Bird Species Recognition using Convolutional Neural Networks with Attention on Frequency Bands

Mario Lasseck

Museum für Naturkunde Berlin, Germany

Abstract

This paper presents a deep learning approach for recognizing bird species in soundscape recordings using Convolutional Neural Networks (CNNs). The proposed method extends CNNs with a classification head that incorporates attention on frequency bands. The models are trained on a large dataset of bird sounds and employ various data augmentation techniques to improve performance and address the domain shift between training and test data. The effectiveness of the approach is evaluated in the BirdCLEF 2023 competition, hosted on Kaggle, where it achieves a macro-averaged mean average precision (cmAP) of 76.3 % on the official test set. This performance positions the method among the top 3 systems to accurately identify birds in wildlife monitoring recordings around Northern Mount Kenya.

Keywords

Bird Species Recognition, Biodiversity Assessment, Soundscapes, Convolutional Neural Networks, Deep Learning, Data Augmentation, Kaggle Competition

1. Introduction

The BirdCLEF 2023 competition focuses on recognizing vocalizing birds in Eastern African soundscape recordings. The main challenges to address in this year's task edition include the imbalance in the number of training files per species, the domain shift between training and test data and the time limit of two hours to identify all birds in a large set of diverse soundscape recordings spanning over 30 hours.

The machine learning algorithms developed within the scope of this competition will help researchers to conduct pilot projects in selected areas of Northern Mount Kenya and evaluate the impact of different management strategies and degradation levels on bird biodiversity in rangeland systems. By accurately monitoring the effects of restoration efforts on biodiversity, they aim to establish financial mechanisms for widespread landscape restoration and protection. The advancements of systems for automated bird recognition will facilitate more effective evaluation of threats and adjustments to conservation actions, benefiting avian populations and supporting long-term sustainability.

Further details about the BirdCLEF 2023 task are given in [1], [2] and [3]. The task is part of the LifeCLEF 2023 evaluation campaign [4,5] and the Conference and Labs of the Evaluation Forum [6,7].

CLEF 2023: Conference and Labs of the Evaluation Forum, September 18–21, 2023, Thessaloniki, Greece
EMAIL: Mario.Lasseck@mfn.berlin

2. Implementation

The implementation of the machine learning based system for bird species recognition presented in this paper builds upon solutions for previous BirdCLEF competitions and similar tasks [8,9,10,11,12]. Further details on own past developments and implementation methods can be found for example in [13], [14] and [15].

2.1. Data Preparation

The official BirdCLEF 2023 training dataset consists of 16940 audio recordings provided by Xeno-canto [16], covering 264 different bird species. To address the class imbalance and limited number of training files per species, the dataset is extended with recordings from previous competitions [17,18,19,20] and additional files from Xeno-canto (XC). Furthermore, soundscapes (SC) without bird activity from the DCASE 2018 Bird Detection Task [21,22] and other sources [23] are included as a 'nocall' class and for noise augmentation.

Table 1 gives an overview on the individual datasets utilized. The extended dataset encompasses a total of 659 classes, comprising 264 species from 2023, additional 394 species from previous years and the 'nocall' class. All in all, 141580 files are collected for training and augmentation with a total accumulated duration of approximately 78 days.

Table 1: Datasets used for training and augmentation

ID	Name	# Classes	# Files	accum. Duration
1	BirdCLEF 2023 XC	264	16940	8d 00h 24m 02s
2	BirdCLEF 2023 XC ext.[2]	264	32729	21d 02h 38m 33s
3	BirdCLEF 2020/21 XC	397	62874	40d 22h 29m 56s
4	BirdCLEF 2020/21 XC ext.	182	6941	3d 05h 08m 04s
5	BirdCLEF 2020/21 SC	49	20	03h 20m 00s
6	BirdCLEF 2019 SC	69	64	2d 02h 17m 11s
7	DCASE 2018	1	22012	2d 13h 08m 40s
	Total	659	141580	78d 01h 26m 26s

The original training set consists of audio files from Xeno-canto only, which were resampled to 32 kHz, converted to mono and compressed to lossy Ogg format. To ensure consistent sampling rates and prevent resampling during training, files from other sources are also converted to 32 kHz. Additional files obtained from Xeno-canto [24] are converted to lossless FLAC format without mono mixing. The duration of each file is added to the training metadata to enable fast access of short audio segments within files at random position during training. Furthermore, the energy of 5-second segments, shifted by one second, is calculated for each file using the root-mean-square (RMS) of the signal amplitude in each segment. This information is later used to weight the selection of audio chunks and increase the probability of finding segments with bird activity.

