

Machine Learning Identification of the Haast Tokoeka Kiwi Bird from Acoustic Readings

Tom Grubb – **09029648**

Supervised by
Dr Jason Anquandah

Table of Contents

Abstract, Acknowledgements & List of Commonly Used Acronyms	3
1. Introduction.....	4
2. Related work.....	5
3. Project Methodology and Objectives	7
3.1 Objectives	7
3.2 Ethical statement	8
3.3 Assessment of User Requirements	8
3.4 Dataset Creation and Augmentation	9
3.5 Audio Transformation	10
3.6 Model Selection and Training	12
3.7 Hyperparameters	14
3.8 Optimisers	15
3.9 Evaluation metrics	16
3.10 Common Issues with CNNs	17
4. Dataset Creation and Preprocessing	18
4.1 Extracting HTK data	18
4.2 Other species and other kiwis	20
4.3 Imbalance in the Dataset	22
5. Results	23
5.1 Hypotheses	23
5.2 Hyperparameters	24
5.3 Spectrogram and Colour Scheme Investigation	24
5.4 Dataset Augmentation Investigation	25
5.5 Optimiser Investigation	28
6. Conclusion, Next Steps and Final Thoughts.....	29
6.1 Discussion and Conclusions	29
6.2 Reflections	30
6.3 Next Steps	31
7. Bibliography	32
8. Link to my Github repository	33

Abstract

Machine learning, particularly convolutional neural networks (CNNs), have been successfully employed across a wide range of image processing applications, from identifying the ripeness of fruit (Olisha et al., 2024) to detecting breast cancer (Das et al., 2023). Recently, the use of neural networks has extended beyond image recognition to the classification of audio signals, including environmental sounds (Nanni et al., 2021). The need for analysing large datasets and understanding complex variable interactions in conservation has also driven the adoption of machine learning in this field (Branco et al., 2023). Given the rapid decline in species populations worldwide, further research in this relatively underexplored area is essential to support conservation efforts (Tuia et al., 2022). This project aims to contribute to these efforts by developing and applying machine learning techniques to identify endangered species of kiwi birds through the classification of audio signals from their calls. The results of this project shows that the use of data augmentation techniques and pretrained architectures are sufficient to create an accurate working model even from a small dataset.

Acknowledgements

I would like to express my sincere gratitude to my supervisor, Dr. Jason Anquandah, for her invaluable support and guidance throughout the course of this project. My good friend Charlie Clarke a constant source of good advice and without whom I may not have even been on this data science journey. I am also deeply thankful to Justin Blaikie and Kate Dobbie from the Department of Conservation in New Zealand for their assistance. Furthermore, I extend my appreciation to the entire Xeno-Canto team, particularly the contributors who uploaded recordings that were crucial for training my model. These include:

James Lidster	Johannes Fischer	Nathan El
Zhang Shen	Fernand Deroussen	Craig Wilson
Dan Lane	Patrick Aberg	Martin Overy
Barry Edmonston	Nick Talbot	Esteban Martinez
David Boyle	David Bradley	Fredes
Lindsey Alexander	Stefan Greif	Romauld Mikusek
Andrew Mccafferty	George Wagner	Ron Van Bemmelen
Tony Fulford	Ken George	Romuald Mikusek
Ralf Wendt	David Welch	Janas Kotlarz
Jonas Kotlarz	Matthias Feuerenger	Daniel Lane
	David Boyle	Chris Harrison

List of Commonly used acronyms

CNN	Convolutional Neural Network
DOC	Department of Conservation (based in New Zealand)
HTK	Haast Tokoeka Kiwi
MFCC	Mel Frequency Cepstral Coefficients
SSiA	Super Signal Augmentation
TSM	Time Scale Modification
SGD	Stochastic Gradient Descent

1. Introduction

The Haast Tokoeka Kiwi (HTK) is one of New Zealand's rarest avian species, with an estimated population of only 400 individuals in 2015. Due to ongoing efforts in population monitoring and predator trapping, projections suggest that the population could increase to 738 by 2030 (Anon., <https://savethekiwi.nz/about-kiwi/kiwi-species/tokoeka/> [Accessed 23 August 2024]). Native to the rugged mountainous regions of Haast, Fiordland, and Rakiura in the South Island of New Zealand (Aotearoa), the HTK's elusive nature presents a unique challenge for conservationists. To ensure the survival of this species, preserving genetic diversity within the existing population is essential.



Figure 1. Haast Tokoeka Kiwi bird (Anon., <https://www.doc.govt.nz/nature/native-animals/birds/birds-a-z/kiwi/tokoeka/> [Accessed 28 July 2024])

Given the difficulty of navigating the bird's habitat, acoustic recorders have become invaluable tools for identifying current population sizes and discovering new breeding pairs of the HTK. These devices capture audio data, providing a potential method to detect the presence of the kiwi within vast and challenging landscapes. Accurate and reliable identification is crucial for determining population sizes and confirming the existence of new populations, which are vital components of the species preservation strategy. It has been shown that even experts can struggle to consistently agree on species identification, underscoring the need for scrupulous data when training computer models and highlighting the advantages of an objective, digital approach in this field (Mortimer and Greene, 2017).

In collaboration with the Department of Conservation in New Zealand (DOC), I am eager to contribute to the analysis of this data. The objective is to employ machine learning methods to enhance the accuracy of identifying the HTK in these sound recordings. This initiative aims to provide conservationists with a robust and efficient tool for monitoring and safeguarding this critically endangered species.

2. Related Work

This study builds on the foundation laid by prior research, such as the extensive investigation into different model architectures, visual representations, and dataset augmentation techniques conducted in the paper “An Ensemble of Convolutional Neural Networks for Audio Classification” (Nanni et al., 2021). In that study, the authors utilised the ESC-50 dataset, which includes recordings of cats, birds, and other environmental sounds, achieving promising results. They employed several methods to enhance the dataset, including Time Scale Modification (TSM) and Super Signal Augmentation (SSiA). The study also explored a variety of CNN architectures, such as VGG16, AlexNet, and GoogleNet, and experimented with different forms of audio visualisation, with the Mel-Spectrogram proving to be the most effective. The combinations of these approaches as well as others were analysed, all of which achieved similarly high levels of accuracy, exceeding 95%.

Some of the first studies that looked at classifying bird song used spectrographic analysis to classify Rose-crested pipits (“Interactive Classification Using Spectrograms and Audio Glyphs”, Cakmak et al., 2018), which included extracting the spectral centroid, spectral bandwidth and the spectral roll-off and displaying their results in audio glyphs which they used to classify the birds. In the more recent paper “Using Various Pre-Trained Models for Audio Feature Extraction in Automated Audio Captioning” (Won et al., 2023), a CNN model was combined with a transformer to generate captions for audio prompts, and multiple CNN architectures were trialled to classify audio signals, including the previously mentioned VGG16. Notable success was found with CNN10, an architecture specifically designed for classifying spectrograms. The use of non-negative spectrogram decomposition has been researched as a more effective way of extracting features from acoustic data. Studies found that this outperformed more traditional approaches such as Mel Frequency Cepstral Coefficients (MFCCs; Ludeña-Choez, Quispe-Soncco and Gallardo-Antolín, 2017). Although the various methods suggested could be adopted, as this is a proof-of-concept study of a new dataset I will adopt more traditional approaches to transform my audio data using spectrograms, since the Mel-Spectrogram proved to yield high results in a more recent works. At a later date I would like to explore the use of the CNN10 architecture, but as this is not a pretrained model that is part of the PyTorch package I would have to manually create the architecture which is beyond the scope of this project.

The Won et al. study highlights the benefits of using the Adam optimiser, which, due to its adaptive nature, often results in faster convergence compared to traditional optimisers like Stochastic Gradient Descent (SGD) and Adagrad. The Adam optimiser adjusts the learning rate for each parameter individually, making it effective in handling sparse gradients and noisy data, thereby enhancing training stability and performance. In practical use, the learning rate for Adam typically starts lower than usual, around 0.001 or 0.0001, to ensure stable convergence. In my model, although I will experiment with the use of the Adam optimiser and with Adagrad, since the machine I will be training my models on is not specially set up for large data processing required for CNNs, SGD will be used initially for baseline models due to its efficiency in both computation and memory, although due to its large variance convergence can be slower (Yang, 2021).

