KANTIPUR ENGINEERING COLLEGE

(Affiliated to Tribhuvan University)

Dhapakhel, Lalitpur

[Subject Code: CT707]

A MAJOR PROJECT MID-TERM REPORT ON

FEATHERFIND : BIRD SPECIES IDENTIFICATION FROM AUDIO

Submitted by:

Gaurav Giri	[Kan077bct034]
Iza K.C.	[Kan077bct039]
Prajwal Khatiwada	[Kan077bct056]
Samrat Kumar Adhikari	[Kan077bct074]

A MAJOR PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR IN COMPUTER ENGINEERING

Submitted to:

Department of Computer and Electronics Engineering

December, 2024

FEATHERFIND : BIRD SPECIES IDENTIFICATION FROM AUDIO

Submitted by:

Gaurav Giri	**[Kan077bct034]**
Iza K.C.	**[Kan077bct039]**
Prajwal Khatiwada	**[Kan077bct056]**
Samrat Kumar Adhikari	**[Kan077bct074]**

A MAJOR PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF BACHELOR IN COMPUTER ENGINEERING

Submitted to:

Department of Computer and Electronics Engineering

Kantipur Engineering College

Dhapakhel, Lalitpur

December, 2024

ABSTRACT

TABLE OF CONTENTS

Abstract **i**

List Of Figures **iv**

Abbreviations **v**

1 Introduction **1**

1.1 Problem Statement 1

1.2 Objectives 2

1.3 Application Scope 2

1.4 Features 3

1.5 Feasibility Study 3

1.5.1 Economic Feasibility 4

1.5.2 Technical Feasibility 4

1.5.3 Operational Feasibility 4

1.5.4 Schedule Feasibility 5

1.6 System Requirements 6

1.6.1 Development Requirements 6

1.6.2 Deployment Requirements 6

2 Literature Review **7**

2.1 Related Works 7

2.2 Related Research 8

3 Methodology **14**

3.1 System Overview 14

3.2 Working Mechanism for Bird Sound Detection 15

3.2.1 Data Collection 15

3.2.2 Data Transformation 17

3.2.3 Feature Extraction 19

3.2.4 Model Architecture 19

3.2.5 Training Process 23

3.2.6 Evaluation 24

3.3 Working Mechanism for Bird Species Classification 25

3.3.1 Data Collection 25

3.3.2 Data Augmentation 26
3.3.3 Data Preprocessing 27
3.3.4 Feature Extraction 28
3.3.5 Model Architecture 29
3.3.6 Training Process 29
3.3.7 Evaluation Metrics 30
3.4 Feature Extraction 31
3.4.1 Mel Spectrogram 31
3.4.2 Mel-Frequency Cepstral Coefficients (MFCC) 32
3.5 Hyperparameter Optimization using Genetic Algorithm 33
4 Results and Analysis **36**
4.1 Bird sound Detection 36
4.2 Bird Species Classification 37
4.3 Genetic Algorithim for CNN-LSTM and Efficient-Net 37
4.3.1 Genetic Algorithim for CNN-LSTM 38
4.3.2 Genetic Algorithim for Efficient-Net 38
4.4 Mobile Application 38
5 Discussion **44**
5.1 Achievements 44
5.2 Limitations 45
5.3 Future Improvements 45
References **45**

LIST OF FIGURES

1.1 Gantt Chart . . . 5

3.1 System Overview . . . 14

3.2 Block diagram for the working mechanism for Bird Detection . . . 16

3.3 Dataset distribution for freefield1010 . . . 16

3.4 Dataset distribution for warblrb10k . . . 17

3.5 Distribution of the merged dataset . . . 17

3.6 Modified ResNet-50 architecture for Bird Sound Detection. The model processes mel spectrogram inputs, extracts deep features using frozen ResNet-50 layers, and classifies them using a custom fully connected layer. . . . 21

3.7 Block diagram for the working mechanism for Bird Species Identification 26

3.8 Dataset before performing augmentation . . . 27

3.9 Dataset after performing augmentation . . . 28

3.10 Feature Extraction Using Spectrogram and MFCC . . . 32

4.1 Confusion Matrix for Bird Sound Detection Model . . . 36

4.2 ROC Curve of the detection model . . . 37

4.3 GA CNN-LSTM. . . . 38

4.4 GA Efficient-Net. . . . 38

4.5 Home Page of FeatherFind. . . . 39

4.6 Mapped Birds in FeatherFind. . . . 40

4.7 Recording using FeatherFind. . . . 41

4.8 Confirmation for Audio Recognition. . . . 42

4.9 Bird Species Identification using FeatherFind. . . . 43

ABBREVIATIONS

CNN	Convolutional Neural Network
DCNN	Deep Convolutional Neural Network
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
GA	Genetic Algorithm
GPS	Global Positioning System
MAP	Mean Average Precision
MLP	Multilayer Perceptron
RGB	Reg Green Blue
ROC	Receiver Operating characteristics
RNN	Recurrent Neural Network
STFT	Short-Time Fourier Transform

CHAPTER 1
INTRODUCTION

Globally, the avian kingdom is vast, with over 11,000 species, showing nature's complexity and evolutionary skill. This number, from the International Ornithological Committee as of April 2023, shows the great bird diversity, with each species having its own role and story.[1]

In Nepal, a country known for its rich nature and varied ecosystems, from the lowland Terai to the high Himalayas, there are more than 887 bird species, according to the Himalayan Nature organization. This is over 8% of the world's known bird species, a big number given Nepal's small size.[2]

Many of these species are endangered due to habitat loss, climate change, and human activities. The National Red List of Nepal's Birds identifies 168 nationally threatened bird species, including 68 Critically Endangered, 38 Endangered, and 62 Vulnerable species, as detailed in a publication by the Journal of Threatened Taxa.[3]

The situation of these endangered species shows the need for conservation efforts. Technologies such as audio recognition provide new methods for identifying and monitoring bird populations. By analyzing bird sounds and pictures, researchers can better understand species distribution, behavior, and threats. These technologies not only help conserve endangered species but also support broader biodiversity research.