Xeno-canto files are weakly labeled, meaning that there is no precise information on the presence or absence of the labeled bird within the recording. However, there is typically a high probability of hearing the labeled bird at the beginning of each audio file, as recordists often trim their recordings accordingly before uploading them. To exploit this characteristic, the first 10 seconds of each recording are duplicated and also added to the training set.

For training and cross-validation, the entire dataset is split into 8 stratified randomized folds, ensuring that the primary species used in the 2021 and 2023 BirdCLEF editions are proportionally represented in each fold.

[2]This dataset also includes files from the official competition dataset (BirdCLEF 2023 XC) but with different preprocessing

2.2. Feature Engineering

The models are trained on 5-second audio chunks represented as spectrograms. The raw 1D audio signal is converted to a 2D log Mel spectrogram image using the *melspectrogram* and *power_to_db* functions of the librosa python library [25] with the following parameters:

- sr = 32000
- n_fft = 2048
- hop_length = 512
- n_mels = 128
- fmin = 40
- fmax = 15000
- power = 2.0
- ref = np.max
- top_db = 100

The resulting spectrogram image is then normalized to the unsigned integer range of 0 to 255, resized to a resolution of 312x128 pixels and converted to a 3-channel RGB image. This preprocessing yield to create images from audio close to the input format most Convolutional Neural Networks are original designed for and pretrained on.

2.3. Training Methods

The training process makes use of various tools and libraries. PyTorch [26] is utilized as the main framework, along with additional libraries such as timm [27] for CNN backbones and pretrained weights, SoundFile [28], librosa [25] and SciPy [29] for audio and signal processing, Audiomentations [30] for data augmentation and scikit-learn [31] for calculating metrics and creating training/validation data splits.

All models adopt a common architecture consisting of a CNN backbone pretrained on ImageNet [32] as a feature extractor, combined with a custom classification head. The classification head is designed based on a modified Sound Event Detection (SED) head, which incorporates attention on frequency bands. This modification aims to leverage the fact, that birds in soundscapes usually occupy species specific frequency ranges. In the original SED architecture [33,34,35,36], feature maps representing frequency bands are aggregated via mean pooling and attention is applied on features representing time frames only. By applying attention to frequency bands instead, the model can better differentiate species vocalizing simultaneously but with different pitches. The modification involves a simple step of rotating the spectrogram image by 90 degrees before feeding it into the original SED network. For feature encoding, EfficientNet CNNs of the first and second generation [37,38] are employed (mainly timm's tf_efficientnet_b0_ns and tf_efficientnetv2_s_in21k). Figure 1 illustrates the variation in class activation outputs between the original and modified SED architecture when presented with an input containing multiple bird species. In this simplified example, species c_2 is better detected with frequency attention, because class activation doesn't interfere with other species, like it would for example with species c_0 in the case of using time attention.

The training process involves multiple steps. Initially, the models are trained on all 8 cross-validation folds. During training, 5-second audio chunks are randomly selected from each file, either without weighting or weighted by signal energy to increase the likelihood of capturing segments with bird activity. The trained models are then used to create pseudo labels for successive 5-second intervals in all training files. This enables the selection of chunks in subsequent training steps not only based on signal energy weighting but also weighted by the probability of the primary (foreground) species assigned to the file. Pseudo labels are also used to add additional labels of possible background species to selected audio chunks during training. Up to 8 pseudo labels are added, depending on species probabilities, either as hard labels (if the species probability is above 0.8) or as soft labels using the probability derived from the pseudo label. If provided by Xeno-canto recordists, background species are included as soft target labels with a value of 0.3.

