Identification of bird sounds in soundscapes with Convolutional Neuronal Network

Luisa Toro Villegas
Universidad EAFIT
Medellín, Colombia
ltorov@eafit.edu.co

Andrés Gómez Arango
Universidad EAFIT
Medellín, Colombia
agomez10@eafit.edu.co

Manuela Guarnizo Sepulveda
Universidad EAFIT
Medellín, Colombia
mguarnizos@eafit.edu.co

Abelino Sepulveda Estrada
Universidad EAFIT
Medellín, Colombia
asepulvede@eafit.edu.co

March 2022

1 Problem statement

Tracking trends in population sizes for wildlife are a great way to help experts determine courses of action for protecting certain species or ecosystems. Birds in particular are a great animal for conservationists because they indicate the general wellness of an environment. Large amounts of information, especially of labeled audio samples, have been collected by professionals ornithologists and amateurs alike. However, field categorization requires long hours of manual labor and the presence of an expert, which is not practical for long-term studies. There arises the need to implement methods for classifying bird species from audio samples.

Bird songs are composed of simple notes which can be combined in particular sequences as syllables, syllables into phrases, and phrases into songs. Thus, can be considered as chains of discrete acoustic elements in a given temporal order, depending on the particular species. The level of variation of the song depends on the species as well (Berwick *et al.*, 2011). Syllables are the structure most used in bird species recognition, though they can be difficult to identify in audio with a lot of background noise, which needs to be reduced using audio feature extraction techniques.

1.1 Research question

For wild birds, can categorization machine learning algorithms be used to greatly improve the duration that this task takes doing manually without worsening the classification rate?

1.2 Project barrier

For the purpose of this project, the aim is to work upon the solution found by Conde *et al.* (2021) and improve upon it.

2 Objectives

Use acoustic pattern recognition algorithms to identify bird species from their calls given long audio recordings of soundscapes from Hawaii.

2.1 Specific objectives

- Interpret audio features by using visual aids such as spectrograms and techniques such as discrete Fourier transforms.
- Perform an exploratory data analysis that captures relationships between data through different metrics and visual aids.
- Implement acoustic recognition algorithms that detect whether a bird call is present in given audio and to identify the bird species.

3 Literature review

The field of machine learning has shown ideal results when evaluating the environment using pictures, but audio classification has not been much explored. One of the best forms of Artificial Neural Network are the Convolutional Neural Networks(CNNs), this method is used primarily on the field of pattern recognition on images and have proven to be a very effective method for image classification per (O'Shea & Nash, 2015). The spectrogram of an audio file is a visual representation of the frequencies over the time. This way, the representation of a given recording is technically an image and therefore, it is possible to use Machine Learning methods that works for image classification to solve this classification problem (Hershey *et al.*, 2017). Piczak (2015) used CNNs to classify short audio clips of environmental sounds using a collection of environmental recordings such as animals, natural soundscapes, water sounds and urban and domestic noise; finding out that the CNN approach seems to lead to a viable solution.

Another interesting approach for audio files and sounds classification was proposed by Park & Lee (2015) who use Convolutional Neuronal Network to music instruments sound classification.

In their approach, they present a new model to musical instrument classification model using CNNs. To create learned features from CNNs, they not only use conventional spectrograms but also propose the use of multi-resolution recurrence plots (MRPs) that contain the phase information of a raw input signal. The advantage of using this method is that they feed the characteristic timbre of the instrument into a neural network, which cannot be extracted from the spectrogram representations. Finally, the results obtained from this research show that the classifier performance of this combined model exceeds the results obtained from the traditional model.

Rong (2016) proposed a methodology which first illustrates the hierarchical structure of audio data, that is made up of four layers: Audio frame, Audio clip, Audio shot, and Audio high-level semantic unit. Secondly, they classify the audio data by support vector machines (SVM). The task of audio data classification is implemented based on data feature extraction of the three types of audio features. In conclusion, The experimental results confirmed that this methodology shows a proper audio classification accuracy using another approach.

4 Methodology

Bio-acoustic monitoring is a non-invasive method of wildlife surveillance by setting up cameras or audio recorders over a certain territory and using the audio obtained to detect what species are present and how many individuals there are. Once the recordings are obtained, the first step is to perform audio feature extraction. A sound event refers to a specific sound from a given source and many of them refer to a sound scene. The challenge when identifying the source of a single sound event is the noise generated from all other sources, and this is where the computational analysis of sound scenes becomes relevant to extract useful information from audio (Virtanen *et al.*, 2018). Most algorithms focus on extracting time-frequency information, and in order to obtain that domain, discrete Fourier transforms (DFT) are used. Fourier transforms work by breaking down a given signal into sinusoidal functions, which represent the signals' frequency, amplitude, and phase.

4.1 Exploratory data analysis and feature extraction

The feature extraction process of the audio can be considered as a data reduction process in which the basic characteristics of the analyzed data are extracted from a small data rate transforming high-dimensional vectors into lower ones (Bahoura, 2009). The Fourier transform of a discrete time-signal is defined as:

$$S[m,k] = \sum_{n} s[n]w[n-m]e^{-j2\pi nk/N} \tag{1}$$

Where N is the number of discrete frequency, $w[n]$ is a short-time windowing function of size L, centered at time location m.

Once the audio is in time-frequency form, Mel Frequency Cepstrum Coefficients (MFCC) can be extracted, which are a certain type of acoustic characteristic that has proven extremely useful for human voice recognition algorithms (Mohan *et al.*, 2014). This process is done by converting individual audio files into a Mel spectrogram, and then extracting the information contained in that image into a given array.

4.2 Multi-class classifier: Convolutional Neural Network

Once sufficient audio features are extracted, a multi-class classifier is to be implemented, first by detecting whether the audio contains a bird call and later by detecting the specific bird species. Some of the most used classifiers for bio-acoustic signals in literature are k-nearest neighbors, Support vector machines, pruned decision trees, multi-layer perception, Adaptive Boosting, and Convolutional Neural Networks. k-nearest neighbors (kNN) is a non-parametric supervised learning method that relies on distance, or similarity for classification, with k the number of clusters that the algorithm is to classify, and it was developed by Fix & Hodges (1951). Support vector machines (SVM) is a linear model for classification and regression problems. Its algorithm creates a line or hyper-plane which separates the data into classes, Cortes & Vapnik (1995) firstly introduced it. Adaptive boosting (AdaBoost) is an ensemble learning method that uses an iterative approach to learn from the mistakes of weak classifiers and convert them into strong classifiers (Freund *et al.*, 1999). Multi-layer perceptron (MLP) is a neural network model in which the mapping between the outputs and inputs is non-linear. Convolutional Neural Networks (CNN) are a type of neural network which consist of three types of layers: the convolutional layer, the pooling layer and the fully-connected layer IBM (2022). For the purpose of this project, which is to classify with audio signals, and due to its proven superior performance against other similar complexity models, CNN's were chosen as the methodology.