1.1 Problem Statement

In Nepal, a hotspot of avian biodiversity, accurately identifying and classifying bird species, particularly those that are endangered, is a critical yet complex task. Traditional observation methods are limited by the vast geographical and ecological diversity of the region, making it challenging to monitor and protect these birds effectively. The necessity for precise identification is paramount for conservation efforts aimed at maintaining ecosystem balance. To address this, there is a pressing need for a method that can overcome these constraints by leveraging advanced technologies capable of distinguishing between the myriad of bird calls and songs, as well as visual markers through image classification. Such a method promises to automate the identification process, enhancing accuracy and efficiency in monitoring endangered species.

1.2 Objectives

i To develop and implement an integrated technological solution that utilizes advanced audio recognition technique for the accurate identification and monitoring of bird species in Nepal, with an emphasis on endangered species.

1.3 Application Scope

1. **Conservation Efforts:**
   This system will enhance conservation by enabling accurate monitoring of bird populations, helping track endangered species and take a step towards habitat protection.
2. **Biodiversity Monitoring:**
   Automated identification will aid biodiversity monitoring by processing large datasets, helping detect species distribution on bird communities.
3. **Ecological Research:**
   Researchers can use the system to study bird migration, and habitat use, providing crucial data for modeling ecosystems and understanding ecological interactions.
4. **Environmental Education and Awareness:**
   Integrated into educational programs, this tool will raise public awareness about biodiversity and conservation, engaging students and scientists in bird identification.
5. **Bird viewing:**
   Bird enthusiasts will benefit from this system as it will enhance bird watching experiences by providing instant identification of bird species

1.4 Features

1. **Species Identification Using Audio:**
   The app allows users to record bird sounds in real-time using their device's microphone or upload pre-recorded audio files. Advanced noise filtering techniques isolate bird calls from background noise, and sound wave analysis helps in identifying distinct frequency patterns. Machine learning algorithms, trained on a vast database of bird calls, match the recorded sound to identify the bird species accurately.
2. **Mapping Identified Bird Habitat:**
   The app tags the location of identified birds using GPS, providing detailed habitat information typical of each species. Integrating with mapping services, it displays bird sightings on an interactive map, generating heat maps to show species density and distribution. Additionally, it tracks and visualizes bird migration patterns over time, helping users understand seasonal movements.
3. **Provide Description About the Birds:**
   For each identified bird species, the app offers detailed profiles that include scientific and common names, physical descriptions, and conservation status. It also provides audio and visual media for reference, along with information on the bird's behavior, diet, and typical habitats, enriching the user's understanding of the species.

1.5 Feasibility Study

Before implementation of project design, the feasibility analysis of the project must be done to move any further. The feasibility analysis of the project gives an idea on how the project will perform and its impact in the real world scenario. So, it is of utmost importance.

1.5.1 Economic Feasibility

Our system is economically achievable as a result of the development of several tools, libraries, and frameworks. Since all the software required to construct it is free and readily available online, this project is incredibly cost-effective. Only time and effort are needed to create a worthwhile, genuinely passive system. The project doesn't come at a substantial cost. From an economic standpoint, the project appears successful in this sense.

1.5.2 Technical Feasibility

The software needed to implement a project can be downloaded from a wide variety of online resources. Technically speaking, the project is feasible as the necessary software is easily available. We were able to learn the information we required for the project through a variety of online sources, including classes. All the libraries and data are accessible online for free because this project does not require any licensing costs. It is technically possible if one has the necessary information and resources.

1.5.3 Operational Feasibility

The project aims to enhance bird species identification through audio classification, making bird sound recognition more accessible and efficient. This solution is particularly beneficial for ornithologists, bird watchers, and environmental researchers who require accurate and quick identification of bird species based on their calls. The project leverages advanced audio signal processing and deep learning algorithms to classify bird sounds, ensuring high accuracy and reliability. Given the widespread availability of mobile devices and recording equipment, the project is operationally feasible, as it can be easily integrated into existing workflows and tools used by bird enthusiasts and professionals. By providing an efficient method for bird sound classification, the project supports a sizable community interested in avian studies and conservation, ensuring practical applicability and ease of use.

1.5.4 Schedule Feasibility

The workload of the project is divided amongst the project members. The scheduling is done according to an incremental model where different modules are planned to be assigned to the group members.So, the project fulfills the schedule feasibility requirements.

