Bioacoustic Classification Framework: Spectral and Cepstral Based Approaches.

Pedro Bonito Baptista
pedrobonitobaptista@tecnico.ulisboa.pt
Instituto Superior Técnico

Cláudia Antunes
claudia.antunes@tecnico.ulisboa.pt
Instituto Superior Técnico

ABSTRACT

The field of bioacoustics plays an important role on preventing and reducing human impact on environment, by enabling the development of tools capable of performing automated analysis of environmental data. Deep learning methods were successful on automating the process of species identification in environmental recordings, requiring nonetheless a large number of training samples per species. Hence, efforts were made to develop high-accuracy methods capable of automating species detection in noisy environments with limited training data. In this document, we address the problem of automating species detection in noisy environments with limited training data, proposing an end-to-end spectral based approach for training a convolutional neural network (CNN) on Mel spectrograms to predict a set of species present in the Rainforest Connection's acoustic recordings. Additionally, we propose a cepstral based framework for training a Long Short-Term Memory (LSTM) network on the Mel-frequency cepstral coefficients (MFCCs), complementing this approach with the motifs extracted by the matrix profile algorithm. Finally, we evaluate the performance of the approaches so that the bioacoustic classification framework can be established.

1 INTRODUCTION

Bioacoustics focuses on the analysis of the sounds produced by or affecting living organisms, especially the ones related to communication. Prior bioacoustic research was heavily dependent on manual labor to segment, detect and label animal activity, present in hours of field recordings. Consequently, recent research overlaps the work developed by Rainforest Connection (NGO) [1] which focuses on developing bioacoustic monitoring systems to ensure the rainforest's conservation, being also a prominent source of environmental audio data.

Deep learning methods have been successful on automatic acoustic identification, through image analysis dedicated architectures, such as convolutional networks. However, they require a large number of training samples per species. This limits applicability to rarer species, which are central to conservation efforts. Thus, the Kaggle competition "Rainforest Connection Species Audio Detection" [2] encouraged contenders to develop solutions capable of automate high-accuracy species detection in noisy soundscapes with limited training data.

In this document, we address the problem of automating species detection in noisy environments with limited training data, thus, we explore two main approaches to build a bioacoustic classification framework. The first, the spectral based one, proposes a framework for training a convolutional neural network (CNN) on Mel spectrograms to predict a set of species present in the Rainforest Connection's acoustic recordings. We leverage transfer learning by using a pretrained model as a way to reduce training requirements, both the amounts of data and time. Finally, we explore several window sizes, data augmentation techniques and predictive thresholds to improve the model's performance. The second, the cepstral based one, proposed an end-to-end pipeline for training a Long Short-Term Memory (LSTM) network on the Mel-frequency cepstral coefficients (MFCCs). Furthermore, we complement this approach with the motifs extracted by the matrix profile algorithm, as a way of improving the performance of the concerned network. Lastly, we explore the standard and the multidimensional implementation of the matrix profile algorithm, experimenting also different window sizes and predictive thresholds.

The best performing approach is the spectral based classification model, both on the chainsaw and on the Kaggle dataset. It includes 5-second-long Mel spectrograms and relies on the SpecAugment method to increase the training set size. Regarding the chainsaw dataset, it achieves an accuracy of 0.97, a mean precision of 0.99 and a mean recall of 0.97. In relation to the Kaggle dataset, it registers an accuracy of 0.97, a mean precision of 0.91 and a mean recall of 0.93.

This paper is organized into six sections. Section 2 describes the concepts addressed by this work and section 3 introduces the work related with automatic bioacoustic analysis and classification. Section 4 presents the proposed methodology, section 5 details the obtained results and section 6 discusses them briefly.

2 BASIC CONCEPTS AND NOTATION

The research direction that this work will concern is *sound event detection*, which labels temporal regions within an audio recording, with their start and end time, as well as with the event's type. Also, a *frame* (or sound clip) indicates the unit of analysis and may contain several events that may overlap in time.

The referred classifiers, in sound event detection, ideally, have each one of the acoustic events instances in the training data, labeled with their start and end time. This type of labels is referred as *strong labels*, nevertheless, acquiring them is a costly process that also requires careful attention to detail by the annotator.

A *sound spectrogram* is an image of the time-varying spectral representation, produced by applying the *short-time Fourier transform (STFT)* to successive overlapping frames of an audio sequence. The horizontal dimension corresponds to time and the vertical dimension corresponds to frequency. The relative spectral intensity of a sound at any specific time and frequency is indicated by the color/grayscale intensity of the image.

Model performance and capability to capture the natural variability of data can be increased with the use of data augmentation

[1] https://rfcx.org/
[2] https://www.kaggle.com/c/rfcx-species-audio-detection/data

techniques. Such signal transformations may include time shifting, volume control or adding additive noise to the acoustic data.

3 RELATED WORK

In recent years, Convolutional Neural Networks (CNNs) have outperformed the former models in visual recognition tasks, namely in large-scale image and video recognition, mostly due to the late availability of large public datasets of images such as the ImageNet (Krizhevsky et al., 2012).

Transfer learning is used to avoid the large amount of training data and time that deep neural networks with initially randomized weights require to achieve reasonable performance. In particular, given the context of environmental data in which labels are costly, one can take advantage of this technique by retraining with new data a model already optimized for a similar dataset to improve performance. The ResNet50 (He et al., 2016) model is a classic neural network used as backbone for many computer vision tasks and it was trained on the ImageNet dataset. Despite not containing spectrograms, models pre-trained on this dataset learn a variety of image features and have been successfully tuned to spectrogram classification (LeBien et al., 2020), (Zhong et al., 2020).

Furthermore, the competition "Rainforest Connection Species Audio Detection", previously referenced, encouraged contenders to develop models that aimed to automate the detection of several species in the RFCx audio files. Thus, this competition provides multiple audio processing methodologies and models architectures which concern species detection in noisy soundscapes with limited training data, such as the ones presented in (LeBien et al., 2020), (Zhong et al., 2020).

4 METHODS

The first approach is centered on the idea that sound signals can be represented by images. Thus, by extracting the spectral audio features, namely the Mel spectrograms, this methodology leverages deep neural networks, such as Convolutional Neural Networks, to perform the aforementioned task. Also, it takes advantage of transfer learning to reduce training requirements, both the amounts of data and time. The obtained results validate the proposed framework, as the proposed model is capable of differentiating the multiple events present in the image representations of sound.

