INSECT SPECIES SOUND CLASSIFICATION USING DEEP LEARNING WITH SMALL DATA

DARIO GANDINI

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
AT THE SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
OF TILBURG UNIVERSITY

STUDENT NUMBER

2061389

COMMITTEE

Supervisor: dr. Dan Stowell
Second Reader: dr. Afra Alishahi

LOCATION

Tilburg University
School of Humanities and Digital Sciences
Department of Cognitive Science &
Artificial Intelligence
Tilburg, The Netherlands

DATE

January 14, 2022

WORD COUNT

8,550 words

Contents

INSECT SPECIES SOUND CLASSIFICATION USING DEEP LEARNING WITH SMALL DATA

DARIO GANDINI

Abstract

Nature is under siege. While human population is constantly growing, many insect populations have been dropping globally in the last century and entomologists and environmentalists have recognized this trend. The connection between climate change and biodiversity is becoming more and more obvious. Deep learning is a good solution to monitor the insect biodiversity through the analysis of their sound. Unfortunately, when it comes to environmental audio data, the amount of data available has always been an issue since the collection and the labelling can be very expensive and this is a problem when using deep learning models as they need huge amount of training data. Therefore, the research question of this thesis is: "how to create a high-quality insect species recognition model despite small data using deep learning?". In order to overcome the problem of having a small dataset, different types of data augmentation and transfer learning using an AudioSet pre-trained model have been implemented which resulted in an increase of accuracy in recognizing the insect species.

1 Introduction

1.1 Context

The evidence for global insect decline is irrefutable. Many insect populations have been dropping globally in the last century and entomologists and environmentalists have recognized this trend. As mentioned in Dirzo et al. (2014), the insect populations in the last 40 years have declined by 45% and in 2020 Wagner, Grames, Forister, Berenbaum, and Stopak (2021) declared that terrestrial insects were declining at a rate close to 1% per year. The main factors causing this decline can be attributed to climate change, deforestation, pollution from light, intense use of pesticides and fragmentation of habitat that simultaneously cause a decline of the world's insect biodiversity. Even though there is an effort in protection of rare and endangered species, the real problem is the decline of abundant insects that has consequences in the ecosystem function (Wagner et al., 2021).

Insects have a pivotal role in the world ecosystem. They represent the highest number in terms of biomass, species and population and have important functions such as pollinating flowers, disposing of dead organisms and waste, pest controllers and forming crucial links in food webs. Many studies have concentrated on specific insect orders or families, despite substantial evidence from a recent global meta-analysis showing insect losses are widespread and affect a variety of taxonomic groupings. These analysis are also limited in scope as such data only focuses on a particular date and location and therefore it makes it harder to demonstrate the shifts in insect populations which is the most important aspect when monitoring the ecosystem (Montgomery et al., 2020). Usually, localization and recognition of species is carried out manually, but this kind of process can be highly complex as insects live in various and complex environments not always accessible to people. In addition, these tasks are mostly done by expert volunteers as they are time consuming and expensive. However, recent progress in signal processing using computer technology have introduced new automatic methods to identify species by capturing images and acoustic signals. Generally, identifying insect species through images has been very challenging due to their small size, high species diversity and population fluctuation. On the other hand, sound produced by insects allows the detection and classification in a non-invasive way, particularly in locations that are hardly accessible (Ganchev & Potamitis, 2007). Ganchev and Potamitis explain that insect sounds are generated as a mean of communication or generated non intentionally. In general, the emission of a

sound by an insect is usually related to two specific behavior mode. The first mode relates to sounds emitted by insects to attract a female insect located in the surrounding area or sound produced by females in order to be located by males or sounds generated to cause congregation of both males and females (cicadas). The second mode relates to sounds emitted: to warn other insects of danger, to mark a territory, to signal the presence of an insect of the same species. (Alexander, 1957) explored the sound production mechanism that insects use to produce sounds. In the mentioned paper it is explained that insects sound is produced in five different ways: the friction of two body parts (stridulation), by striking some body part such as the feet, head or abdomen against the substrate usually heard as tapping or drumming (percussion), by oscillating body parts (vibration), by contracting and releasing the tymbal muscles or by the ejection of air or fluid through a body constriction (air explosion).

A recent technology that has been successful in bioacustic audio tasks that allows to get results in a more efficient and affordable way is deep learning. Deep learning generally tries to extract the sound characteristics from a spectrogram generated from an insect raw audio and compares it to other insect species (Ganchev & Potamitis, 2007). However, the scarcity of environmental audio labeled data, has impeded the exploitation of this models as they are dependent on the availability of huge amount of training data in order to learn a non-linear function that generalizes well and allows to obtain high accuracy when performing classification on unseen data. In fact, one of the main disadvantages of deep learning is the need of huge amount of training data which still represents a challenge in the data science community as the performance of neural network often improves with the amount of data available. In simple terms, the amount of data required is proportional to the number of learnable parameters in the model and the number of parameters is proportional to the complexity of the task (Nanni, Maguolo, & Paci, 2020). This problem is generally solved using data augmentation techniques. Data augmentation is a powerful tool to artificially create new training data from existing training data which has the benefits to reduce data overfitting, create variability in data, increase model generalization and help resolve class imbalance issues in classification. Another well-known solution for limited data used in deep learning is transfer learning. Transfer learning has a simple basic premise: transfer the knowledge obtained from a model trained on a large dataset to the small dataset. In other words, the model is trained on unrelated categories in a huge dataset and the extracted knowledge is implemented on the small dataset model to extract useful features. Even though transfer learning methods could be used to great effect, the challenges involved

in making a pre-trained model to work for specific tasks are not always simple.

1.2 Research Questions

Even though data augmentation and transfer learning are the most common techniques used in machine learning when dealing with small amount of data, using these methods with deep learning to classify insect species sounds has not been investigated. This highlights the relevance and originality of this thesis. Therefore, the following problem statement is formulated:

> *How to create a high-quality insect species recognition model despite small audio data using deep learning?*

To answer this question, other 3 sub-questions have been formulated:

RQ1 *What are the data augmentation techniques that can be applied to small insects audio datasets?*

The goal is to demonstrate whether data augmentation with such a small dataset can improve the model in recognizing the different species. Different technics have been implemented directly on the raw waveforms in order to increase the size of training data available and as a result increase the generalizability of the classifier.

RQ2 *How effective is transfer learning as a solution to small insect audio datasets?*

In this sub-question an AudioSet pre-trained model has been used as a starting point to transfer knowledge to the insect species classification task.

RQ3 *How effective are raw waveforms compared to spectrogram based convolutional neural network?*

This sub question will demonstrate how effective is using raw audio data as input to convolutional neural network compared to the spectrogram

based convolutional neural network which is the most common method used to analyze audio data. To explore various solutions and make a fair comparison the raw waveform model has been implemented both with the initial amount of audio data and with the augmented data obtained from the first research question. This will also help to understand if the performance of the raw waveform model has a relevant difference in accuracy based on the amount of training data.

1.3 Findings

The results obtained from these experiments show that data augmentation can be a valid technique for small datasets when dealing with insects sound data. In particular, the accuracy of the model using augmented data showed an increase in accuracy of about 35% compared to a baseline model built using only the original data. The same conclusion has been obtained with transfer learning which led to an increase in accuracy of around 22%. Since both data augmentation and transfer learning resulted in an increase of accuracy, a combination of these two techniques has also been implemented. The combined model, which consists of the transfer learning model applied on the augmented dataset, resulted in an increase of 3% above the transfer learning model using the original dataset. Similar conclusion can be drawn when using raw audio as input to the neural network without any pre-processing work. Compared to the baseline model, the accuracy in this case improved by almost 5% when using the initial limited amount of data and by 17% when implemented on the augmented dataset. These results indicate that skipping the pre-processing stage can be a valid alternative especially when the amount of data is large enough.

2 Related Work

This section serves to give an introduction to related works in the literature. Initially, the background of environmental sound classification will be introduced and then previous related works will be described explaining the methodologies used and the results achieved.
The classification of sound is mainly applied in three different disciplines: Music Information Retrieval (MIR), Automatic Speech Recognition and

Environment Sound Classification (ESC) which is the area this thesis is based on.