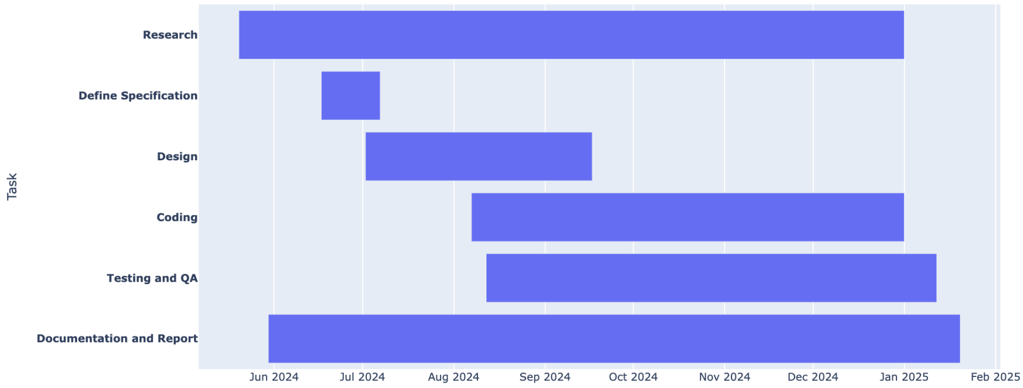

Figure 1.1: Gantt Chart

1.6 System Requirements

1.6.1 Development Requirements

Table 1.1: Development Requirements

Software Requirements	**Hardware Requirements**
Programming Language: Python, Dart, Javascript	RAM: >= 8 GB Megapixels
Design Tools: Figma, Canva, Draw.io	CPU: i5 10th
Libraries: Librosa, PyAudio, Pytorch	GPU: P100 (Recommended)
Framework: Flutter, Django RestFramework	Storage: >= 50 GB
IDEs: VSCode, Android Studio	

1.6.2 Deployment Requirements

Table 1.2: Deployment Requirements

Software Requirements	**Hardware Requirements**
Android: >= 10	RAM: > 4 GB
Read/Write FileSystem	Storage: >= 20 GB
Internet Accessibility	Recording Quality >= 256 Kbps, 48 KHz
Database: Sqlfite	

CHAPTER 2
LITERATURE REVIEW

This literature review explores the progression of methodologies and technologies in the field, with a particular focus on the use of Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks for audio-based bird species identification. The review also examines the challenges associated with dataset quality and diversity, and the innovative strategies employed to address these issues, providing a comprehensive overview of the current state of research and future directions in avian bioacoustics.

2.1 Related Works

BirdNET is a cutting-edge research platform developed through collaboration between the K. Lisa Yang Center for Conservation Bioacoustics at the Cornell Lab of Ornithology and the Chair of Media Informatics at Chemnitz University of Technology. Its primary aim is to detect and classify bird sounds using machine learning technologies, serving both experts and citizen scientists in their efforts to monitor and protect bird populations.

BirdNET can identify around 3,000 of the world's most common bird species, with plans to expand this number. Features such as a live submissions map and a Twitter bot are included to engage the community and share real-time data. The project is supported by donations and collaborations, offering opportunities for researchers and developers to contribute to its growth. BirdNET serves as an invaluable tool for bird enthusiasts, conservationists, and biologists alike, providing innovative solutions for large-scale acoustic monitoring and contributing to the conservation and understanding of avian biodiversity.

The BirdCLEF 2023 competition on Kaggle is a significant data science challenge that falls under the broader LifeCLEF initiative, aimed at pushing the boundaries of species identification and biodiversity monitoring through technological innovation. This particular competition focuses on the development of machine learning models that can identify bird species based on audio recordings. It presents a complex and realistic challenge due to the diversity of the audio recordings, which are collected from various environments and feature a wide range of bird species.

2.2 Related Research

The study[4] ”Audio Classifier for Automatic Identification of Endangered Bird Species of Nepal” focuses on using deep learning techniques to identify endangered bird species from audio recordings. The dataset, collected from xeno-canto.org, comprises 2215 audio recordings of 41 bird species, 38 of which are endangered. This dataset was expanded to 6733 recordings through 10-second audio splitting and Gaussian noise augmentation, with 5407 recordings used for training, 639 for validation, and 687 for testing. The methodology involved handling imbalanced class distribution through data augmentation, employing Mel spectrograms and Mel-Frequency Cepstral Coefficients (MFCCs) for feature extraction, and developing a custom Convolutional Neural Network (CNN) model and an EfficientNet model. The hyperparameters of these models were optimized using a genetic algorithm. The Mel spectrograms were created using Short-Time Fourier Transform, converting amplitudes to decibel scale, and applying Mel filter banks to the spectrograms. Similarly, MFCCs were derived by framing the audio signals, applying Discrete Fourier Transform, logarithmic scaling, Mel scaling, and Discrete Cosine Transform. The EfficientNet architecture utilized compound scaling for network depth, width, and resolution. The findings indicated that the proposed approach achieved satisfactory results in classifying the bird species. Model I, using Mel Spectrogram and EfficientNet, achieved an F1-score of 79%, while Model II, using Mel Spectrogram and Custom CNN, achieved 64%, and Model III, using MFCC and EfficientNet, achieved 72%. However, limitations include the relatively small dataset size and the need for further enhancement in model robustness and accuracy.

Sevilla et al.[5] introduced an innovative approach to bird sound classification with their study ‘Audio Bird Classification with Inception-v4 extended with Time and Time-Frequency Attention Mechanisms’. The datasets employed include various bird sounds, prominently from the BirdClef2017 challenge, consisting of 1500 bird species recordings. The methodology revolves around treating bird sound classification as an image classification problem through transfer learning. The Inception-v4 model, initially pretrained on ImageNet, was adapted to process time-frequency representations of bird sounds by converting these sounds into RGB images using three log-spectrograms generated via fast Fourier transform at different scales (128, 512, 2048 bins).The findings demonstrate that the model, termed 'Soundception', integrates time and time-frequency attention mechanisms effectively, significantly improving classification accuracy. The results highlight Soundception's outstanding performance, achieving a mean average precision (MAP) of 0.714 in classifying 1500 bird species, 0.616 MAP for background species, and 0.288 MAP for soundscapes with time-codes, making it the top model in the BirdClef2017 challenge across multiple tasks. However, limitations include the incomplete convergence of the model due to computational constraints and the extensive GPU resources required for training. The paper concludes with a discussion on future improvements, such as exploring different scalable optimizations and incorporating stacked GRU layers for better audio-to-image representation learning, underscoring the potential of transfer learning from advanced image classification models to acoustic domains.