Mortimer and Greene highlight the challenges in classifying bird songs, noting that even experienced ornithologists only reached a 50% agreement rate when identifying Tui bird calls from audio recordings. In the best-case scenario, this agreement peaked at just 85%. (Mortimer and Greene, 2017) These findings underscore the importance of rigorous scrutiny of the training data for my model and set a clear benchmark for success. To be considered effective, my model must not only surpass human performance in terms of speed but also achieve an accuracy rate which compares to the best human performances of 85%.

Building upon previous research, this study aims to adopt traditional methods to achieve proof of concept results for a newly created dataset specifically designed for classifying HTK calls. Given that the dataset will be smaller than what is typically used for CNNs, I will employ data augmentation techniques to improve model performance (Nanni et al., 2021). Additionally, I will utilise pre-trained models, as these approaches have been shown to enhance model accuracy (Knyshov, Hoang and Weirauch, 2021). Audio files will be transformed into Mel-Spectrograms, which have proven effective in similar studies (Nanni et al., 2021). I will also compare the use of MFCCs, as they are widely utilised in audio classification (Ke and Southwest Jiaotong University, 2023). Furthermore, I will investigate whether using different colour schemes in the visualisation of these spectrograms can enhance classification results. Although there is no existing evidence to suggest that this would be beneficial, it remains an area of exploration that has yet to be fully examined.

3. Project Methodology and Objectives

The objective of this study is to develop a CNN model capable of performing multiclass classification to identify various bird species native to the Haast and Fiordland regions of New Zealand, including the HTK. CNNs have been widely used for image classification tasks due to their ability to automatically learn spatial hierarchies of features and adapt to diverse datasets (Kondratyuk et al., 2021). In this study, a bespoke dataset will be created, comprising labelled audio data collected from publicly available online databases, along with audio data provided by the DOC from bird sanctuaries. The audio data will be processed and transformed into spectrograms, which will then be used to train the classifier model. The process and methodology are described in detail in the relevant sections below.

3.1 Objectives

My project to create a model that can detect a HTK from an audio file can be summarised in the following objectives:

- Create, clip, and augment a labelled audio dataset. Data will need to be labelled, obtained from credible sources and of reasonable quality. Due to the lack of data that is available for the HTK, data augmentation techniques to synthesise data will be adopted to increase the number of audio recordings. This objective will be met if I am able to meet the minimum data requirements of using a pretrained model quantified as 32 images per class. (Mortimer and Greene, 2017)
- Transform the audio data into Mel-Spectrograms and MFCCs. Due to CNNs need for standardised image sizing all images generated will be of a uniform size and as high a resolution as possible, this will require the cropping of images and the removal of any scales or legends that are usually created with the spectrogram. This objective will be met by the creation of a Python notebook able to load and transform audio data into spectrograms.
- Train and evaluate pretrained CNN models for identifying HTK and other bird species. This objective will be met by the creation of a Python notebook that will allow me to load my spectrogram images, split the data into train and test, be adaptable to change hyperparameters and optimisers. My evaluation will consider both the overall accuracy of the model but also consider the ability to predict HTK in particular. I will conclude that there is a proof of concept if the model achieves correct accuracies more often than not ($> 50\%$) and is very effective if the model is able to predict kiwis at a level which exceeds that of experts ($>85\%$). (Mortimer and Greene, 2017)

3.2 Ethical statement

When acquiring data from the DOC, ensuring the security and confidentiality of the data was of paramount importance, particularly due to the potential risk of poachers obtaining and misusing the data. Such unauthorised access would directly conflict with the conservation efforts aimed at protecting the kiwi population. To mitigate this risk, the DOC provided the data on physical flash drives, avoiding the vulnerabilities associated with online data transfer. Consequently, all data utilised in this research must be stored and handled securely to maintain this level of protection.

Additionally, the Xeno-Canto recordings referenced in this study have been modified and are subject to Creative Commons licences. While these recordings are used strictly within an educational context and are not being distributed, the DOC should evaluate its legal obligations concerning the use of the resulting model from this research.

3.3 Assessment of User Requirements

The primary user of this project is the DOC, with the central objective being the conservation of the HTK population. A key requirement for the DOC is that the developed solution must reduce the time needed to analyse data related to HTK conservation efforts. The success of this project will be measured by its ability to streamline the analysis process, thereby allowing the DOC to allocate resources more effectively. The project will be considered highly successful if the solution can perform as well as, or better than, human experts in identifying patterns and insights crucial to conservation efforts (Branco, Correia and Cardoso, 2023). Additionally, to ensure the tool is accessible to end users who may not have a technical background, consideration will be given to developing a user-friendly front-end interface that simplifies interaction with the underlying model and data.

3.4 Dataset Creation and Augmentation

The dataset for this project is compiled from two primary sources:

DOC

This source provides audio recordings of HTK, which is crucial for our primary dataset. Due to the limited number of these critically endangered birds the data acquired is invaluable for this study. The data provided by the DOC was in its rawest form (as uncategorised audio recordings of 15 minutes in length from a bird sanctuary) and so several preprocessing steps were required to isolate the calls details of which are provide in section 4.

Xeno-Canto

This open-source database (<https://xeno-canto.org/>) offers a wide variety of animal sounds, including calls from other bird species native to the Haast and Fiordland regions of New Zealand. This supplementary data enriches the dataset by incorporating calls from additional local species.

To extract the largest amount of data from the audio files I collected, several preprocessing steps are employed:

Audio Clipping

Audio files are clipped into 5-second increments. Given the brief duration of individual bird calls, these 5-second segments are still likely to include multiple calls, while at the same time increasing the amount of training data for my model, enhancing the dataset's diversity and robustness.

Data Augmentation

To further augment the dataset and improve model performance, two augmentation techniques, as described by (Nanni et al. (2021)), are applied:

TSM

This technique involves duplicating and stretching audio clips by factors of 0.9 and 1.1. This modification effectively increases the dataset size by introducing variations in the temporal domain. By applying this transformation, the dataset size is effectively doubled, which helps in capturing more variability in the data.

SSiA

This technique involves combining similar audio files to create new variations. Each new file is generated by adding:

- Randomised Gain: A gain adjustment within the range of $[-0.5, 0.5]$ to introduce variability in audio intensity.
- Randomised Pitch Shift: A pitch shift within the range of $[-0.5, 0.5]$ to simulate different pitch variations.

- Noise Addition: Varying levels of noise are added to mimic real-world recording conditions and further diversify the dataset.

This augmentation technique can triple the dataset size by creating diverse and realistic variations of the original recordings.

When both augmentation techniques are applied simultaneously, the dataset can potentially increase up to sixfold. This substantial increase in data volume is expected to enhance the robustness and generalisation of the trained models, improving their performance in identifying HTK and other native bird species.

3.5 Audio Transformation

Spectrograms are widely used as a visual representation of audio data in machine learning tasks, particularly for training CNNs. They convert audio signals into images that depict the frequencies present over time, allowing CNNs to effectively learn patterns from audio data.

I will use the Librosa library in Python for audio data augmentation and Matplotlib's Pyplot module to generate spectrograms. Due to the need for CNNs to use uniform sized imagery all spectrograms were clipped to the following sizes 1800 x 900. In order to generate the highest resolution image, the frequency range was cropped to between 256 and 4096 Hz. Most kiwi calls lie at around 2.5 kHz (Department of Conservation New Zealand, 2024) so this range should be sufficient to capture the required features for classification.

In this project, I will utilise Mel-Spectrograms and MFCCs as the primary methods for audio representation. I will compare the effectiveness of models trained using these visualisations to determine which method provides better performance for audio classification tasks.

Mel-Spectrograms

Mel-Spectrograms are a variant of spectrograms where the frequency axis is mapped to a non-linear Mel scale. This scale is designed to better align with human auditory perception, where lower frequencies are spaced more closely together than higher frequencies, reflecting the human ear's heightened sensitivity to low-frequency sounds. The Mel scale is particularly effective for audio processing tasks that aim to replicate human classification abilities, such as distinguishing subtle differences in sounds (Abraham, Khan and Shahina, 2023). This is especially important for applications like identifying bird calls, where precise acoustic cues are essential for accurate classification.

MFCCs

MFCCs are coefficients that together form a Mel-frequency spectrum, representing the short-term power spectrum of a sound. They are derived by converting the audio signal into a Mel-Spectrogram, applying a logarithmic transformation to the Mel-Spectrogram, and then performing a Discrete Cosine Transform (DCT) on the logarithmic values. The resulting coefficients, known as MFCCs, offer a compact representation of the audio's spectral properties. MFCCs serve as a discrete version of the Mel-Spectrogram and have been widely used in various speaker recognition systems (Abraham, Khan, and Shahina, 2023).