2.1 Background

Recently, environmental sound classification projects have received increasing attention from the data science community and to date, many machine learning techniques have been tested. In particular, convolutional neural networks have been a popular technique adopted for this kind of tasks thanks to their ability in capturing energy pattern when using spectrograms as input to the neural network. In addition, small filters size are capable of learning different sound classes based on their spectro-temporal patterns (Salamon & Bello, 2017). Spectrograms are bidimensional graph with time on the x-axis and frequency on the y-axis representing sequences of spectra while the colors represent the strength of the frequency at a determined time frame. They have the advantage of retaining more information than hand-crafted features and have a lower dimension compared to raw waveforms (Wyse, 2017). Researchers working with environmental sound classification tasks investigated different feature extraction techniques and machine learning models. To cope with this challenge researchers consider popular ESC datasets such as: ESC-10 (Piczak, 2015b), ESC-50 (Piczak, 2015b) and Urbansound8k (Salamon, Jacoby, & Bello, 2014). Piczak (2015b) adopted a convolutional neural network obtaining exceptional results which have increased the accuracy by 7.8% compared to the baseline accuracy for ESC-10 which consists of only 400 data points with 10 classes. These results showed how convolutional neural network, which are mostly known for classifying images, can also be considered when dealing with environmental audio data. In other words, an audio can be converted into a two-dimensional spectrum and therefore be considered as an image to use as input into a convolutional neural network. However, as explained in the next section, in many deep learning projects on environmental sound classification, the amount of data available has always been a problem. Machine learning researchers and practitioners have been working on various solutions to successfully handle the problem of small datasets.

2.2 Small dataset problem

The main problem with small datasets is that models don't have the ability to adapt to previously unseen data drawn from the same distribution. In other words, the model doesn't generalize well from the test set and therefore the model will suffer from overfitting. In data science, overfitting is a concept that occurs when a model fits the data too well and learns in detail the training data to the extent that it has a low performance on unseen data. As a result, the model will have low bias and high variance (Vemuri, 2020, p. 50). To overcome overfitting there are several methods. The simplest include adding a regularization term on the weights that discourages learning a more complex or flexible model. Another method consists in adding a dropout layer that randomly drops neurons at each iteration along with all its incoming and outgoing connections obtaining a slightly different architecture (Vemuri, 2020, p. 51). Another popular technique is batch normalization, that reparametrizes the model to standardize the input in a layer (Perez & Wang, 2017). Data augmentation is a tool to artificially create new training data from existing training data which has the benefits to reduce data overfitting, create variability in data, increase model generalization and help resolve class imbalance issues in classification. (Burkov, n.d.)

2.3 Data Augmentation

Data augmentation has been proposed also for the audio domain. A key concept of audio data augmentation is that new synthetic data obtained with the application of one or more deformation to the data points of the training data don't change the semantic meaning of the labels. As a result, the neural network becomes invariant to these deformations and will perform better with unseen data. Moreover Salamon and Bello (2017) applied various data augmentation methods on the UrbanSound8K dataset and demonstrated how each type of data augmentation influences the model classification accuracy. In particular, since the considered dataset had different kinds of classes, (air conditioner, dog bark, siren, street music, etc.) each audio class accuracy behaved differently based on the augmentation technique adopted. In Wei, Zou, Liao, et al. (2020) a comparison of the main data augmentation techniques are explained. In the audio domain, four types of data augmentation are generally implemented:

- Noise injection: adding to a recording random noise obtained from another audio source containing background sounds
- Time shifting shifts left (fast forward) or right (back forward) the audio and replaces the shifted part with silence.
- Time stretching speeds up or slows down the audio while keeping the pitch unchanged. The audio sample is time stretched with values greater than 1 to speed it up or lower than 1 to slow it down.
- Pitch shifting: like stretching, this method raises or lowers the pitch of an audio sample while keeping the duration unchanged. it uses semitones as values to shift the pitch of the audio.

Many researchers tried different combination of feature extraction methods, CNN architectures and data augmentation techniques to help boost the performance of a classifier for ESC projects. As mentioned by Takahashi, Gygli, Pfister, and Van Gool (2016) when working with convolutional neural networks, a large number of training data is fundamental to train such networks. One of the data augmentation techniques suggested consists in mixing sounds together coming from the same class, as the resulting sound will still belong to the same class. One way of doing this is by randomly mixing two same class with randomly selected timings which will add more variation in the data. Pandeya and Lee (2018) demonstrated that accuracy and F1 score of a model when using audio data can be increased by simply implementing techniques such as: time stretching, pitch shifting, dynamic range compression and insertion of noise. In general, when using one to three clones per single audio results in a better performance.

Mushtaq, Su, and Tran (2021) adopts the most used ESC datasets to demonstrate how data augmentation techniques commonly used for image classification such as zoom range, width shift, brightness range, rotation angle, height shift, shear range are inefficient if applied to audio data. On the other hand applying data augmentation methods that have a physical meaning and maintain the semantic meaning of the audio can improve the performance of the model and help overcome the overfitting problem. The techniques applied by Mushtaq et al. which are similar to the ones adopted by Pandeya and Lee (2018) increased the accuracy of the classifier by circa 20% compared to the ones commonly applied in image classification.

2.4 Transfer Learning

Another elegant way of dealing with small datasets and avoid overfitting is transfer learning. Transfer learning is a machine leaning method where an already available pre-trained model is reused for another new task. It is commonly used in deep learning for computer vision and natural language processing tasks. In particular part or all of the knowledge gained from a model trained on a huge labeled dataset is transferred to another model. The benefits of using transfer learning are mainly three: lack of data, reduces training time and most of the times it increases the accuracy of the model. As a result, it is often used in environmental sound classification due to the lack of data (Burkov, n.d.).

Palanisamy, Singhania, and Yao (2020) applies transfer learning on the ESC-50, UrbanSound8k and the GTZAN datasets and demonstrates that even transfer learning with a model pre-trained on an image dataset like ImageNet can boost the model accuracy. Even if audio spectrograms and images have almost nothing in common, the assumption of transfer learning still hold firmly. In fact, using ImageNet pre-trained model allow to achieve state of the art results when fine-tuned on audio datasets. Moreover, Palanisamy et al. tested different types of audio representations such as Log-Spectrograms, LogMelspectrograms and MFCC and found out that Log MelSpectrograms were the best feature representation for that task. The results of their experiments show that using pre-trained weights increased accuracy by 20% on the ESC-50 dataset, by 10% on the UrbanSound8k dataset and 5% on the GTZAN dataset compared to the baseline model where weights where randomly initialized.

Ntalampiras (2018) proposed the use of a music genre dataset to pre-train a model in order to transfer the learned weights to a dataset consisting of the sound of ten species of birds in order to classify them with a higher accuracy. The paper compares the classification accuracy reached by the proposed transfer learning framework and the methodology without transfer learning. The results of the experiment show the superiority of the model with transfer learning in classifying bird species reaching a classification accuracy of 92.5% compared to the 81.3% accuracy reached without transfer learning. This shows how this technique can be useful also to transfer knowledge between task that are in different but related domains.

Another related work that uses transfer learning to overcome the difficulty of lack of sufficient training data is Zhang, Wang, Bao, Wang, and Xu (2019).

This paper explains the advantage of reusing and fine-tuning pretrained weights from a problem already solved instead of training the CNN from scratch which would be computationally expensive and entails a huge amount of training samples. With transfer learning in fact the low-level semantic features learned can be reused to solve another problem as they are constant for many classification tasks in computer vision. On the other hand, the high level semantic features which are generated in the top layers need to be tuned to the specific problem. Similarly to Palanisamy et al. (2020), this paper explores how CNN models pre-trained on images can be applied to audio classification tasks and also determines that the best time-frequency feature representation for audio classification are log-mel features as demonstrated by Huzaifah (2017) . The results obtained with the pre-trained model showed an increase in accuracy of around 4% compared to the model trained from scratch.

Bian et al. (2019) demonstrated that using transfer learning with a small dataset can significantly improve the accuracy of the model but the results also depends on the architecture adopted and on the number of parameters of the model. Moreover, according to (Barman et al., 2019) there are many advantages of applying transfer learning: it requires less data, less computational power, less time in prediction and it does not require heavy GPU.