The study, conducted by Chandu B et al.[6], outlines a robust methodology for identifying bird species from audio recordings, leveraging a combination of meticulously curated datasets and machine learning techniques. The dataset was manually compiledThe working mechanism for bird identification from audio utilizes the dataset compiled from Xeno Canto, housing a diverse collection of avian vocalizations. Our methodology revolves around the utilization of EfficientNet, a state-of-the-art convolutional neural network architecture known for its balance between accuracy and efficiency.

The primary objective of this study is to achieve high levels of accuracy in identifying a broad spectrum of 41 distinct bird species. Here lies the detailed explanation of the methodology, from the working mechanism to model training. from both local recordings and online resources such as xeno-canto.org, which apart from bird songs also contains ambient noise and human voices to simulate real-world conditions. Pre-processing techniques including pre-emphasis, framing, silence removal, and reconstruction were applied to the audio clips to enhance the relevant frequency components and eliminate unnecessary noise, ensuring the purity of the dataset. Spectrograms of these processed clips were generated and used as input for training a convolutional neural network (CNN), specifically AlexNet, chosen for its high accuracy in image classification tasks. Through transfer learning, AlexNet was adapted to recognize bird species from the spectrograms, achieving a classification accuracy of 97% in controlled environments. However, recognizing the variability of real-world conditions, the researchers retrained the model with datasets containing ambient noise, achieving a real-time classification accuracy of 91%. Despite these promising results, the study acknowledges limitations such as the relatively small size of the dataset and the need for further tuning of performance parameters to improve robustness.

In[7] 'An Ensemble of Convolutional Neural Networks for Audio Classification' delves into a comprehensive study on CNN classification using different architectures, data augmentation techniques, and audio signal representations, aimed at enhancing audio classification tasks across various datasets. The study employs three datasets: BIRDZ, CAT, and ESC-50, each offering unique challenges in audio classification. The methodology involves training five convolutional neural networks (CNNs) with four audio representations combined with six different data augmentation methods, resulting in thirty-five subtypes of ensembles. The audio representations include techniques such as the Discrete Gabor Transform (DGT), Waveform Similarity OverLap Add (WSOLA), and Phase Vocoder. The data augmentation methods encompass procedures like short spectrogram augmentation, random time shift, and frequency masking. The CNN architectures are pre-trained models fine-tuned with these augmented datasets to boost classification accuracy. The findings reveal that the ensemble method outperforms standalone networks, achieving 97% accuracy on the BIRDZ dataset, 90.51% on the CAT dataset, andThe working mechanism for bird identification from audio utilizes the dataset compiled from Xeno Canto, housing a diverse collection of avian vocalizations. Our methodology revolves around the utilization of EfficientNet, a state-of-the-art convolutional neural network architecture known for its balance between accuracy and efficiency.

The primary objective of this study is to achieve high levels of accuracy in identifying a broad spectrum of 41 distinct bird species. Here lies the detailed explanation of the methodology, from the working mechanism to model training. 88.65% on the ESC-50 dataset. The study also highlights that the best-performing CNNs are VGG16 and VGG19, with DGT as the most effective signal representation. However, the study acknowledges limitations, such as the computational cost of training ensembles and the variability in performance across different augmentation techniques.

In[8] 'Analysis of bird call datasets sourced from Xeno-Canto', comprising 72,172 samples from 264 bird species in 16-bit wav format with a 16 kHz sampling rate. The methodology involved preprocessing the audio data to filter out low-frequency noise and normalize signal amplitude, followed by generating Mel Spectrograms and Mel-Frequency Cepstral Coefficients (MFCCs) as inputs for deep learning models. The Mel Spectrograms were produced using discrete Fourier transform (DFT), and the MFCCs were derived by applying discrete cosine transform (DCT) to the Mel Spectrogram. The study employed various metrics to evaluate the performance of these methods, including ROC analysis to visualize model effectiveness. Findings indicated that the proposed models showed significant promise in identifying bird species from their calls, with improvements in classification accuracy compared to previous approaches. However, limitations were noted, including potential biases in the dataset due to uneven sample distribution across species and the challenge of background noise affecting signal quality. Future work suggested enhancing noise reduction techniques and exploring more sophisticated neural network architectures to further improve model robustness and accuracy.

The study[9] by Pahuja et al. conducted an in-depth analysis of bird species recognition through acoustic monitoring, utilizing a robust dataset of bird sound samples, meticulously annotated and validated for accuracy. The dataset, referred to as SD, comprises multispecies bird sound recordings, each labeled with species name and sample ID, along with corresponding metadata, providing a comprehensive foundation for model training and evaluation. Methodologically, the research employed a spectrogram-based feature extraction approach, leveraging Short-Time Fourier Transform (STFT) to capture the intricate temporal and spectral characteristics of bird sounds. This was followed by the application of a Multilayer Perceptron (MLP) classifier to distinguish between different bird species. The findings reveal that the proposed model achieved high recognition accuracy, with some species being identified with perfect precision, recall, and accuracy (100%), though the performance varied across species, with a few showing lower recognition rates (86.9%) and precision/recall values ranging between 5075%. The results demonstrated an overall classification accuracy of 96%, with cross-validation accuracy standing at 81.4%, highlighting the model's robustness yet indicating room for improvement in generalizability across diverse datasets. Despite the promising results, the study acknowledges several limitations, including the variability in recognition accuracy among different species and the potential influence of environmental noise on model performance. Future work is suggested to explore feature and model fusion techniques, integrate the model with cloud-based systems for real-time recognition, and expand the dataset to include a broader range of bird species to enhance the model's applicability and accuracy in practical scenarios.