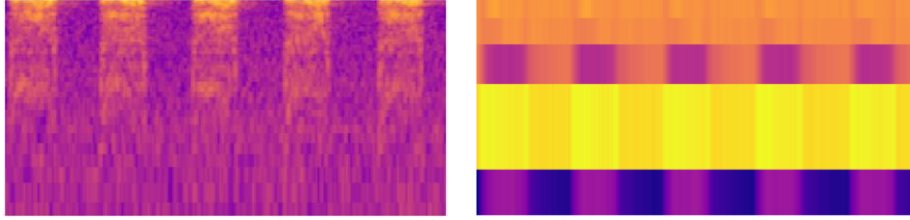


Figure 2. Examples of kiwi calls represented as a Mel-Spectrogram(left) and MFCC (right)

Colour Schemes in Spectrogram Visualisation

In addition to comparing Mel-Spectrograms and MFCCs, I will explore the impact of different colour schemes on the rendering of spectrograms to assess whether they influence model accuracy. Although CNNs are generally robust to colour variations, different colour maps may affect how features are perceived and learned.

For this investigation, I will focus on three perceptually uniform sequential colour maps: Viridis, Plasma, and Cividis. These colour maps are selected for their ability to provide consistent visual representation across various shades and brightness levels, ensuring that features are neither unintentionally emphasised nor obscured (Bergman, Rogowitz and Treinish, 1995).



Figure 3. Examples of colour schemes (Bergman, Rogowitz and Treinish, 1995)

While I do not anticipate significant impacts on CNN performance due to the colour map choice, this area remains underexplored and warrants investigation. Understanding whether the choice of colour scheme affects model accuracy could provide valuable insights into the data preprocessing phase for audio classification tasks.

3.6 Model Selection and Training

CNNs are a type of deep learning algorithm that excels in multiclass image-based recognition tasks. A CNN learns from a set of training images by identifying patterns and assigning weights to them based on their importance in distinguishing between different classes.

Typical CNN architectures consist of two main functions: feature extraction and classification.

In the feature extraction stage, CNNs generally employ three core operations, although the number and type of these operations can vary between different architectures as described below.

Convolution Operation

This operation involves applying filters to the input image to produce feature maps that highlight specific features, such as edges or textures. The filter slides across the image and performs a dot product between the input values and the filter values at each position. The result is a feature map that captures the presence of the features detected by the filter shown below.

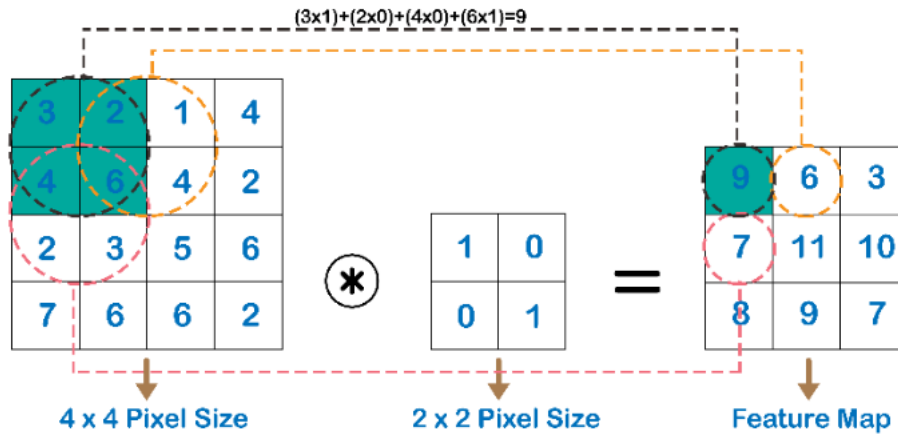


Figure 4. An example of a convolutional process (Fu'adah et al., 2021)

Rectified Linear Unit (ReLU)

The ReLU function introduces non-linearity into the model by applying an operation that replaces all negative values in the feature maps with zero, while positive values remain unchanged. One benefit of its use is to avoid the “vanishing gradient problem”, (described in detail in section 3.8).

$$ReLU(x) = \max(0, x)$$

Pooling

Pooling reduces the spatial dimensions of the feature maps, which helps in summarising the features and reducing computational complexity. Common pooling methods include max pooling where the maximum value of the feature map is selected and average pooling where an average is used. In the classification stage, CNNs typically include two main components which are the following.

Fully Connected Layer

This layer connects every neuron in the previous layer to every neuron in the current layer. It aggregates the features extracted and learns to map them to the output classes.

Activation Function

The activation function in the classification stage, often a softmax function, converts the outputs of the fully connected layer into probabilities, allowing the network to assign a class label to the input image.

By combining these operations, CNNs effectively learn and classify images, making them highly effective for various image recognition and classification tasks.

All of these steps are summarised in figure 5

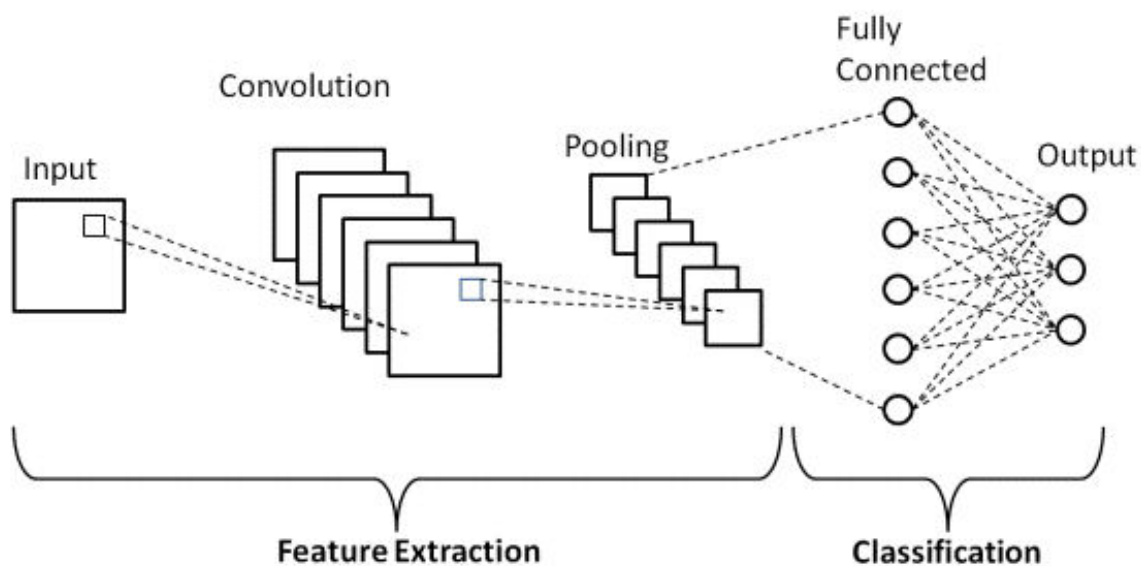


Figure 5 Visual representation of a typical CNN architecture (Anon. (2024) *upGrad blog*. 19 June 2024 [online]. Available from: <https://www.upgrad.com/blog/basic-cnn-architecture/> [Accessed 24 August 2024].)

Although CNNs traditionally require large datasets to achieve high performance, recent advancements have demonstrated the effectiveness of pretrained models on smaller datasets. Studies have shown that pretrained models can maintain high accuracy even with datasets averaging as few as 32 images per class (Knyshov, Hoang and Weirauch, 2021). Building on these insights, this project utilises pretrained models to enhance performance, despite the limited size of the dataset. I will also be using a pretrained CNN model AlexNet, which features eight hidden layers: five convolutional layers followed by three fully connected layers (figure 6). This model has been pretrained on the ImageNet dataset, which provides a diverse range of categories to improve their generalisation capabilities. These pretrained models will be implemented using the PyTorch library in Python.