Alternatively, the second methodology aims to reduce the procedures associated with an image-based approach. Thus, the second approach presents an alternative procedure, that explores different audio features and a distinct network, namely a Long Short-Term Memory (LSTM) network, to identify the given species in the multiple recordings. The obtained results reveal that the models trained on the cepstral features, namely the Mel-frequency cepstral coefficients (MFCCs), achieve better performance, nevertheless, the results are relatively worse than the ones attained by the spectral based one. In this sense, we complement this procedure with the motifs extracted by the matrix profile algorithm, as a way of improving the performance of the concerned network. Nonetheless, we only present this additional step in the master's thesis document due to space limitations.

Finally, it is important to note that the classifications models that result from both approaches were trained and evaluated with a fixed training and test set. In particular, the results presented in section 5 reflect this condition, mainly because the definition of both methodologies results from numerous experiments, in which we considered different network architectures, training configurations and feature extraction and processing techniques. We only use cross-validation to train and test the developed classification models in section 5.6, when both methodologies are well defined and matured.

4.1 Spectral Based Classification Model

Our first proposal was a bioacoustic classification framework using transfer learning of deep neural networks. Thus, this section focuses on detailing each step of the suggested end-to-end pipeline, a process that results in a **classification model**, as represented in Fig. 1.

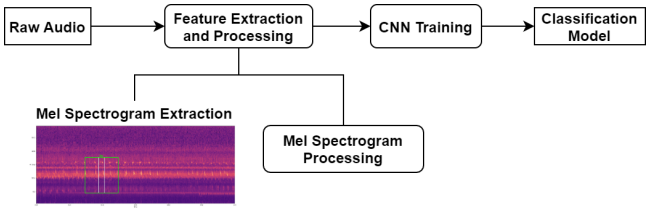

Figure 1: Spectral based approach flowchart.

The starting point consists of converting the sound sequences (**raw audio)**, that is, the time series, into audio features that can capture the distinctive properties of each event. Given the results obtained by deep neural networks in image classification problems our **feature extraction** step focuses on the extraction and processing of the Mel spectrograms, that are image representations of sound. So, we explore the learning ability of deep neural networks, namely Convolutional Neural Networks, describing their **training** process with the mentioned spectral shape features.

4.1.1 Feature Extraction and Processing.

In audio processing and analysis, the frame length is critical to the neural network's performance, as it must be set long enough to preserve the meaningful events but not so long that temporal variations disappear. In this regard, this report proposes a window function and evaluates the effect of different window (frame) lengths on the model's results.

The proposed window function defines the frame's center (window center) as the sum of half of the maximum label interval (maximum delta) of a given dataset, to the interval's beginning (interval start) of the concerned label. The start (window start) and end (window end) of the frame are the result of subtracting and adding to the center, respectively, the selected frame length (window duration) divided by two.

window center = interval start + (maximum delta / 2)

window start = window center - (window duration / 2)

window end = window center + (window duration / 2)

Additionally, it is important to note that the sampling rate by which the audio is extracted must be taken into consideration when extracting the mentioned window. All in all, the window function

allows for a training set composed only by frames that are linked to a given event.

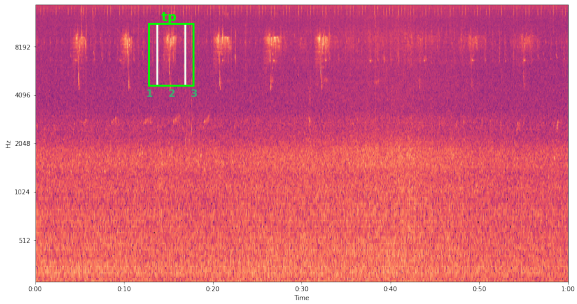

Figure 2: Window extraction example - Mel spectrogram: window start (1) window center (2) window end (3).

The extracted Mel spectrogram (**Mel spectrogram extraction**) is represented in Fig. 2 by the green box, being the white box the representation of the event's labelled time interval. Each Mel spectrogram is computed using the *librosa* Python package with the default settings (sampling rate = 48 kHz, NFFT = 2048, hop length = 512, window length = 2048, Hann window), specifying however the number of mel bands (n_mels = 224) and if available the minimum and maximum frequency. The frequency interval corresponds to the minimum and maximum value registered in the dataset, with a 10% margin to increase the considered interval.

The resulting Mel spectrograms, as a part of the **Mel spectrogram processing** step, are converted to units of decibel (dB), resized to the dimensions supported by the pre-trained model, that for the ResNet50 case correspond to 224x224 images, and normalized with the min-max scaling. Finally, the spectral features are converted to color images, that is, images with RGB channels and given the transfer learning setting, the spectrogram is processed to the adequate image format of the selected backbone model. For the ResNet50 model, for instances, the images are converted from RGB to BGR, then each color channel is zero-centered with respect to the ImageNet dataset, without scaling.

4.1.2 *Convolutional Neural Network Architecture.*

The proposed model uses the pre-trained ResNet50 weights used for ImageNet classification, and includes only the feature extraction layers of this model, excluding the remaining layers, often referred as the network "top". Hence, the knowledge obtained in image classification, namely the detection of basic image features, can be transferred (**transfer learning**) to the task at hand by using the weights of the optimized model. In this sense, by freezing some layers of the pre-trained model and only training the last several layers, the model can be fine-tuned to our problem. In addition to ResNet50, our work also evaluates different backbone models, such as EfficientNetB0, InceptionResNetV2 and VGG19.

We propose two network architectures for the introduced methodology, which have as reference the networks introduced in (Zhong et al., 2020). The first **model architecture** comprises the pre-trained model and two *fully connected (FC)* layers. The first consists of 512 nodes and uses the "Relu" activation function that converts negative inputs to 0. This layer is followed by a batch normalization and drop-out layer, the latter with a drop-out rate of 50% in which each node is ignored with a 50% probability, helping prevent overfitting. The final layer, given the binary classification setting, consists of one node that passes through the sigmoid function.

Additionally, we propose the addition of a LSTM layer to complement the above model, as a way of improving the general model's performance. Hence, the second **model architecture** includes the pre-trained network (ResNet50), and is followed by a Flatten and LSTM layer, being the latter composed by 512 neurons. Then, the subsequent layers follow the structure from the previous network, tuning however some parameters. In detail, we include two *fully connected (FC)* layers, the first with 1024 nodes and that uses the "Relu" activation function and the last layer which comprises only one neuron. Likewise, between the aforementioned fully connected layers there are a batch normalization and a dropout layer to help prevent overfitting.