The dataset used in this study[10] comprises recordings labeled by species from California and Nevada, USA. It includes 91 species, with 30 audio samples per species, amounting to a total of 2,730 MP3 files. The methodology adopted involves three main steps: pre-processing, feature extraction, and deep learning modeling. For feature extraction, MFCCs were obtained using the Python library python_speech_features, with parameters such as sample rate, 13 cepstrum coefficients, 26 filterbank filters, and an FFT size of 512. Mel spectrograms were extracted using the Librosa library, employing parameters such as a sample rate, an FFT window size of 2048, a hop length of 512, and 128 Mel bands. In the deep learning modeling, CNNs and LSTMs were compared for their effectiveness in classifying bird sounds. CNNs demonstrated superior training accuracy with Mel spectrogram features, achieving 99.05% and 98.76% accuracy for 3-second and 1.5-second spectroThe working mechanism for bird identification from audio utilizes the dataset compiled from Xeno Canto, housing a diverse collection of avian vocalizations. Our methodology revolves around the utilization of EfficientNet, a state-of-the-art convolutional neural network architecture known for its balance between accuracy and efficiency.

The primary objective of this study is to achieve high levels of accuracy in identifying a broad spectrum of 41 distinct bird species. Here lies the detailed explanation of the methodology, from the working mechanism to model training.grams, respectively. In contrast, LSTMs achieved lower training accuracies of 75.85% and 73.29% under similar conditions.These results highlight the superior ability of CNNs to leverage the spatial and frequency-related patterns in Mel spectrograms for accurate bird species classification.

Finally, in the study[11], 'Acoustic Bird Detection with Deep Convolution Neural Network', DCNNs pretrained on ImageNet, have been used for bioacoustic classification, with preprocessing and augmentation playing crucial roles in model performance. The preprocessing pipeline typically involves applying a shallow high-pass filter (Q = 0.707) with a 2 kHz cutoff, resampling to 22,050 Hz, and extracting around 4-second audio chunks, which are transformed into mel spectrograms with 310 mel bands (160-10,300 Hz). Further processing includes removing extreme frequencies, normalizing power spectrograms to decibel units, resizing to 224*224, and converting grayscale spectrograms to RGB for compatibility with ResNet. To improve generalization, data augmentation techniques such as jittering chunk duration, extracting chunks from random positions, adding noise from unrelated audio files, and applying random amplitude scaling are employed. Additional augmentations in the frequency domain include frequency shifting/stretching, piecewise time/frequency resizing, and color jittering (brightness, contrast, saturation, hue). Among these, the most effective methods are noise addition from random files, piecewise time and frequency stretching, and time interval dropout, which enhance robustness by simulating real-world variations in bird vocalizations.

CHAPTER 3

METHODOLOGY

This chapter describes the overall system including the bird sound detection and bird species classification.

3.1 System Overview

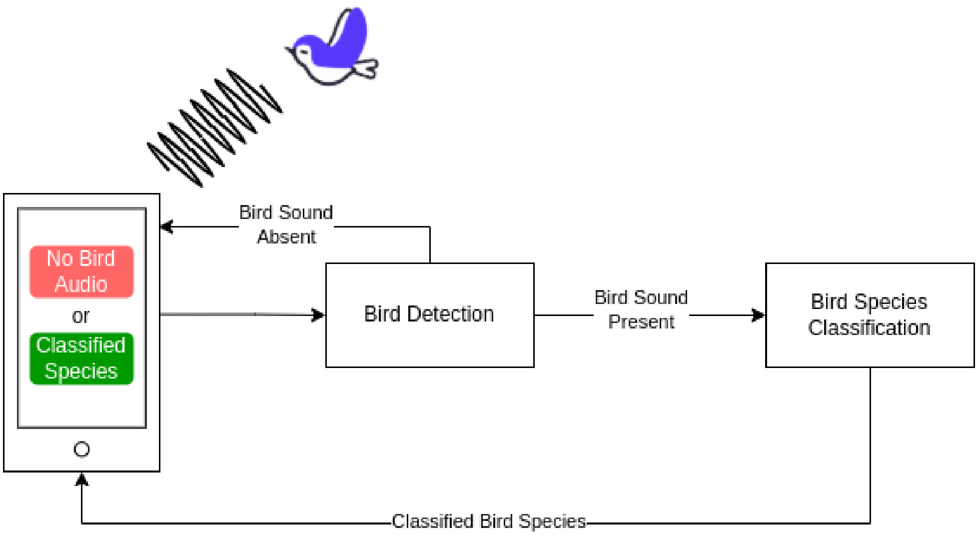

Figure 3.1: System Overview

Our project leverages a mobile application to capture audio recordings, which are then processed to identify bird species. The mobile app records the sound and visualizes it with waveforms during the recording process. Once the audio is captured, it is sent to our backend system for further analysis.

The backend system, built using Django and Django Rest Framework, first passes the audio through a bird sound detection model. This model determines whether the recorded audio contains bird sounds. If bird sounds are detected, the audio is then forwarded to the bird species classifier model. The classifier identifies the specific bird species present in the recording.

Upon successful classification, the system maps the identified bird species to the location where the audio was recorded, utilizing the device's GPS data. This information is stored in a MySQL database, which handles all data management tasks. If the bird sound detection model does not detect any bird sounds, the recorded audio is labeled accordingly, indicating the absence of bird sounds.

This approach ensures that only relevant audio recordings are processed for species classification, enhancing the accuracy and efficiency of the system. The integration of Django, Django Rest Framework, and MySQL provides a robust and scalable backend infrastructure to support the application's functionality as shown in Figure 3.1.