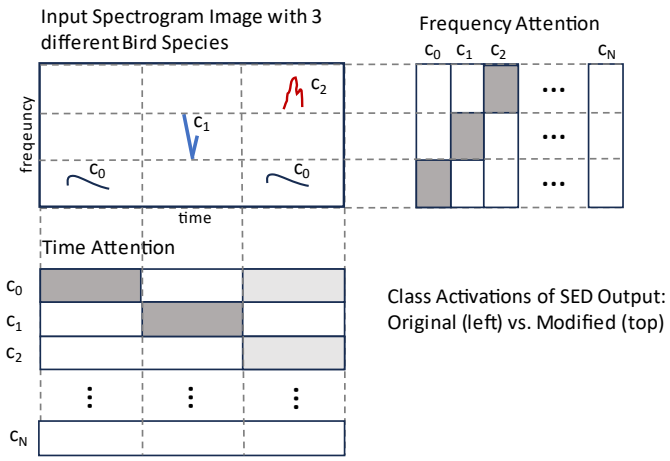

Figure 1: Example of SED output (class activation) using either time or frequency attention

During training, binary cross-entropy (*BCEWithLogitsLoss*) [39] serves as loss function, while Adam [40] is employed as optimizer. The learning rate scheduling follows the one-cycle *CosineAnnealingLR* [41] policy, starting with an initial learning rate of 0.001. Validation is performed using the first 5 seconds of files in the validation set and learning progress is tracked using evaluation metrics LRAP (*label_ranking_average_precision_score*) [42], cmAP [43] and F1 score [44].

For inference, predictions are reduced to the 2023 species set (264 classes) and multiple models are ensembled by averaging their predictions without weighting.

2.4. Data Augmentation

Several data augmentation techniques are applied during training, especially to address the challenges posed by weak and noisy labels, as well as to compensate for the domain shift between training and test data. Many of these techniques have been successfully employed in previous approaches. For a more detailed description of each method and its impact on model performance, please refer to [13] and [15]. Here is a concise overview of the augmentation methods used in this competition:

- Random cyclic shift of the audio signal
- Application of audio signal filter with random transfer function
- Mixup in time domain by adding chunks of same species, random species and nocall/noise
- Random gain adjustment of signal amplitude for individual chunks before mixing
- Random gain adjustment for the mixed signal
- Pitch shift and time stretch (both local and global in time and frequency domain)
- Addition of Gaussian/pink/brown noise
- Insertion of short noise bursts
- Addition of reverb (see remarks below)
- Utilization of different interpolation filters for spectrogram resizing
- Application of color jitter (brightness, contrast, saturation, hue) to the spectrogram image

A novel augmentation method introduced in this year's task is reverb augmentation. In soundscapes, recordings often capture birds from a large distance, resulting in weaker sounds with more reverb and attenuated high frequencies compared to the typically cleaner sounds in Xeno-canto files, where the microphone is directly targeted at the bird. To address this difference between training and test data, reverb is added to the training files using impulse responses recorded from the Valhalla Vintage Verb audio plugin [45]. During training, randomly selected impulse responses are convolved with the original audio signal with a probability of 20%, employing a dry/wet mix control that ranges from 0.2 (almost dry signal) to 1.0 (only reverb). Figure 2 provides examples, illustrating the influence of reverb augmentation on the resulting spectrogram image.