In CNNs, the first layer is the convolutional layer, which requires an input data, a filter and a feature map. In our case, the input data will consist of the arrays containing the Mel spectrogram information, which account for the pixel information for the height, the width and the depth, or RGB. The filter or feature detector goes over an image checking whether a given feature is present, which is known as a convolution. The chosen convolution for the project is *2D convolution*, in which with a kernel, is a matrix of weights of a set size, in our case 3×3, that moves through a two-dimensional space taken from the input, and multiplies it element-wise with a filter, and then sums up the values, shown below on Figure 1.

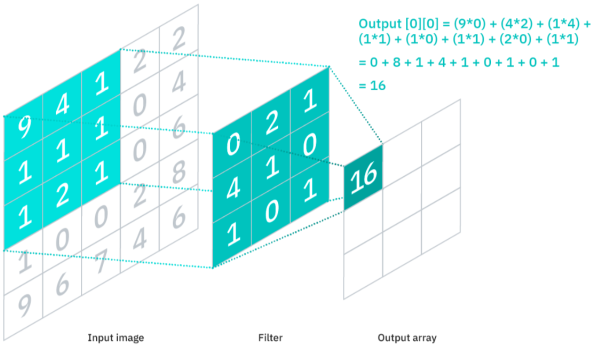

Output [0][0] = (9*0) + (4*2) + (1*4) + (1*1) + (1*0) + (1*1) + (2*0) + (1*1)
= 0 + 8 + 1 + 4 + 1 + 0 + 1 + 0 + 1
= 16
16
Input image
Filter
Output array

Figure 1: Convolutional layer, IBM (2022)

Batch normalization is a technique for training neuronal networks that standardizes the inputs to a network. This allows to stabilize the learning process and drastically reduce the number of training epochs required to train deep networks, reducing the generalization error.

During the training phase, the equation used for Batch normalization in the *TensorFlow* library is as follows:

$$\frac{\gamma(\text{batch} - \text{mean(batch)})}{\sqrt{\text{var(batch)} + \epsilon} + \beta} \tag{2}$$

Similarly, during the testing phase, the equation uses the means and the variances calculated during the training phase.

After the convolutional layer a pooling layer is added, which generates a new set of the same number of pooled feature maps. There are two common functions used in this layer, the *Maximum Pooling* and the *Average Pooling*. These pooling operations calculate the largest value and the average for every patch on the feature map respectively. The resulting output size is shown in 3, where the pool size used is $(2, 2)$.:

$$\text{size} = \left\lfloor \frac{\text{inputSize - poolSize}}{\text{strides}} \right\rfloor + 1 \tag{3}$$

Furthermore, the *Dense Block* is built to connect all the mentioned layers directly with each other, in order to compact the network. Finally, the *Classification layer* is constructed to make the bird classification. Once the model is created it will be compiled, by defining the metrics, the optimizer and the loss function to train the model. The metric used is the *F1 score*, which is the harmonic mean of the precision and recall of a given classifier, shown in equation 4. It gives a measure of a model accuracy, where the highest value is 1, representing perfect precision and recall, and the lowest value is 0, which is either zero recall or precision.

$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{4}$$

The optimizer used is Adaptive moment estimation or *Adam* optimization, a type of stochastic gradient descent introduced in 2014 that is good for sparse data like in the present case, because of its adaptive nature. The results obtained are usually better than other algorithms and has a lower computational cost. Finally, the loss function chosen was cross-entropy which is used for multi-class problems. The objective of the optimization problem stated is to minimize the loss function. Defining entropy of a random variable X, as the level of uncertainty inherent in the variables possible outcome, Koech (2021). For $p(x)$, the probability distribution of random variable X, the entropy $H(X)$ is defined as equation 5:

$$H(X) = \begin{cases} -\int_x p(x) \log p(x), & \text{if } X \text{ is continous} \\ -\sum_x p(x) \log p(x), & \text{if } X \text{ is discrete} \end{cases} \tag{5}$$

Cross-entropy loss is used when adjusting model weights during training, and the aim is to minimize the loss, or to get as close to 0. The cross-entropy loss, or L_{CE} is defined in equation 6:

$$L_{\mathrm{CE}} = -\sum_{i=1}^{n} t_i \log\left(p_i\right), \text{ for } \mathrm{n} \text{ classes} \tag{6}$$

4.3 Model validation and testing

The main objective in supervised learning is to train the model with the known data so when it is given a new input, in can classify it properly. To understand the accuracy of your models predictions, its a good idea to first measure certain metrics under already labelled data. Train test split is a model validation technique which takes your training data (labelled data) and separates it randomly into a training set and a validation set, consisting of 75% and 25% of the data. If the data is not separated, the metrics calculated will be completely unreliable.

4.3.1 Multi-class Classification Metrics

Calculating metrics for a classifier can be useful for both tuning the model and comparing it with other models. The following section provides a brief introduction to some commonly used metrics in the multi-class case that will be used to evaluate the performance of our model, according to Grandini *et al.* (2020) and Koech (2021):

- *Multi-class confusion matrix:* N by N matrices, where each position represents the proportion of true positives between total positives for each class. It shows a comparison between actual and predicted values.
- *Accuracy:* This metric is useful to evaluate the performance of classification models giving the accuracy of the prediction model, this is the fraction of predictions that the model got right.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} \tag{7}$$

This metric can also be calculated in terms of positives and negatives, using the following formula:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \tag{8}$$

- *ROC - AUC:* The Area Under the Receiver Operating Characteristic Curve is important to evaluate a multi-class classification model, this metric determine the model ability to distinguish between classes, it also can be interpreted as the probability that the model ranks a random positive example more highly than a random negative example. The Area Under the Curve gave us information about the degrees or the measure of separability between classes and the Receiver Operating Characteristics that is a probability curve. The ROC curve is plotted with the metric FPR (False positive rate) on the y-axis against the TPR (True positive rate) on the x-axis, we can calculate this rates with the following formulas:

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{9}$$

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{10}$$

  The AUC values are normalized between 0 and 1, when the models predictions are completely incorrect it has a AUC of 0, when the prediction are 100% correct takes a value of 1.

- *Cohen's Kappa score:* Is a statistic that is used to measure inter-rater reliability (and also intra-rater reliability) for qualitative (categorical) items Koech (2021). This score is a more complete version of the accuracy, and its given by the following equation:

$$\text{k} = \frac{P_0 - P_e}{1 - P_e} \tag{11}$$

  Where P_0 is the observed proportional agreement and P_e is the probability that true values and false values agree by chance.

- *Matthew's correlation coefficient*

  This coefficient is an alternative for the accuracy, and it is expressed as follows:

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{12}$$

  This metric represents the correlation between true values and the predicted ones. It can go from -1 to 1, meaning that the closer it is to 1 the better the classifier. But if the metrics is around 0 it means that the classifier is no better than randomly choosing. Unlike precision and recall, MCC is perfectly symmetric so if the positive and negatives classes are swapped it shows the same score

5 Data and analysis

The train and test data is provided by Kaggles's BirdCLEF competition. The training set consists of short audio recordings of 152 bird species and 1 audio of a soundscape, each species containing a different amount of audios, and they are both given in an *.ogg* audio file format.

