Contents lists available at ScienceDirect

Ecological Informatics

journal homepage: www.elsevier.com/locate/ecolinf

A pipeline for identification of bird and frog species in tropical soundscape recordings using a convolutional neural network

Jack LeBien[a,*], Ming Zhong[b], Marconi Campos-Cerqueira[a], Julian P. Velev[c], Rahul Dodhia[b], Juan Lavista Ferres[b], T. Mitchell Aide[a,d]

[a] Sieve Analytics, San Juan, PR 00911, USA
[b] AI for Good Research Lab, Microsoft, USA
[c] Department of Physics, University of Puerto Rico, San Juan, PR 00931, USA
[d] Department of Biology, University of Puerto Rico, San Juan, PR 00931, USA

ARTICLE INFO

Keywords:
Acoustic monitoring
Bioacoustics
Sound classification
Convolutional neural network
Deep learning

ABSTRACT

Automated acoustic recorders can collect long-term soundscape data containing species-specific signals in remote environments. Ecologists have increasingly used them for studying diverse fauna around the globe. Deep learning methods have gained recent attention for automating the process of species identification in soundscape recordings. We present an end-to-end pipeline for training a convolutional neural network (CNN) for multi-species multi-label classification of soundscape recordings, starting from raw, unlabeled audio. Training data for species-specific signals are collected using a semi-automated procedure consisting of an efficient template-based signal detection algorithm and a graphical user interface for rapid detection validation. A CNN is then trained based on mel-spectrograms of sound to predict the set of species present in a recording. Transfer learning of a pre-trained model is employed to reduce the necessary training data and time. Furthermore, we define a loss function that allows for using true and false template-based detections to train a multi-class multi-label audio classifier. This approach leverages relevant absence (negative) information in training, and reduces the effort in creating multi-label training data by allowing weak labels. We evaluated the pipeline using a set of soundscape recordings collected across 749 sites in Puerto Rico. A CNN model was trained to identify 24 regional species of birds and frogs. The semi-automated training data collection process greatly reduced the manual effort required for training. The model was evaluated on an excluded set of 1000 randomly sampled 1-min soundscapes from 17 sites in the El Yunque National Forest. The test recordings contained an average of ~3 present target species per recording, and a maximum of 8. The test set also showed a large class imbalance with most species being present in less than 5% of recordings, and others present in > 25%. The model achieved a mean-average-precision of 0.893 across the 24 species. Across all predictions, the total average-precision was 0.975.

1. Introduction

Acoustic monitoring has gained widespread interest as an ecological tool for wildlife population assessment, conservation, and biodiversity research. Many species emit regular vocalizations or other acoustic signals that are species-specific, which enables monitoring via sound recognition. Advances in automated acoustic recorders have reduced prices and enabled data collection for months at a time. This has resulted in enormous data sets; however, like the evolution of many data-driven approaches, including camera traps and eDNA, methods for data collection are progressing faster than those for effective analysis and interpretation. In many cases, acoustic analyses are done manually, and this often limits the analyses to a subset of the complete datasets (Potamitis et al., 2014; Priyadarshani et al., 2018; Swiston and Mennill, 2009). To enable analysis of entire datasets, accurate, automated sound recognition methods are paramount.

Efforts to automate species identification in audio recordings have spanned diverse fauna such as birds, amphibians, bats, insects, fish and marine mammals (e.g. Ganchev, 2017). Recently, machine learning approaches have been successfully applied to acoustic data for identifying multiple species (Priyadarshani et al., 2018). Deep learning models such as convolutional neural networks (CNNs) have achieved remarkable performance (Florentin et al., 2020; Kahl et al., 2019; Ruff et al., 2019). Deep learning models have the advantage of incorporating

[*] Corresponding author.
E-mail address: jlebien@uno.edu (J. LeBien).

https://doi.org/10.1016/j.ecoinf.2020.101113
Received 29 January 2020; Received in revised form 14 April 2020
Available online 08 June 2020

feature learning in the training process, which eliminates or reduces the manual feature selection required. They are also often capable of scaling to a high number of classes. For example, in the 2016 BirdCLEF challenge, models were trained to recognize 999 bird species and evaluated on both omnidirectional soundscapes (i.e. ambient field recordings) containing multiple species and monodirectional recordings targeting single species (Goëau et al., 2016). CNNs achieved a significant performance increase in the challenge, compared to other methods that were mostly based on nearest neighbors or decision trees (Goëau et al., 2016; Sprengel et al., 2016). CNNs are a type of deep neural network that have achieved state-of-the-art performance on many image-recognition tasks (Aloysius and Geetha, 2017). Their network structure is characterized by layers designed for spatially-invariant image feature extraction. CNNs have also been successfully applied to many other sound recognition tasks, including bioacoustic recognition for taxonomic groups other than birds, and human voice recognition (Colonna et al., 2016; Kao et al., 2018). For sound recognition, CNNs typically use audio spectrograms as input, whereas mel-frequency cepstral coefficients (MFCCs) and other spectral statistics are common for other sound recognition models (Priyadarshani et al., 2018). Single CNNs can model many classes, as demonstrated in BirdCLEF and benchmark image recognition tasks (e.g. ImageNet, Deng et al., 2009) (Goëau et al., 2016; He et al., 2015). This is of particular interest for biodiversity monitoring efforts with many target species. These points highlight the potential of CNNs as an important tool for the ecological community.

Sprengel et al. (2016) used a six-layer single-label CNN trained on audio spectrograms to classify audio recordings targeting foreground bird species. They used data augmentation techniques such as combining same-class audio samples and samples of noise. Ruff et al. (2019) collected training samples directly from soundscape recordings and implemented a six-layer convolutional neural network for detection and classification of six owl species, incorporating a noise class to account for audio containing no target species. Florentin et al (2019) repurposed several popular pre-trained CNN image classifiers for the detection and classification of birds in soundscape recordings. They also used a noise class and, notably, found that including previously-identified false detections of target signals in the noise class improved performance. They also found that deeper network architectures generally performed the best on field (soundscape) data. Incze et al. (2018) repurposed a pre-trained MobileNet CNN (Howard et al., 2017) for single-species classification of bird audio recordings, and found that mapping grayscale input spectrograms to a color scheme improved performance (Incze et al., 2018). Other studies have found that combining various types of spectrograms into 3-channel images have improved performance for CNNs pre-trained for image classification (Sevilla et al., 2017; Xie et al., 2018; Xie et al., 2019). Sevilla et al. (2017) achieved relatively high-performance classification of single-species recordings and soundscapes by using an Inception-type CNN architecture (Szegedy et al., 2015) with time and time-frequency attention mechanisms. These studies have used single-label models for classification, and aggregated predictions over segments of soundscape recordings to make multi-label predictions. However, other studies have reported improved performance in soundscape classification with multi-label prediction models (Kahl et al., 2017; Zhang et al., 2016). As soundscape recordings can contain simultaneous occurrences of different target signals, multi-label prediction is a natural choice. However, the number of multi-label soundscape classification studies is limited due to the increased difficulty of acquiring multi-label training data. The best-performing team of the 2019 BirdCLEF challenge, which focused on soundscape recordings containing multiple species, used pre-trained ResNet (He et al., 2015) and Inception-type CNN models repurposed for classification of mel-scaled spectrograms of audio clips, and achieved the best performance with ensembled single-label and multi-label models (Kahl et al., 2019). Various data augmentation techniques were also found to significantly improve performance, such as time and frequency shifting and stretching of target signals, and adding Gaussian noise or noise from soundscapes (Koh et al., 2019; Lasseck, 2019). This competition's training data consisted primarily of recordings targeting single foreground species, but also a smaller validation set of annotated soundscapes like those in the test set. Results from this challenge showed a significant increase in performance for submissions that incorporated the validation soundscape data in training (Kahl et al., 2019). Specifically, Lasseck (2019) found that adding background noise from the validation soundscapes to training samples significantly improved performance on the test soundscapes. This indicates that acoustic monitoring systems can benefit significantly from location-specific training data.