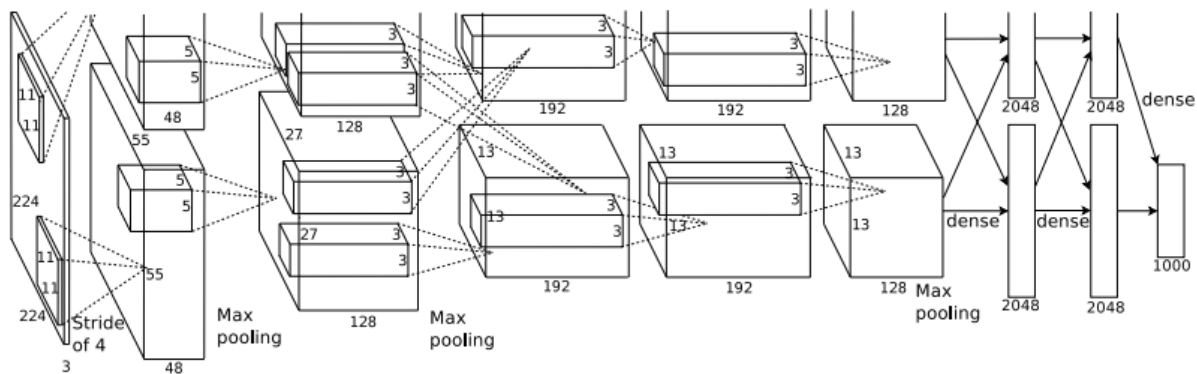


Figure 6 An illustration of the architecture of AlexNet (Krizhevsky, Sutskever and Hinton, 2012)

3.7 Hyperparameters

As well as exploring these different architectures, several parameters require tuning to achieve the most accurate and robust model.

Epochs

An epoch is one complete pass through the entire training dataset. In general, increasing the number of epochs allows the model to see the data more times, which can help it learn more features and improve performance. However, while more epochs typically lead to better model accuracy, there are potential downsides:

Overfitting: With more epochs, the model may begin to memorise the training data rather than generalising from it. This can lead to overfitting, where the model performs well on the training data but poorly on unseen validation or test data.

Training Time: Increasing the number of epochs will also increase the time required to train the model. This can become impractical, especially with very large datasets or complex models with many layers.

Batch Size

Batch size refers to the number of training examples used in one forward/backward pass of the training process. The choice of batch size can impact model performance and training efficiency.

Larger batch sizes can make training more stable and efficient by averaging out noise, but they also require more memory and can slow down each epoch. Smaller batch sizes, on the other hand, may lead to noisier updates but can provide a form of regularisation and potentially faster convergence.

Learning Rate

The learning rate determines how much to adjust the model's weights with respect to the gradient of the loss function. A higher learning rate can speed up training but may overshoot the optimal weights, leading to instability. Conversely, a lower learning rate provides more precise updates but can make training very slow and potentially get stuck in local minima. Typical learning rates are 0.01, 0.001, and 0.0001.

3.8 Optimisers

Optimisers are algorithms used to adjust the model's weights based on the gradients computed during backpropagation. Different optimisers can have a significant impact on training performance in my study the following optimisers will be tested.

SGD

The basic optimiser that updates weights based on the average gradient of a batch. It is simple and effective but may require careful tuning of the learning rate and may converge slowly.

AdaGrad

An adaptive gradient algorithm that adjusts the learning rate for each parameter based on the magnitude of past gradients. This means that parameters with frequently large gradients get smaller updates, while those with smaller or infrequent gradients get larger updates. AdaGrad is particularly well-suited for problems with sparse data as it effectively deals with different feature frequencies (Chakrabarti and Chopra, 2021). However, because the learning rate is continually decreasing and accumulates over time, it can lead to overly small updates and thus slow convergence in the later stages of training.

Adam

An advanced optimiser that combines the advantages of two other extensions of SGD, namely AdaGrad and RMSProp. It adapts the learning rate for each parameter and uses momentum for faster convergence. In practice, when using Adam, typical learning rates are lower compared to optimisers like SGD.

3.9 Evaluation metrics

In order to properly assess my model I will be using a number of accuracy measures which I have defined below that will assess both the overall effectiveness of the model and of identifying HTK. Above 50% will show proof of concept whereas 85% will indicate a working model that could be used by the DOC.

Accuracy

The most basic measure of performance indicating the percentage of correctly identified cases when compared to all possible cases.

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

Precision

Calculates the proportion of true positive predictions among all positive predictions made by the model.

$$Precision = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}$$

Higher precision indicates that my model will have fewer false positives meaning that if my model identifies a kiwi it is more likely to be a correct identification.

Recall

Measures the proportion of true positive predictions out of all actual positives in the dataset.

$$Recall = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

High recall indicates that the model is effective at identifying positive instances meaning that it is less likely to miss a kiwi call in a recording.

F1 Score

The harmonic mean of precision and recall, providing a single metric that balances both.

$$F1\ Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

High F1 Score indicates that the model has a good balance between both precision and recall.

Cross validation

Due to the level of natural variation that can appear when training a model, in order to achieve reliably reportable results, I will be adopting a 5-fold cross-validation for the majority of my experiments. This method selects a random subset of the data to serve as a validation set while the remaining data is used for training. This process is repeated five times, with each fold using a different subset of data for validation. The results from each fold are then averaged to provide a more robust and reliable measure of the model's performance, reducing the impact of any particular data split and ensuring that the evaluation is less biased by specific data distributions or anomalies within a single fold. Although a greater number of folds could yield even more reliable results, a 5-fold cross-validation approach is chosen to balance the need for rigorous evaluation with the time required to explore various optimisers, architectures, and dataset augmentation strategies. This approach allows for a comprehensive investigation of these factors while managing computational resources effectively.

3.10 Common Issues with CNNs

Vanishing Gradient Problem

This issue occurs when the gradients used to update the weights of a neural network become extremely small during backpropagation. As a result, the weights are updated very slowly, or not at all, causing the network to train very slowly or get stuck in a local minimum. To mitigate this issue in my project, I am using ReLU activation functions, which help maintain larger gradients and improve the training process.

Exploding Gradient Problem

The exploding gradient problem is the opposite of the vanishing gradient problem. It occurs when gradients become excessively large during training, causing the model weights to update too much and diverge, leading to unstable training and poor model performance. While this issue is less common than the vanishing gradient problem, my initial models on the dataset did encounter it, necessitating the use of gradient clipping to stabilise the training process and achieve a successful model.

Large Training Times

CNNs, especially those that are deep and have many parameters, can require significant time to train due to their computational complexity. This is particularly true when dealing with large datasets or high-resolution images, where the volume of data and the number of calculations needed per layer can greatly extend training times. To address this, techniques like transfer learning (using pre-trained models), choosing more efficient architectures, utilising powerful hardware (such as GPUs or TPUs), or employing parallel processing are commonly employed. However, since I will be using a standard laptop setup without a GPU, it will be necessary to set realistic expectations for the number of epochs, complexity of architecture and cross-validation folds, reducing them to allow sufficient time for achieving baseline testing.

4 Dataset Creation and Preprocessing

The creation of a bespoke dataset was pivotal for the success of this project. A significant amount of time was invested in curating and preparing this dataset to ensure its quality and relevance. Below, I detail the process of dataset creation and the subsequent augmentation techniques applied.

4.1 Extracting HTK Data

The creation of a bespoke dataset was pivotal for the success of this project. A significant amount of time was invested in curating and preparing this dataset to ensure its quality and relevance. Below, I detail the process of dataset creation and the subsequent augmentation techniques applied.

To clean this data, I developed code to preprocess the audio files. The code identified segments where the audio signal's amplitude exceeded a threshold of 0.05. This threshold was determined through trial and error to effectively capture the kiwi calls while minimising interference from other environmental sounds. The decision to use a higher amplitude threshold aimed to ensure the selection of high-quality calls.