2.5 Raw Waveform

Besides exploring how effective data augmentation and transfer learning are for small datasets on insects, this thesis will also explore using raw waveform based convolutional neural network. In fact when working with audio classification tasks, a valid alternative to spectrograms is to feed directly the raw audio to the convolutional neural network. Dieleman and Schrauwen (2014) compares raw waveform based CNN and spectrogram based CNN demonstrating that using raw audio performs very well but has the disadvantage of requiring more training data to allow the algorithm learn the right representations. The main difference of this approach compared to spectrogram based CNN is that the pre-processing stage of converting the audio into other forms of input can be completely skipped as it is done automatically by the classifier. While hand crafted features are designed by humans considering the auditory perception, the end to end systems implement feature extraction automatically together with the classification task which allows the extraction of new features that

humans are not able to design. As a result, this methods could improve the classification performance with new features representation information that are not captured with spectrograms (Tokozume & Harada, 2017). Recently, many researchers implemented this end to end approach and compared it to the handcrafted features such as the log-mel feature.

In contrast with spectrogram based CNNs which most of the times achieve a high performance with only 2 convolutional layers, Dai, Dai, Qu, Li, and Das (2017) proposes to use deep CNN with up to 34 layers with raw waveforms as input for speech recognition and other time-series modeling. They demonstrate that using a CNN with 18 convolutional layers could outperform a CNN with 3 convolutional layers by around 15% achieving almost the same accuracy reported by Piczak (2015a) using a CNN with spectrograms as input. This result was achieved also thanks to the combination of batch normalization, residual learning, and down-sampling. In addition, they compared the proposed deep fully convolutional networks with fully connected layers CNNs. However, fully connected layers CNNs didn't improve the accuracy and concluded that having dense layers in the network might discourage the model to learn, obtaining poor results.

Another related work is (Zhang et al., 2019) which tries to classify whales vocal calls from a large open-source dataset. This study uses the transfer learning method on both the 1D raw waveforms and 2D log-mel features achieving a higher accuracy with the latter. This shows that end to end CNN are not always better than the spectrogram based CNN but it depends on the dataset, the task and the model architecture.

Also (Tokozume & Harada, 2017) proposed an end to end system to classify environmental sounds data and compared the performance to a CNN with log-mel features. The results showed an increase of performance of 5,1% using a simplified version of the EnvNet architecture. They also explored the best architecture to use by changing the number of convolutional layers and number of filters in the model. The best performance was achieved using 64 as filter size and 3 convolutional layers.

While most related works used ESC datasets to perform audio classification and in some cases data augmentation was implemented to increase the training set size, this thesis will focus on specific type of sound with only a few recordings available for each class. Data augmentation and transfer learning will be explored to understand how these methods can improve the performance with insect sounds and with such a little amount of data. In addition, compared to most previous related work that used an image dataset like ImageNet to perform the transfer learning task, this thesis will

show how efficient is using the YAMNet pre-trained deep neural network which is trained on the AudioSet dataset. Moreover, a comparison with an end-to-end system will demonstrate if raw waveforms as input are a valid option in this case. As mentioned, implementing CNN with raw waveforms is still a challenging problem as it also requires substantial amount of training data to let the network discover the right features. In many previous works this approach has achieved an accuracy close to the traditional CNN with spectrogram. However, most state-of-the-art results have been obtained using hand crafted features.

3 Method

This section will provide a description of the general approach implemented through the description of the mathematical models and algorithm implemented.

3.1 Data Transformation

When working with audio data, there are many popular ways to convert an audio waveform into a feature representation. The most popular way is to convert the audio to a 2D time-frequency representation. In this thesis log mel spectrogram representation will be used as input to the CNN classifiers. Firstly, the audio signal is mapped from a time domain to a frequency domain through the fast Fourier transformation function. Then, the frequency is converted to a log scale to generate a spectrogram. A spectrogram is a way to visualize frequencies spectrum of a signal. A mel scale is simply a non linear transformation of the frequency scale. Since humans don't perceive frequencies on a linear scale the Mel scale mimics the human ear. As a result, a mel spectrogram is a spectrogram converted into a mel scale. This approach is part of the pre-processing stage which allows the data to be transformed into a format similar to images which is accepted by a convolutional neural network as input (Figure 1).

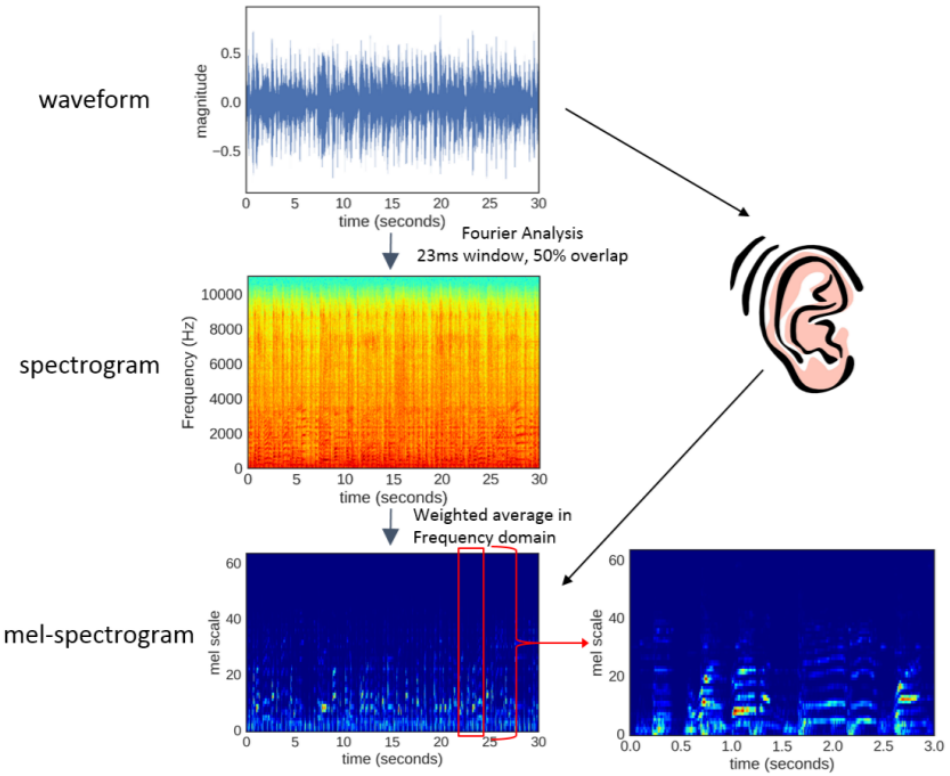

Figure 1: Convert waveform into log mel-spectrogram. Mel-spectrogram replicates human ear, with high precision in low frequency band and low precision in high frequency band. Source: https://arxiv.org/pdf/1802.09697.pdf

Once the data is pre-processed it is ready to be fed into the network. This study involved 6 models of convolutional neural networks:

1. Baseline model
2. Data augmentation model
3. Transfer Learning model
4. Transfer Learning and data augmentation combined model
5. Raw waveform model
6. Raw waveform and data augmentation combined model

Each of these models have a different CNN architecture based on the amount of data and if the features are hand crafted or extracted automatically. Since tuning manually the parameters would take a considerable

amount of time and resources, most of the parameters have been tuned using GridsearchCV, which is a function that allows to automatically loop through a pre-defined list of values of different parameters and fit the model in order to obtain the optimal combination of parameters that generate the best model. Moreover, this function allows to determine the number of splits for the cross-validation for each set of hyper-parameters to obtain more robust results. In particular, the hyper-parameters tuned are: the number of neurons, the activation function, the optimizer, the learning rate, the regularization term, the dropout value and the number of epochs.

Firstly, a baseline model is built to make comparisons with the data augmentation and transfer learning models. Both the baseline model and the data augmented model are built based on the LeNet-5 architecture. As shown in Figure 2, LeNet-5 consists of 2 convolutional layers, each followed by an averaging pool layer and 2 fully connected layers before using the Softmax function for classification. Due to the limited amount of data available the architecture has been modified to fit the data in order to avoid overfitting. In particular, drop out layers and regularization have been added and the number of filters has been reduced in order to decrease the complexity of the model and obtain a model with fewer parameters. These additional steps increase the generalization ability of the model.