4.1.3 *Model Training.*

Given the binary classification setting, the **training** step consists of training the network on the spectral features to obtain a classification model. The optimizer uses the Adam optimization method with a learning rate of $1 * 10^{-4}$ and decay of $1 * 10^{-7}$. Moreover, the binary cross entropy loss function is utilized and 30 epochs are applied. These parameters result from a fine-tuning process in which we analysed the values who favored the model's performance. For instances, a higher number of epochs did not contribute to a significant improvement on the performance, ending up in an overfitting situation in the cases that did. Oppositely, a lower number of epochs usually did not lead to a convergence point, being the proposed value a trade-off between both scenarios.

Model performance and capability to capture the natural variability of data can be increased with the use of **data augmentation** techniques. Thus, two approaches are followed as a way of increasing the training set's effectiveness: the first randomly adds one of the two additive noises, Gaussian or Pink, to the audio signal, time shifting and controlling its volume afterwards; the second applies the SpecAugment technique to the Mel spectrogram.

4.2 Cepstral Based Classification Model

This section complements the research on the bioacoustic classification framework as it presents an alternative approach to the one proposed in section 4.1, the spectral based one. Despite having the same goal, as it also aims to obtain a model capable of learning the distinctive characteristics of the concerned events, it explores the use of different audio features, such as the Mel-frequency cepstral coefficients (MFCCs), the root mean square (RMS), the zero-crossing rate (ZCR) attributes, and even the raw audios. Nevertheless, it is important to remark that we focus our research on the cepstral ones.

Hence, the objective is to develop a simpler approach, in comparison to the previous one, in terms of the required feature extraction and processing steps. In this sense, the Convolutional Neural Network (CNN) was replaced by a Long Short-Term Memory network (LSTM), changing also the concerned features by the previously mentioned ones. In detail, we change the network to determine if the LSTM's remembering and forgetting nature contributes to

the learning of the distinctive traits of the events present in our bioacoustic classification problem.

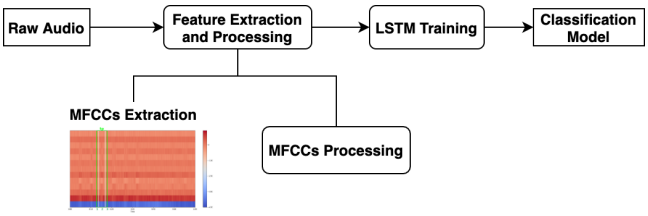

Figure 3: Cepstral based approach flowchart.

Analogously, the starting point of the proposed procedure consists of transforming the **raw audios** into audio features that can be used to train the **classification model**. We describe the **extraction and processing** of the aforementioned features, as well as the **training** of the concerned classification model.

4.2.1 *Feature Extraction and Processing.*

In this section, we cover the procedure that transforms the raw audios into the features used to train the developed model. We assume the window function proposed in section 4.1.1, as we will also have a training set composed only with frames associated with the presence or absence of a given event. So, the difference in this step lies on the extracted features and in their processing.

The procedure introduced in this section focuses on the cepstral features (MFCCs), however, as previously noted, it also addresses other audio attributes such as the ZCR, the RMS, and the raw recordings. The *librosa* Python package once again enables the extraction of these features.

As the lower order MFCCs contain most of the information present in the recordings we only extract 13 MFCCs. Also, although we initially tried normalizing this attribute, we ended up not performing this step as it did not benefit the model's performance. The toy problem's MFCCs attribute is represented in Fig.4. Lastly, the other features did not undergo through any additional processing.

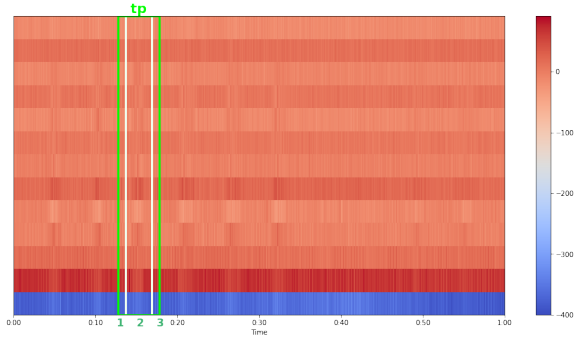

Figure 4: Window extraction example - MFCC: window start (1) window center (2) window end (3).

4.2.2 *Long Short-Term Memory network architecture.*

The **model architecture** consists of one LSTM layer, that comprises 512 nodes and assumes the default activation function, the hyperbolic tangent (tanh). This layer is responsible for handling the input features and it is followed by three fully connected layers, the first with 256 neurons and the second with 128, both with a "Relu" activation function. The final layer, given the binary classification setting, has one neuron that goes through the sigmoid activation function.

Also, it is important to note that the proposed architecture is the result of multiple experiments, in which we adjusted the configuration according to the attained results. The goal was to maintain the model as simple as possible without compromising its performance.

4.2.3 *Model Training.*

The **training** step is similar to the one described in section 4.1.3, differing only in the sense that it trains each model on the cepstral features, instead of the spectral ones. Similarly, the model is trained with the Adam optimization method, with a learning rate of $1 * 10^{-4}$ and decay of $1 * 10^{-7}$. Also, the binary cross entropy loss function is utilized, due to the binary classification setting, and 30 epochs are applied.

5 RESULTS

5.1 Case Studies

5.1.1 *Kaggle Competition Dataset.*

The "Rainforest Connection Species Audio Detection" [3] is a Kaggle competition that concerns the classification of 24 bird and amphibian species which inhabit the tropical mountains. It provides 6719 audio files (.flac) that include sounds from numerous species and two files, the first has data about the true positive events registered in all recordings, having a labelled interval which refers to the specie call; the second has data about the false positive events, detailing by opposition the intervals where a certain specie does not appear. Furthermore, both files also provide data about the specie present in the audio sample, the sound´s song type as well as the frequency and time interval of the event.