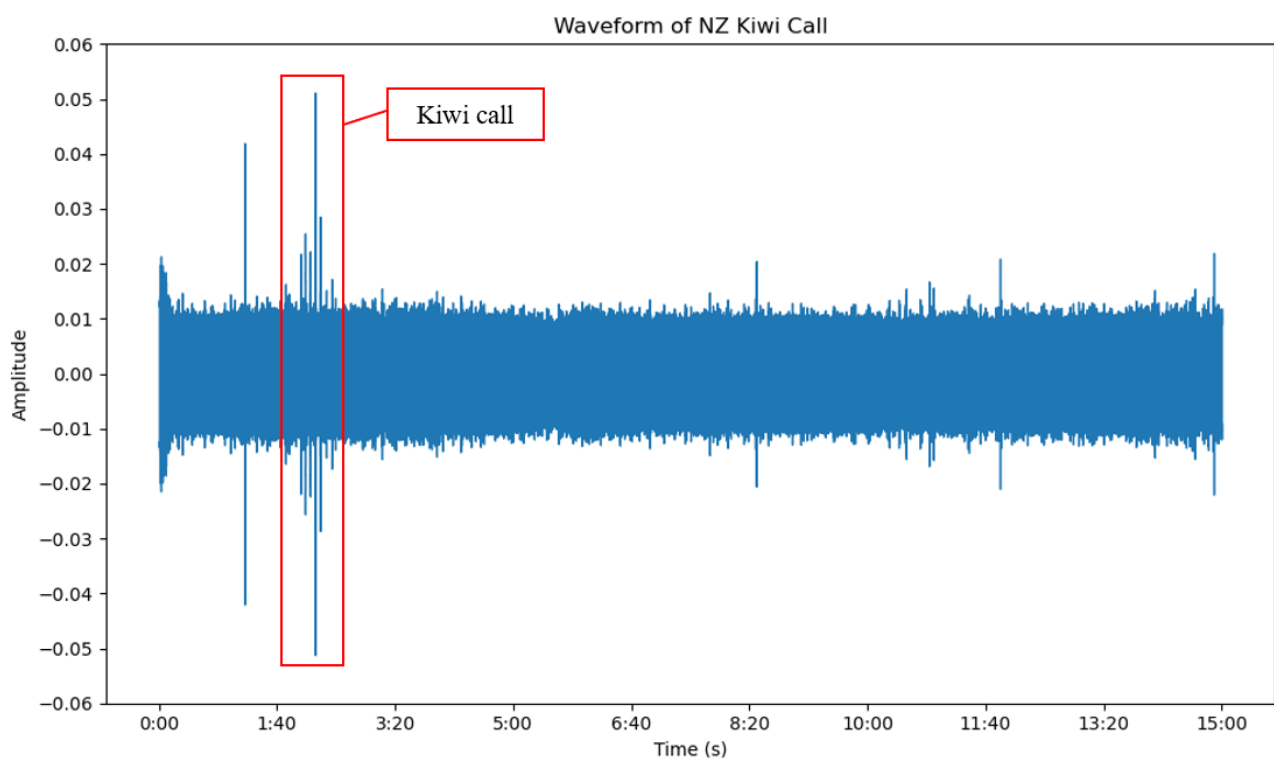


Figure 6. Typical waveform chart of the data provided by the sanctuary, with the location of the kiwi call highlighted in red.

The code iterated through each audio file, pinpointing sections where the amplitude surpassed the threshold. For each detected instance, it extracted a 10-second window of audio, which was then divided into two 5-second segments for further analysis. The output of this script is illustrated in Figure 7.

```
File: 20200628_003020.wav
Number of potential calls: 1
Time periods:
['01:50', '02:00']
Saved: C:\Users\44778\OneDrive\Desktop\UWE Docs\7. Dissertation\Trimmed Audio\20200628_003020.wav_sample_1.wav
Saved: C:\Users\44778\OneDrive\Desktop\UWE Docs\7. Dissertation\Trimmed Audio\20200628_003020.wav_sample_2.wav
```

Figure 7. Example of code output.

Through this automated process, the 17 hours of audio were reduced to approximately 2 hours of relevant content, resulting in 1,947 individual 5-second WAV files. This subset of data was further refined manually by listening to the recordings and categorising them into files with identifiable bird calls and those without. Ultimately, 46 5-second files containing HTK calls were isolated. Additionally, 100 5-second recordings without calls were retained to provide a background noise dataset referred to in this paper as the class “blank”. If my model is successful further mining of data from these files could be carried out by machines saving a lot of time and effort.

4.2 Other Species and Other Kiwis

In collaboration with the DOC, a list of native bird species from the Haast and Fiordland regions was compiled. Audio recordings for these species were sourced from the Xeno-Canto database (<https://xeno-canto.org/>). The species included are shown in figure 8 below







Common Name (English)	Scientific Name	Picture	Number of audio files in the data set
New Zealand Bellbird (Korimako)	<i>Anthornis melanura</i>		10
New Zealand Kaka	<i>Nestor meridionalis</i>		11
Kea	<i>Nestor notabilis</i>		10
Morepork (Ruru)	<i>Ninox novaeseelandiae</i>		10
Paradise Shelduck (Putakitaki / Putangitangi)	<i>Tadorna variegata</i>		10
Australasian Swamphen (Pukeko)	<i>Porphyrio melanotus</i>		8

Figure 8. Species of birds native to the Haast Fiordland region in the southern island of New Zealand, all images collected from the website (www.doc.govt.nz) Accessed[13/08/2024]

In addition to this a selection of other kiwi species that can be found all around New Zealand not limited to the South Island were also added a detailed list of these species is provided in figure 9.






Common Name (English)	Scientific Name	Picture	Number of audio files in the data set
Southern Brown Kiwi	<i>Apteryx australis</i>		1
Great Spotted Kiwi	<i>Apteryx haastii</i>		3
North Island Brown Kiwi	<i>Apteryx mantelli</i>		21
Okarito Kiwi (Rowi)	<i>Apteryx rowi</i>		4
Little Spotted Kiwi	<i>Apteryx owenii</i>		5

Figure 9. Species of other kiwi birds used in the dataset, all images collected from the website (www.doc.govt.nz) Accessed[13/08/2024]

After preprocessing the audio files into 5 second clips and applying the data augmentation techniques mentioned in my methodology 4 datasets were created. The number of 5 second files for each class are shown in the table below.

Table 1: Size and proportional representation of each class in the dataset after each augmentation technique was applied

Augmentation technique	Bellbird	Blank	HTK	Kaka	Kea	Morepork	Other kiwi	Shelduck	Swamphen	Total dataset size
NoAug	149 (16%)	100 (11%)	45 (5%)	94 (10%)	113 (12%)	87 (10%)	154 (17%)	96 (11%)	67 (7%)	905
TSM	298 (16%)	200 (11%)	90 (5%)	188 (10%)	226 (12%)	174 (10%)	308 (17%)	192 (11%)	134 (7%)	1810
SSiA	447 (16%)	300 (11%)	135 (5%)	282 (10%)	339 (12%)	261 (10%)	462 (17%)	288 (11%)	201 (7%)	2715
TSM/SSiA	894 (16%)	600 (11%)	270 (5%)	564 (10%)	678 (12%)	522 (10%)	924 (17%)	576 (11%)	402 (7%)	5430

4.3 Imbalance in the Dataset

As can be seen from the table above, there is considerable imbalance in the dataset, with only 5% of all the training data belonging to the HTK. This may cause issues in model training, such as the model becoming biased towards the majority classes and failing to accurately learn and predict the characteristics of the minority class in this case HTK. Such imbalances can lead to poor generalisation and low recall for the minority class, reducing the model's overall effectiveness in identifying HTK sounds. To mitigate this, several solutions can be adopted:

Over-sampling the Minority Classes

This involves artificially increasing the number of samples in the minority class by duplicating existing samples or generating synthetic data points. Techniques such as Synthetic Minority Over-sampling Technique (SMOTE) can be used to create synthetic examples by interpolating between existing samples in the minority class. This helps the model receive more training data for the underrepresented class, potentially improving its ability to learn and generalise features specific to the HTK.

Under-sampling the Majority Classes

To balance the dataset, another approach is to reduce the number of samples in the majority classes. This can be done by randomly removing samples from the majority class to match the size of the minority class. However, this method can risk losing valuable information from the majority classes and may not always be the best solution, especially if the majority classes contain significant variability.

Data Augmentation / Synthesising Data

Data augmentation involves creating new training samples by applying various transformations to the existing data. For audio data, this could include adding noise, shifting time, changing pitch, or even time-stretching audio recordings. This process can help to expand the training dataset and provide more diverse examples for the model to learn from.

Although addressing dataset imbalance is crucial, it is beyond the scope of my project to explore all possible methods for balancing the dataset. Instead, I will adopt a straightforward, proof-of-concept approach. I will use the maximum amount of data generated from my HTK recordings and, for each other class, select an augmentation method that produces a similar amount of data to match the HTK data. The table below provides an overview of the dataset after applying these methods.