Several challenges remain to be addressed for effective application of deep learning in acoustic monitoring. First, CNNs often require many training samples for each class. Large-scale species recognition efforts, such as for the BirdCLEF challenge, often use crowd-sourced public training data from various geographic sources (e.g. Xeno-Canto dataset, Vellinga, 2020;), and thus far these public datasets focus on birds and consist mainly of recordings targeting single foreground species. However, as mentioned above, geographic variation in soundscapes could require training data collected at a local or regional scale for high performance. This, however, would greatly increase the data labeling effort. Second, the non-directional nature of soundscape recordings demands accurate detection as well as classification for species recognition. Existing studies have demonstrated the difficulty of species recognition in soundscapes, compared to recordings that target a single foreground signal (Goëau et al., 2015; Goëau et al., 2016; Goëau et al., 2017; Goëau et al., 2018; Kahl et al., 2019). While many studies have evaluated models for classification of prior signal detections, methods that integrate both detection and classification of multiple species in noisy soundscapes are scarcer (Priyadarshani et al., 2018). Soundscape recordings often contain multiple species with calls overlapping in time and frequency, and a variety of environmental noises. Thus, models are challenged with achieving accurate multi-label recognition of simultaneous sounds, and a low false detection rate despite a variety of input noise. Third, although multi-label prediction is naturally desired for soundscapes containing different target signals that frequently overlap in time, collection of multi-label training data is typically more difficult than single-label data due to the demand of fully-labeling all target signals in each sample. Furthermore, class imbalance can be an issue for multi-label data because the number of positive instances of each class is more difficult to control than single-label data.

Considering these challenges, methods that mitigate region-specific training data collection effort and optimize model precision (minimizing false positives) in soundscape recordings are desired. To address this, we developed a pipeline for training CNNs for multi-species multi-label classification of soundscape recordings using single-label true and false-positive detections of each species. For the purpose of training data collection, we created an efficient implementation of a template-based sound detector and a graphical user interface for post-detection validation. A custom training loss was used to enable multi-label learning from single-label training data, and incorporate false-positive detections into training, which improved model precision in the presence of simultaneous and frequency-overlapping call types. This approach to model building: 1) mitigates the manual effort in collecting multi-label training data directly from soundscapes, 2) incorporates relevant absence information for each class by using false-positives, and 3) allows for controlling the number of positive and negative examples of each class in multi-label learning. In a related study (Zhong et al., 2020), we evaluated several CNN architectures and training methods using 2-s single-labeled audio clips for classification. In this paper, we apply the best-performing method to multi-label classification of soundscape recordings, and describe an end-to-end pipeline for model creation, starting from unlabeled soundscapes. The template-based detection and validation tools were used to generate the CNN training dataset in a semi-automated fashion, and greatly reduced the manual effort required. ResNet50 transfer learning was applied to reduce the necessary training data and time. High precision and recall were achieved in predicting the presence of 24 tropical bird and frog species from 1-min soundscape recordings in the El Yunque National Forest. The presented pipeline is highly generic and expected to be applicable to many call types from diverse habitats.

2. Methods

2.1. Data acquisition

Soundscape recordings previously collected throughout Puerto Rico for other projects were used to create training and test datasets. AudioMoth recorders (Hill et al., 2018) were used to collect acoustic data. Recorders were placed on trees at the height of 1.5 m and programmed to record 1 min of audio every 10 min for a total of 144 recordings per day at a sampling rate of 48 kHz. A small portion (~10%) of recordings were sampled at 44.1 kHz. All recordings were stored in the ARBIMON web-based platform (Aide et al., 2013). In total, 97,900 1-min soundscape recordings were weakly or fully annotated with species-specific time-frequency bounding boxes as detailed in the next section. These recordings came from 749 sites across the island, however, 49% of recordings with annotations came from 152 sites throughout the El Yunque National Forest (Fig. 1). Recordings from other sites and older years were searched to increase the training sample size for certain species. 78% of recordings were collected in 2019, 11% in 2018, and 11% collected between 2015 and 2018. Most recordings were collected in the months March and April, a period of high acoustic activity.

One thousand recordings collected in April 2019 were randomly selected from 17 sites in the El Yunque National Forest, and used as a test set. The 17 sites were selected to cover species-rich habitats, and various elevations and monitoring transects throughout the forest.

2.2. Training data collection

Twenty-four species of bird and amphibian were chosen to be included in the CNN sound recognition model (Table 1). The species in this study included species of "Great Conservation Need" according to the State Wildlife Action Plan (Puerto Rico State Wildlife Action Plan, 2015) and regionally common species (e.g., *Eleutherodactylus coqui, Margarops fuscatus, Turdus plumbeus, Patagioenas squamosa*). For each species, examples of the one or two most common call types were collected for the purpose of CNN training (Fig. 2). To collect CNN training data for each species, we applied a template-based signal

Table 1
Numbers of true and false-positive template-based detections used for model training.

Species	Abbreviation	Taxon	True positives (*tp*)	False positives (*fp*)
Eleutherodactylus unicolor	ELUN	Frog	18,810	2209
Eleutherodactylus brittoni	ELBR	Frog	9369	6003
Eleutherodactylus wightmanae	ELWI	Frog	7690	10,222
Eleutherodactylus coqui	ELCO	Frog	6007	1966
Eleutherodactylus hedricki	ELHE	Frog	4687	10,318
Eleutherodactylus gryllus	ELGR	Frog	3682	8718
Eleutherodactylus richmondi	ELRI	Frog	3327	10,087
Eleutherodactylus portoricensis	ELPO	Frog	2550	4087
Eleutherodactylus locustus	ELLO	Frog	2492	2523
Eleutherodactylus antillensis	ELAN	Frog	1513	8685
Leptodactylus albilabris	LEAL	Frog	812	6505
Vireo altiloquus	VIAL	Bird	7745	9942
Loxigilla portoricensis	LOPO	Bird	3391	15,804
Patagioenas squamosa	PASQ	Bird	2162	2276
Spindalis portoricensis	SPPO	Bird	2024	10,080
Nesospingus speculiferus	NEES	Bird	1771	9226
Megascops nudipes	MENU	Bird	1634	3100
Margarops fuscatus	MAFU	Bird	1576	8054
Setophaga angelae	SEAN	Bird	1261	11,099
Turdus plumbeus	TUPL	Bird	976	7466
Melanerpes portoricensis	MEPO	Bird	932	26,290
Todus mexicanus	TOME	Bird	906	6143
Coereba flaveola	COFL	Bird	859	1748
Coccyzus vieilloti	COVI	Bird	476	6357

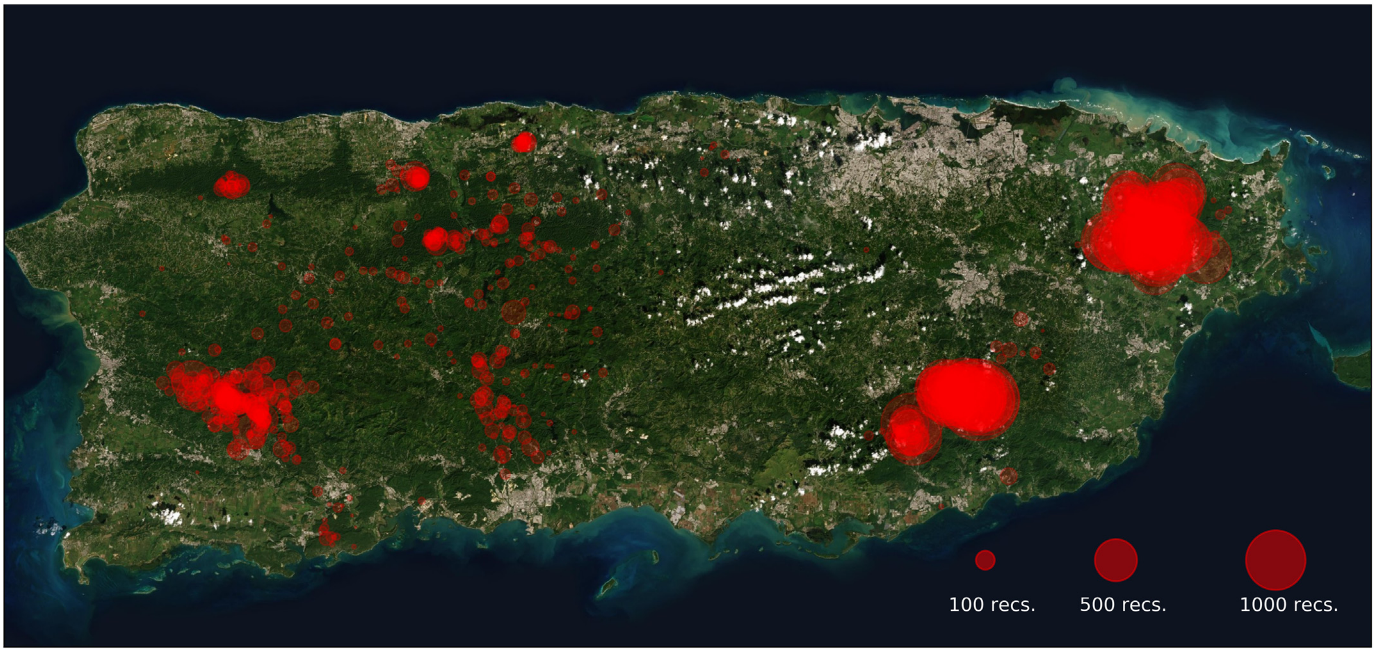


Fig. 1. Spatial distribution of recordings used for training and testing. Circles represent recorder sites with diameter indicating the number of recordings used. Most recordings came from El Yunque National Forest (large cluster in upper right) and the test recordings were sampled from this area.