Some of the most relevant fields in the train data are:

- **primary_label:** the code for the bird species.

- **secondary_label:** background species as annotated by the recordist.

- **filename:** the audio file

The recordings also contain certain metadata, such as location, which can prove useful because most species of birds are either endemic or specific to certain areas, and it is unlikely to find this species in other places. However, migratory birds are the exception, so in case they are catalogued as such, that parameter can be overlooked when classifying them.

5.1 Exploratory data analysis

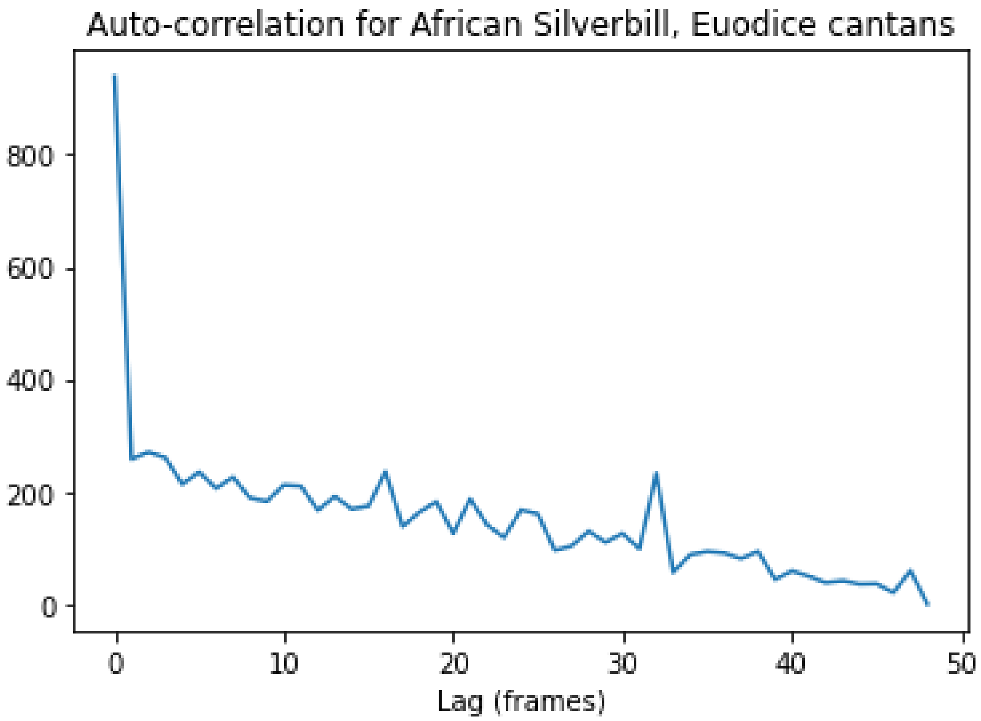
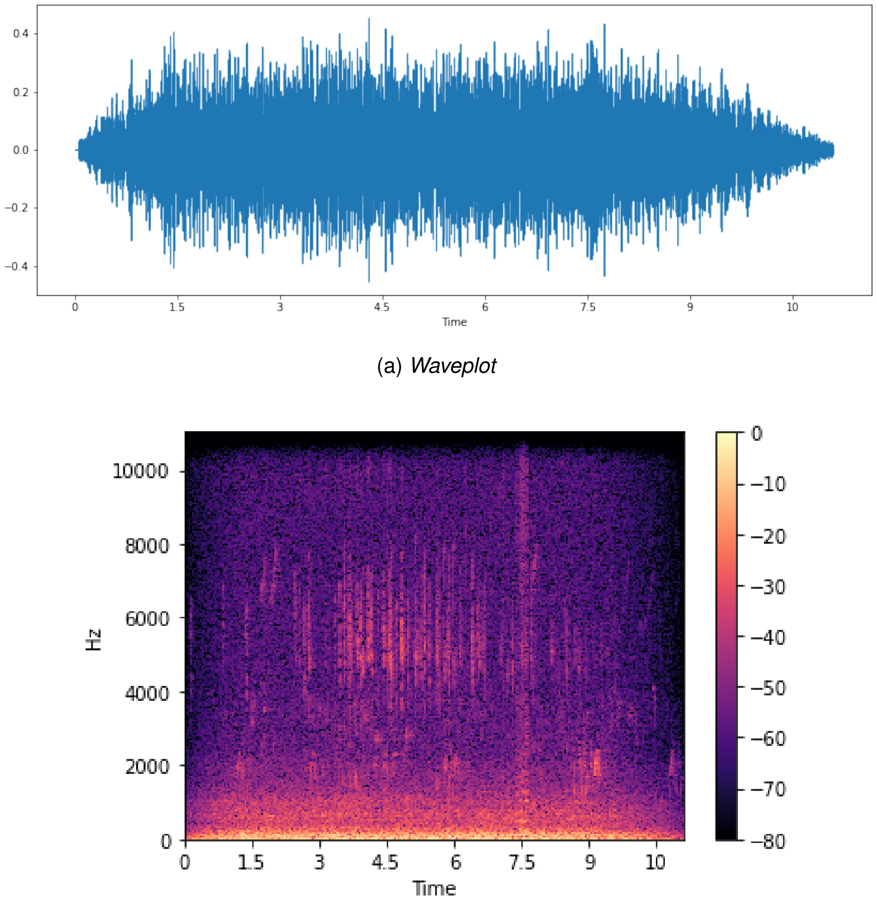


Figure 2: Auto-correlation

In Figure 3 two different types of data visualization are shown, wave plots and spectrograms. Wave plots are the usual type of representation, where a wave amplitude is plotted over time. This data can only be captured by computers in a discrete time, and a hyper parameter called *sampling frequency* or *sampling rate (sr)* is used to define said capturing. For the purpose of this study, the *sampling rate* is defined as *32000 Hz*. On the other hand, spectrograms show frequency over time, and they are generated using the *librosa* library.



(a) *Waveplot*

(b) *Mel spectrogram*

Figure 3: Visualization for audio sample for African Silverbill, *Euodice cantans*

5.2 Audio features

Spectral-domain features are directly extracted from the power value of a spectrum. The spectral centroid, spectral bandwidth, and spectral roll-off are calculated using documentation found in Verma (2021).

The spectral centroid measures the spectral shape and position of the spectrum. In other words it measures the shape of the wave signals spectrum. While higher the value of the centroid corresponds to a greater energy of the signal that is concentrated in higher frequencies. In Figure 4 it can be observed the spectral centroid for one of the training data. Expressed as:

$$C(i) = \frac{\sum_{k=0}^{N-1} k|Xi(k)|}{\sum_{k=0}^{N-1} |X_i(k)|} \tag{13}$$

Where $x_i(n)$, $n = 0, 1, \ldots, N{-}1$ is the sample of the i-th frame, with $X_i(k)$, $k = 0, 1, \ldots, N{-}1$ the discrete Fourier transform (DFT) coefficients of the sequence.