Table 2: Size and proportional representation of each class in the dataset after rebalancing

Dataset Name	Bellbird	Blank	Haast kiwi	Kaka	Kea	Morepork	Other kiwi	Shelduck	Swamphen	Total dataset size
<i>Data Augmentation method applied</i>	TSM	SSiA	TSM/SSiA	SSiA	TSM	SSiA	TSM	SSiA	SSiA	
Balanced	298 (12%)	300 (12%)	270 (11%)	282 (12%)	226 (9%)	261 (11%)	308 (13%)	288 (12%)	201 (8%)	2434

5. Results

5.1 Hypotheses

The hypotheses investigated in this study are as follows:

- a. Mel-Spectrograms will yield higher classification accuracy compared to MFCCs.
- b. The choice of colour scheme will have a negligible impact on the classification accuracy of the models.
- c. Applying data augmentation techniques will significantly improve the classification accuracy of the model.
- d. Rebalancing the dataset will result in better performance metrics for the HTK.
- e. Pre-trained models will achieve excellent classification performance even with a relatively small dataset (accuracy > 85%).

5.2 Hyperparameters

Due to the time required to train the models, all hyperparameter tuning was conducted using the AlexNet architecture, which is a simple and quick model that is well-suited for establishing initial baseline performance (Krizhevsky, Sutskever and Hinton, 2012). The batch size and the number of epochs were varied during testing, with a batch size of 16 and 8 epochs being selected. This combination resulted in a more stable model that was less prone to errors in the code while also achieving good training efficiency.

Subsequent tests were conducted to determine the optimal learning rates for each optimiser, using 5-fold cross-validation to validate the results standard for such studies (Sejuti and Islam, 2023). The table below presents the outcomes of these tests. Significant differences in performance were observed when using standard learning rates of 0.01, 0.001, and 0.0001, with each optimiser performing best at a different rate. For each model tested thereafter, the optimal learning rate was chosen based on these results. Given its faster training times and high accuracy, SGD with a learning rate of 0.01 was selected as the base optimiser for testing further hypotheses.

Table 3: Hyperparameter Tuning for Each Optimiser

Each model was trained using the AlexNet architecture with 8 epochs and a batch size of 16. The mean accuracy was calculated over a 5-fold cross-validation. The most significant values are highlighted in bold.

Optimiser	lr = 0.01	lr = 0.001	lr = 0.0001
SGD	0.8	0.71	0.42
Adam	0.16	0.4	0.79
AdaGrad	0.24	0.8	0.77

5.3 Spectrogram and Colour Scheme Investigation

Table 4: Accuracy of Each Colour Map Used for Generation of Data
Each model was trained using the AlexNet architecture with 8 epochs, a batch size of 16, and a learning rate of 0.01. The mean accuracy was calculated over a 5-fold cross-validation. The most significant values are highlighted in bold.

	Viridis	Plasma	Cividis	AVG accuracy
Mel-Spectrogram	0.78	0.79	0.76	0.78
MFCC	0.76	0.73	0.77	0.75
AVG Accuracy	0.77	0.76	0.765	

Table 2 shows that there is little evidence to suggest that the colour schemes used in generating spectrograms have a significant impact on the overall accuracy of the model. While there is some indication that models trained using Mel-Spectrograms achieve higher accuracy, this trend is not consistent across all colour schemes. Consequently, more data is needed to draw a definitive conclusion regarding the influence of colour schemes on model performance. Despite the inconclusive results, the Viridis colour scheme and Mel-Spectrogram will be adopted for future modelling due to their relatively better performance in preliminary tests.

5.4 Dataset Augmentation Investigation

Table 5: Overall Accuracy, Precision, Recall, and F1 Score for Each Data Augmentation Method
Each model was trained using the AlexNet architecture with the SGD optimiser, 8 epochs, a batch size of 16, and a learning rate of 0.01. The metrics were measured over a 5-fold cross-validation. The most significant values are highlighted in bold.

	No_Aug	TSM	SSiA	TSM-SSiA	Balanced
Dataset size	905	1810	2715	5430	2434
Accuracy	0.79	0.88	0.89	0.9	0.89
Precision	0.79	0.88	0.88	0.9	0.89
Recall	0.79	0.88	0.88	0.9	0.89
F1	0.79	0.88	0.88	0.9	0.89
% increase in accuracy	0%	11%	13%	14%	13%

The table demonstrates that the data augmentation techniques employed have positively impacted the model's accuracy, showing an 11-14% increase in the accuracy of the augmented datasets compared to the original raw data, consistent with results found in related works (Nanni et al., 2021). Increasing the dataset size influences accuracy, as evidenced by the data; however, balancing the dataset also appears to have a significant effect on overall performance. The balanced dataset of 2,434 instances outperforms the SSiA dataset of 2,715 instances in terms of precision, recall, and F1 score. A closer look at class-specific accuracies (Table 4) provides further evidence of the strong performance of the balanced dataset, particularly when comparing the underrepresented HTK class. In this case, both the balanced and fully augmented TSM-SSiA datasets were the strongest performers, achieving high overall accuracy as well as high precision, recall, and F1 scores for the HTK class.

Table 6: Precision, Recall, and F1 Score for Each Class Across the Data Augmentation Methods
Each model was trained using the AlexNet architecture with 8 epochs, a batch size of 16, and a learning rate of 0.01. The metrics were measured over a 5-fold cross-validation. The most significant values are highlighted in bold.

		No_Aug	TSM	SSiA	TSM-SSiA	Balanced
	<i>Overall accuracy</i>	0.72	0.88	0.89	0.90	0.89
<i>Bellbird</i>	Precision	0.72	0.90	0.89	0.9	0.85
	Recall	0.88	0.90	0.91	0.91	0.90
	F1-Score	0.79	0.90	0.90	0.91	0.88
	(Proportion of the dataset)	(16%)	(16%)	(16%)	(16%)	(12%)
<i>Blank</i>	Precision	0.93	0.90	0.9	0.93	0.91
	Recall	0.97	1.00	0.98	0.97	0.94
	F1-Score	0.95	0.95	0.94	0.95	0.92
	Proportion of the dataset	(11%)	(11%)	(11%)	(11%)	(12%)
<i>Haast Kiwi</i>	Precision	0.62	0.81	0.76	0.83	0.82
	Recall	0.62	0.72	0.70	0.82	0.88
	F1-Score	0.62	0.76	0.73	0.93	0.85
	Proportion of the dataset	(5%)	(5%)	(5%)	(5%)	(11%)
<i>Kaka</i>	Precision	0.87	0.81	0.82	0.89	0.83
	Recall	0.88	0.83	0.84	0.84	0.80
	F1-Score	0.87	0.82	0.83	0.87	0.81
	Proportion of the dataset	(12%)	(12%)	(12%)	(12%)	(9%)
<i>Kea</i>	Precision	0.66	0.94	0.95	0.92	0.90
	Recall	0.65	0.87	0.88	0.93	0.93
	F1-Score	0.66	0.90	0.91	0.92	0.92
	Proportion of the dataset	(10%)	(10%)	(10%)	(10%)	(12%)
<i>Morepork</i>	Precision	0.89	0.92	0.98	0.95	0.95
	Recall	0.83	0.93	0.95	0.95	0.94
	F1-Score	0.86	0.92	0.97	0.95	0.94
	Proportion of the dataset	(10%)	(10%)	(10%)	(10%)	(11%)
<i>Other conserv</i>	Precision	0.73	0.83	0.87	0.88	0.87
	Recall	0.62	0.80	0.83	0.85	0.79
	F1-Score	0.67	0.81	0.82	0.87	0.83
	Proportion of the dataset	(17%)	(17%)	(17%)	(17%)	(13%)
<i>Shelduck</i>	Precision	0.94	0.95	0.91	0.93	0.97
	Recall	0.84	0.96	0.97	0.97	0.97
	F1-Score	0.89	0.96	0.94	0.95	0.97
	Proportion of the dataset	(11%)	(11%)	(11%)	(11%)	(12%)
<i>Swamphen</i>	Precision	0.79	0.88	0.81	0.86	0.93
	Recall	0.82	0.93	0.85	0.87	0.88
	F1-Score	0.8	0.90	0.83	0.86	0.90
	Proportion of the dataset	(7%)	(7%)	(7%)	(7%)	(8%)