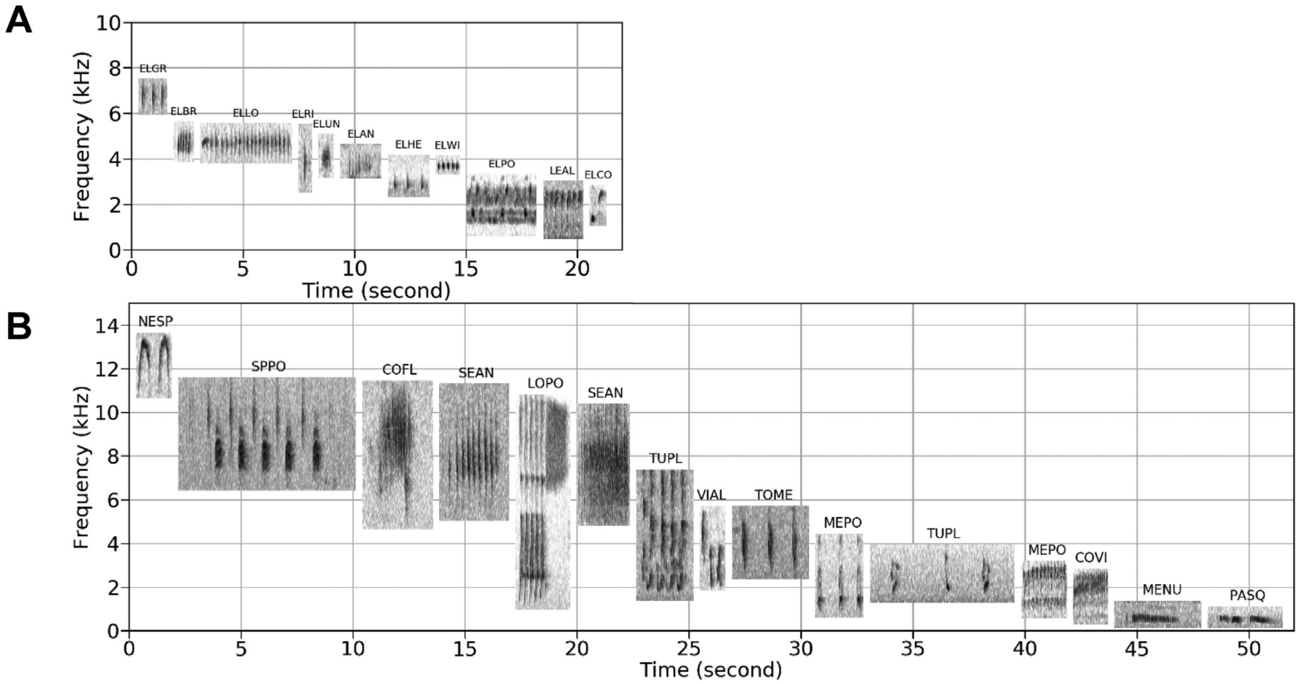


Fig. 2. Templates for each target species of frog (A) and bird (B). Species abbreviations are the same as in Table 1. The spectrograms show the pattern of acoustic energy in time and frequency of each call type, with darker values indicating higher amplitude. The template position with respect to the vertical axis indicates the frequency range of the call. The horizontal extent of the templates represents their time duration. The axes of the two plots are shown in proportion.

detection algorithm. This process consisted of three primary steps:

2.2.1. Template creation

Based on prior knowledge of the target species' sound repertoires, one or two of the most common call types were chosen as template signals for each species. The ARBIMON Visualizer feature was used to search audio spectrograms for a high signal-to-noise-ratio example of each call type. Time-frequency bounding boxes were drawn around the identified template calls and labeled by species and call-type. This metadata was stored in the platform for later analyses.

2.2.2. Template match detection

Using the ARBIMON Pattern Matching feature, developed as part of this study, each template was used to search through a playlist of recordings for calls that matched the template. This procedure detects time-localized signals with a correlation equal to or greater than a threshold assigned by the user. The correlation is computed in the spectrogram (time-frequency) domain.

The function *spectrogram* from the Python package *SciPy* is used for spectrogram generation (Oliphant, 2007). Spectrograms are computed from Hann-windowed 512-sample (~10 ms) segments of audio data, with 50% overlap between segments, and 512 Fast Fourier Transform (FFT) coefficients per segment. Time-frequency bin amplitudes are log-scaled. Note that these spectrogram parameters used for template match detection are different from those used to create the CNN inputs, which are described below.

The stored time-frequency coordinates of the template are used to extract it from its source recording. If necessary, playlist recordings are re-sampled to the sampling rate of the template's source recording. For each spectrogram in the playlist, frequency bins outside the template's frequency range are discarded, resulting in an image height equal to that of the template. For each remaining frequency bin, the median intensity over time is subtracted. This "flattening" step reduces the influence of acoustic processes that are approximately stationary throughout the recording (e.g. some insects, rain) in the search for time-localized signals (e.g. vocalizations).

The normalized cross-correlation (NCC) is then computed between the template T and the cropped, flattened spectrogram S, defined as follows:

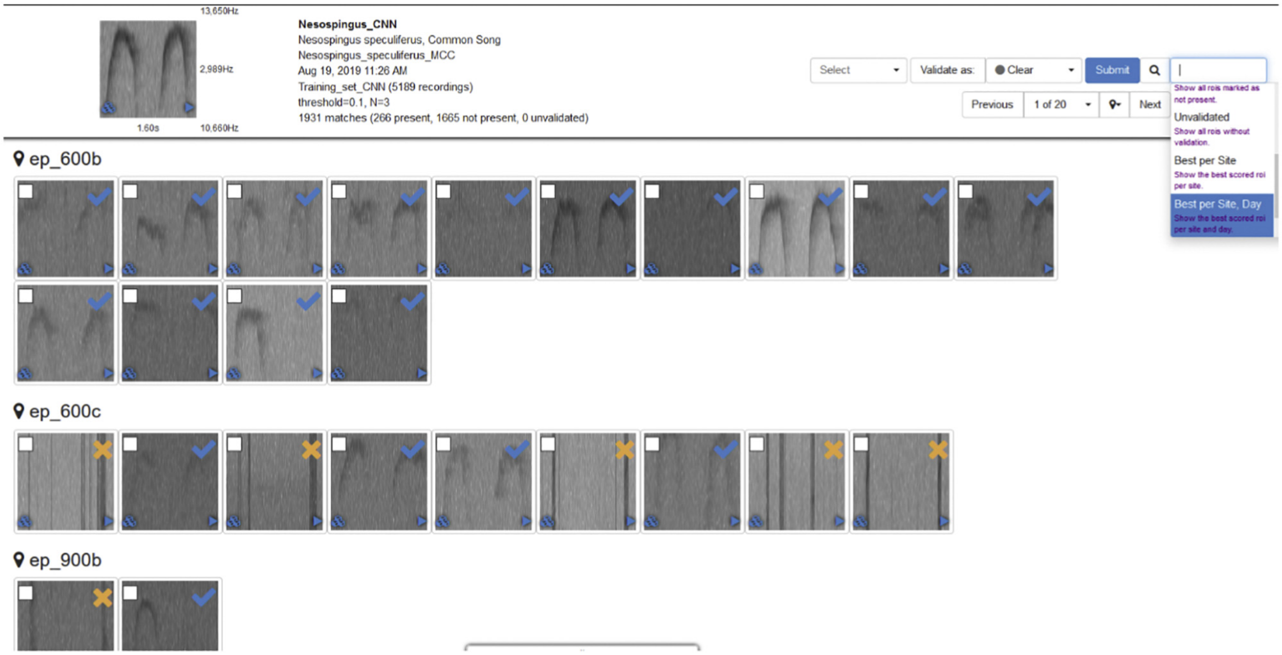
$$\gamma(u) = \frac{\sum_{t,f}[S(t,f)-\overline{S}_u][T(t-u,f)-\overline{T}]}{\left\{\sum_{t,f}[S(t,f)-\overline{S}_u]^2\sum_{t,f}[T(t-u,f)-\overline{T}]^2\right\}^{1/2}} \quad -1 \le \gamma(u) \le 1$$

where the sums are over t, f (time, frequency) under the window containing the template T shifted by u time bins; $\overline{S}_u$ is the mean of S under the window containing the shifted template; and $\overline{T}$ is the mean of the template. The normalization of each window of S and T to unit length eliminates the correlation's dependence on acoustic amplitude. The NCC is computed using the fast NCC algorithm, wherein the cross-correlation is computed as a pointwise product in the Fourier domain (Lewis, 1995). Note that the cross-correlation is computed along the time axis only. The NCC is computed using the function *match_template* from the python package *scikit-image* (van der Walt et al., 2014).