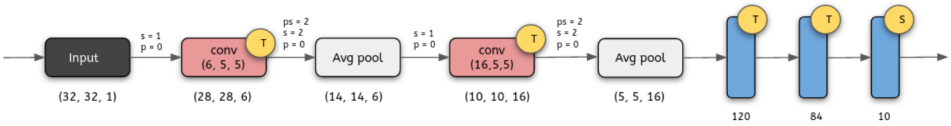

Figure 2: LeNet-5 architecture. Source: LeNet-5: Summary and Implementation.

Another change made to the architecture is to replace the Tanh activation function with ReLu in the hidden layers. ReLu is the most popular activation function used for deep learning used in most of the related works. The mathematical expression of how ReLu activation function works is showed below:

$$Relu(x) = max(0, x) \tag{1}$$

It is linear for all positive values and zero for all negative values. With this approach it eliminates the vanishing gradient problem observed in the earlier types of activation function. It has the advantage of being computationally cheap which results in less training time and fast convergence. It also offers a better performance and generalization in deep learning compared to the Sigmoid and Tanh activation functions. The main disad-

vantage of this function is that it can cause some gradients to die which doesn't allow those weights to update during backpropagation (Nwankpa, Ijomah, Gachagan, & Marshall, 2018).

In the last layer the model uses the Softmax function. It is used for multiclass classification tasks, and it returns the probability of each class where the highest probability corresponds to the target class. The main characteristic of this function is that it produces probabilities that are in a range between 0 and 1 and the sum of these probabilities is equal to 1 (Nwankpa et al., 2018). The mathematical definition is shown below:

$$\sigma(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{2}$$

where sigma is the Softmax, z is the input vector, the numerator is the standard exponential function for input vector and the denominator is the sum of standard exponential function for output vector for K classes.

As for the transfer learning task, the YAMNet pre trained model has been used. YAMNet employs the MobileNet architecture to classify 521 classes from the AudioSet corpus. The main purpose of YAMNet is to extract high level features and use this embedding to feed the actual model consisting of few dense layers. This network consists of 28 layers with learnable weights: 27 convolutional layers and 1 fully connected layer.

On the other hand, the raw waveform model is based on the EnvNet architecture also implemented by Tokozume and Harada (2017). This architecture consists of 2 convolutional layers followed by 3 consecutive maxpool layers, 2 dense layers and a final Softmax output layer (Figure **??**).

3.2 Evaluation

To evaluate the performance of the all the models Adam optimizer and a categorical cross entropy as loss function have been chosen. Adam is an optimization algorithm that combines RMSprop which uses squared gradients to scale the learning rate and Stochastic Gradient Descent with momentum which moves the average of the gradient instead of gradient itself. A key factor of Adam is that it uses an adaptive learning rate method that computes individual learning rate for different parameters (Kingma &

Ba, 2014). The loss function used for multi classification tasks is Categorical cross entropy loss function. It is computed using the following function:

$$Loss = -\sum_{i=1}^{K} y_i * log\hat{y}_i \tag{3}$$

where $\hat{}y$ is the i^{th} scalar value in the model output, y_i is the corresponding target value, and output size is the number of scalar values in the model output. In general, cross entropy builds upon entropy form the information theory field and measure the difference between two probability distributions for a given variable. With this loss function each predicted class is compared to the actual class and based on how far the two values are, it calculates a score that penalizes the probability (Ho & Wookey, 2019).

4 Experimental Setup

4.1 Dataset description

The dataset used for this thesis is the European orthoptera dataset recorded by Baudewijn Odé. The European orthoptera is a dataset of sound recordings of around 360 species, of which only 9 species consisting of 158 recordings are made publicly available. In Table 1 the name of these 9 species, the number of recordings per species and the average duration of the recordings per species are shown. The recordings are in their original resolution and unfiltered state. In most cases hardly any other sound can be heard in the background and in some species recordings, variation is present being the result of different temperatures or social conditions (resulting in aberrant rivalry or courtship songs). This limited dataset is suitable for this thesis to understand to what extend data augmentation and transfer learning can improve the model accuracy in identifying the different species of orthoptera.

4.2 Pre-processing

Before starting to build the CNN models, few pre-processing steps has been made on the orthoptera dataset using the Librosa library. Firstly, the

Table 1: Dataset description

id	Species	Recordings	Average Duration (seconds)
0	Chorthippus biguttulus	20	10
1	Chorthippus brunneus	16	11
2	Gryllus campestris	22	9
3	Nemobius sylvestris	22	30
4	Oecanthus pellucens	15	15
5	Pholidoptera griseoaptera	15	7
6	Pseudochorthippus parallelus	17	6
7	Roeseliana roeselii	13	5
8	Tettigonia viridissima	18	5
Total:		158	

sample rate of the recordings needs to be the same for all the recordings. Some recordings are in old-school CD-quality with a frequency of 44.1kHz, but others are in full resolution with frequencies up to about 100kHz. Since the majority of the recordings (70%) have a frequency of 44.1kHz and the rest of the recordings have higher or lower frequencies, all the audios have been standardized to a 44.1kHz sampling rate through Librosa library when loading the recordings. Moreover, all sound files are converted to monaural 16-bit WAV files.
Secondly, all recordings have different duration which means that some recordings had to be cut and others had to be extended in order to have all audios of the length of 5 seconds.
Thirdly, all recordings have been transformed to log mel spectrograms which produced images of height 128 and width 431. Finally, a metadata excel spreadsheet containing all the names of the audios and the insect species has been created in order to facilitate the manipulation of the recordings, for the evaluation of the models and to keep track of the augmentation process. In addition, to reduce the computation complexity the target feature represented by "species" have been encoded obtaining a 0 to 8 range representing the 9 species of orthoptera.

Another crucial step before starting to train the model is to shuffle the data and split the data into sets. To achieve robust results k-fold cross validation has been implemented with 5 folds. Once all these steps are completed, Keras library has been used to build the baseline model.

4.3 Baseline model

As mentioned, the baseline model has been built taking the LeNet-5 architecture as a reference. Compared to LeNet-5 few changes have been done to reduce the total number of trainable parameters to reduce the model complexity and deal with the small amount of data available. The accuracy obtained with this model is 75% in the training set and only 55% in the test set. The summary of the baseline model architecture is shown in Table 2.

Table 2: Baseline Model Architecture

Layer Type	Operation	Filters	Filter Size	Output Shape	Parameters
Convolution Layer 1	ReLu, L2(0.001)	2	(3,3)	(126, 429, 2)	20
Pooling Layer	MaxPooling	1	(2,2)	(63, 214, 2)	0
Dropout Layer	Dropout	1	0.5	(63, 214, 2)	0
Convolution Layer 2	ReLu, L2(0.001)	4	(3,3)	(61, 212, 4)	76
Pooling Layer	MaxPooling	1	(2,2)	(30, 106, 4)	0
Dropout Layer	Dropout	1	0.5	(30, 106, 4)	0
Convolution Layer 3	ReLu, L2(0.001)	8	(3,3)	(28, 104, 8)	296
Pooling Layer	MaxPooling	1	(2,2)	(14, 52, 8)	0
Dropout Layer	Dropout	1	0.5	(14, 52, 8)	0
Flatten Layer	Flatten	-	-	(5,824)	0
Dense Layer	L2(0.001)	16	-	(16)	93,200
Dropout Layer	Dropout	1	0.5	(16)	0
Output Layer	Softmax	9	-	(9)	153
Total parameters: 94,017					

4.4 Augmented model

Next step is to perform data augmentation on the raw waveforms. To increase the size of the dataset, time stretch with rates of 0.8 and 0.9, pitch shift with number of steps of -1, 1 and 2 and time shift with a rate of 0.2 have been implemented on the original dataset. In addition, to make this experiment more realistic, environmental background noise has been added using the Audiomentations library. Audios tagged with "insect" from the ESC-50 dataset has been used to add background noise

with a probability of 100% to all the original dataset. Once a dataset with background noise is obtained, time stretch with a rate of 0.8 is implemented again to increase the size of the data with background noise.

As a result, the dataset increased to 1,422 recordings which corresponds to 8 times the size of the original dataset with 158 recordings. Also for the data augmentation model, cross validation with 5 folds has been implemented to achieve more reliable results.

The architecture used for the augmented data is similar to the baseline model but in this case another convolutional layer has been added to increase the complexity of the model and extract more features (Table 3). The hyper-parameters tuned with GridsearchCV to find the optimal parameters are the L2 regularization term (0.001, 0.01 and 0.1) and the optimizer (Adam and RMsprop). Moreover, the number of units increases in the convolutional layers to capture larger combinations of patterns. The accuracy obtained with the augmented dataset is 95% in the train set and 90% in the test set.