3.2 Working Mechanism for Bird Sound Detection

3.2.1 Data Collection

The dataset used for the Bird Sound Detection Model includes Field recordings, worldwide (”freefield1010”) - a collection of 7,690 excerpts from field recordings around the world, gathered by the FreeSound project, and then standardised for research. This collection is very diverse in location and environment, and for the BAD Challenge, it has been annotated for the presence/absence of birds.

Another dataset used is the Crowdsourced dataset, UK (”warblrb10k”) - 8,000 smartphone audio recordings from around the UK, crowdsourced by users of Warblr, the bird recognition app. The audio covers a wide distribution of UK locations and environments and includes weather noise, traffic noise, human speech, and even human bird imitations.

Instead of using the datasets separately, we merged them to create a single dataset comprising 15,690 audio recordings—7,710 without bird sounds and 7,980 with bird sounds. The merged dataset was then partitioned into training, testing, and validation sets in an 80:10:10 ratio, ensuring a comprehensive and balanced dataset for model training and evaluation.

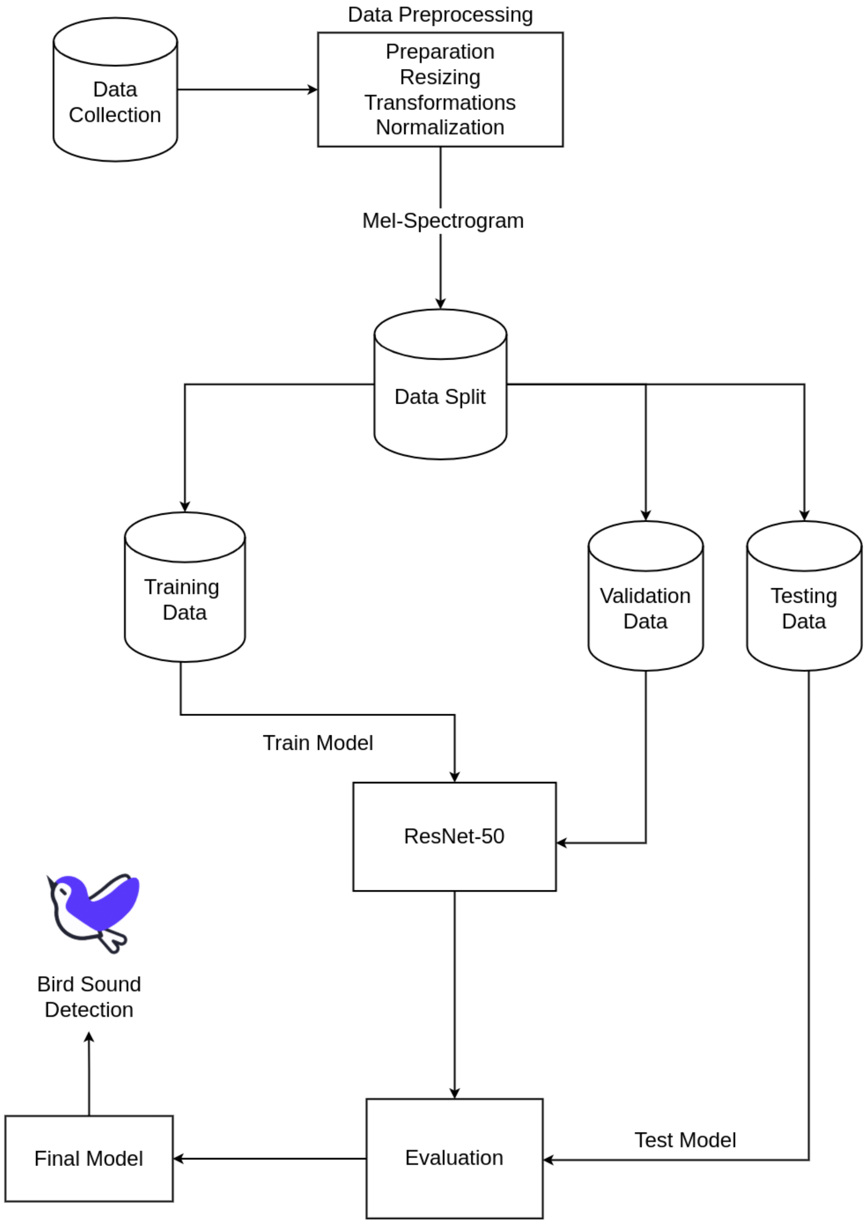

Figure 3.2: Block diagram for the working mechanism for Bird Detection

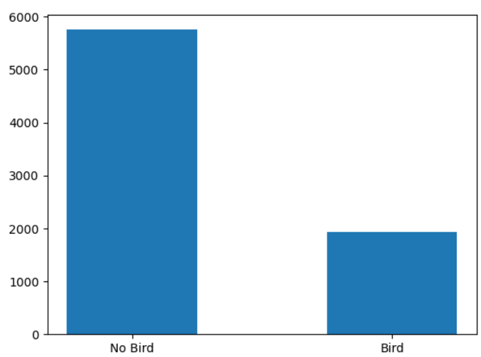

Figure 3.3: Dataset distribution for freefield1010

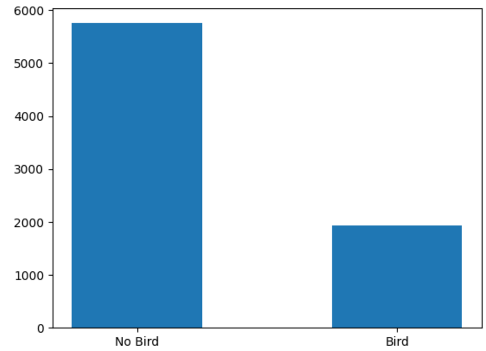

Figure 3.4: Dataset distribution for warblrb10k

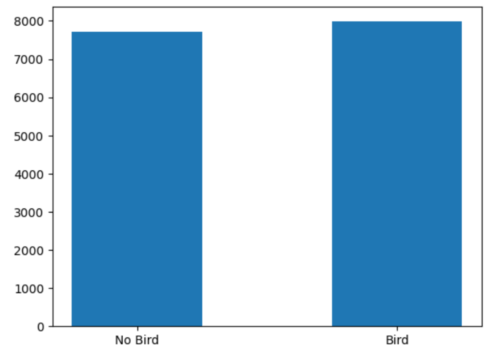

Figure 3.5: Distribution of the merged dataset

3.2.2 Data Transformation

To enhance the robustness and generalization of our bird sound detection model, we applied a series of transformations on both the raw audio signals and their corresponding Mel Spectrograms. These transformations include:

- **Add Gaussian Noise (Raw Audio):**
  This transformation adds Gaussian noise to the raw audio signal. The noise is generated with a specified mean and standard deviation, and it is added to the audio signal with a certain probability. This helps in making the model robust to noisy environments.

$$\text{noise} = \mathcal{N}(\mu, \sigma^2) \tag{3.1}$$

- **Random Volume Scaling (Raw Audio):**
  This transformation scales the volume of the raw audio signal by a random factor within a specified range. The scaling is applied with a certain probability, making

the model resilient to variations in recording levels.

$$\text{audio} = \text{audio} \times \text{gain} \tag{3.2}$$

- **Time Stretch (Raw Audio):**

  This transformation stretches or compresses the time axis of the raw audio signal by a random factor within a specified range. The transformed audio is then converted into a Mel Spectrogram, which helps the model to handle variations in speed.

$$\text{audio} = \text{TimeStretch}(\text{audio}, \text{rate}) \tag{3.3}$$

- **Frequency Masking (Mel Spectrogram):**

  This transformation masks a portion of the frequency bins in the Mel Spectrogram. The maximum number of frequency bins to be masked is specified, and the masking is applied with a certain probability. It encourages the model to focus on different frequency components.

$$\text{mel\_spec}[f : f + \text{max\_mask}] = 0 \tag{3.4}$$

- **Time Masking (Mel Spectrogram):**