All local maxima in the resulting NCC vector are detected, and then filtered based on given criteria. First, correlation peaks below a given magnitude threshold are discarded. Peaks are then filtered based on a minimum time-distance threshold. Detected peaks must be separated by at least half the duration of the template. For this filtering step, all correlation peaks are iterated over in order of descending magnitude. For each peak visited, lower-magnitude peaks within the distance threshold (to the left or right) are discarded from the set. Correlation peaks are detected and filtered using the function *find_peaks* from the python package *SciPy* (Oliphant, 2007). The time coordinates of the correlation peaks satisfying the given criteria are returned as positive detections.

For most template matching analyses in this study, the NCC threshold was chosen to be low (i.e. 0.1). This resulted in a high number of false positives (*fp*'s; we use lowercase notation for template match errors to distinguish them from CNN prediction errors), though the number of false negatives (*fn*'s) was considered negligible.

A parallelized and scalable cloud-based implementation of the template matching algorithm was developed for the ARBIMON Pattern Matching feature, available at (https://arbimon.sieve-analytics.com). Input playlists are broken into batches with size determined by the



Nesospingus_CNN
Nesospingus speculiferus, Common Song
Nesospingus_speculiferus_MCC
Aug 19, 2019 11:26 AM
Training_set_CNN (5189 recordings)
threshold=0.1, N=3
1931 matches (266 present, 1665 not present, 0 unvalidated)

13,650Hz
2,989Hz
1.60s
10,660Hz
Select
Validate as:
● Clear
Submit
Previous
1 of 20
Next
Show all rois marked as not present.
Unvalidated
Show all rois without validation.
Best per Site
Show the best scored roi per site.
Best per Site, Day
Show the best scored roi per site and day.
ep_600b
ep_600c
ep_900b

Fig. 3. ARBIMON template match visualizer, allowing for sorting, filtering, and validation of detections, which can be used as CNN training data. In the example screenshot, we show the best matches per day for a *Nesospingus speculiferus* template for three sites.

playlist. Each batch is processed with a separate CPU with the ability to launch 1000s of simultaneous CPUs. At typical sampling rates (44.1 or 48 kHz), more than 50,000 1-min recordings (~1 month of audio) can be analyzed for detections in less than five minutes.

2.2.3. Template match validation

The resulting matches can be visually browsed in the ARBIMON platform (Fig. 3). The matches are displayed as time-frequency bounded spectrogram images with the same dimensions as the respective template. Individual detections can also be reviewed with frequency-filtered audio playback, and the source recordings can be easily accessed for audio-visual analysis. The detections can then be validated by checking each image as a true positive (*tp*) or *fp*. Preset queries such as the best *N* matches per recording, site and/or day can accelerate the assessment of the presence or absence of a species at a location.

Altogether, the ARBIMON Visualizer and Pattern Matching features described above allow for highly efficient training data creation (Fig. 3, 4a). Compared to manual inspection and annotation of bounding boxes in spectrograms, many potentially true-positive bounding boxes can be generated automatically in little time, requiring only post-validation.

In total, 512,471 soundscape recordings were searched for call detections using the above three steps. Different recording subsets were searched for different species. This resulted in 86,652 *tp* and 188,908 *fp* annotated time-frequency bounding boxes for 24 species. (Table 1). Note that *fp*'s of a given species *A* that contained the call of a different target species *B* were not re-classified as *tp*'s for *B*. For model training, all *fp*'s were simply used as examples of absence, as detailed in the next section. These true and false detections were used to create the CNN training samples as described below.

2.3. Model training

2.3.1. Training data preprocessing

The CNN model used in this study requires equally sized input images. We chose a time-frequency input window size of 2 s as it is near the mean and median template duration across target call types (Fig. 2). Most call types have a duration below 1 s, and for those above 2 s, important features can still be captured within 2 s. For the case of input frequency bandwidth, we chose to use the entire range of 24 kHz. This was chosen over a smaller, more focused bandwidth for several reasons. Firstly, we approach training as a multi-label classification problem. In other words, for each input audio segment, the model is trained to predict the set of all species present, rather than a single foreground class. This eliminates the need to focus on single target calls in the input. Also, many species' calls highly overlap in frequency (Fig. 2). So, while a focused bandwidth could separate several call types by frequency and avoid their presence together in the same input, many calls would still potentially be present together. A large bandwidth also reduces the number of predictions required to cover a 1-min recording. Furthermore, it increases model generalizability because the optimal bandwidth would likely be specific to the set of target species.

From each template-based detection, we extracted spectrogram images representing two seconds of audio time-centered on the detection and spanning 24 kHz (Fig. 4a). If needed, audio files were resampled to 48 kHz before spectrogram computation. Spectrograms were computed from Hann-windowed 1024-sample (~20 ms) segments of audio data, with 50% segment overlap, and 2048 FFT coefficients per segment. Mel-scaling is a technique commonly applied in acoustic time-frequency analysis. The mel scale refers to a perceptual scale of pitch based on an empirical study of human hearing (Stevens et al., 1937). The conversion from *f* Hertz to *m* “mels” is commonly approximated as:

$$m = 2595\log_{10}\left(1 + \frac{f}{700}\right)$$

Mel-scaling emphasizes lower frequencies while reducing high-frequency resolution, similar to log-scaling the frequency axis. Many of our target call types occupied a low-frequency band ($<$ 6 kHz), which motivated the use of the mel scale. The 1025 frequency bands in our training samples were converted to 128 mel-scaled frequency bands, using the *Librosa* Python package (McFee et al., 2015) (Fig. 4a). Training sample spectrograms were mapped to a color space using the *Librosa* function *specshow*.

2.3.2. Model architecture

We implemented the CNN using the Keras application programming interface to the Tensorflow Python library (Abadi et al., 2016; Chollet,