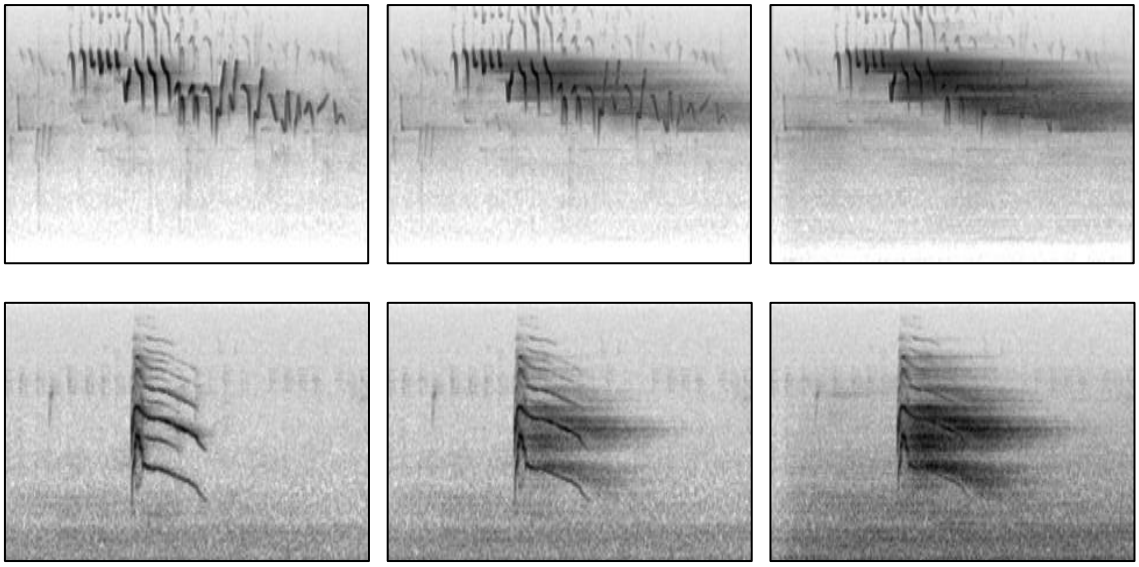

Figure 2: Reverb augmentation examples (left: original, mid.: dry/wet mix 0.5, right: dry/wet mix 0.8)

3. Results

The approach described in this paper secured the 3rd place among a total of 1189 participating teams and the 1st place on public leaderboard, representing a smaller subset of the test data. Final scores on private and public leaderboard (LB) and ranking of the top 10 teams are presented in Table 2. By combining several diverse models (including different CNN backbones, training hyperparameters and dataset folds) a macro-averaged mean average precision score (cmAP) of 76.3 % was achieved on the complete test set (see team 'adsr' in Table 2).

Due to differences in Kaggle's hardware (particularly CPU types) used to run inference notebooks, the number of models that could be ensembled to identify all birds in the test set in the given time varied. To prevent submission errors, a timer was implemented in the notebook to ensure completion within the 2-hour limit. If the timer reached approximately 118 minutes, inferencing was halted and results were collected for all models and predicted file parts up to that point. Predictions from unfinished models or file parts were masked before averaging. This approach makes it difficult to determine the exact number of models that can be ensembled. Initially, only 3 models could be ensembled without risking timeouts. Later, inference speed was prioritized over model diversity by using models with the same inputs (consistent FFT size, number of Mel bands, etc.). This allowed pre-calculation and saving of Mel spectrogram images to memory for all files in advance, which were then reused for each model. Additionally, models were converted to TorchScript and the preprocessing of test files was parallelized using multiple CPU threads. With these optimizations, at least 7 models could be ensembled, depending on the backbone architecture, without setting a timer (e.g., 4 EfficientNetB0 + 3 EfficientNetV2s).

A fairly good cmAP of 74.2% on the complete test set (Kaggle's private leaderboard) can be achieved with a single, small and very fast EfficientNetB0-based model (see M6 in Table 3). The best single model, which utilizes a ResNet50 backbone, achieves a score of 74.8% on the private leaderboard (M8 in Table 3). On public leaderboard, the highest score is achieved by a model with an EfficientNetV2s backbone (M9 in Table 3). The overall best system was not submitted for the final ranking. It achieves a cmAP score of 76.4 % with an ensemble of 8 models (5 EfficientNetB0 + 3 EfficientNetV2s).

Table 2: Competition results of the top 10 teams (with solution of team adsr describes in this paper)

Rank	Team Name on Kaggle	cmAP [%] (priv. LB)	cmAP [%] (publ. LB)
1	Volodymyr	**76.392**	84.444
2	griffith	76.369	84.292
3	adsr	76.309	**84.735**
4	atfujita	75.688	84.096
5	Yevhenii Maslov	75.498	83.847
6	anonamename	75.384	83.391
7	MSU+YSDA+HSE	75.347	83.442
8	furu-nag	75.285	83.735
9	Synergy	75.201	83.474
10	LeonShangguan	74.962	83.181