Table 3: Data Augmentation Model Architecture

Layer Type	**Operation**	**Filters**	**Filter Size**	**Output Shape**	**Parameters**
Convolution Layer 1	ReLu, L2(0.001)	8	(3,3)	(126, 429, 8)	80
Pooling Layer	MaxPooling	1	(2,2)	(63, 214, 8)	0
Convolution Layer 2	ReLu, L2(0.001)	16	(3,3)	(61, 212, 16)	1,168
Pooling Layer	MaxPooling	1	(2,2)	(30, 106, 16)	0
Convolution Layer 3	ReLu, L2(0.001)	32	(3,3)	(28, 104, 32)	4,640
Pooling Layer	MaxPooling	1	(2,2)	(14, 52, 32)	0
Convolution Layer 4	ReLu, L2(0.001)	64	(3,3)	(12, 50, 64)	18,496
Pooling Layer	MaxPooling	1	(2,2)	(6, 25, 64)	0
Flatten Layer	Flatten	-	-	(9,600)	0
Dense Layer	L2(0.001)	64	-	(64)	614,464
Output Layer	Softmax	9	-	(9)	585
Total parameters: 639,433					

4.5 Transfer learning model

The third model that answers the third research question consists in implementing transfer learning using the AudioSet dataset for the pre-training part. AudioSet is an audio event dataset that consists of over 2 million human annotated Youtube videos of 10 seconds length. These videos are annotated based on a hierarchical ontology of 632 classes. This dataset is only used to pre-train the YAMNet network in order to transfer the knowledge and classify the European orthoptera dataset achieving good results even without requiring a lot of labeled data. In particular, the first layers of the network learn the high-level features and output the embeddings which are used for transfer learning. The YAMNet input features are then fed into a shallower model, which is used to classify insects, consisting of one hidden dense layer with 512 neurons and a ReLu activation function and a dropout layer with 0.5 probability. Similarly to the previous models the last layer consists of a Softmax layer with 9 output units totalling 529,417 trainable parameters. With transfer learning the training accuracy increased to 77% and the test set accuracy increased to 75%.

Since both the data augmentation model and the transfer learning model resulted in a higher accuracy compared to the baseline model, a combination of these two models has also been implemented. In other words, the augmented dataset obtained with the second model consisting of 1,422 recordings has been implemented with the transfer learning model. With this approach, the test accuracy increased to 80% which corresponds to an increase of 3% compared to the transfer learning model implemented on the limited amount of data. This shows the importance of having a large enough amount of training data in order to get a model that generalizes better and gives better results and how data augmentation is a valid solution when the amount of data is limited.

4.6 Raw waveform model

Finally, a comparison between the spectrogram CNN model and the raw waveform CNN model has been performed. The main difference is that with raw waveforms the feature extraction step in the pre-processing stage can be skipped because it is completed automatically by the CNN. To achieve a good performance, EnvNet architecture used also in (Tokozume

& Harada, 2017) has been adopted with few changes. The architecture used for this model consists of two 1D convolutional layers with 8 and 16 neurons and a kernel size of 5. As for the previous models, ReLu activation function has been used. The convolution layers are followed by 3 maxpool layers of size 2 and a fully connected layer with 16 neurons (Table 4).
In this case, a Lambda layer has also been added after the dense layer to decrease the number of parameters, reduce the complexity of the model and decrease the computational cost by calculating the mean values of the tensors. In a neural network, a Lambda layer is a layer with its own function used to transform the data in between the modelling before applying that data as input to any of the existing layers.
Also in this case GridsearchCV has been applied to select the best parameters. The hyper-parameter tuned included the optimizer (adam and RMspop) and the number of units in every convolutional layer (8, 16, 32). As mentioned before, also for this model categorical cross entropy has been used as the loss function and accuracy is used to evaluate the performance of the classifier. The last layer is the Softmax output layer which has as many neurons as the number of classes.

The last model implemented is a combination of the raw waveform model and the data augmentation model to understand what are the outcomes when using a larger training set with an end to end approach. Due to the higher amount of training data the number of epochs has been increase to 400 to let the model converge. Compared to the raw waveform model this approach surprisingly increased the test accuracy by 22%.

Table 4: Raw Waveform Models Architecture

Layer Type	**Operation**	**Filters**	**Filter Size**	**Output Shape**	**Parameters**
Convolution Layer 1	ReLu	8	5	(220496, 8)	48
Convolution Layer 2	ReLu	8	5	(220496, 16)	656
Pooling Layer 1	MaxPooling	1	2	(110244, 8)	0
Pooling Layer 2	MaxPooling	1	2	(55122, 8)	0
Pooling Layer 3	MaxPooling	1	2	(27561, 8)	0
Dense Layer	-	16	-	(27561, 8)	272
Lambda Layer	Mean	-	-	(8)	0
Output Layer	Softmax	9	-	(9)	153
Total parameters: 1,129					

4.7 Software and Hardware

All the experiments are done using the python programming language with version 3.7 on Windows 10 operating system. The main libraries used for the implementation of this thesis are Anaconda, Keras and Librosa which is the most crucial library for this project. Anaconda is an open-source library used for python which already includes various packages like NumPy, pandas and matplotlib. It also offers the option to create different environment for specific tasks using specific packages. Keras is the library used to build all the CNN models in this thesis by adding layers on top of each other. Librosa is a python package used for music and audio analysis. It is mainly used for feature extraction, data augmentation, plotting and manipulating recordings. In terms of hardware, the thesis has been conducted with a processor core i5-1035G1 CPU @ 1.00GHz, 1.19 GHz with a RAM of 8 GB and a 512 GB SSD.

5 Results

In this thesis, 6 models have been experimented to classify insect species sounds with limited amount of data.
The first model is used as baseline and implements a CNN with an architecture similar to LeNet-5 with the data available in the original dataset consisting of 158 raw audios on 9 species of insects.
The second one implements data augmentation with various techniques such as time shift, pitch shift time stretch and background noise to increase the training data to 1,422 data points and uses an architecture similar to the baseline model to understand the effect of data augmentation.
The third model adopts transfer learning with the YAMNet model and an additional dense layer to train the limited insects data consisting of 158 recordings.
Since both the data augmentation model and the transfer learning model achieved good results, a combination of these two models has been implemented. This way transfer learning has been applied to the increased dataset consisting of 1,422 recordings.
The last two model are based on an end to end approach where a raw waveform model is implemented both using the original size of the orthoptera dataset and the augmented dataset on the EnvNet architecture.
The results of these model are shown in Table 5.

Table 5: Models comparison based on test accuracy

Model	Data Size	Epochs	Train Accuracy	Test Accuracy
Baseline	158	30	0.75	0.55
Data Augmentation	1,422	10	0.95	0.90
Transfer Learning	158	30	0.77	0.75
Transfer Learning and Data Aug.	1,422	30	0.80	0.80
Raw Waveforms	158	400	0.66	0.50
Raw Waveforms and Data Aug.	1,422	100	0.75	0.72

As expected, the baseline model performed poorly due to the small amount of data. In fact, based on the results of the 5-fold cross validation, it appears that in some cases the class prediction are done quite randomly due to the lack of data to be trained. The accuracy achieved on the training data was high obtaining a 75% accuracy but the performance on the test set was poor, meaning that the model was not able to generalize well (Appendix A).
In terms of class classification, as shown in the confusion matrix in Appendix A, in the baseline model only two species could be identified with a high accuracy on the test set (class 2 with 91% and class 8 with 83%) while all the other species have been misclassified.

On the other hand, the effect of data augmentation is surprising with a 90% accuracy achieved on the test set. Increasing the dataset size by 8 times was enough to obtain a high-performance classifier. The confusion matrix (Appendix B) shows how each species could be identified accurately compared to the baseline model. Also in this case the training accuracy was very high reaching almost 95% accuracy but differently from the baseline model, this model has a high capacity to generalize which led to a high accuracy.