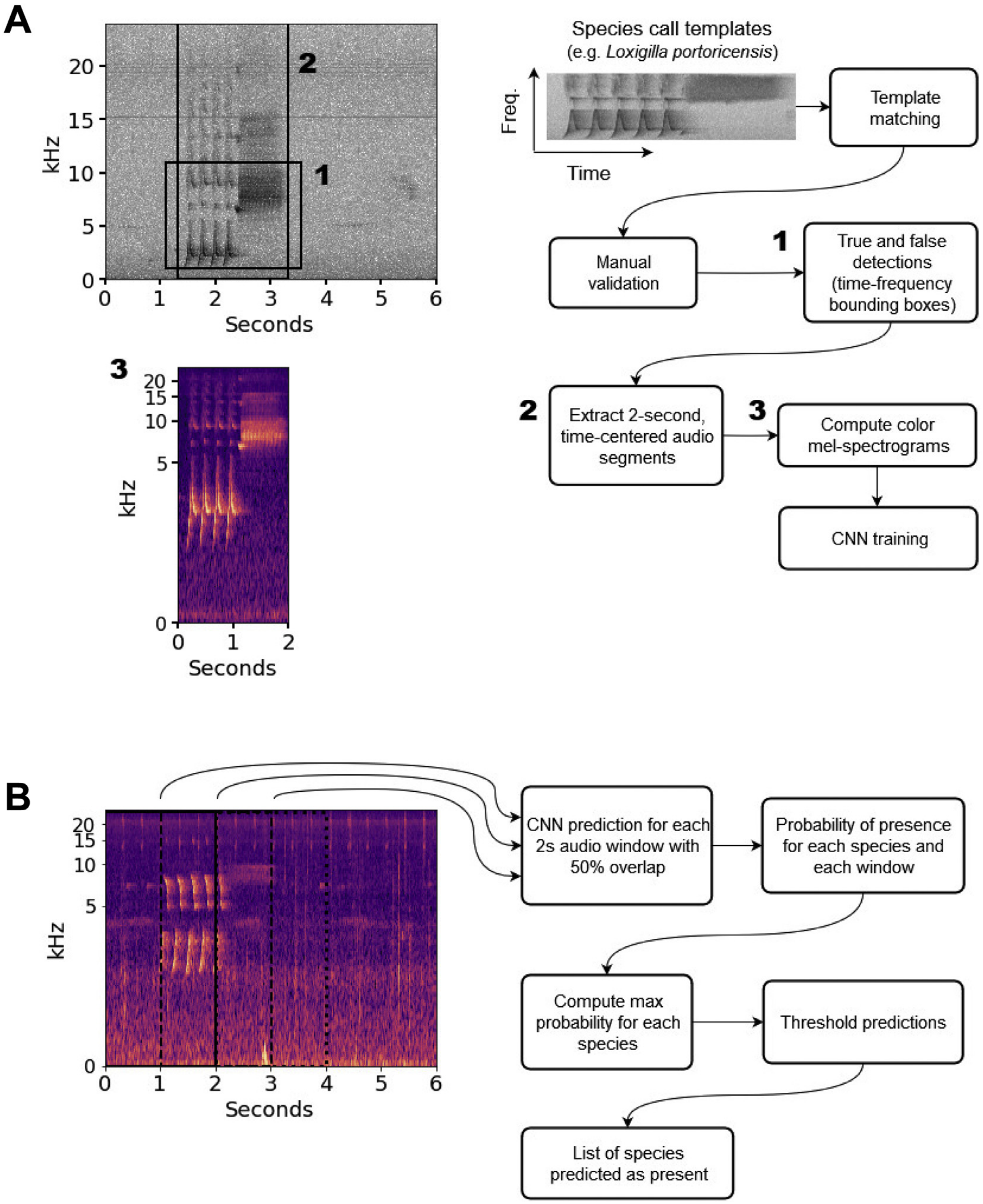

A
2
1
3
kHz
Seconds
Species call templates
(e.g. *Loxigilla portoricensis*)
Freq.
Time
Template matching
Manual validation
True and false detections (time-frequency bounding boxes)
Extract 2-second, time-centered audio segments
Compute color mel-spectrograms
CNN training
B
CNN prediction for each 2s audio window with 50% overlap
Probability of presence for each species and each window
Compute max probability for each species
Threshold predictions
List of species predicted as present

Fig. 4. (Color) Flowcharts for (A) training and (B) prediction portions of the species identification pipeline.

2019). Code for the CNN training and prediction method described in the paper are available at https://github.com/Sieve-Analytics/arbimon2-cnn. The implementation described here builds on an evaluation of several CNN models for classification of 2-s true and false positive detections of the same species (Zhong et al., 2020). In this case, we choose the best-performing method and apply it to multi-label classification of 1-min soundscape recordings using single-label training data. The training data size and spectrogram parameters are also adjusted, and we did not include a pseudo-labeling step during training. Deep neural networks with initially randomized weights typically require large amounts of training data and time to achieve adequate performance. Improved performance is often obtained from transfer learning, wherein a model already optimized for a similar dataset is retrained with new data. We used a ResNet50 model, pre-trained on the ImageNet dataset, which contains over one million photo images across 1000 classes (Deng et al., 2009). Although ImageNet does not contain spectrograms, models pre-trained with the dataset learn a variety of image features that have been successfully tuned to spectrogram classification previously (Lasseck, 2019; Florentin et al., 2020). While previous studies have found optimal performance using ensembles of multiple CNN models (Kahl et al., 2017; Lasseck, 2019; Florentin et al., 2020), in the interest of prediction efficiency and reasonable memory requirements, we chose to evaluate a single ResNet50 model.

Our implementation only included the feature extraction layers of ResNet50, excluding the remaining layers used for ImageNet classification, referred to as the network “top”. The network top consists of fully connected (FC) layers for learning a predictive model from the input features. An FC layer consists of a set of nodes, each of which takes a weighted sum of the input's values and passes it through a transfer function. The weights of an FC layer are learnable. In our case, we used two FC layers separated by a drop-out layer. The drop-out layer causes nodes in the previous FC layer to be probabilistically ignored in each training iteration, such that a different subset of nodes is connected to the final FC layer at each iteration. This emulates training a group of different models and reduces the chance of overfitting. The first FC layer consisted of 512 nodes and used the common "ReLU" activation function, which simply converts negative inputs to 0. The following drop-out layer was assigned a drop-out rate of 0.5, such that each node was ignored with a 50% probability. The final layer consists of 24 nodes, corresponding to the number of target species, and each node was assigned a sigmoid activation function. The sigmoid function S, defined below, maps the input to the range [0,1], and is commonly used for the prediction of binary outcomes. In our case, an independent output score within [0, 1] for each species was desired, to allow for multi-species prediction of presence in audio segments. Thus, our model outputs a vector of 24 scores, representing the predicted probability of presence for each species.

$$S(x) = \frac{1}{1 + e^{-x}}$$

2.3.3. Loss and optimization

A custom training loss function was defined to leverage both true and false detection training data. Typically, for multi-label prediction, the training target vectors consist of 0's and 1's indicating the presence or absence of each class in the input. This assumes fully labeled training data (i.e. all labels are known). In our case, each training sample is only labeled for presence or absence of a single species, based on validation of the associated detection as *tp* or *fp*. Therefore, each target vector consisted of a 1 or 0 at the position of the species determined to be present or absent, and unknown values represented by NaN ("Not a Number") for all other species. The custom loss for class c, given the presence label y_c and predicted score p_c is defined

$$L_c = \begin{cases} -\left(y_c\log(p_c) + (1 - y_c)\log(1 - p_c)\right) & y_c \in \{0,1\} \\ 0 & y_c = NaN \end{cases}$$

The total loss for a sample is simply the loss for the single labeled species. This allows for multi-label learning based on single labels at a time, but requires examples of both presence and absence for each class. This training regime is therefore compatible with true and false-positive detections from other detectors as training data.

The Adam optimization method was used with a learning rate of 1×10^{-4} and decay 1×10^{-7} (Kingma and Ba, 2017). The Adam method allows for an adaptive learning rate during training and is commonly used for deep neural network training. A 10% validation split was used to compare the loss between training samples and unseen samples during training. Based on comparing training and validation loss at the end of each epoch, 5 epochs of training were applied.

2.4. Prediction and evaluation

Performance evaluation was based on the ability to predict the set of species present in each of 1000 randomly sampled 1-min test soundscape recordings from 17 sites in the El Yunque National Forest. As mentioned above, the test sites were selected to cover species-rich habitats, and a variety of elevations and monitoring transects in the forest. To create present-species labels for the test recordings, we used the template matching procedure described in Section 2.2. To do this, for each test recording and each call type, the three highest-correlating template matches were validated as *tp* or *fp*. Due to the low correlation threshold of 0.1, *fn*'s were assumed to be negligible. Therefore, the *tp* detections were used to create present species labels for the test recordings to evaluate model predictions.

The test recordings showed a large class imbalance (Table 2). Frequencies of call presence ranged from 0.3% of recordings (*Melanerpes portoricensis*) to 92.3% of recordings (*Eleutherodactylus coqui*). More than half of the species were in less than 5% of recordings and a quarter were in greater than 20% of the recordings. The number of species in test recordings ranged from 0 to 8, with a mode of 3. Only 1% of test recordings contained no calls, demonstrating the high acoustic activity in the study region.

2.4.1. Sliding window

Prediction was performed using a sliding window approach. Predictions were made for every 2-s time-window of each audio recording with a 1-s shift between windows (50% window overlap) (Fig. 4b). Spectrograms of each audio window were computed in the same way as for training data. For a species and recording, the highest predicted score across audio windows was used to predict presence. We also tested using the average of the two highest scores across windows as the predictive score.

In practice, a threshold τ is applied to the model's predicted scores to make a binary prediction of presence or absence. For a species and recording with predicted score p, we predict presence if $p \geq \tau$, and absence otherwise.

2.4.2. Evaluation metrics

Let $TP \equiv TP(\tau)$ represent number of true-positive CNN predictions (i.e. correct predictions of species presence). Similarly, let TN represent number of true negatives (i.e. correct predictions of absence), let FP be number of false positives (i.e. incorrect predictions of presence), and let FN be number of false negatives (i.e. incorrect predictions of absence). All counts are dependent on the chosen threshold τ.