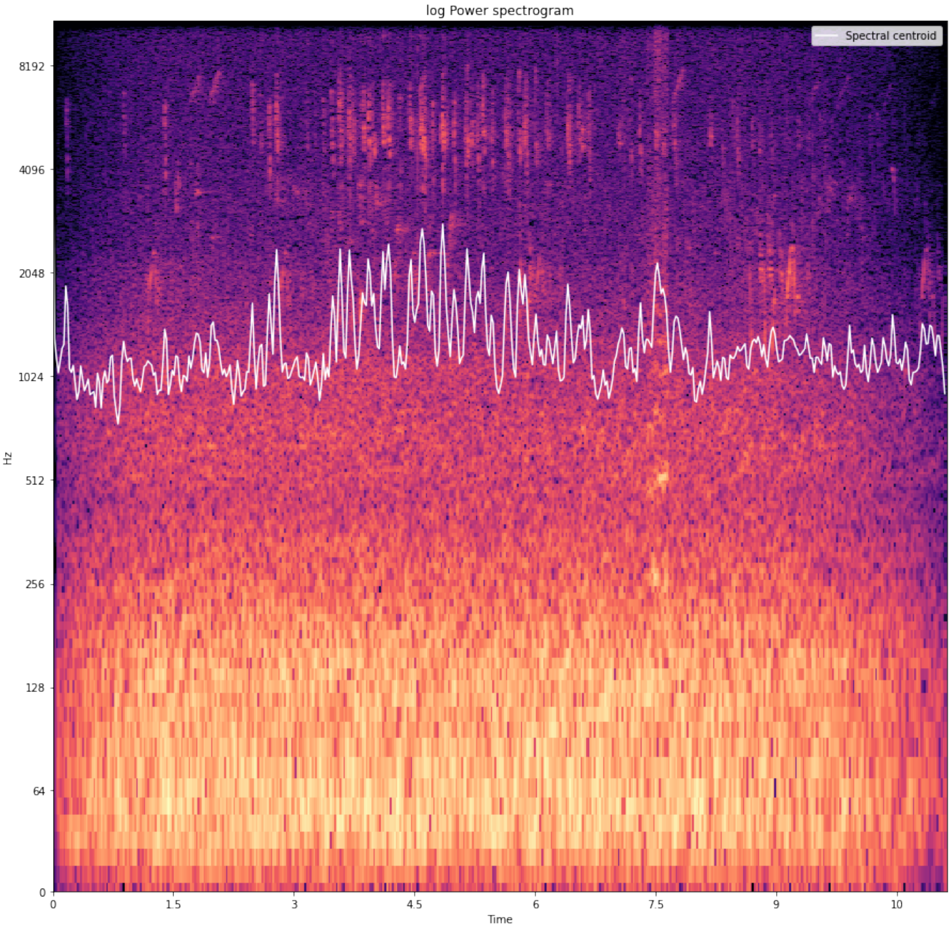


Figure 4: Spectral centroid for African Silverbill, *Euodice cantans*

In Figure 4 we can see that in the time interval from 4 to 6 the spectral centroid takes the highest values of the observation, that means that in this interval we can analyze higher energy signals from the observation.

In Figure 5, the spectral bandwidth of the chosen audio sample represent the difference between the lower and upper frequencies in a continuous band. The blue area in the image below show the largest deviation of the signal at every time frame by covering the area from blue colour. For example if the lower frequency of a signal is $120HZ$, and the upper frequency $1000HZ$ the bandwidth of the signal will be $880HZ$

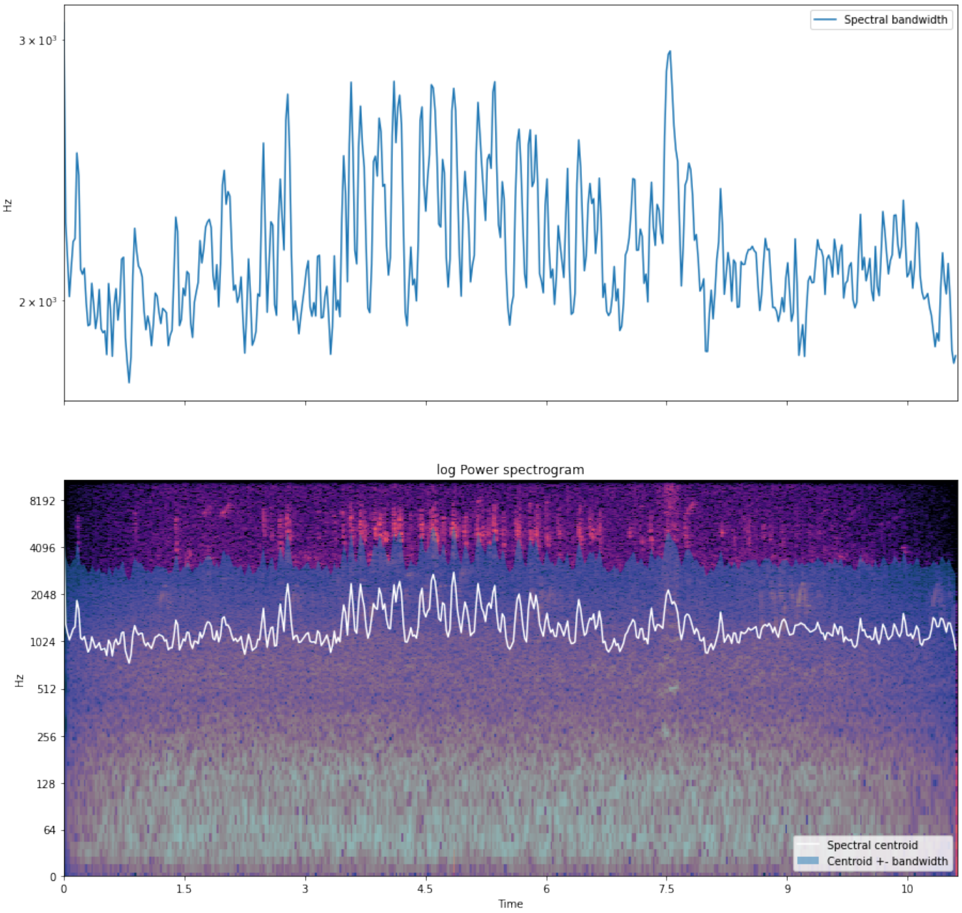


Figure 5: Spectral bandwidth for African Silverbill, *Euodice cantans*

In Figure 6, the spectral roll-off of the chosen audio sample is shown, which filters frequencies out of a specified range, which for the figure bellow was chosen to cut off the lower and upper 5%. Is it computed as follows:

$$\sum_{k=0}^{k(i)} |X_i(k)| = \frac{P}{100} \sum_{k=0}^{N-1} |X_i(k)|$$

where $xi(n)$, $n = 0, 1, \ldots N{-}1$ is the sample of the i-th frame, with $Xi(k)$, $k = 0, 1, ...N{-}1$ the discrete Fourier transform (DFT) coefficients of the sequence.