4. Ablation Study

Table 3 illustrates the contributions of important aspects and novel approaches described in this paper on model performance. Model M1 serves as a baseline for comparison, utilizing an EfficientNetB0 backbone with the modified SED head mentioned earlier. Initially, audio chunks were selected without any weighting and only official competition data from this and previous years were used for training. As the model progressed from M1 to M2 and M4, by incorporating pseudo labels and adopting weighted audio chunk selection, there was a noticeable increase in performance. The use of EfficientNetV2s backbones improved the score on the public but not necessarily on the private leaderboard (M3 vs. M2 & M9 vs. M6). The inclusion of additional files from Xeno-canto in the training data slightly contributed to score improvement (M5 vs. M4). Notably, the introduction of reverb augmentation significantly enhanced performance and proved to be an effective method to compensate for the domain shift between Xeno-canto files and soundscapes (M6 vs. M5). The modified SED version, with attention on frequency bands, surpassed the original SED architecture, which focused on time frames (M6 vs. M7). Although ResNet-based models achieved commendable scores (M8), they were not included in the final ensembles due to less favorable tradeoffs between model accuracy and inference time compared to EfficientNet-based architectures.

Table 3: Influence of individual methods and network architectures on model performance

ID	Description	cmAP [%] (priv. LB)	cmAP [%] (publ. LB)
M1	Baseline (EfficientNetB0, chunk selection unweighted)	71.553	80.949
M2	M1 w. chunk selection weighted 75 % RMS, 25 % pseudo label	71.786	81.289
M3	M2 with EfficientNetV2s	71.713	81.853
M4	M3 w. chunk selection weighted 45 % RMS, 15 % pseudo label	73.010	82.299
M5	M4 with EfficientNetB0 and extended 2023 Xeno-canto data	73.733	82.808
M6	M5 with reverb augmentation	74.246	83.164
M7	M6 with original SED head using attention on time frames	74.002	82.897
M8	M6 with ResNet50	**74.820**	83.288
M9	M6 with EfficientNetV2s	74.104	**83.386**

5. Discussion

The BirdCLEF 2023 task introduced some interesting and welcome changes compared to previous years. The competition's focus on cmAP as the evaluation metric eliminated the need for threshold tuning, while the inference time constraint encouraged the development of efficient models with a good balance between accuracy and speed.

While this paper discusses successful approaches, several other methods were explored but didn't yield satisfactory results. These included for example experimenting with different loss functions, creating model soups [46], applying knowledge distillation, finetuning models with data containing only species of the test set or further optimizing CPU inference by converting models to ONNX [47] or OpenVINO [48] formats. Advanced postprocessing techniques, such as adjusting probabilities in neighboring audio segments or weighting model predictions in the ensemble, also did not lead to further score improvements.

Modifying the original SED architecture to incorporate frequency instead of time attention proved effective in recognizing birds in soundscapes, where multiple species vocalize at the same time but in different frequency ranges. However, the frequency dimensionality of the SED output is much smaller than the frequency resolution of the input spectrogram due to multiple max pooling operations in the CNN encoder. Increasing the frequency resolution by adjusting preceding layers in the feature encoder could further improve identification performance for recordings with a high degree of overlapping sounds. Exploring models that combine time and frequency attention within the same network would be also a promising avenue for future research. In addition to pure classification, such models would allow to annotate individual sound events in both time and frequency within the spectrogram.

The performance of models heavily relies on the quantity and quality of the training data. If the model is deployed to identify birds in soundscapes, the training data should be representative of that scenario. But unlike recordings with only a few birds and good signal to noise ratio, annotating soundscapes can be very time-consuming and requires expert knowledge. In addition to mixing audio segments with different sounds and noise characteristics, incorporating reverb into the audio signal can help to bridge the gap between clean recordings and soundscapes. If a system is designed to identify birds in a specific area or habitat it might be worth to create impulse responses of the target location and use those for reverb augmentation during training to simulate the characteristics of sound propagation in that area.

A model similar to the ones developed for this competition, trained to identify European bird species, is available at https://code.naturkundemuseum.berlin/tsa/birdid-europe254-2103. The model adopts the modified SED architecture and many of the training methods described in this paper. It has already been successfully implemented in various projects to assess avian biodiversity [49, 50,51,52] and is part of Naturblick [53], a mobile application for discovering and learning about nature in urban areas.