Performance was quantified by several metrics described below. The precision $P(\tau)$ is the fraction of predictions of presence that are correct.

$$P(\tau) = \frac{TP}{N_P'} = \frac{TP}{TP + FP}$$

where $N_P{'}$ is the number of predicted presences. Recall $R(\tau)$, also known as sensitivity or true-positive rate, measures the fraction of presences that are correctly identified.

$$R(\tau) = \frac{TP}{N_P} = \frac{TP}{TP + FN}$$

where N_P represents the number of true presences. A precision-recall curve consists of the points in precision-recall space achieved at each possible threshold. Recall is typically the horizontal axis and precision the vertical axis. Based on the precision-recall curve we also measured the average-precision (AP) of predictions. The AP is defined:

$$AP = \sum_{i=2}^{N} (R(\tau_i) - R(\tau_{i-1}))P(\tau_i)$$

where $(\tau_1, \tau_2, \ldots, \tau_N)$ are the different thresholds to be evaluated, sorted in descending magnitude. Typically, the chosen thresholds $(\tau_1, \tau_2, \ldots, \tau_N)$ are the sorted predicted scores. The AP is the weighted sum of precisions at each threshold, using the increase in recall from the previous threshold as the weight. It approximates the integral of, or area under, the precision-recall curve. The AP is independent of the chosen threshold τ, so it is commonly used for model comparison. The mean-average-precision (mAP) across classes is commonly used in multi-label prediction evaluation.

$$mAP = \frac{1}{N_s}\sum_{s} AP_s$$

where N_s is the number of target species. The false-positive rate (FPR) is the fraction of true absences incorrectly predicted as presences.

$$FPR(\tau) = \frac{FP}{N_A} = \frac{FP}{FP + TN}$$

where N_A represents the number of true absences.

Table 2
Species-specific and summary evaluation scores for a selected prediction threshold of 0.99. The total scores are computed from all predictions across all species and test recordings. The mean scores are computed for each species separately before averaging.

	Presences	Absences	TP	FP	TN	FN	Precision	Recall	FPR
Eleutherodactylus richmondi	123	877	120	2	875	3	0.98	0.98	0.002
Eleutherodactylus coqui	923	77	843	0	77	80	1.00	0.91	0.000
Eleutherodactylus unicolor	474	526	418	1	525	56	1.00	0.88	0.002
Eleutherodactylus portoricensis	255	745	224	1	744	31	1.00	0.88	0.001
Eleutherodactylus locustus	44	956	44	2	954	0	0.96	1.00	0.002
Eleutherodactylus gryllus	113	887	96	1	886	17	0.99	0.85	0.001
Vireo altiloquus	149	851	119	0	851	30	1.00	0.80	0.000
Spindalis portoricensis	72	928	70	4	924	2	0.95	0.97	0.004
Coereba flaveola	213	787	164	0	787	49	1.00	0.77	0.000
Leptodactylus albilabris	44	956	33	0	956	11	1.00	0.75	0.000
Eleutherodactylus antillensis	46	954	34	0	954	12	1.00	0.74	0.000
Eleutherodactylus brittoni	204	796	152	1	795	52	0.99	0.75	0.001
Turdus plumbeus	6	994	4	0	994	2	1.00	0.67	0.000
Patagioenas squamosa	222	778	148	1	777	74	0.99	0.67	0.001
Eleutherodactylus wightmanae	18	982	13	1	981	5	0.93	0.72	0.001
Loxigilla portoricensis	26	974	15	0	974	11	1.00	0.58	0.000
Nesospingus speculiferus	23	977	13	0	977	10	1.00	0.57	0.000
Eleutherodactylus hedricki	53	947	33	2	945	20	0.94	0.62	0.002
Melanerpes portoricensis	3	997	1	0	997	2	1.00	0.33	0.000
Megascops nudipes	14	986	11	6	980	3	0.65	0.79	0.006
Todus mexicanus	17	983	9	6	977	8	0.60	0.53	0.006
Setophaga angelae	41	959	8	0	959	33	1.00	0.20	0.000
Margarops fuscatus	23	977	17	24	953	6	0.41	0.74	0.025
Coccyzus vieilloti	4	996	1	7	989	3	0.13	0.25	0.007
Total	–	–	–	–	–	–	0.98	0.83	0.003
Mean	129.6	870.4	–	–	–	–	0.90	0.71	0.003

2.4.3. Annotation review

Cases where the CNN-predicted score differed from the annotation by 95% or greater were reviewed. For the annotations of presence, 16 cases were found (where the CNN-predicted score was ≤ 0.05) across all species and recordings, of which only 1 was determined to be an annotation error. For the annotations of absence, 294 errors were found (where the CNN-predicted score was ≥ 0.95). Most errors (43%) were caused by the template matching method confusing the target with another call type with higher signal-to-noise-ratio (SNR), causing an *fn*. In 21% of cases, no matches for the target template were found, typically due to low SNR. Other errors were manual annotation errors, mostly based on difficulty validating noisy detections. The number of annotation errors was seen to be positively correlated with difference between annotation and prediction. Furthermore, 78% of all predicted scores for the results shown were below 0.05 or greater than 0.95. Therefore, most annotation errors were assumed to be accounted for. These errors accounted for only 1.2% of annotations. The results presented are based on the revalidated annotation.

3. Results

The average-precision *AP* varied among the species from 1 (*Melanerpes portoricensis*) to 0.28 (*Coccyzus Vielloti*) (Fig. 5). The mean-average-precision across species was 0.893. When computing precision and recall over all species and recordings, the total average-precision (AP_{total}) was 0.975 (Fig. 6, left). This indicates that the species with more frequent calls tended to have higher scores, because each species' contribution to the AP_{total} is proportional to its number of presences. Only 3 of the 24 species had an *AP* below 0.80. Excluding these three species, the mean-average-precision for the rest of the species is 0.955. These results indicate that the model's predicted scores strongly distinguish cases of presence and absence for most species.

Species-specific performance did not have a clear association with training sample size (Table 1, Fig. 5). Many classes with varying sample sizes achieved a similar strong performance. *Melanerpes portoricensis*, *Leptodactylus albilabris*, *Coereba flaveola* each had fewer than 1000 positive training samples and $AP > 0.97$. Thus, depending on the call type and potentially confounding signals and noise in the environment, strong performance is achievable with several hundreds of call examples.

A high predictive threshold greater than 0.90 was seen to yield an optimal balance of precision and recall (Fig. 6). A balanced precision and recall are desired if false positives and false negatives are equally significant, though if false positives are more costly, precision has greater importance. As the threshold neared 1, the recall began to drop rapidly, particularly for species-average scores. Using the mean of the two highest scores across audio windows to predict presence was seen to provide minor improvements to species-average precision and recall in some cases. However, simply using the highest-scoring window to determine presence theoretically allows for time-localizing the calls, though this was not evaluated. All results presented correspond to a prediction based on the highest score across audio windows for a species and recording.

For a selected prediction threshold of 0.99, the mean precision and recall across species were 0.90 and 0.71, respectively (Table 2). However, the total precision and recall were 0.98 and 0.83, respectively, further demonstrating that model performance is correlated with frequency of species presence. This may be due in part to the slight correlation between species frequency and training sample size, though strong performance was achieved for some species without a large sample size (eg. MEPO, LEAL, COFL). The association between performance and frequency of presence could be further explained by the higher balance between presences and absences. Rarer species with relatively few presences require stronger robustness to noise to achieve a low *FPR* and high precision. Comparing the average-precision scores to the precision and recalls at the chosen threshold of 0.99, we find that some species have a high *AP* but moderate *P* and *R* for the chosen threshold, which suggests that performance could be further improved with species-specific detection thresholds.

Three species with relatively poor scores ($AP < 0.80$) were further investigated: *Margarops fuscatus, Todus mexicanus, Coccyzus vieilloti*. The template call for *Margarops fuscatus* has an approximately two-second, two-syllable structure, which resulted in CNN training samples with call features only near the left and right borders of the image. The training

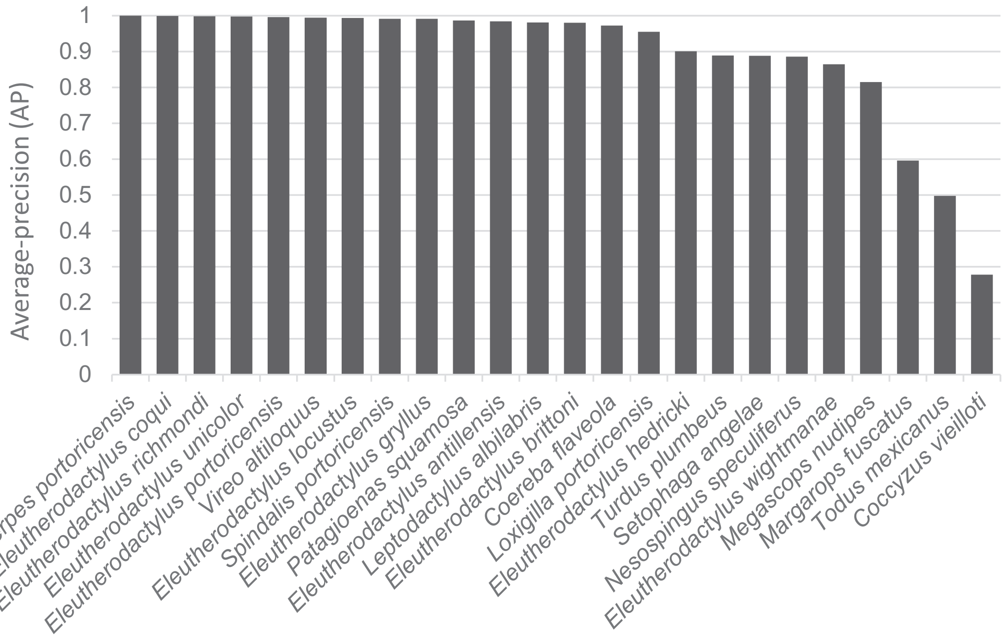


Fig. 5. Average-precision (*AP*) scores for all species.