5.1.2 *Chainsaw Dataset.*

This dataset refers to the data provided by Hitachi Vantara through its partnership with Rainforest Connection. It also includes the VisBig project (PTDC/CCI-CIF/28939/2017), being important to remark that both connections enable more data to be considered. Despite that, our work only considers recordings from January 2020, having the concerned dataset 6567 audio files (.flac and .wav). These particular files have been preprocessed, a procedure in which files smaller than 1.1MB and with sampling rates lower than 12,000 Hz were filtered out. Also, the remaining audios are approximately 90 seconds-long.

Nonetheless, we consider a subset of these recordings, as only 1091 of these audio files possess annotations regarding chainsaw events. The labels were obtained by a manual confirmation process that validated the output of a model, developed by Huawei, that detects chainsaw events. In detail, each labelled recording can encompass multiple events, registering a total of 7885 confirmed and 3274 rejected chainsaw events, each one annotated with the corresponding event's time interval.

[3] Rainforest Connection Species Audio Detection

5.2 Spectral Based Model - Kaggle

There are 24 annotated species in the provided dataset, which would suggest a 24 multi-label classification setting. Nevertheless, two species have more than one song type, having both type 1 and 4, revealing the need of two additional labels. As a starting point, the created training set disregards the song type 4 for the mentioned species.

In this sense, our approach transforms the 24 multi-label classification setting into 24 distinct classification problems, where in each we train a model so that it can learn the presence or absence of a given specie. Moreover, the upcoming sections describe several experiments in which the concerned models follow the architecture described in section 4.1.2. In particular, note that the experiments' results represent the average of each specie related model's score, taking as an example the scores displayed in Fig. 5, that refer to the average of the accuracy scores across all 24 species. It is also important to remark that from these results, the ones presented in section 5.2.2 refer to each specie related model as this analysis discriminates all species.

Also, the baseline training set includes the maximum number of true positive events for each specie, that for the majority of species corresponds to approximate 50 samples. Additionally, other variants may encompass different quantities of true positive augmented samples, as further described in section 5.2.1. Lastly, a subset of the available 350 false positive samples is extracted, for each specie, in the same quantity as the true positive subset, that may contain augmented instances. For example, if we complement the baseline approach with data augmentation, one specie that has 50 true positive samples, will also have 50 augmented samples and 100 false positive samples, resulting in a balanced training set for each specie.

5.2.1 *Window Size and Data Augmentation.*

The first approach aims to assess the effect of different window sizes and data augmentation techniques on the performance of each model. Thus, the considered frame lengths were 2, 5 and 10-second-long, as more than 80 percent of the recorded events have intervals smaller or equal to 4 seconds. Also, by including the 2 second window one can verify if smaller frames can capture enough image traits to conduct automate species detection. In addition, for each window size, we trained a model with a training set that did not include augmented samples (baseline) and compared it to two models whose training set contained samples augmented by the two techniques described in Section 4.1.3.

As detailed in the previous section, the training set that does not takes advantage of data augmentation techniques includes the maximum number of available true positive samples, having the same number of negative samples. Conversely, both training sets with augmented instances, from the two aforementioned augmentation techniques, differ from the latter by having augmented samples in the same number as the true positive calls, thus enabling the use of more negative instances. Finally, the evaluation metrics were *accuracy*, which corresponds to the percentage of correct predictions (tp + tn) over the total number of instances evaluated (tp + tn + fp + fn); *precision*, that measures the fraction of an identified event correctly classified (tp ÷ (tp + fp)); and *sensitivy or recall*, which measures the fraction of positive patterns correctly classified (tp ÷ (tp + fn), being the test set classified with a threshold score of 0.6. Once again, it is also relevant to stress that the evaluation metrics represent the average of the scores of each individual model, excluding those who fail to learn the distinguishing characteristics of the audio features.

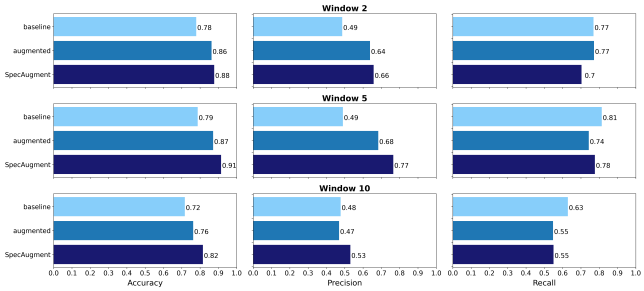

Figure 5: Effect of different window sizes and data augmentation techniques on accuracy, precision and recall.

As depicted in Fig. 5, by including the augmented samples in the training set we increased the accuracy scores across all windows sizes. The model trained on the 10-second-long window failed to capture the data's variability, leading to the worse results in terms of precision and recall. The 5 second window obtained a significant accuracy increase, registering the best precision and recall score (0.77 and 0.78) with the SpecAugmented spectrograms. Furthermore, the smallest concerned frame obtained similar results in comparison to the 5 second window in terms of accuracy, achieving, nevertheless, lower precision and recall scores.

All in all, the results confirm the well-known precision-recall relation, in which generally an increase in precision leads to a decrease in recall, and vice-versa. Consequently, a balance is desired if false positives and false negatives are equally significant, which is not the case in our problem's spectrum as recall is slightly more important because false negatives are more costly. From this experiment, both the 2 and 5-second long frames seem to be able to capture the distinctive traits of each Mel spectrogram. Nonetheless, as the model trained on the 5-second-long windows performed slightly better, this is the frame length concerned from this point forward.

5.2.2 *Dynamic Window Sizes.*

The results obtained in the previous section are strongly marked by the models that fail to differentiate both classes, difficulty amplified with the 10 second frame. So, in order to assess if each model would perform better with a tailored window size, a different approach was experimented. More concretely, each specie related spectrogram was obtained by taking into consideration the mean time interval of each specie call with a one second margin, which implied that, for instances, a specie with an average interval call of 2 seconds would have a 3-second-long Mel spectrogram.

Definition 5.1. **Average-precision:** Summarizes the precision-recall curve as the weighted mean of precisions achieved at each threshold, with the increase in recall from the previous threshold used as the weight. P_n and R_n correspond to the precision and recall at the nth threshold.

$$AP = \sum_n (R_n - R_{n-1})\, P_n \tag{5.2.1}$$

Hence, the average-precision, presented in the definition 5.1 was the metric used to compare the model trained on the dynamic windows with the one trained on the fixed window size (5 seconds). Also, both models had recordings augmented with the SpecAugment method.