6. Acknowledgements

I would like to thank Sohier Dane, Stefan Kahl, Tom Denton, Holger Klinck and all involved institutions (Kaggle, Chemnitz University of Technology, Google Research, K. Lisa Yang Center for Conservation Bioacoustics at the Cornell Lab of Ornithology, LifeCLEF, NATURAL STATE, OekoFor GbR and Xeno-canto) for organizing this competition. I also want to thank the Museum für Naturkunde and the team of the Animal Sound Archive Berlin [54] in particular Karl-Heinz Frommolt, Olaf Jahn and Benjamin Werner for supporting my work. The research was partly funded by the BMEL (Bundesministerium für Ernährung und Landwirtschaft) within the project “Machbarkeitsstudie - Integration (bio-)akustischer Methoden zur Quantifizierung biologischer Vielfalt in das Waldmonitoring” (FKZ: 2221NR050B).

7. References

[1] https://www.kaggle.com/competitions/birdclef-2023
[2] https://www.imageclef.org/BirdCLEF2023
[3] Kahl S, Denton T, Klinck H, Reers H, Cherutich F, Glotin H, Goëau H, Vellinga WP, Planqué R, Joly A (2023) Overview of BirdCLEF 2023: Automated bird species identification in Eastern Africa. In: Working Notes of CLEF 2023 – Conference and Labs of the Evaluation Forum
[4] https://www.imageclef.org/LifeCLEF2023
[5] Joly A, Botella C, Picek L, Kahl S, Goëau H, Deneu B, Marcos D, Estopinan J, Leblanc C, Larcher T, Chamidullin R, Šulc M, Hrúz M, Servajean M, Glotin H, Planqué R, Vellinga WP, Klinck H, Denton T, Eggel I, Bonnet P, Müller H (2023) Overview of LifeCLEF 2023: evaluation of ai models for the identification and prediction of birds, plants, snakes and fungi. In: International Conference of the Cross-Language Evaluation Forum for European Languages, Springer, 2023.
[6] Aliannejadi M, Faggioli G, Ferro N, Vlachos M (Ed.) (2023) Working Notes of CLEF 2023 - Conference and Labs of the Evaluation Forum
[7] Arampatzis A, Kanoulas E, Tsikrika T, Vrochidis S, Giachanou A, Li D, Aliannejadi M, Vlachos M, Faggioli G, Ferro N (Ed.) (2023) Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2023)
[8] Sprengel E, Jaggi M, Kilcher Y, Hofmann T (2016) Audio based bird species identification using deep learning techniques. In: CEUR Workshop Proceedings.
[9] Kahl S, Wilhelm-Stein T, Hussein H et al. (2017) Large-Scale Bird Sound Classification using Convolutional Neural Networks. In: CEUR Workshop Proceedings.
[10] Grill T, Schlüter J (2017) Two Convolutional Neural Networks for Bird Detection in Audio Signals. In: 25th European Signal Processing Conference (EUSIPCO2017). Kos, Greece. https://doi.org/10.23919/EUSIPCO.2017.8081512
[11] Sevilla A, Glotin H (2017) Audio bird classification with inception-v4 extended with time and time-frequency attention mechanisms. In: CEUR Workshop Proceedings.
[12] Stowell D, Stylianou Y, Wood M, Pamuła H, Glotin H (2018) Automatic acoustic detection of birds through deep learning: the first Bird Audio Detection challenge. In: Methods in Ecology and Evolution
[13] Lasseck M (2019) Bird Species Identification in Soundscapes. In: CEUR Workshop Proceedings.
[14] Lasseck M (2018) Acoustic Bird Detection with Deep Convolutional Neural Networks. In: Plumbley MD et al. (eds) Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018), pp. 143-147, Tampere University of Technology.
[15] Lasseck M (2018) Audio-based Bird Species Identification with Deep Convolutional Neural Networks. In: CEUR Workshop Proceedings.
[16] https://xeno-canto.org/
[17] Kahl S, Stöter FR, Goëau H, Glotin H, Planqué R, Vellinga WP, Joly A (2019) Overview of BirdCLEF 2019: Large-Scale Bird Recognition in Soundscapes. In: Working Notes of CLEF 2019 – Conference and Labs of the Evaluation Forum
[18] https://www.kaggle.com/competitions/birdsong-recognition
[19] Kahl S, Denton T, Klinck H, Glotin H, Goëau H, Vellinga WP, Planqué R, Joly A (2021) Overview of BirdCLEF 2021: Bird call identification in soundscape recordings. In: Working Notes of CLEF 2021 – Conference and Labs of the Evaluation Forum
[20] https://www.kaggle.com/c/birdsong-recognition/discussion/159970
[21] https://dcase.community/challenge2018/task-bird-audio-detection
[22] Stowell D, Stylianou Y, Wood M, Pamuła H, Glotin H (2018) Automatic acoustic detection of birds through deep learning: the first bird audio detection challenge. In: Methods in Ecology and Evolution, https://arxiv.org/abs/1807.05812, doi: 10.48550/arXiv.1807.05812
[23] https://www.kaggle.com/datasets/theoviel/bird-backgrounds
[24] https://www.kaggle.com/datasets/mariotsaberlin/xeno-canto-extended-metadata-for-birdclef2023
[25] McFee, Brian, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, and Oriol Nieto. "librosa: Audio and music signal analysis in python." In Proceedings of the 14th python in science conference, pp. 18-25. 2015.
[26] Paszke A et al. (2017) Automatic differentiation in PyTorch. In: NIPS-W
[27] Wightman R (2019) PyTorch Image Models. GitHub repository: https://github.com/rwightman/pytorch-image-models, doi: 10.5281/zenodo.4414861
[28] https://pypi.org/project/soundfile/
[29] https://scipy.org/