Also transfer learning can be considered a good option for insect sounds classification with an increase of accuracy of 25% compared to the baseline model. From the accuracy chart (Appendix C), it is possible to see how the model tends to overfit due to the limited size of the training set. In fact, only 1 class could achieve a relatively high accuracy of 82% (Appendix C). The application of this model with the increased dataset obtained with data augmentation showed an increase in accuracy of about 3%. This explains the importance of having a large training set and how efficient is data augmentation when dealing with a small insects audio dataset (Appendix D).

As for the raw waveform model, as already mentioned, the data needed to achieve good results is usually much higher than the spectrogram based models. This model achieved a training accuracy of 66% and test accuracy of only 50%, which makes it the worst model of all in terms of performance. Moreover, compared to the other models the training time needed to converge for an end to end approach is much higher. This is why the number of epochs for both the raw waveform models are higher than the other models (Appendix E).
However, when the raw waveform model is combined with the augmented dataset, the accuracy increases to 75% on training set and 72% on test set. Compared to the other models, some classes where classified with very high accuracy (100%) while other classes are misclassified or have low accuracy (33% for class 2). (Appendix F)

In conclusion, it is quite clear that for insect species sound classification data augmentation is a very valuable option when the data available is small.

6 Discussion

Many insect populations have been dropping globally in the last century and Deep learning is a good solution to monitor the insect biodiversity through the analysis of their sound. Unfortunately, the amount of environmental data available has always been an issue since the collection and the labelling can be very expensive. The purpose of this thesis is to understand how to use deep learning to classify insect species when the amount of data available is limited. Various techniques have been explored such as data augmentation and transfer learning to overcome this problem. Moreover, a comparison of performance between spectrogram based CNN and raw audio CNN have been experimented to understand the effectiveness of skipping the pre-processing stage, which is automatically done by the algorithm. All the techniques adopted showed how the small dataset problem can be solved in an efficient way obtaining a high-quality classifier compared to the baseline model.

The first model implemented is with the data augmentation technique which is the model with the highest accuracy compared to all the other approaches. This method increased by 8 times the original dataset size and increased the test set accuracy to 90%. Compared to the related works where data augmentation increased the classifier performance by 5% to 10%, in this thesis the accuracy increased by around 35%. The reason

behind this huge improvement can be partially attributed to the fact that the sounds emitted by the orthoptera insects are very different between species and data augmentation highlighted this difference by increasing the number of recordings for each species. In fact, the number of species adopted in this thesis could be too small to really understand how efficient data augmentation is. However, the results achieved with this method are still a representation of reality and based on the performance of this model, data augmentation can be considered a valid solution when the data available is limited. An advantage of data augmentation for the classification of insect species is that it makes the model more applicable in practical uses. For instance, adding background noise related to the environment to the audio dataset creates new recordings that are closer to the real word recording of an insect. As a result, this makes the model more reliable when new recordings of insect species with background noise need to be classified.

Also transfer learning can be consider a good solution when dealing with small insects audio dataset. The results suggest how the relevant features to classify insect sounds can be also extracted from another dataset like AudioSet. Moreover, with only 158 data points the accuracy could reach a test accuracy of 75% which is surprising if compared to the baseline model. On the other hand, applying transfer learning with the augmented dataset led only to a small increase of performance of about 3%. This means that applying transfer learning on an increased dataset is not as effective as a model trained from scratch. This could be because the weights of the network are already trained on the AudioSet data and re-used as a starting point in the training stage and adopted to the insects dataset. In addition, from the confusion matrix it is possible to notice that only few classes (class 1 and 2) have been poorly classified while all the other classes have an accuracy of at least 73% and a maximum accuracy of 96% (Appendix D). The results obtained with this combined model are very similar to those obtained by (Palanisamy et al., 2020). The accuracy with transfer learning increased by 25% above the baseline model. In any case these results suggest that transfer learning is a valid solution to small datasets.

Similarly to transfer learning, two raw waveform model have been experimented: a model implemented on the initial 158 data points and a model implemented on the augmented dataset.
The first raw waveform model achieved the lowest performance compared to all the other models. As reported by Dieleman and Schrauwen (2014), the end to end approach perform very well for audio classification tasks but the it has the disadvantage of requiring more training data to allow the algorithm learn the right representations. This helps to understand

why this model has a lower accuracy than the baseline model which is a spectrogram based approach.
The effect of increasing the dataset size are explained on the second raw waveform model which increased the accuracy by 17%. For this classification task the raw waveform model combined with data augmentation achieved an accuracy similar to the transfer learning model applied on the augmented dataset but Tokozume and Harada (2017) showed how the performance of an end to end approach can vary based on the dataset size and type of dataset. This could also mean that since a raw waveform model needs more training data than a spectrogram based model, with further data augmentation it might also achieve high-quality performance but unfortunately, this approach is computationally expensive since the feature extraction is performed automatically together with the classification task.

7 Conclusion

Since the amount of environmental audio data has always been an issue because the collection can be very expensive and time consuming, this thesis suggests some effective methods that can be applied to overcome this problem. In particular, it explores how to create a high-quality insect species recognition model despite small audio data using deep learning. Three sub-questions were formulated to help answer this main research question:

RQ1 *What are the data augmentation techniques that can be applied to small insects audio datasets?*

Different types of data augmentation techniques on the raw audio have been proposed and compared to the baseline model built on the original small dataset. Data augmentation techniques such as time shift, pitch shift, time stretch and environmental noise injection have been implemented. Then, a convolutional neural network that takes as input spectrograms was trained, demonstrating that data augmentation can considerably improve the accuracy performance. Moreover this approach also helps to make the model more realistic and applicable in real world scenarios thanks to the injection of background environmental noise. As a results this thesis suggest data augmentation as a solution to small insects audio dataset.

RQ2 *How effective is transfer learning as a solution to small insect audio datasets?*

Transfer learning using AudioSet dataset was experimented with the YAMNet architecture to transfer knowledge to the insects dataset. This method has been experimented both with the original limited dataset and in combination with the augmented dataset. Both models led to an increase in performance compared to the baseline model but the improvement is not as significant as the data augmentation approach. Therefore, when classifying insect species using audio data, building a classifier from scratch can give better results than using a pre-trained model to transfer knowledge. However, based on the increase of performance of the transfer learning model it is clear that also this approach is a valid solution.

RQ3 *How effective are raw waveforms compared to spectrogram based convolutional neural network?*

A raw waveform CNN, which takes as input directly the raw audio, was trained. This recent method compared to spectrogram based CNN has the advantage of skipping the pre-processing stage but on the other hand needs more training data. Also this approach has been experimented both with the limited dataset and the augmented dataset on the EnvNet architecture. The results showed that with the small amount of data used for the first model the performance achieved is even lower than the baseline model. On the other hand, when the same approach is applied to the augmented dataset the performance increases considerably. The advantage of this approach is that the feature extraction is done automatically by the algorithm but at the same time it can be computationally and time expensive when the size of the dataset is increased. In addition to a high-quality insect species recognition model, the amount of data needed would be higher than a spectrogram based model.

This work can be further developed to understand which type of data augmentation are more beneficial when dealing with insects data and which dataset can be used with transfer learning to transfer more knowledge and increase the performance. Also, with enough computational capacity this work can be extended by investigating the combination of data augmentation, transfer learning and raw waveform.

8 Acknowledgements

The author thanks Dr. Dan Stowell for supporting this work and Baudewijn Odé for providing the European Orthoptera dataset.

9 Code

The jupyter nothebook with python code for the methods presented in this thesis is freely available for comparison at https://github.com/darione10/insects

References

Alexander, R. D. (1957). Sound production and associated behavior in insects.

Bian, W., Wang, J., Zhuang, B., Yang, J., Wang, S., & Xiao, J. (2019). Audio-based music classification with densenet and data augmentation. In *Pacific rim international conference on artificial intelligence* (pp. 56–65).

Burkov, A. (n.d.). The hundred-page machine learning book (2019). *ISBN-13*, 978–1999579500.

Dai, W., Dai, C., Qu, S., Li, J., & Das, S. (2017). Very deep convolutional neural networks for raw waveforms. In *2017 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 421–425).

Dieleman, S., & Schrauwen, B. (2014). End-to-end learning for music audio. In *2014 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 6964–6968).

Dirzo, R., Young, H. S., Galetti, M., Ceballos, G., Isaac, N. J., & Collen, B. (2014). Defaunation in the anthropocene. *science*, *345*(6195), 401–406.