  This transformation masks a portion of the time frames in the Mel Spectrogram. The maximum number of time frames to be masked is specified, and the masking is applied with a certain probability. This helps the model to focus on various temporal features.

$$\text{mel\_spec}[:, t : t + \text{max\_mask}] = 0 \tag{3.5}$$

- **Random Content Mixing (Mel Spectrogram):**

  This transformation mixes the Mel Spectrogram with another randomly selected noise Mel Spectrogram. The mixing ratio is chosen randomly within a specified range, and the mixing is applied with a certain probability. This aids in making the model robust to background noises.

$$\text{mel\_spec} = (1 - \text{mix\_ratio}) \times \text{mel\_spec} + \text{mix\_ratio} \times \text{noise\_spec} \tag{3.6}$$

- **Time Interval Dropout (Mel Spectrogram):**
  This transformation randomly drops intervals of time frames in the Mel Spectrogram. The number of intervals and the width of each interval are chosen randomly within specified ranges, applied with a certain probability. This helps the model to cope with missing or corrupted audio segments.

$$\text{mel\_spec}[:, t : t + \text{width}] = 0 \tag{3.7}$$

3.2.3 Feature Extraction

The Mel Spectrogram transforms raw audio signals into a time-frequency image that captures the essential spectral characteristics required for bird sound detection. As demonstrated by Lasseck et al.[11], this approach efficiently distinguishes recordings containing bird vocalizations from those without. For a comprehensive explanation of the computation, please refer to Section **??**.

3.2.4 Model Architecture

The bird sound detection model utilizes a pre-trained ResNet-50 architecture as a feature extractor, which has been modified to process mel spectrogram representations of audio signals. This section provides a detailed explanation of the ResNet-50 architecture and the modifications applied to adapt it for bird sound detection.

3.2.4.1 ResNet-50: A Detailed Overview

ResNet-50, or Residual Network-50, is a deep convolutional neural network (CNN) designed to overcome the vanishing gradient problem that commonly arises in deep networks. It achieves this through the use of residual learning, where shortcut connections (skip connections) allow gradients to bypass certain layers, ensuring efficient training even in very deep architectures.

3.2.4.2 Architecture of ResNet-50

ResNet-50 consists of multiple convolutional layers arranged into residual blocks. These blocks introduce identity mappings that help the model learn transformations efficiently. The network follows a hierarchical structure where initial layers focus on low-level features such as edges and textures, while deeper layers extract high-level representations.

The architecture is composed of four main stages, each containing bottleneck residual blocks. Each block consists of three convolutional layers: a 1×1 convolution for dimensionality reduction, a 3×3 convolution for feature extraction, and another 1×1 convolution to restore dimensionality. By utilizing these bottleneck layers, ResNet-50 maintains computational efficiency while preserving representational power.

Another key component of ResNet-50 is batch normalization, which standardizes feature maps at each layer, leading to stable gradient flow and faster convergence. Additionally, ReLU activation functions introduce non-linearity, allowing the network to learn complex patterns in the data.