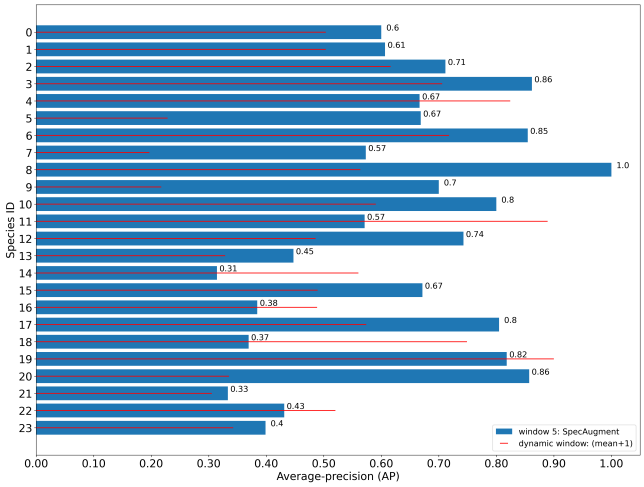

Figure 6: Average-precision of dynamic and fixed window size approaches.

In particular, Fig. 6 demonstrates the difference in average-precision between each model trained on the dynamic window size (red) and those trained on the 5 second window (blue). The mean average-precision scores for the dynamic and fixed size approach are 0.52 and 0.63, respectively. Furthermore, species with a mean window size smaller than 5, such as 11 and 18, for instances, are the ones who benefit the most from the dynamic approach. Also, it is possible to understand the impact that models with lower scores have on the metrics depicted in Fig. 5. To sum up, the goal of this approach was to understand if a small combination of window sizes, as a large one would be extremely costly in the prediction step, would favor the model's results. Despite the aforementioned improvement on the species that register smaller calls, the overall performance was not sufficient to justify the cost that a windowed approach would require.

5.2.3 Predictive threshold.

The predictive threshold represents the probability value by which a given sample is classified, that is, if the probability returned by the model is superior to the defined threshold the sample will be classified as belonging to the class, and vice-versa. On that account, the previous experiments considered a predictive threshold of 0.6, achieving a precision of 0.77 and a recall of 0.78 with the best performing model. Nonetheless, one can try to improve the precision score by increasing the predictive threshold.

Hence, Fig. 7 displays the mean precision and recall variation with different thresholds, so that the influence of the threshold value on the obtained results could be determined.

The increase in the threshold value leads to higher precision scores, nevertheless, this increment also results in a significant decrease

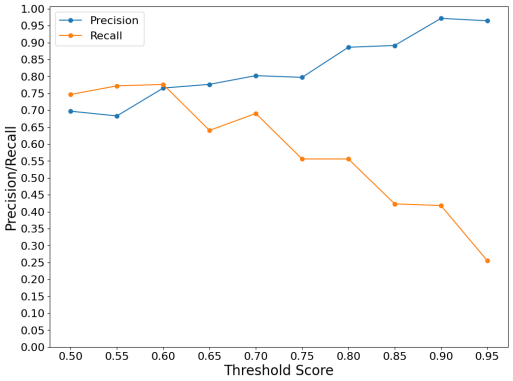

Figure 7: Precision/Recall threshold curve of the model trained on a 5-second-long window with SpecAugmented samples.

in recall. The precision-recall balance is achieved somewhere between the 0.50 and the 0.65 predictive threshold value, with the 0.60 threshold registering the optimal value for the develop approach, with a precision of 0.77 and a recall of 0.78.

5.2.4 Convolutional Neural Network combined with a Long Short-Term Memory network.

As referenced in the master thesis document, similar classification problems were addressed with a hybrid architecture, that combined Convolutional Neural Networks with Long Short-Term Memory networks. In this sense, we complement the previous architecture with an LSTM layer, as an attempt to improve the general performance of the developed model.

The network architecture described in 4.1.2 stems from several experiments in which we tried to establish the optimal combination to the problem at hand. In depth, we started by adding an LSTM layer with 512 neurons between the pretrained model and the fully connected layer with 512 neurons. Despite being the initial experiment, it remained the best performing one, achieving an accuracy score of **94%** and a precision and recall score of **83%** and **84%**, respectively.

According to our experiments, an increase in the number of LSTM's neurons led to a scenario where we ended up with higher recall scores and slightly lower precision scores, such as 86% and 80%, for example. Conversely, an increment in the number of neurons of the fully connected layer resulted in lower precision (82%) and recall scores (79%). Finally, we also tested multiple settings where we tried several combinations of LSTM and fully connected layers, nonetheless none of them improved the results from the best performing one.

5.3 Spectral Based Model - Chainsaw

In view of the results attained in the previous sections, we evaluated the proposed framework on the dataset, previously introduced in section 5.1.2, that includes the recordings labelled with the chainsaw events. The main difference to the annotations concerned in the previous dataset (5.1.1) lies on the information regarding the event's frequency interval, as the labels from this dataset do not detail the

mentioned interval. Thus, as we analysed the Mel spectrograms of the different recordings we noticed that chainsaw events, for the most part, took place in the lower frequencies. Despite being possible to find other animal sounds in this frequency range, we also observed that events such as bird sounds, would generally assume higher frequencies in comparison to the chainsaw sounds. In this sense, as our goal is to detect chainsaw events, we reduced the previous sampling rate of 48 kHz to 22kHz since there was no need to concern such high frequencies when training our model, imposing also a minimum and maximum frequency of 0.08 and 3kHz, respectively, for the extracted Mel Spectrogram.

Furthermore, the amount of available labelled recordings is considerable larger, in comparison to the previous dataset, favoring the model greatly as it allows for a bigger training set. However, due to hardware limitations we restricted the training set to 1600 positive and 1600 negatives samples. So, apart from the aforementioned modifications, the training process of this particular classification model was similar to the one described in section 4.1.

5.3.1 *Transfer Learning and Fine-Tuning.*

The transfer learning setting enables the use of multiples models which can be used for prediction, feature extraction, and fine-tuning. The ResNet50 network was the selected to complement the developed baseline architecture, as it is often utilized in similar image classification problems, being the model used as a start point in numerous solutions. Nevertheless, this section aims to compare the performance of the previously mentioned model with other networks that were also trained on the ImageNet dataset.