samples for this species thus had a low signal-to-noise ratio. Similarly, *Todus mexicanus* has a multi-syllable call with a variable number of syllables, of which only one fits in a single CNN input. The multi-syllable structure is a significant feature of the call, though since only a single syllable is seen by the CNN at a time, this presumably impacts the detectability and makes the call more easily confounded with other signals. *Coccyzus vielloti* had clearer training sample features, but the fewest positive training samples of any species (Table 1).

4. Discussion

Soundscape recordings, which capture omnidirectional ambient sound in an area, are widely used for ecological research. While soundscape data is often used to analyze long-term changes in soundscape composition and richness, another interest is to document species-specific site occupancy over time, thus providing quantitative information for species conservation and management decisions. However, due to the non-directional nature of soundscape recordings, species-specific detection methods are often plagued by high false positive rates. Furthermore, multi-species detection models have required high complexity, and thus a high number of training samples to achieve adequate performance.

The work presented here addresses the challenge of acquiring multi-species detections from raw soundscape recordings and demonstrates a high model precision for the study species. Although only bird and frog taxa are considered here, our approach is expected to generalize well to other target signals. No model parameters are specific to the target call types aside from the CNN input width (2 s). The input spans a large frequency bandwidth from 0 to 24 kHz, which can account for many call types. Considering the high variation in the time-frequency extents and characteristics of the target call types in this study, we expect that many vocalizations or other transient signal types (i.e. up to several seconds) within the range of 24 kHz frequency would be appropriate for the pipeline.

The manual effort in training data creation was reduced to template creation and validation of template-based detections in a graphical user interface (Fig. 3). This addresses an important need for more accessible training data from study sites to leverage deep learning for acoustic monitoring. Our evaluation demonstrates that strong classification performance can be achieved using data collected from the study region, without relying on crowd-sourced public datasets. The pipeline thus increases the potential for region-optimized acoustic monitoring systems. Furthermore, the training data collection pipeline could accelerate the collection of data for rare species.

Our training scheme allowed for multi-label learning from single-label training data by defining a custom training loss and including

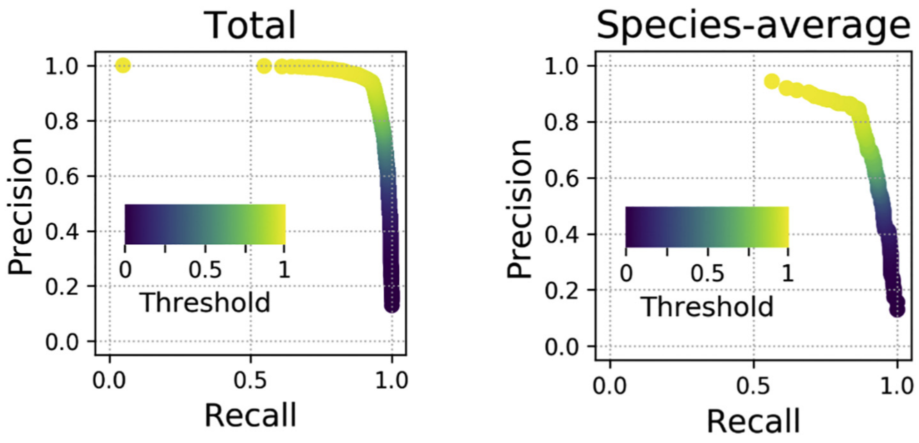


Fig. 6. (Color) (A) Precision-recall curve for all predictions across all species and test recordings. Total precision and recall (i.e. P_{total} and R_{total}) were computed for thresholds from 0 to 1 with an increment of 2.5×10^{-4}. (B) Precision-recall curve where precision and recall are averaged across species at each tested threshold. Mean precision and mean recall were computed for each threshold from 0 to 0.998 with an increment of 0.002. Higher threshold values are not shown in this case because with higher thresholds, all predictions become negative for some species (no presence predicted) and their precision becomes undefined.

both positive and negative examples for each species in training. Instead of using random ambient noise for the absence training samples, we used false template-based detections (*fp*'s). Because *fp*'s were detected based on correlation with the template, this method forces the model to learn to distinguish each call type from similar-looking noise (i.e. unknown) signals as well as from the other target signals. This improved the accuracy compared to tests with a standard approach, in which training was based only on positive examples of each call (Zhong et al., 2020). This is also supported by previous research that reported improved soundscape classification performance when false-positives identified from previous trials were included in the training data for a noise class (Florentin et al., 2020). This method also enables control over the number of positive and negative instances of each class in multi-label learning. Furthermore, this method allows for using the CNN to filter *fp* s from subsequent template-based detections. In general, the CNN can be trained to distinguish positive and negative detections for multiple classes, making it compatible for use in combination with other detectors.

Our mean-average-precision of 0.893 for the 24 species compares favorably to other CNN classification studies. A recent study of CNNs applied to acoustic recognition of six avian species achieved a mean-average-precision of 0.541 (Ruff et al., 2019). The latest BirdCLEF challenge (2019) yielded a maximum *mAP* of 0.407 for soundscape recordings in North America and 0.293 for soundscape recordings from Colombia, though in this case models were trained to identify > 600 species (Kahl et al., 2019). Notably, though, the results in this case were from multi-CNN ensembles, while our results are from a single CNN.

To expand this approach to the broader community, we have identified three important challenges for future research. First, future developments should account for the large variability in the size of target calls (i.e. templates). Introducing recurrent connections in the CNN, or other architecture modifications could potentially reduce the negative effects of window size. Second, previous studies have found data augmentation to significantly improve performance (Kahl et al., 2019). In these cases, training data was mainly based on monodirectional recordings of single species, and data augmentation (i.e. noise addition) apparently helped to emulate the conditions of soundscape recordings. The effect may be reduced for training data collected directly from soundscapes, as in this study. Still, data augmentation may be necessary to increase the training data size for rare species. Thus, future efforts should investigate optimal data augmentation methods for bioacoustic recognition. Third, an important challenge will be to maintain high accuracy while increasing the efficiency of prediction. This will require increasing the prediction speed and decreasing the memory footprint of the model by investigating other network architectures and reducing the number of parameters.

5. Conclusions

The presented pipeline enables training convolutional neural networks for multi-species multi-label classification of soundscape recordings, starting from raw unlabeled recordings. A high-accuracy model for 24 species in the El Yunque National Forest was obtained using training data collected from the study area, without relying on public, labeled bioacoustic datasets. Semi-automated training data collection improves the potential for creating region-specific CNNs for large-scale biodiversity monitoring. We show that single-label true and false-positive detections from a more rudimentary sound detector can be effectively used to train a CNN model for multi-class multi-label sound recognition. False detections, which contain examples of potentially confounding signals from each target, were found to improve performance when incorporated in the training process. Based on our evaluation of the model with 1000, 1-min soundscape recordings, CNNs are a viable solution for automated acoustic monitoring of many species using a single model.

Acknowledgements

Some of the field data were collected for projects funded by U.S. Forest Service (#12F43018C0014) to Sieve Analytics and the National Science Foundation (#1831952) to the University of Puerto Rico. The authors would like to thank Giovany Vega and Michael Haas for their help implementing the research in the ARBIMON platform.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., ... Zheng, X., 2016. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. http://arxiv.org/abs/1603.04467.

Aide, T.M., Corrada-Bravo, C., Campos-Cerqueira, M., Milan, C., Vega, G., Alvarez, R., 2013. Real-time bioacoustics monitoring and automated species identification. PeerJ 1, e103. https://doi.org/10.7717/peerj.103.

Aloysius, N., Geetha, M., 2017. A review on deep convolutional neural networks. In: 2017 International Conference on Communication and Signal Processing (ICCSP), pp. 0588–0592. https://doi.org/10.1109/ICCSP.2017.8286426.