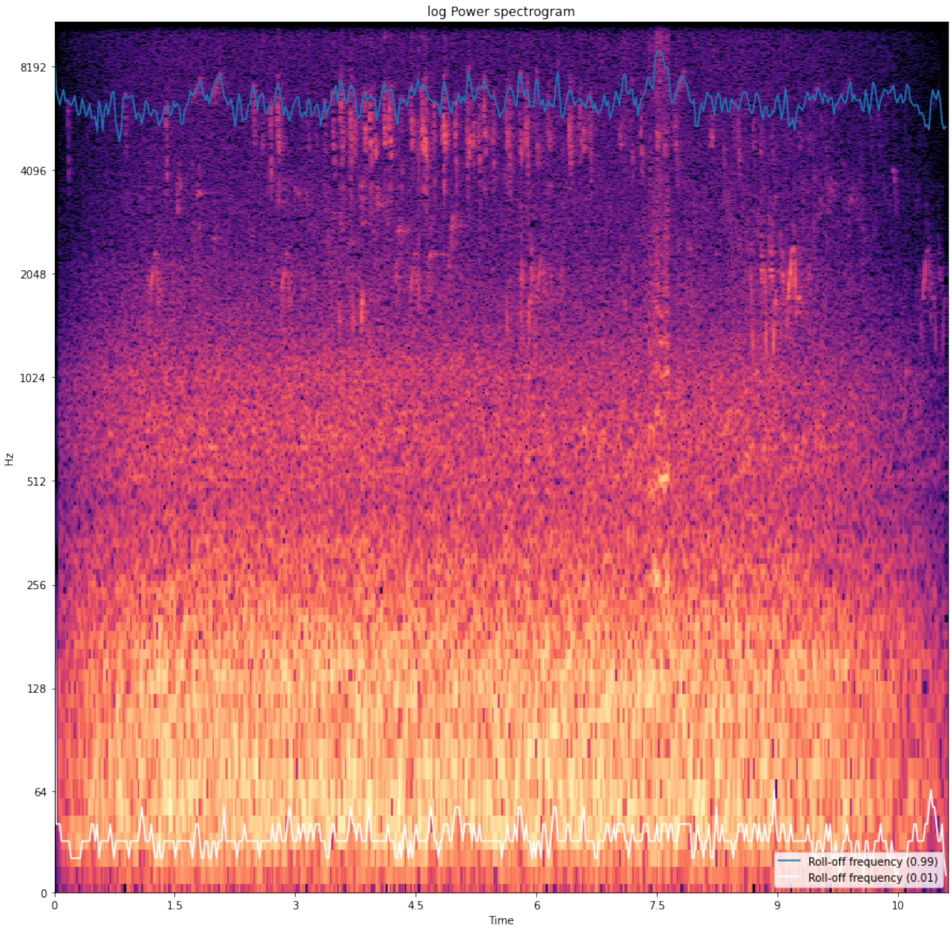


Figure 6: Spectral roll-off for African Silverbill, *Euodice cantans*

The figure shows the hi-pass roll-off in the blue line and low pass roll-off in the white line, each represent the upper and lower bounds, respectively.

6 Mapping

In this section the training data is explored to identify different characteristics of the recordings, first it is recognized the locations where the audios were recorded, then it is analyzed the number of species in the data and the most and least recorded birds species.

In order to have a global picture of the regions that provide the recordings a heat map is used. For this purpose the Python library called Folium is implemented, this makes easy to turn the data into a map an visualize it. In Figure 7 it can be observed the heat map of the recording locations.

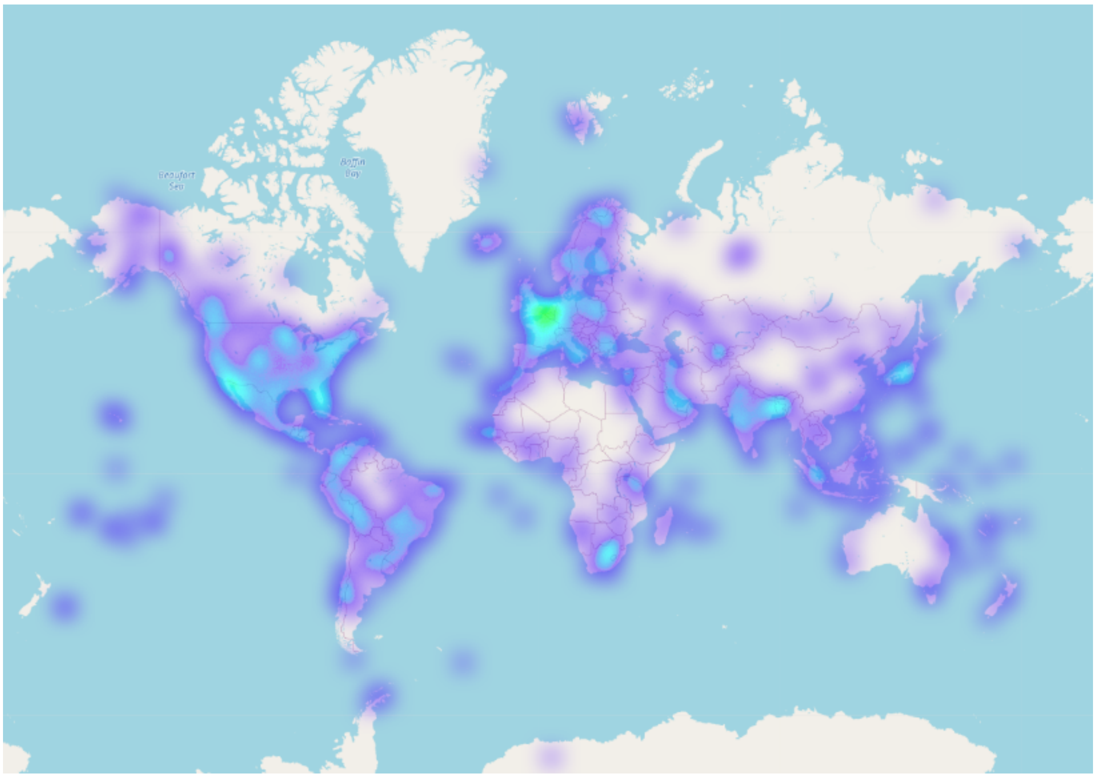


Figure 7: Heat map

To identify the most and least recorded bird species in the audio files a bar graph was made, the results showed that the most recorded type of bird was the Barn Owl as shown in Figure 8 and the least one was the Maui Parrotbill that has only one audio file showed in Figure 10, also the *Folium* library was implement to observe where this species were recorded, this tool was very useful to recognize the biodiversity of species in the regions and the location of the main data providers.