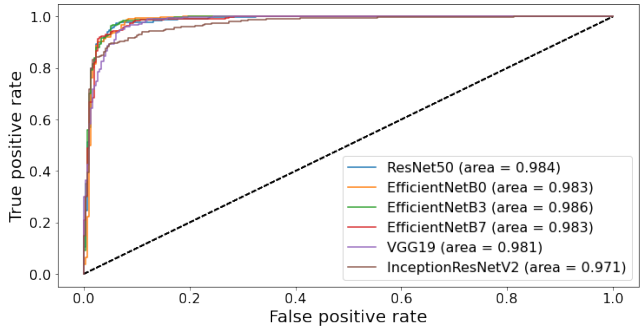

Figure 8: Comparison between the performance of different pretrained models.

Consequently, we experimented 3 other pretrained networks, EfficientNetB0, InceptionResNetV2 and VGG19, as the backbone of our proposed architecture. We then compared the obtained results with the ones attained with the initial architecture, which comprised the ResNet50 model. The used training set was the one described in the above section (5.3), being showcased in Fig 8 the comparison between the aforementioned models' performance.

The different networks present similar AUC (area under the ROC curve), being the Resnet50 model the best performing one, along with the EfficientNetB3. Even though the ResNet50 network (as in 50 weight layers) is much deeper than VGG19, its model size is substantially smaller due to the usage of the global average pooling layers rather than the fully-connected ones, which makes it preferable to the latter.

Furthermore, it is also possible to compare all models against the family of EfficientNets (EfficientNetB0 to EfficientNetB7), considering only, for simplicity, the B0, B3 and B7 variants. The performance of the EfficientNetB0 network is very similar to the ResNet50 one, being also possible to notice that the use of other EfficientNet variants does not increase significantly the obtained results.

Also, the combination of the Inception architecture with residual connections, present in the InceptionResNetV2 network, does not justify the use of this specific architecture as it does not register a better performance when comparing with the ResNet50 network.

Hitherto, the layers from the pretrained models were frozen, that is, they were not trained during the training process to avoid destroying any of the information they contained. Nevertheless, in this setting one can take one last optional step, referred to as fine-tuning, that consists of unfreezing the entire model, or a part of it, and retraining it on the new data, with a very low learning rate. Despite multiples attempts, we were not able to attain better results when fine-tuning our model, as all our experiments ended up with poorer performance.

5.4 Cepstral Based Model - Chainsaw

As stated in section 4.2, this approach aims to provide an alternative to the spectral based classification model (4.1), by exploring a different network architecture, namely the Long Short-Term Memory network, and different audio features. In particular, it focuses on features such as the root mean square (RMS), a reliable indicator for silence detection, the zero-crossing rate (ZCR), useful for discriminating periodic signals from those marked by noise, to understand if it is possible to perform biacoustic classification without all the processing related with an image-based approach. Moreover, we also explore the MFCCs as this feature is widely used in similar problems, as previously referenced.

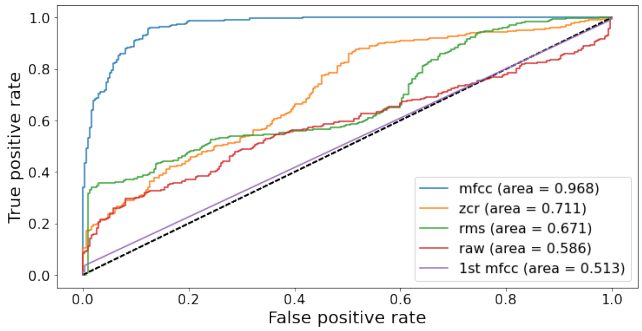

Figure 9: LSTM network's performance with different audio features.

The first experiment sets the baseline for the concerned approach, as it compares the performance of the LSTM network when trained with the different referenced audio features. Additionally, it follows the windowed approach introduced in section 4.1.1. Given the results presented in Fig.9, it is possible to conclude that the multidimensional feature (MFCCs) outperforms the one-dimensional ones,

obtaining a similar performance to the spectral based approach, in this particular setting.
In addition, the networks trained with the raw recordings and with one of the MFCCs failed to learn the unique properties of the labelled events, being the worse performing models. Also, the models trained with the RMS and the ZCR features registered a significant improvement in performance when comparing to the ones trained on the aforementioned attributes, being, nonetheless, quite far from achieving similar results as the ones from the network trained on the cepstral features.

All in all, given the results, from this point forward we will only concern the MFCCs and the ZCR attribute, being the latter considered as a way of confirming the above results, since we also evaluate the framework on the Kaggle dataset.

5.5 Cepstral Based Model - Kaggle

In light of the results described above, the classification model presented in this section, was obtained by following two different approaches. The first trains each specie related LSTM network with the MFCCs and the second uses the ZCR attributes instead.

5.5.1 Window Size.

LSTM's networks can keep track of arbitrary long-term dependencies in the input sequences, thus, in this sense, the time component can play a major part on the outcome of the developed solution. So, we complemented our research with the study of the window size's effect on the model's performance, as we did in section 5.2.1. In particular, we want to determine if this type of network benefits more from longer frames or smaller ones.

The results, displayed in Fig. 10, reveal the discrepancies in relation to the previous experiment, as the networks trained on this dataset have lower scores in comparison to the ones obtained by the models trained on the chainsaw one. Also, it is important to remark that the starting point of this analysis is significantly worse than the one described in section 5.2.1.

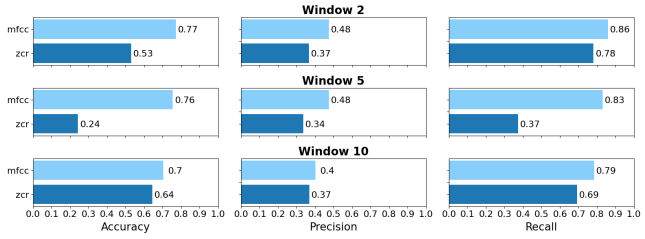

Figure 10: Effect of different window sizes on the LSTM's accuracy, precision and recall scores.

Moreover, from all the networks trained with the different frame lengths, it is possible to conclude that the ones trained with the MFCCs achieved better results. When concerning only this feature, the accuracy scores were very homogeneous, with a slight decrease in the one attained by the network trained with the 10-second-long window. Oppositely, when regarding the ZCR feature, the model who stands out in terms of accuracy is the one which considered the 10-second-long frame, as it achieved the highest score.