Chollet, F., 2019. Keras. Online. Available at https://keras.io.

Colonna, J., Peet, T., Ferreira, C., Jorge, A., Gomes, E., Gama, J., 2016. Automatic classification of anuran sounds using convolutional neural networks. In: Proceedings of the Ninth International C* Conference on Computer Science & Software Engineering, pp. 73–78. https://doi.org/10.1145/2948992.2949016.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. ImageNet: A large-scale hierarchical image database. In: Proceedings of IEEE CVPR 2009, pp. 2009.

Florentin, J., Dutoit, T., Verlinden, O., 2020. Detection and identification of European woodpeckers with deep convolutional neural networks. Ecol. Informa. 55, 101023.

Ganchev, T., 2017. Computational Bioacoustics: Biodiversity Monitoring and Assessment. https://doi.org/10.1515/9781614516316.

Goëau, H., Glotin, H., Vellinga, W.P., Planqué, R., Rauber, A., Joly, A., 2015. LifeCLEF Bird Identification Task 2015. Working Notes of CLEF 2015. http://ceur-ws.org/Vol-1391/156-CR.pdf.

Goëau, H., Glotin, H., Vellinga, W.P., Planqué, R., Joly, A., 2016. LifeCLEF bird identification task 2016: The arrival of deep learning. In: Working Notes of CLEF 2016, pp. 440–449. http://ceur-ws.org/Vol-1609/16090440.pdf.

Goëau, H., Glotin, H., Vellinga, W.P., Planqué, R., Joly, A., 2017. LifeCLEF Bird Identification Task 2017. Working Notes of CLEF 2017. http://ceur-ws.org/Vol-1866/invited_paper_8.pdf.

Goëau, H., Kahl, S., Glotin, H., Vellinga, W.P., Planqué, R., Vellinga, W.P., Joly, A., 2018. Overview of BirdCLEF 2018: Monospecies Vs. Soundscape Bird Identification. Working Notes of CLEF. pp. 2018. http://ceur-ws.org/Vol-2125/invited_paper_9.pdf.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep residual learning for image recognition. In: Proceedings of CVPR 2016, . http://arxiv.org/abs/1512.03385.

Hill, A.P., Prince, P., Covarrubias, E.P., Doncaster, C.P., Snaddon, J.L., Rogers, A., 2018. AudioMoth: evaluation of a smart open acoustic device for monitoring biodiversity and the environment. Methods Ecol. Evol. 9 (5), 1199–1211. https://doi.org/10.1111/2041-210X.12955.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. CoRR. (vol. abs/1704.04861).

Incze, Á., Jancsó, H.-B., Szilágyi, Z., Farkas, A., Sulyok, C., 2018. Bird sound recognition using a convolutional neural network. In: Proceedings of IEEE 16th Int. Symp. Intell. Syst. Inform. (SISY).

Kahl, S., Willhelm-Stein, T., Hussein, H., Klinck, H., Kowerko, D., Ritter, M., Eibl, M., 2017. Large-Scale Bird Sound Classification Using Convolutional Neural Networks. Working Notes of CLEF. pp. 2017. http://ceur-ws.org/Vol-1866/paper_143.pdf.

Kahl, S., Stöter, F.-R., Goëau, H., Glotin, H., Planque, R., Vellinga, W.-P., Joly, A., 2019. Overview of BirdCLEF 2019: Large-Scale Bird Recognition in Soundscapes. Working Notes of CLEF. pp. 2019. http://ceur-ws.org/Vol-2380/paper_256.pdf.

Kao, C.-C., Wang, W., Sun, M., Wang, C., 2018. R-CRNN: Region-based convolutional recurrent neural network for audio event detection. In: Proceedings of Interspeech 2018, pp. 2018. http://arxiv.org/abs/1808.06627.

Kingma, D.P., Ba, J., 2017. Adam: A method for stochastic optimization. In: Proceeding of ICLR 2015, . http://arxiv.org/abs/1412.6980.

Koh, C.-Y., Chang, J.-Y., Tai, C.-L., Huang, D.-Y., Hsieh, H.-H., Liu, Y.-W., 2019. Bird Sound Classification Using Convolutional Neural Networks. Working Notes of CLEF. pp. 2019. http://ceur-ws.org/Vol-2380/paper_68.pdf.

Lasseck, M., 2019. Bird Species Identification in Soundscapes. Working Notes of CLEF 2019. http://ceur-ws.org/Vol-2380/paper_86.pdf.

Lewis, J., 1995. Fast Normalized Cross-Correlation. Industrial Light and Magic.

McFee, B., Raffel, C., Liang, D., Ellis, D., McVicar, M., Battenberg, E., Nieto, O., 2015. Librosa: Audio and music signal analysis in Python. In: Proceedings of the 14th Python in Science Conference 2015, pp. 2015. https://doi.org/10.25080/Majora-7b98e3ed-003.

Oliphant, T., 2007. Python for scientific computing. Comput. Sci. Eng. 9, 10–20. https://doi.org/10.1109/MCSE.2007.58.

Potamitis, I., Ntalampiras, S., Jahn, O., Riede, K., 2014. Automatic bird sound detection in long real-field recordings: applications and tools. Appl. Acoust. 80, 1–9. https://doi.org/10.1016/j.apacoust.2014.01.001.

Priyadarshani, N., Marsland, S., Castro, I., 2018. Automated birdsong recognition in complex acoustic environments: a review. J. Avian Biol. 49. https://doi.org/10.1111/jav.01447.

Puerto Rico State Wildlife Action Plan, 2015. Ten year review. In: Puerto Rico Department of Natural and Environmental Resources, . http://drna.pr.gov/wp-content/uploads/2015/10/PRSWAP-2015.pdf.

Ruff, Z., Lesmeister, D., Duchac, L., Padmaraju, B., Sullivan, C., 2019. Automated identification of avian vocalizations with deep convolutional neural networks. Remote Sens. Ecol. Conserv. https://doi.org/10.1002/rse2.125.

Sevilla, A., Bessonne, L., Glotin, H., 2017. Audio bird classification with inception-v4 extended with time and time-frequency attention mechanisms. In: Working Notes of CLEF 2017 (Cross Language Evaluation Forum).

Sprengel, E., Jaggi, M., Kilcher, Y., Hofmann, T., 2016. Audio Based Bird Species Identification Using Deep Learning Techniques. Working Notes of CLEF. pp. 2016. http://ceur-ws.org/Vol-1609/16090547.pdf.

Stevens, S.S., Volkmann, J., Newman, E.B., 1937. A scale for the measurement of the psychological magnitude pitch. J. Acoust. Soc. Am. 8, 185–190. https://doi.org/10.1121/1.1915893.

Swiston, K., Mennill, D., 2009. Comparison of manual and automated methods for identifying target sounds in audio recordings of pileated, pale-billed, and putative ivory-billed woodpeckers. J. Field Ornithol. 80, 42–50. https://doi.org/10.1111/j.1557-9263.2009.00204.x.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. Proceedings of IEEE Conf. Comput. Vision Pattern Recognition 2015 2015, 1–9.

van der Walt, S., Schönberger, J.L., Nunez-Iglesias, J., Boulogne, F., Warner, J.D., Yager, N., Gouillart, E., Yu, T., scikit-image, contributors., 2014. scikit-image: image processing in Python. PeerJ 2, e453. https://doi.org/10.7717/peerj.453.

Vellinga, W., 2020. Xeno-Canto - Bird Sounds from around the World. Xeno-Canto Foundation for Nature Sounds. Occurrence Dataset. https://doi.org/10.15468/qv0ksn accessed via GBIF.org on 2020-01-22.

Xie, J., Ding, C., Li, W., 2018. Audio-only bird species automated identification method with limited training data based on multi-channel deep convolutional neural networks. arXiv 1803.01107.

Xie, J., Hu, K., Zhu, M., Yu, J., Zhu, Q., 2019. Investigation of different CNN-based models for improved bird sound classification. IEEE Access. 7, 175353–175361. https://doi.org/10.1109/ACCESS.2019.2957572.

Zhang, L., Towsey, M., Xie, J., Zhang, J., Roe, P., 2016. Using multi-label classification for acoustic pattern detection and assisting bird species surveys. Appl. Acoust. 2, 91–98.

Zhong, M., LeBien, J., Campos-Cerquiera, M., Dodhia, R., Ferres, J.L., Velev, J.P., Aide, T.M., 2020. Multispecies bioacoustic classification using transfer learning of deep convolutional neural networks with pseudo-labeling. Appl. Acoust. 166, 107375. https://www.sciencedirect.com/science/article/abs/pii/S0003682X20304795.