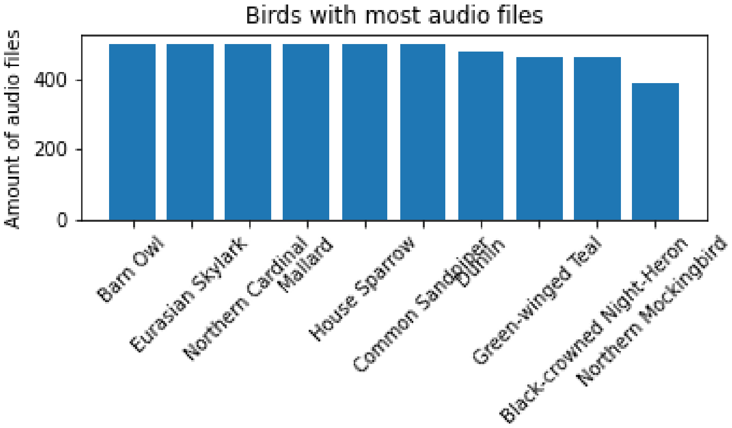


Figure 8: Bar graph for bird species with most sightings

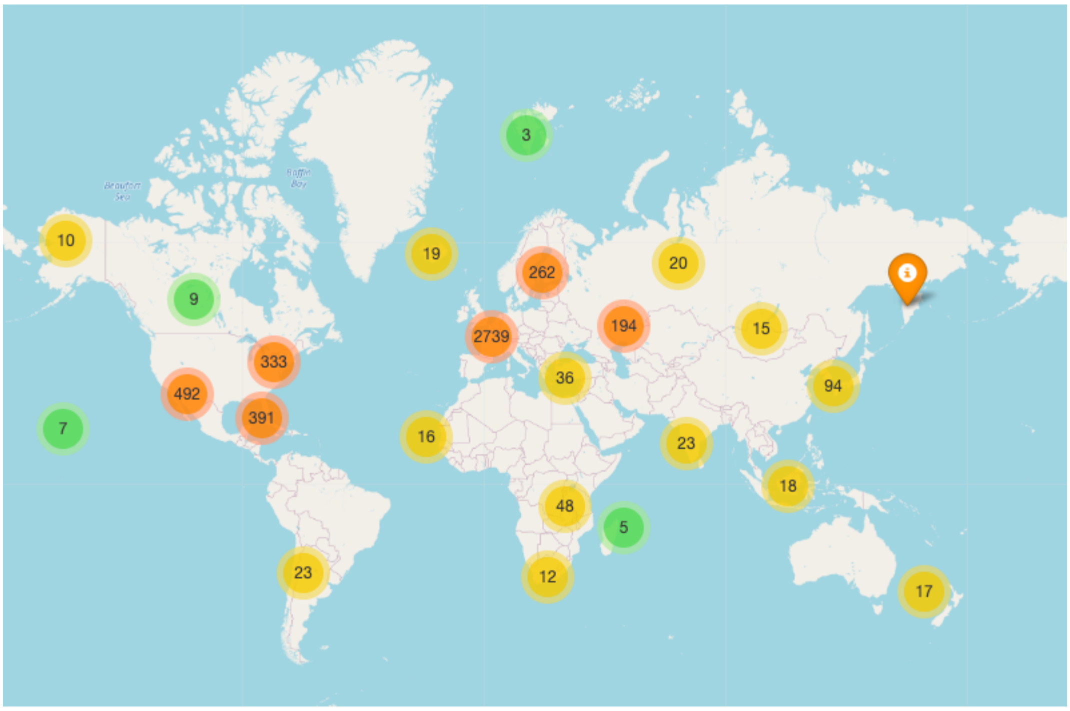


Figure 9: Locations for the most recorded species

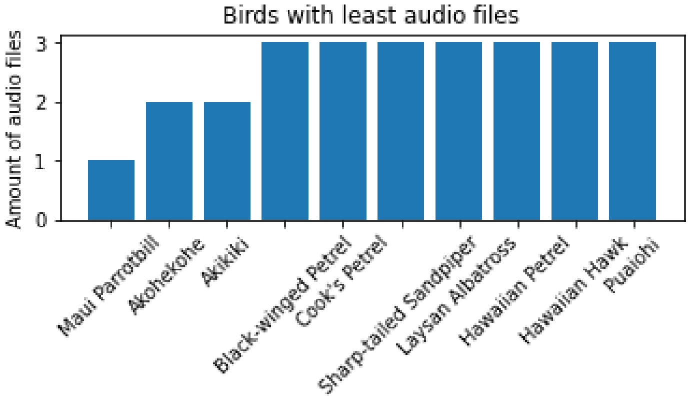


Figure 10: Bar graph for bird species with least sightings

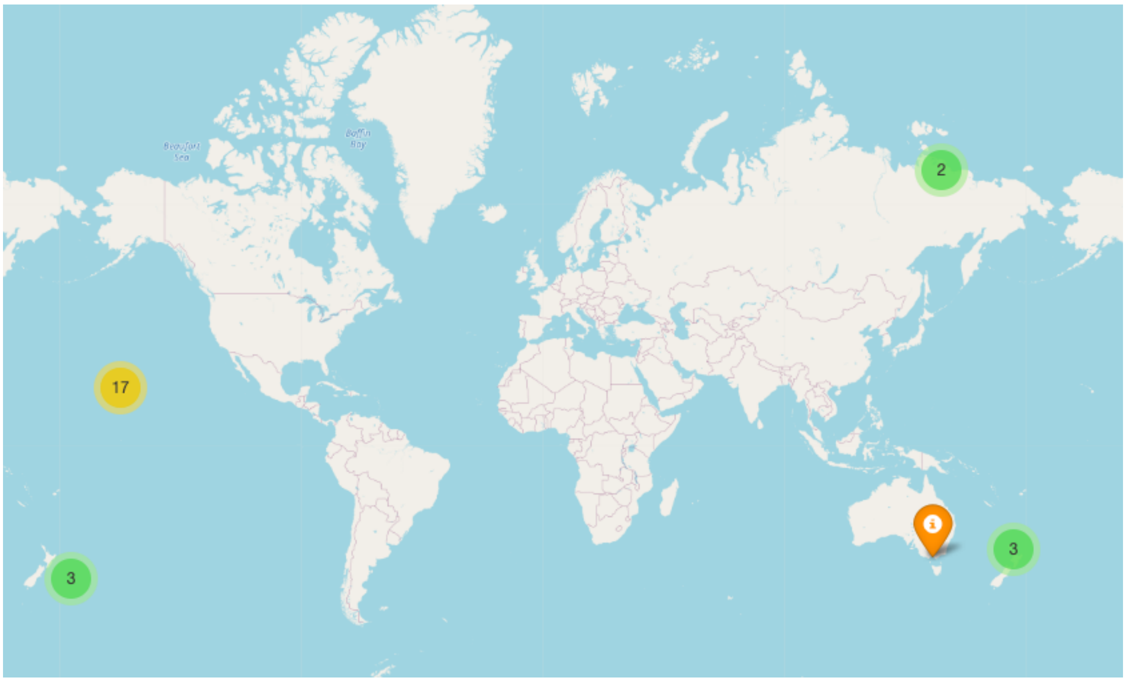


Figure 11: Locations for the least recorded species

7 Detailed plan

- **Literature review:** There will be a 2 week period to read about the problem and all the information regarding bird sounds classification needed. This is the first step of the project.
- **Data analysis:** Then, a study of the data available will be done in order to check that there is enough data, it is consistent and precise.
- **Model design:** Once the data has been studied, the first version of the model will be designed.
- **Model testing:** Having a working version of the model, in this part, it is going to be tested to see if it works accordingly to the literature and the data fit well.
- **Article writing:** Once the model has been designed and tested, it is time to begin the documentation of all the process for the article. The tests, corrections, data distributions will be described in the article.
- **Model adjusting:** In parallel, the model will be fitted to fix possible flaws found in the model testing.
- **Presentation preparation:** Finally, having the model and article almost done, the preparation for the final exposition will take place