In terms of precision, all models got similar results when regarding the same feature. However it is important to mention that we registered multiple specie related networks that failed to distinguish both classes, when considering the models trained with the 5-second-long ZCR features, as their output was made only of negative instances, a case which we do not consider.

In relation to recall, it is possible to observe that all models that concern the ZCR feature achieve lower scores in comparison to the ones trained with the MFCCs, noting also the considerable low result of the network trained with the 5-second-long ZCR attribute. In addition, the recall scores of the models trained with the MFCCs were very similar, with the 2 and 5 frame lengths standing out in relation to the other.

To sum up, when concerning the cepstral features, the 2 and 5-second-long window sizes are the frame lengths which favour the learning capability of each model, similarly to in section 5.2.1. Nevertheless, there is a significant gap between the performance of the cepstral based approach and the spectral one, as the first attained worse results. As a consequence, further research will attempt to improve this approach, focusing on the network trained with the 5-second MFCCs, as the difference to model trained with the 2-second-long frame is relatively small.

5.5.2 Predictive Threshold.

Following the results of the above section, it is possible to notice the considerable difference between the recall scores and the precision ones, being the first substantially higher. Thus, as introduced in section 5.2.3, one can attempt to diminish this gap by changing the predictive threshold value, that for the previous analysis held a value of 60%.

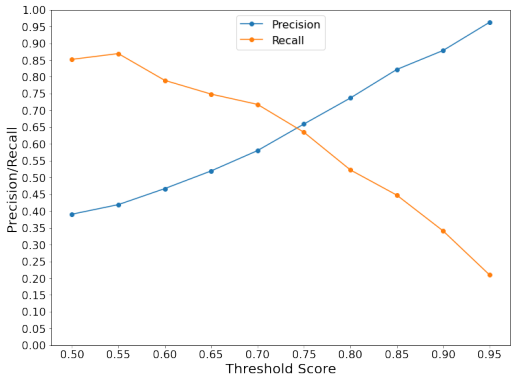

Figure 11: Precision/Recall threshold curve of the LSTM network trained with the 5-second-long MFCCs.

From Fig. 11, it is possible to observe the effect that the predictive threshold holds on the precision and recall scores. As previously noted, our experiments reveal that an increase in the threshold value leads to higher precision values and to lower recall ones. Moreover, as opposed to section 5.2.3, the balance in both scores is not attained with a 60% threshold value but with a 75% one, achieving a precision of **66%** and a recall of **63%**.
Thus, this is the predictive threshold considered from this point forward, in this approach, as it is the one that favours the developed

model, as it increases the precision score without compromising immensely the recall one.

5.6 Bioacoustic Framework Appreciation and Best Configuration Results

So far, we have only estimated the models' performance, as the concerned metrics were measured in a set of known records. Despite the insight given by those performance measures there is still uncertainty regarding the models' behaviour when facing unseen objects. So, this section focuses on determining the confidence bounds which detail how much the attained estimate may deviate from the true value. The mentioned process will only concern the best configurations of each approach, as the goal of this work is to establish the best bioacoustic framework possible.

In this regard, to compute the aforementioned intervals we apply the stratified k-fold cross-validation technique to both datasets, in which each dataset is divided in k equal-sized parts (folds), that preserve the percentage of samples for each class. Afterwards, we train each model under the multiple proposed configurations on the different folds. The concerned metrics are obtained regarding also their respective confidence bound. We considered 5 folds for both datasets ($k = 5$) and we used the T-student (95%) distribution to compute the confidence intervals. Note that the mentioned computation considers the average of each metric across the 5 folds.

5.6.1 *Chainsaw Dataset.*

With the chainsaw dataset, both approaches performed well due to the considerable amount of available labelled recordings. In this sense, regarding the spectral based approach, we only complement the configuration described in section 5.3 with the use of cross-validation.

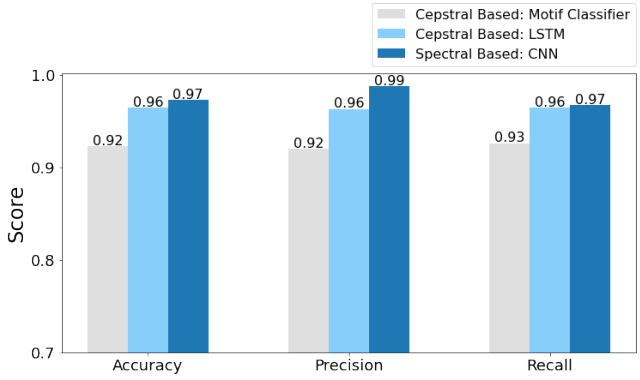

Figure 12: Cepstral and spectral based classification models performance (Chainsaw dataset).

The Convolutional Neural network is trained with 5-second-long Mel spectrograms and leverages the use of the SpecAugment data augmentation technique to increase its training set. In relation to the cepstral based approach, we expand the work developed in section 5.4, with the introduction of the cross-validation technique. In detail, the Long Short-Term Memory network is trained with the 5-second-long MFCCs. Also, both procedures use a predictive threshold of 60%.

In Fig 12, it is possible to compare the accuracy, precision, and recall scores from both approaches. From this figure, we can conclude that the spectral based classification model is the one that achieves the best performance. Nevertheless, in this particular setting, the cepstral based classification network registers similar results, stressing the scores obtained by the motif classifier as this alternative approach was able to approximate the performance of the other two.

Model	Acc. Low.	Acc. Up.	Prec. Low.	Prec. Up.	Rec. Low.	Rec. Up.
Spectral (CNN)	0.95	0.99	0.96	0.99	0.97	0.99
Cepstral (LSTM)	0.90	1.00	0.89	1.00	0.89	1.00
Cepstral (Motif C.)	0.92	0.93	0.92	0.93	0.92	0.93

Table 1: Spectral and cepstral based classification model accuracy, precision and recall T-student (95%) confidence intervals (Chainsaw dataset).

Additionally, in Table. 1 it is possible to observe the confidence bounds for each metric (accuracy, precision and recall), obtained at the final step of this analysis. Note that across all metrics, the spectral based approach is the one with higher confidence bounds. Not only it attains better results as the higher confidence intervals support our performance estimation.

All in all, the considerable amount of available labelled recordings is the key factor that contributes to the good performance of the developed models.

5.6.2 *Kaggle Dataset.*

In this particular dataset, the number of labelled recordings is very small, as a consequence, both approaches present different techniques to address this problem. As in the previous section, we expand the analysis concerned up until this point, with the introduction of the cross-validation technique.