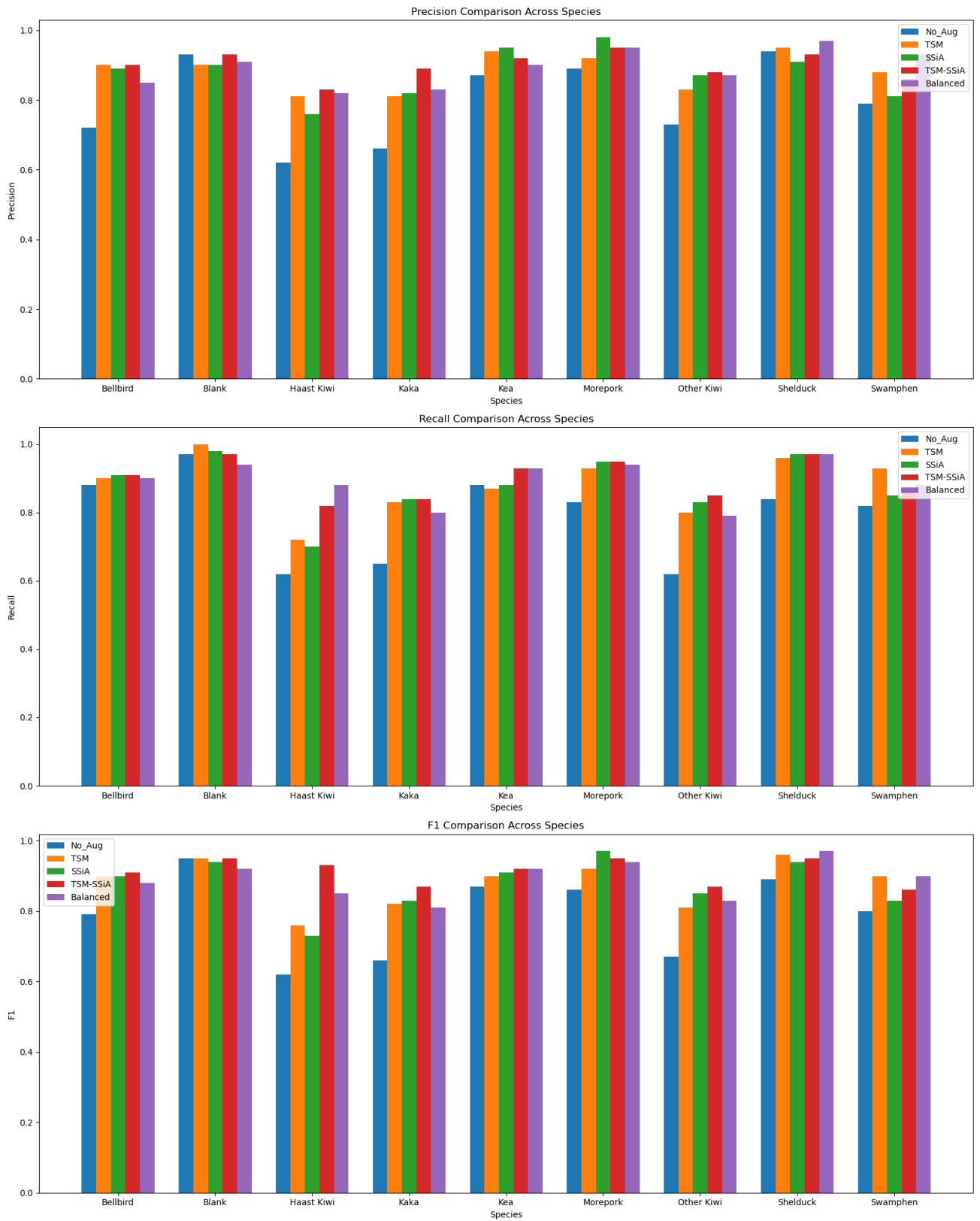


Figure 10. Visualisation of table 6 showing the summary of precision, recall and $F1$ score of each Class Across various data augmentation methods

5.5 Optimiser Investigation

Further tests were conducted to evaluate the performance of different optimisers on the Balanced and TSM-SSiA datasets, with the aim of identifying the best overall model. As shown in Table 7, all models exceeded the benchmark accuracy of 85%, with the best model achieving an overall accuracy of 0.94. This level of performance was also consistently observed across all other metrics.

Table 7 Precision, Recall and F1 score of different optimisers combined with balanced and TSM-SSiA augmentation techniques. Each model was trained using the AlexNet architecture with 8 epochs and a batch size of 16 (Metrics averaged over a 5-fold cross-validation) most significant values are shown in bold.

Optimiser		SGD (lr = 0.01)		AdaGrad (lr = 0.001)		Adam (lr = 0.0001)	
		Overall	HTK	Overall	HTK	Overall	HTK
Balanced	Accuracy	0.89		0.91		0.92	
	Precision	0.89	0.82	0.91	0.86	0.92	0.85
	Recall	0.89	0.88	0.92	0.94	0.92	0.87
	F1 Score	0.89	0.85	0.92	0.90	0.92	0.86
TSM-SSiA	Accuracy	0.9		0.94		0.93	
	Precision	0.9	0.83	0.94	0.87	0.93	0.85
	Recall	0.9	0.82	0.94	0.84	0.93	0.79
	F1 Score	0.9	0.93	0.94	0.85	0.92	0.82

6. Conclusion, Next Steps and Final Thoughts

6.1 Discussion and Conclusions

The overall objective of this project was to employ machine learning techniques to detect the endangered HTK from audio recordings, thereby assisting in the monitoring of their population. To achieve this, a small dataset was sourced from both the DOC and Xeno-Canto website both reputable sources for data and subsequently augmented into standardised chunks of audio. The size of this dataset exceeded the minimum required amount of data and was able to be used to train successful models fulfilling all the criteria for my first objective.

These audio files were then transformed into high resolution, uniform Mel-Spectrograms and MFCCs using Python packages and were successfully loaded into a CNN model fulfilling the criteria for my second objective.

Pre-trained models based on the AlexNet architecture were created and evaluated, achieving overall accuracies exceeding the 85% benchmark set by expert human classification consensus. Data augmentation played a crucial role in surpassing this benchmark, with the use of TSM and SSiA techniques increasing the data size by a factor of six and resulting in a 13% improvement in overall accuracy. This is a significant outcome for conservation efforts, particularly when data on endangered species is scarce compared to more common species. Not only did my models show proof of concept they were accurate enough to potentially be used by the DOC in their conservation work.

Balancing the imbalanced dataset notably improved the accuracy of minority classes without affecting training time. Specifically, the F1 score for HTK increased by 50% compared to the non-augmented data. The final model using a still imbalanced but much larger dataset having undergone both TSM and SSiA achieved an overall accuracy of 0.94 across all classes, with precision and recall scores of 0.87 and 0.84, respectively for the HTK class.

Due to the success of my model, I aim to develop a front-end application that processes a 15-minute audio segment, similar to the data provided by DOC, to generate timestamps for potential bird calls and identify the species. This tool would be designed to assist experts by reducing the time needed to manually review recordings, particularly in sections with minimal relevant data.

6.2 Reflections

My personal goal for this project was to gain experience with deep learning models, a technique not covered in depth in the data science course. The project has been both challenging and incredibly rewarding. Despite facing numerous challenges and obstacles, which sometimes constrained the project's scope, the experience has been invaluable.

One major issue was time related. Since the data I was working with came from the DOC, it took time for this to be gathered and sent to me from New Zealand, and I received it only three weeks before the deadline. As such I created a data set ahead of this using online resources, then included this dataset once received. Once I began generating models, the training speed was significantly hampered by the lack of a GPU on my laptop. This limitation was particularly disappointing as it prevented me from fully exploring different architectures and led to restrictions on epoch size and the number of folds in cross-validation, ultimately affecting the comprehensiveness of my findings.

This was a project which had moments which moved incredibly quickly for example the creation of the dataset and the transformation of audio files. But also times of terrible despair and anxiety when I wasn't able to install the PyTorch packages and also when my initial CNN models would not train due to exploding gradients resulting in one class achieving 100% and all other classes 0%. After discussions and encouragement from my supervisor I was able to overcome this challenge through my own research and resilience. My weekly meetings with my supervisor were initially slow paced while I waited for data to arrive but grew in purpose as I created my own dataset from Xeno-Canto in anticipation of the DOC data, but I always found sessions productive where I was able to take stock and plan the tasks and goals I would achieve in the week ahead.