Ganchev, T., & Potamitis, I. (2007). Automatic acoustic identification of singing insects. *Bioacoustics*, *16*(3), 281–328.

Ho, Y., & Wookey, S. (2019). The real-world-weight cross-entropy loss function: Modeling the costs of mislabeling. *IEEE Access*, *8*, 4806–4813.

Huzaifah, M. (2017). Comparison of time-frequency representations for environmental sound classification using convolutional neural networks. *arXiv preprint arXiv:1706.07156*.

Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Montgomery, G. A., Dunn, R. R., Fox, R., Jongejans, E., Leather, S. R., Saunders, M. E., ... Wagner, D. L. (2020). Is the insect apocalypse upon us? how to find out. *Biological Conservation*, *241*, 108327.

Mushtaq, Z., Su, S.-F., & Tran, Q.-V. (2021). Spectral images based environmental sound classification using cnn with meaningful data augmentation. *Applied Acoustics*, *172*, 107581.

Nanni, L., Maguolo, G., & Paci, M. (2020). Data augmentation approaches for improving animal audio classification. *Ecological Informatics*, *57*, 101084.

Ntalampiras, S. (2018). Bird species identification via transfer learning from music genres. *Ecological informatics*, *44*, 76–81.

Nwankpa, C., Ijomah, W., Gachagan, A., & Marshall, S. (2018). Activation functions: Comparison of trends in practice and research for deep learning. *arXiv preprint arXiv:1811.03378*.

Palanisamy, K., Singhania, D., & Yao, A. (2020). Rethinking cnn models for audio classification. arxiv 2020. *arXiv preprint arXiv:2007.11154*.

Pandeya, Y. R., & Lee, J. (2018). Domestic cat sound classification using transfer learning. *International Journal of Fuzzy Logic and Intelligent Systems*, *18*(2), 154–160.

Perez, L., & Wang, J. (2017). The effectiveness of data augmentation in image classification using deep learning. *arXiv preprint arXiv:1712.04621*.

Piczak, K. J. (2015a). Environmental sound classification with convolutional neural networks. In *2015 ieee 25th international workshop on machine learning for signal processing (mlsp)* (pp. 1–6).

Piczak, K. J. (2015b). Esc: Dataset for environmental sound classification. In *Proceedings of the 23rd acm international conference on multimedia* (pp. 1015–1018).

Salamon, J., & Bello, J. P. (2017). Deep convolutional neural networks and data augmentation for environmental sound classification. *IEEE Signal processing letters*, *24*(3), 279–283.

Salamon, J., Jacoby, C., & Bello, J. P. (2014). A dataset and taxonomy for urban sound research. In *Proceedings of the 22nd acm international conference on multimedia* (pp. 1041–1044).

Takahashi, N., Gygli, M., Pfister, B., & Van Gool, L. (2016). Deep convolutional neural networks and data augmentation for acoustic event detection. *arXiv preprint arXiv:1604.07160*.

Tokozume, Y., & Harada, T. (2017). Learning environmental sounds with end-to-end convolutional neural network. In *2017 ieee international conference on acoustics, speech and signal processing (icassp)* (pp. 2721–2725).

Vemuri, V. K. (2020). *The hundred-page machine learning book: by andriy burkov, quebec city, canada, 2019, 160 pp.; isbn 978-1999579517.* Taylor & Francis.

Wagner, D. L., Grames, E. M., Forister, M. L., Berenbaum, M. R., & Stopak, D. (2021). Insect decline in the anthropocene: Death by a thousand cuts. *Proceedings of the National Academy of Sciences*, *118*(2).

Wei, S., Zou, S., Liao, F., et al. (2020). A comparison on data augmentation methods based on deep learning for audio classification. In *Journal of physics: Conference series* (Vol. 1453, p. 012085).

Wyse, L. (2017). Audio spectrogram representations for processing with convolutional neural networks. *arXiv preprint arXiv:1706.09559*.

Zhang, L., Wang, D., Bao, C., Wang, Y., & Xu, K. (2019). Large-scale whale-call classification by transfer learning on multi-scale waveforms and time-frequency features. *Applied Sciences*, *9*(5), 1020.