	Weeks										
Activity	1	2	3	4	5	6	7	8	9	10	11
Literature review	■	■									
Data Analysis			■	■	■						
Model design						■	■	■			
Model testing									■		
Article writing									■	■	
Model adjusting										■	
Presentation preparation											■

8 Results

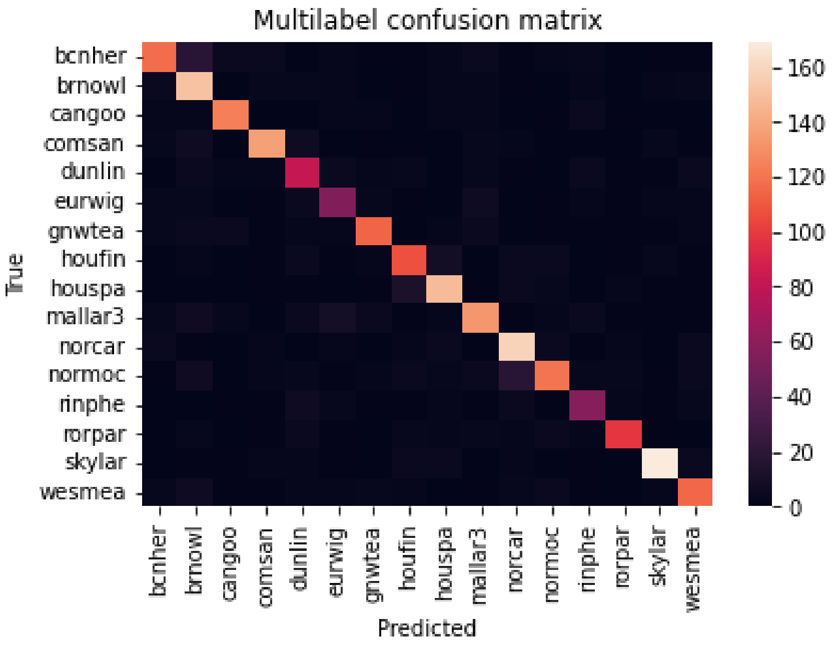
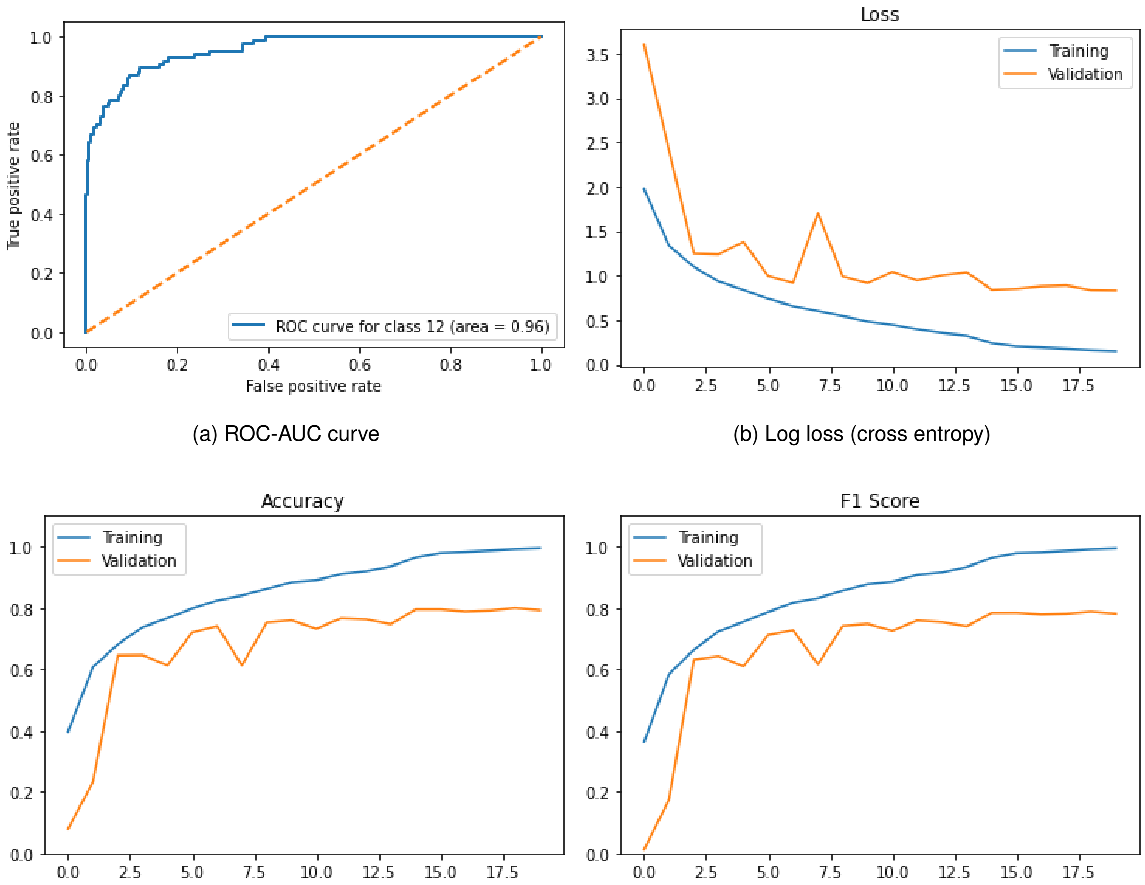


Figure 12: Multi-class confusion matrix

The figure 12 shows the confusion matrix. It can be seen that the values on the diagonal of the matrix, which are the values correctly estimated by the model, maintain a very high scale with few exceptions, while the values off the diagonal are very low, which means that the model is making good predictions.



(a) ROC-AUC curve

(b) Log loss (cross entropy)

(c) Accuracy

(d) F1 score

Figure 13: Metrics for validation

In the figure 13 it is shown the different metrics implemented for model validation. The ROC curve shows the growth of the performance of the model as the epochs of the neural network increase, which was the expected behavior, since generally, the performance tends to improve as the epochs increase, also the area under the curve (AUC) takes a very high value and close to 1 so it is possible to say that the model has a good performance.

Furthermore, the behavior of the Accuracy and F1 Score curves have similar behavior to that of the ROC curve, however, in these two it can be observed the training and validation accuracy, from which the gap between training and validation accuracy is not considerable, and it can indicate the presence of a low over-fitting.