Although all these setbacks limited what I could achieve, I am still very pleased with the results and the knowledge I have gained. The project has provided valuable insights into both the theoretical aspects of implementing a CNN and the practical challenges involved. It has been a learning experience that has not only expanded my understanding but also raised further questions and potential avenues for continued exploration in my learning journey.

6.3 Next Steps

There are numerous opportunities for further investigation and enhancement in this project. Regarding the dataset, as outlined in the construction description, initial bird calls were identified using high amplitude values. With a functioning model now established, there is potential to re-examine the data and extract more information from the audio files. This could address the issue of data imbalance observed in this project, which has been shown to negatively impact model accuracy. Other methods to resolve the imbalance issues such as the use of SMOTE could also be investigated.

Additionally, my preliminary research suggests several alternative data transformations that could improve model performance. For instance, non-negative spectrograms (Ludeña-Choez, Quispe-Soncco and Gallardo-Antolín, 2017) and time-frequency matrix feature classification (Ghoraani and Krishnan, 2011) have been proposed as methods that outperform traditional spectrogram transforms.

While this project has focused on traditional architectures and optimisers, there are numerous pre-trained models available that could potentially offer more accurate results than the AlexNet architecture. Notable examples include VGG16 (Nanni et al., 2021) and CNN10 (Won et al., 2023), with CNN10 demonstrating exceptional performance in audio classification tasks.

Additionally, future projects could explore the monitoring of other endangered species and their predators, contributing to broader conservation efforts.

7. Bibliography

- Abraham, J.V.T., Khan, A.N. and Shahina, A. (2023) A deep learning approach for robust speaker identification using chroma energy normalized statistics and mel frequency cepstral coefficients. *International Journal of Speech Technology* [online]. 26 (3), pp. 579–587.
- Anon. (2024) *upGrad blog*. 19 June 2024 [online]. Available from: <https://www.upgrad.com/blog/basic-cnn-architecture/> [Accessed 24 August 2024].
- Anon. (no date) *Save the Kiwi* [online]. Available from: <https://savethekiwi.nz/about-kiwi/kiwi-species/tokoeka/> [Accessed 23 August 2024a].
- Anon. (no date) [online]. Available from: <https://xeno-canto.org/> [Accessed 29 February 2024b].
- Bergman, L.D., Rogowitz, B.E. and Treinish, L.A. (1995) *A rule-based tool for assisting colormap selection*. In: *Proceedings Visualization '95* [online] Visualization '95. Atlanta, GA, USA: IEEE Comput. Soc. Press, pp. 118–125. Available from: <http://ieeexplore.ieee.org/document/480803/> [Accessed 24 August 2024].
- Branco, V.V., Correia, L. and Cardoso, P. (2023) The use of machine learning in species threats and conservation analysis. *Biological Conservation* [online]. 283, p. 110091.
- Cakmak, E., Schlegel, U., Miller, M., Buchmüller, J., Jentner, W. and Keim, D.A. (2018) *Interactive Classification Using Spectrograms and Audio Glyphs*. In: *2018 IEEE Conference on Visual Analytics Science and Technology (VAST)* [online] 2018 IEEE Conference on Visual Analytics Science and Technology (VAST). pp. 110–111. Available from: <https://ieeexplore.ieee.org/document/8802500> [Accessed 29 February 2024].
- Chakrabarti, K. and Chopra, N. (2021) Generalized AdaGrad (G-AdaGrad) and Adam: A State-Space Perspective. *arXiv.org* [online]. Available from: <https://www.proquest.com/docview/2536120501?pq-origsite=primo&sourcetype=Working%20Papers> [Accessed 30 August 2024].
- Das, H.S., Das, A., Neog, A., Mallik, S., Bora, K. and Zhao, Z. (2023) Breast cancer detection: Shallow convolutional neural network against deep convolutional neural networks based approach. *Frontiers in Genetics* [online]. 13, p. 1097207.
- Fu'adah, Y.N., Wijayanto, I., Pratiwi, N.K.C., Taliningsih, F.F., Rizal, S. and Pramudito, M.A. (2021) Automated Classification of Alzheimer's Disease Based on MRI Image Processing using Convolutional Neural Network (CNN) with AlexNet Architecture. *Journal of Physics: Conference Series* [online]. 1844 (1), p. 012020.
- Gao, Z., Liu, T., Zhu, M., Li, J., Ning, Y. and Wang, Z. (2023) *Environmental Sound Classification Using CNN Based on Mel-spectrogram*. In: *2023 2nd International Conference on Artificial Intelligence and Blockchain Technology (AIBT)* [online] 2023 2nd International Conference on Artificial Intelligence and Blockchain Technology (AIBT). pp. 41–45. Available from: <https://ieeexplore-ieee.org.uwe.idm.oclc.org/document/10248097/?arnumber=10248097> [Accessed 24 August 2024].
- Ghoraani, B. and Krishnan, S. (2011) Time–Frequency Matrix Feature Extraction and Classification of Environmental Audio Signals. *IEEE Transactions on Audio, Speech, and Language Processing* [online] IEEE Transactions on Audio, Speech, and Language Processing. 19 (7), pp. 2197–2209.
- Ke, L. 1 1 U. of L. and Southwest Jiaotong University, C. (2023) Virtual human speech emotion recognition based on multi-channel CNN: MFCC, LPC, and F0 features. [online]. p. 012011.
- Knyshov, A., Hoang, S. and Weirauch, C. (2021) Pretrained Convolutional Neural Networks Perform Well in a Challenging Test Case: Identification of Plant Bugs (Hemiptera: Miridae) Using a Small Number of Training Images. Jockusch, E., ed. *Insect Systematics and Diversity* [online]. 5 (2), p. 3.

Kondratyuk, D., Yuan, L., Li, Y., Zhang, L., Tan, M., Brown, M. and Gong, B. (2021) *MoViNets: Mobile Video Networks for Efficient Video Recognition* [online]. Available from: <http://arxiv.org/abs/2103.11511> [Accessed 30 August 2024].

Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012) *ImageNet Classification with Deep Convolutional Neural Networks*. In: *Advances in Neural Information Processing Systems* [online]. 25, Curran Associates, Inc. Available from: https://papers.nips.cc/paper_files/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html [Accessed 30 August 2024].

Ludeña-Choez, J., Quispe-Soncco, R. and Gallardo-Antolín, A. (2017) Bird sound spectrogram decomposition through Non-Negative Matrix Factorization for the acoustic classification of bird species. [online]. p. e0179403.

Mortimer, J.A.J. and Greene, T.C. (2017) Investigating bird call identification uncertainty using data from processed audio recordings. *New Zealand Journal of Ecology*. 41 (1), pp. 126–133.

Nanni, L., Maguolo, G., Brahnam, S. and Paci, M. (2021) An Ensemble of Convolutional Neural Networks for Audio Classification. *Applied Sciences* [online]. 11 (13), pp. 5796–.

Olisah, C.C., Trehwella, B., Li, B., Smith, M.L., Winstone, B., Charles Whitfield, E., Fernández, F.F. and Duncalfe, H. (2024) Convolutional neural network ensemble learning for hyperspectral imaging-based blackberry fruit ripeness detection in uncontrolled farm environment. *Engineering Applications of Artificial Intelligence* [online]. 132. Available from: <https://uwe-repository.worktribe.com/output/11669512> [Accessed 22 August 2024].

Sejuti, Z.A. and Islam, M.S. (2023) A hybrid CNN–KNN approach for identification of COVID-19 with 5-fold cross validation. *Sensors International* [online]. 4, p. 100229.

Tuia, D. *et al.* (2022) Perspectives in machine learning for wildlife conservation. *Nature Communications* [online]. 13 (1), p. 792.

Won, H., Kim, B., Kwak, I.-Y. and Lim, C. (2023) Using various pre-trained models for audio feature extraction in automated audio captioning. *Expert Systems with Applications* [online]. 231, p. 120664.

Yang, X. (2021) *Stochastic Gradient Variance Reduction by Solving a Filtering Problem* [online]. Available from: <http://arxiv.org/abs/2012.12418> [Accessed 30 August 2024].

Department of Conservation New Zealand, 2024. *Kiwi, call characteristics*. [email attachment]. Sent by K. Dobbie to T. Grubb on 31 May 2024.

8. Link to my Github repository

[tgrubb550/CNN-kiwi-call-classification: My dissertation project 08/24 \(github.com\)](https://github.com/tgrubb550/CNN-kiwi-call-classification)