A Appendix: Baseline model

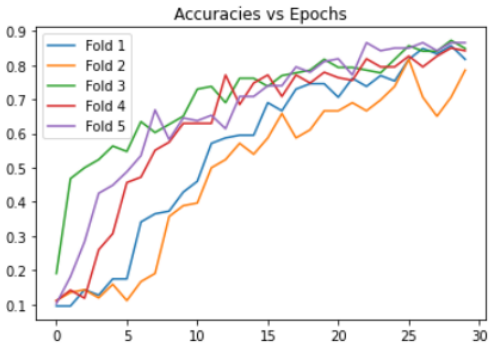

Figure 3: Baseline 5-folds accuracies

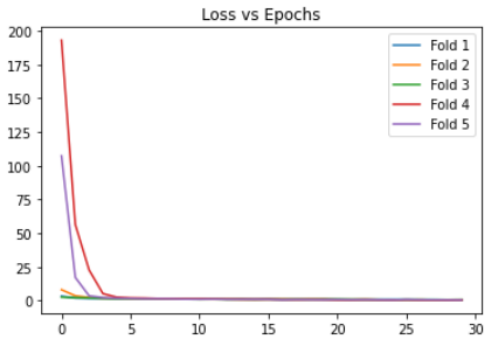

Figure 4: Baseline 5-folds Loss

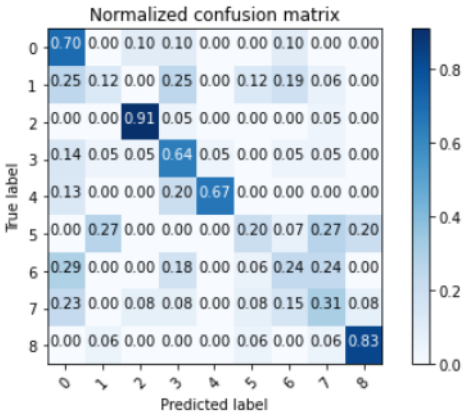

Figure 5: Baseline confusion matrix

B Appendix: Data Augmentation model

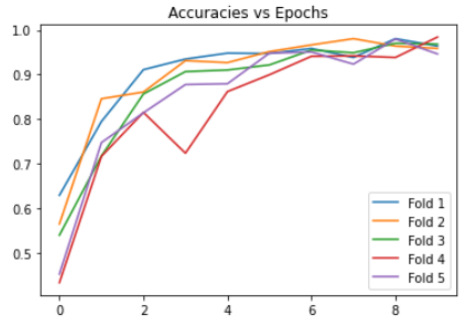

Figure 6: Data Augmentation 5-folds accuracies

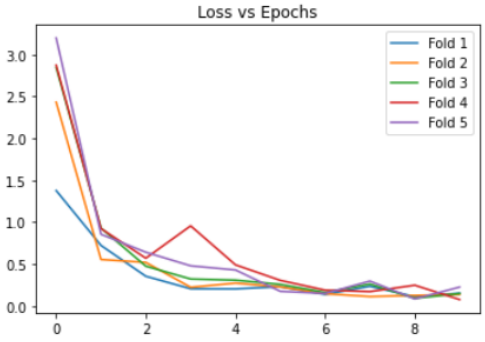

Figure 7: Data Augmentation 5-folds Loss

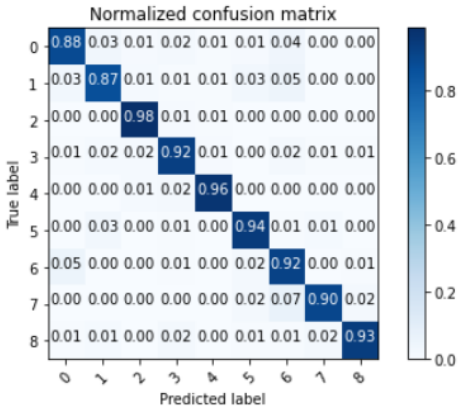

Figure 8: Data Augmentation confusion matrix

C Appendix: Transfer Learning Model

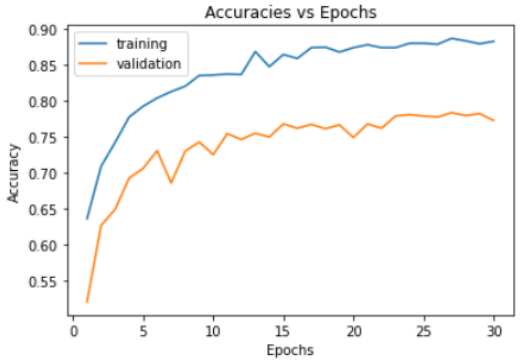

Figure 9: Transfer Learning accuracy

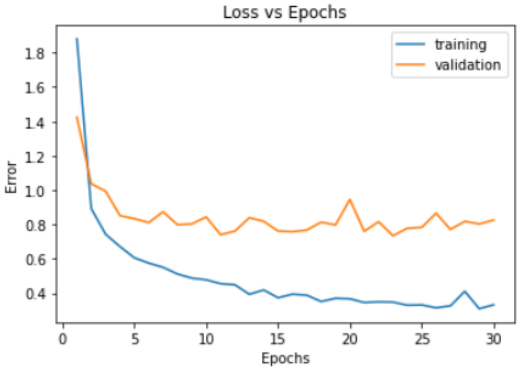

Figure 10: Transfer Learning Loss

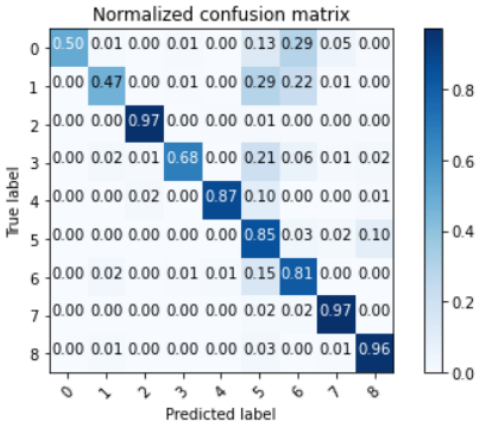

Figure 11: Transfer Learning confusion matrix

D Appendix: Transfer Learning and Data Augmentation

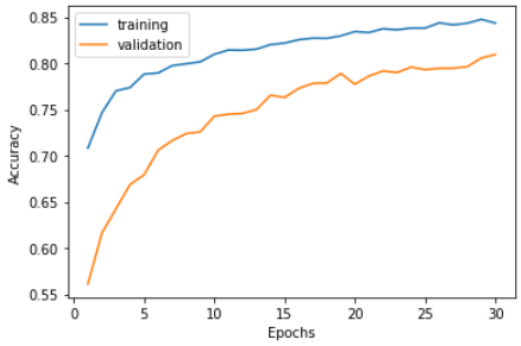

Figure 12: Transfer Learning and Data Augmentation accuracy

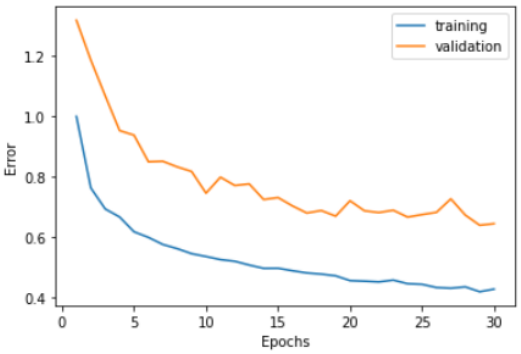

Figure 13: Transfer Learning and Data Augmentation Loss

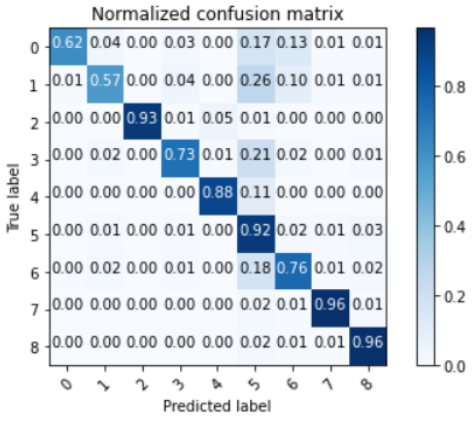

Figure 14: Transfer Learning and Data Augmentation confusion matrix

E Appendix: Raw Waveform Model

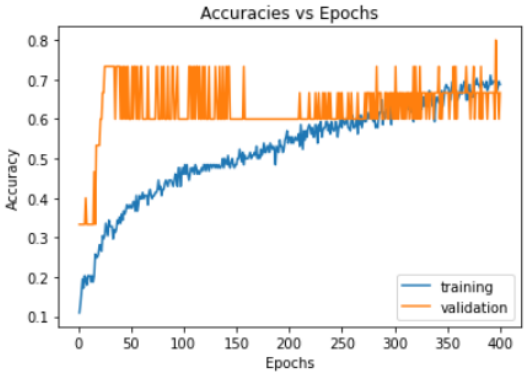

Figure 15: Raw Waveform accuracy

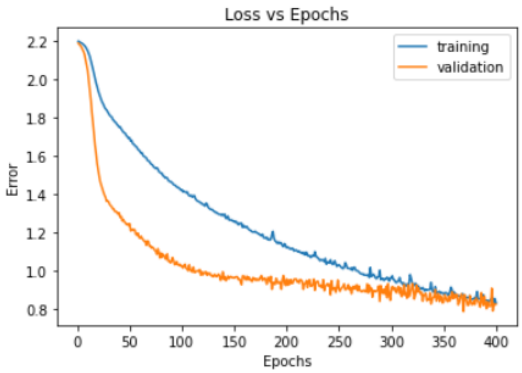

Figure 16: Raw Waveform Loss

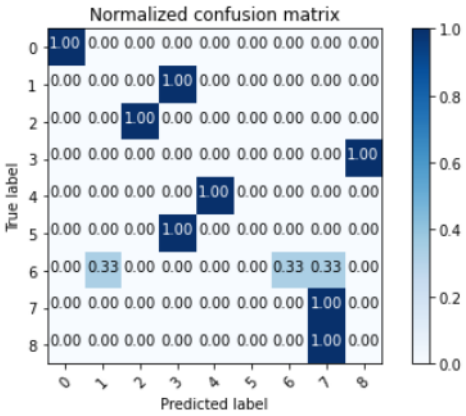

Figure 17: Raw Waveform confusion matrix

F Appendix: Raw Waveform and Data Augmentation Model

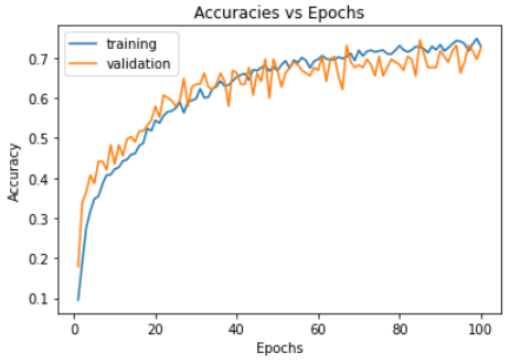

Figure 18: Raw Waveform and Data Augmentation accuracy

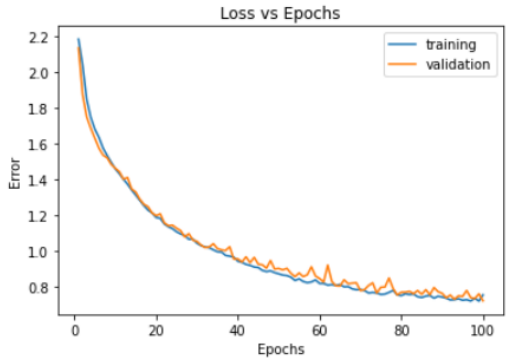

Figure 19: Raw Waveform and Data Augmentation Loss

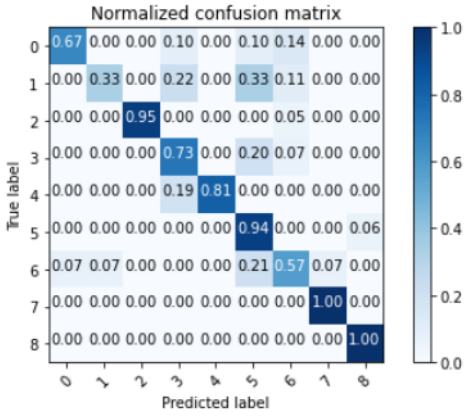

Figure 20: Raw Waveform and Data Augmentation confusion matrix