[30] https://github.com/iver56/audiomentations
[31] Pedregosa et al. (2011) Scikit-learn: Machine Learning in Python, JMLR 12, pp. 2825-2830, 2011.
[32] Deng J et al. (2009) Imagenet: A largescale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, 2009. pp. 248–255
[33] Kong, Qiuqiang, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D. Plumbley. "PANNs: Large-Scale Pretrained Audio Neural Networks for Audio Pattern Recognition." arXiv preprint arXiv:1912.10211 (2019).
[34] https://github.com/qiuqiangkong/audioset_tagging_cnn/
[35] S. Adavanne, H. Fayek & V. Tourbabin, "Sound Event Classification and Detection with Weakly Labeled Data", Proceedings of the Detection and Classification of Acoustic Scenes and Events 2019 Workshop (DCASE2019), pages 15–19, New York University, NY, USA, Oct. 2019
[36] https://www.kaggle.com/code/hidehisaarai1213/introduction-to-sound-event-detection/notebook
[37] Tan M, Le QV (2020) EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, arXiv:1905.11946, doi: 10.48550/arXiv.1905.11946
[38] Tan M, Le QV (2021) EfficientNetV2: Smaller Models and Faster Training. arXiv:2104.00298, doi: 10.48550/arXiv.2104.00298
[39] https://pytorch.org/docs/stable/generated/torch.nn.BCEWithLogitsLoss.html
[40] https://pytorch.org/docs/stable/generated/torch.optim.Adam.html
[41] https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.CosineAnnealingLR.html
[42] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.label_ranking_average_precision_score.html
[43] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.average_precision_score.html
[44] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html
[45] https://valhalladsp.com/shop/reverb/valhalla-vintage-verb/
[46] Wortsman M, Ilharco G, Gadre SY et al. (2022) Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In: arXiv:2203.05482, doi: 10.48550/arXiv.2203.05482
[47] https://github.com/onnx/onnx
[48] https://docs.openvino.ai/2023.0/home.html
[49] Wägele JW, Bodesheim P, Bourlat SJ, Denzler J et al. (2022) Towards a multisensor station for automated biodiversity monitoring. In: Basic and Applied Ecology (59), 105-138. doi: 10.1016/j.baae.2022.01.003
[50] https://ammod.de/
[51] Stehle M, Lasseck M, Khorramshahi O, Sturm U (2020) Evaluation of acoustic pattern recognition of nightingale (Luscinia megarhynchos) recordings by citizens. In: Research Ideas and Outcomes 6: e50233. doi: 10.3897/rio.6.e50233
[52] https://www.idmt.fraunhofer.de/en/institute/projects-products/projects/devise.html
[53] https://naturblick.museumfuernaturkunde.berlin/?lang=en
[54] https://www.museumfuernaturkunde.berlin/en/science/animal-sound-archive