Finally, the Log loss curve shows the learning rate of the neural network as the epochs grow, as shown in the figure 13, the training curve shows that the network has a good learning rate since it decreases at a good pace while in the curve test it shows that the model fit is a little lower, however, it still has very good behavior. In conclusion, all the metrics for validation confirm a good performance of the model.

9 Conclusions and future work

After the model testing and reviewing the results it is possible to affirm that the project has successfully proven that categorization machine learning algorithms can be used to classify birds using only audio as input. This has proven to be very useful in the field due to the great time improvement it shows when classifying wild birds and the possibility to attend the task without needing a professional in the matter being present in every instance.

It is important to say that the model training was a key factor to its results. This because firstly the model could not determine properly whether a bird was from an species or another, and that could be due to a low trained process, then it was necessary to explore more options and give more information to the model at first so it could learn better, but constantly making sure the model was not over fitted.

Finally the confusion matrix is the perfect chart to see how the model behaves and it shows a great classification, making the purpose of this project real: save classification time without worsening the classification rate.

For future work it could be interesting to expand this project to human voice classification for example. thanks to the nature of CNNs and how the model receives and interpret the audio, it could be possible to have any kind of audios as inputs, it would only require a proper training.

10 Plan execution

The execution plan was followed as intended but some changes were made caused by the difficulty of some tasks in reality. This is how it went:

	Weeks										
Activity	1	2	3	4	5	6	7	8	9	10	11
Literature review	■	■									
Data Analysis			■	■							
Model design					■	■					
Model testing							■	■	■		
Article writing									■	■	
Model adjusting										■	
Presentation preparation											■

As the chart shows, the Data analysis required less time than expected and the model design as well, in the other hand, the model testing was the task that needed more attention during the process, due to little changes in the code and some parameter-change testing. Therefore, it is important to take this into consideration for future work and other projects, in order to work more efficiently and make the time distribution more accurate.

11 Ethical implications

If the model proves useful and accurate for detecting bird species in a soundscape, it can mean a big step towards ensuring the continued survival of said species. However, when endangered animals are concerned, there is also the possibility that said technology can lead to illegal wildlife trade by making rare or exotic birds, or animals in general, easier to find for poachers. Furthermore, the automation of the identification of bird sounds can lighten the tasks for bird and animal conservationists by saving them hours of having to sit through soundscapes and manually labeling data. If the model is inaccurate it can lead to false classification and thus make wrong decisions regarding which species or areas require more action.

12 Legal and commercial aspects

The relevance of this project resides in the potential help that the model could bring to the animal protection entities or even governments. This way, the classification of birds, the possibility to know how many individuals of a specific species are in a certain area and even the possible recognition of new species will represent a great contribution to the field.

Legally, the problem and data that this project is based on are from Kaggle. Even though this project itself will not be submitted into Kaggle as participation, the data used belongs to the problem stated in the platform. Because of this, the intention of the project if it results suitable for the real-life problem, is to deliver it as open source.

References

Bahoura, Mohammed. 2009. Pattern recognition methods applied to respiratory sounds classification into normal and wheeze classes. *Computers in biology and medicine*, **39**(9), 824–843.

Bardeli, Rolf, Wolff, Daniel, Kurth, Frank, Koch, Martina, Tauchert, K-H, & Frommolt, K-H. 2010. Detecting bird sounds in a complex acoustic environment and application to bioacoustic monitoring. *Pattern Recognition Letters*, **31**(12), 1524–1534.

Berwick, Robert C, Okanoya, Kazuo, Beckers, Gabriel JL, & Bolhuis, Johan J. 2011. Songs to syntax: the linguistics of birdsong. *Trends in cognitive sciences*, **15**(3), 113–121.

Conde, Marcos V, Shubham, Kumar, Agnihotri, Prateek, Movva, Nitin D, & Bessenyei, Szilard. 2021. Weakly-Supervised Classification and Detection of Bird Sounds in the Wild. A BirdCLEF 2021 Solution. *arXiv preprint arXiv:2107.04878*.

Cortes, Corinna, & Vapnik, Vladimir. 1995. Support-vector networks. *Machine learning*, **20**(3), 273–297.

Fix, Evelyn, & Hodges, Joseph Lawson. 1951. Discriminatory analysis. Nonparametric discrimination: Consistency properties. *International Statistical Review/Revue Internationale de Statistique*, **57**(3), 238–247.

Freund, Yoav, Schapire, Robert, & Abe, Naoki. 1999. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, **14**(771-780), 1612.

Frommolt, Karl-Heinz, & Tauchert, Klaus-Henry. 2014. Applying bioacoustic methods for long-term monitoring of a nocturnal wetland bird. *Ecological Informatics*, **21**, 4–12.

Grandini, Margherita, Bagli, Enrico, & Visani, Giorgio. 2020. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756*.

Hershey, Shawn, Chaudhuri, Sourish, Ellis, Daniel PW, Gemmeke, Jort F, Jansen, Aren, Moore, R Channing, Plakal, Manoj, Platt, Devin, Saurous, Rif A, Seybold, Bryan, *et al.* 2017. CNN architectures for large-scale audio classification. *Pages 131–135 of: 2017 ieee international conference on acoustics, speech and signal processing (icassp)*. IEEE.

IBM, Cloud Education. 2022. *What are convolutional neural networks?*

Koech, Kiprono Elijah. 2021 (Nov). *Cross-entropy loss function*.

Mohan, Bhadragiri Jagan, *et al.* 2014. Speech recognition using MFCC and DTW. *Pages 1–4 of: 2014 International Conference on Advances in Electrical Engineering (ICAEE)*. IEEE.

Mporas, Iosif, Ganchev, Todor, Kocsis, Otilia, Fakotakis, Nikos, Jahn, Olaf, Riede, Klaus, & Schuchmann, Karl L. 2012. Automated acoustic classification of bird species from real-field recordings. *Pages 778–781 of: 2012 IEEE 24th International Conference on Tools with Artificial Intelligence*, vol. 1. IEEE.

O'Shea, Keiron, & Nash, Ryan. 2015. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*.

Park, Taejin, & Lee, Taejin. 2015. Musical instrument sound classification with deep convolutional neural network using feature fusion approach. *arXiv preprint arXiv:1512.07370*.

Piczak, Karol J. 2015. Environmental sound classification with convolutional neural networks. *Pages 1–6 of: 2015 IEEE 25th international workshop on machine learning for signal processing (MLSP)*. IEEE.

Rong, Feng. 2016. Audio classification method based on machine learning. *Pages 81–84 of: 2016 International conference on intelligent transportation, big data & smart city (ICITBS)*. IEEE.

Verma, Yugesh. 2021 (Nov). *A tutorial on spectral feature extraction for Audio Analytics*.

Virtanen, Tuomas, Plumbley, Mark D, & Ellis, Dan. 2018. Introduction to sound scene and event analysis. *Pages 3–12 of: Computational analysis of sound scenes and events*. Springer.