In regard to the spectral based approach, our analysis includes two different architectures, both presented in section 4.1.2. The main difference between them lies in the introduction of an LSTM layer in the second architecture. Moreover, both networks are trained with 5-second-long Mel spectrograms, the training set of the two is increased with the "SpecAugment" technique, and the used predictive threshold value is 60%. Oppositely, the cepstral approach is trained with the 5-second-long MFCCs and according to the previous results, this procedure uses a predictive threshold of 75%.

The attained results are depicted in Fig. 13, and from a general point of view the spectral based classification model performed better than the cepstral one, supporting the idea that this approach is the more suitable to a bioacoustic classification task. In depth, apart from recall, it achieved the highest scores, with the network that includes the LSTM layer standing out from the other one, and justifying the introduction of this layer. The cepstral based classification model benefited from the cross-validation technique, as it registered a significant improvement in all scores, nevertheless, despite having a higher recall score, all the other metrics are lower than the ones obtained by the spectral based classification model.

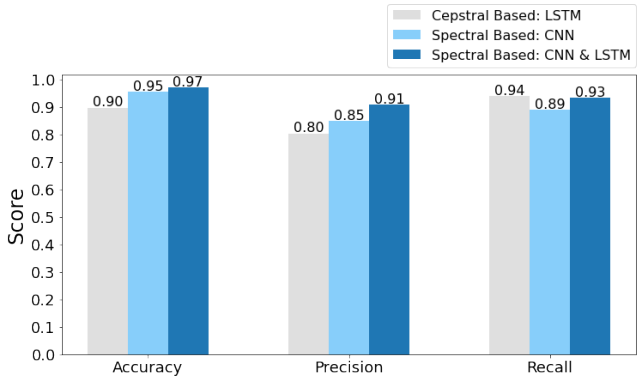

Figure 13: Cepstral and Spectral based classification models' performance comparison (Kaggle dataset).

Moreover, in Table. 2 we showcase the confidence bounds for the previously presented results. Once again, our performance estimation is much stronger for the spectral based classification model. Nonetheless, in both approaches, the lower confidence bound value is much smaller in comparison to the ones obtained in the previous setting. In a sense, the attained confidence intervals align with the learning difficulties faced by the developed networks, since their training relied on a limited set of labelled recordings. The spectral based classification model that includes the LSTM layer is the one that provides more certainty regarding our performance estimation, being the cepstral based classification model the one with the lower confidence bounds.

Model	Acc. Low.	Acc. Up.	Prec. Low.	Prec. Up.	Rec. Low.	Rec. Up.
Spec. (CNN)	0.89	1.00	0.64	1.00	0.70	1.00
Spec. (with LSTM)	0.90	0.99	0.74	1.00	0.75	1.00
Ceps. (LSTM)	0.74	1.00	0.56	1.00	0.76	1.00

Table 2: Spectral and cepstral based classification model accuracy, precision and recall T-student (95%) confidence intervals (Kaggle dataset).

To sum up, once again the spectral based approach seems to be the more adequate approach to a bioacoustic classification task, however, further research needs to focus on trying to attain higher confidence bounds when using the proposed methodology with datasets that have limited training data.

6 CONCLUSION

The field of bioacoustics is key to ensure the conservation of rainforests and their wildlife, as it helps reducing human impact on the environment. In this sense, Rainforest Connection emerges as a prominent source of environmental audio data, contributing to this cause by encouraging the development of bioacoustic monitoring systems. Deep learning methods have been successful on automating the process of species identification in environmental recordings, requiring nonetheless a large number of training samples per species. Thus, recent research focused on developing solutions capable of automate high-accuracy species detection in noisy soundscapes with limited training data.

Our work proposes a bioacoustic classification framework that achieved encouraging results, presenting capable solutions for the problem at hand. In depth, it details two different approaches to address this task, and it evaluates different concepts and procedures to determine the most suitable one. The first leverages off the transfer learning setting to reduce the training requirements, both the amounts of data and time, and relies on the Mel spectrograms to train the developed classification model (CNN). Conversely, the second uses the MFCCs to train the developed classification model (LSTM), proposing also an additional network trained on the matrix profile motifs to complement the proposed methodology.

We have demonstrated that both approaches are able to automate this process and can be included in bioacoustic monitoring systems. The spectral based approach performed better than the cepstral one, in both datasets. In particular, it achieved an accuracy of 0.97, a mean precision of 0.99 and a mean recall of 0.97, with the chainsaw dataset. With the Kaggle dataset, it registered an accuracy of 0.97, a mean precision of 0.91 and a mean recall of 0.93. The cepstral based approach aimed to present an alternative to the previous methodology, as it concerned other audio features and other network type. Additionally, it attempted to improve the results obtained by this procedure, by exploring a setting in which a classifier was trained with the motifs extracted by the matrix profile algorithm.

All in all, we can state that all the goals set for this work were fully met, namely the definition of a capable bioacoustic classification framework.

7 ACKNOWLEDGEMENTS

This work was supported by national funds by Fundação para a Ciência e Tecnologia (FCT) through project VizBig (PTDC/CCI-CIF/28939/2017).

REFERENCES

Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. Imagenet classification with deep convolutional neural networks. *Neural Information Processing Systems*, 25, 01 2012. doi: 10.1145/3065386.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Jack LeBien, Ming Zhong, Marconi Campos-Cerqueira, Julian P. Velev, Rahul Dodhia, Juan Lavista Ferres, and T. Mitchell Aide. A pipeline for identification of bird and frog species in tropical soundscape recordings using a convolutional neural network. *Ecological Informatics*, 59:101113, 2020. ISSN 1574-9541. doi: https://doi.org/10.1016/j.ecoinf.2020.101113. URL https://www.sciencedirect.com/science/article/pii/S1574954120300637.

Ming Zhong, Jack LeBien, Marconi Campos-Cerqueira, Rahul Dodhia, Juan Lavista Ferres, Julian P. Velev, and T. Mitchell Aide. Multispecies bioacoustic classification using transfer learning of deep convolutional neural networks with pseudo-labeling. *Applied Acoustics*, 166:107375, 2020. ISSN 0003-682X. doi: https://doi.org/10.1016/j.apacoust.2020.107375. URL https://www.sciencedirect.com/science/article/pii/S0003682X20304795.