.5281/zenodo.7078499

[31] E. Dufourq, I. Durbach, J. P. Hansford, A. Hoepfner, H. Ma *et al.*, "Automated detection of Hainan gibbon calls for passive acoustic monitoring," *Remote Sensing in Ecology and Conservation*, vol. 7, no. 3, pp. 475–487, 2021.

[32] A. N. Allen, M. Harvey, L. Harrell, A. Jansen, K. P. Merkens *et al.*, "A convolutional neural network for automated detection of humpback whale song," *Frontiers in Marine Science*, vol. 8, p. 607321, 2021.

[33] L. Symes, S. Madhusudhana, S. Martinson *et al.*, "Neotropical forest soundscapes with call identifications for katydids," Dataset, 2023. [Online]. Available: https://datadryad.org/dataset/doi:10.5061/dryad.zw3r2288b

[34] E. Sarkar and M. Magimai.-Doss, "Can self-supervised neural representations pre-trained on human speech distinguish animal callers?" in *Proceedings of INTERSPEECH 2023*, 2023, pp. 1189–1193.

[35] Y.-J. Zhang, J.-F. Huang, N. Gong, Z.-H. Ling, and Y. Hu, "Automatic detection and classification of marmoset vocalizations using deep and recurrent neural networks," *The Journal of the Acoustical Society of America*, vol. 144, no. 1, p. 478, 2018.

[36] L. M. Chronister, T. A. Rhinehart, A. Place, and J. Kitzes, "An annotated set of audio recordings of eastern north american birds containing frequency, time, and species information," *Ecology*, vol. 102, no. 6, p. e03329, 2021.

[37] S. Lapp, J. L. Larkin, H. A. Parker, J. T. Larkin, D. R. Shaffer *et al.*, "Automated recognition of ruffed grouse drumming in field recordings," *Wildlife Society Bulletin*, vol. 47, no. 1, p. e1395, 2023.

[38] S. Lapp, T. C. Smith, R. A. Knapp, A. Lindauer, and J. Kitzes, "Aquatic soundscape recordings reveal diverse vocalizations and nocturnal activity of an endangered frog," *The American Naturalist*, vol. 203, no. 5, pp. 618–627, 2024, pMID: 38635364.

[39] O. S. Kirsebom, F. Frazao, Y. Simard, N. Roy, S. Matwin, and S. Giard, "Performance of a deep neural network at detecting north atlantic right whale upcalls," *The Journal of the Acoustical Society of America*, vol. 147, no. 4, pp. 2636–2646, 2020.

[40] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins *et al.*, "BEATs: audio pre-training with acoustic tokenizers," in *Proceedings of the 40th International Conference on Machine Learning*, ser. ICML'23, 2023.

[41] M. Hagiwara, "AVES: Animal vocalization encoder based on self-supervision," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[42] I. Turc, M. Chang, K. Lee, and K. Toutanova, "Well-read students learn better: The impact of student initialization on knowledge distillation," *CoRR*, vol. abs/1908.08962, 2019.