3.2.4.3 Components of ResNet-50 Architecture

The ResNet-50 model consists of several essential components that contribute to its powerful feature extraction and classification capabilities:

- **Convolutional Layers:** These layers apply convolution operations to extract spatial features from input images. Early layers capture basic features (edges, textures), while deeper layers learn more abstract patterns.
- **Batch Normalization:** This technique normalizes activations at each layer to stabilize training, improve convergence, and prevent internal covariate shifts.
- **ReLU Activation:** A non-linear activation function that introduces sparsity and helps the network learn complex representations.
- **Residual Blocks:** The defining characteristic of ResNet, these blocks contain skip connections that allow gradients to bypass certain layers, mitigating the vanishing gradient problem.

resnet50_architecture.png

Figure 3.6: Modified ResNet-50 architecture for Bird Sound Detection. The model processes mel spectrogram inputs, extracts deep features using frozen ResNet-50 layers, and classifies them using a custom fully connected layer.

- **Bottleneck Layers:** Each residual block consists of three convolutional layers: a 1×1 convolution to reduce dimensionality, a 3×3 convolution to extract features, and another 1×1 convolution to restore the original depth.
- **Global Average Pooling:** Reduces the dimensionality of feature maps before passing them to the final classification layer, making the network more efficient.
- **Fully Connected Layer:** The final layer of the network that performs classification based on extracted features.
- **Softmax/Sigmoid Activation:** Applies a probability distribution to the output for classification, with softmax used for multi-class classification and sigmoid used for binary classification.

3.2.4.4 Adaptation for Bird Sound Detection

While ResNet-50 is originally designed for image classification, it can be adapted for processing spectrograms, which are visual representations of sound frequencies over time. In this model, several modifications have been applied to tailor ResNet-50 for the bird sound detection task (see Figure 3.6):

- **Input Modification:** Since spectrograms are typically represented as grayscale images with a single channel, the original ResNet-50 architecture, which expects three-channel RGB images, has been adjusted to accept single-channel inputs. This allows the network to effectively process spectrogram data without requiring redundant channels.
- **Feature Extraction:** The pre-trained ResNet-50 model, trained on large-scale image datasets, is used as a feature extractor. The earlier layers remain unchanged, as they capture fundamental patterns useful for various classification tasks. These convolutional layers are frozen, meaning their weights are not updated during training. This strategy ensures that the model benefits from pre-learned visual features while focusing its learning on the final classification task.
- **Modified Classification Layer:** The original fully connected classification layer of ResNet-50 has been replaced with a custom classification head optimized for binary classification (presence or absence of bird sound). This new layer stack introduces additional non-linearity and dropout mechanisms to prevent overfitting, ultimately improving the model's ability to generalize to new data.

3.2.4.5 Training and Generalization

By freezing the early convolutional layers and training only the newly added classification layers, the model efficiently learns to distinguish bird sounds from other environmental noises. This transfer learning approach significantly reduces the training time and the amount of labeled data required, making the model robust in real-world scenarios.

The final model is designed to process mel spectrogram inputs and produce a probability score indicating the presence of bird sounds. The use of residual learning allows for deep feature extraction, making the model highly effective in distinguishing between subtle variations in audio patterns.

3.2.5 Training Process

The training process for the bird sound detection model was carefully designed to ensure optimal learning and generalization. The process incorporated several key components and strategies:

- **Training Parameters:**
  - Number of epochs: 20
  - Batch size: 32
  - Initial learning rate: 0.001
  - Weight decay: 0.0001
- **Loss Function:** Binary Cross-Entropy Loss (BCE) was chosen as the loss function due to its effectiveness in binary classification tasks. For input predictions $\hat{y}$ and true labels y, the BCE loss is calculated as:

$$L_{BCE} = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(\hat{y}_i) + (1 - y_i)\log(1 - \hat{y}_i)] \tag{3.8}$$

- **Optimization:**
  - The Adam optimizer was employed with the following parameters:
    * Learning rate: 0.001
    * Weight decay: 0.0001
  - Learning rate scheduling was implemented using ReduceLROnPlateau with:
    * Reduction factor: 0.2
    * Patience: 3 epochs
- **Monitoring and Validation:**
  - Training and validation losses were monitored each epoch
  - Learning rate adjustments were made based on validation loss plateaus
  - Early stopping was implemented to prevent overfitting

- Model performance was evaluated on the test set after training completion

3.2.6 Evaluation

To assess the effectiveness of our model in distinguishing between the target classes, we analyzed its performance using the Receiver Operating Characteristic (ROC) curve. Following the benchmark established by Lasseck et al.[11] for the freefield1010 and warblrb10k datasets, the Area Under the Curve (AUC) metric was chosen as the primary evaluation criterion.

The ROC curve provides a comprehensive visualization of the model's discriminative ability by plotting the True Positive Rate (TPR) against the False Positive Rate (FPR) at various classification thresholds. For our binary classification task:

- **True Positive Rate (Sensitivity):**

$$TPR = \frac{TP}{TP + FN} \tag{3.9}$$

  Measures the proportion of actual bird sounds correctly identified.

- **False Positive Rate (1 - Specificity):**

$$FPR = \frac{FP}{FP + TN} \tag{3.10}$$

  Represents the proportion of non-bird sounds incorrectly classified as bird sounds.

The Area Under the ROC Curve (AUC) provides a single scalar value representing the model's overall classification performance. An AUC of 1.0 represents perfect classification, while 0.5 indicates random chance performance.

The ROC curve analysis reveals several important aspects of our model's performance:

- **Threshold Independence:** The ROC curve demonstrates the model's performance across all possible classification thresholds, providing a more robust evaluation than single-threshold metrics.

- **Class Imbalance Handling:** Unlike accuracy, the ROC curve remains effective even when dealing with imbalanced datasets, making it particularly suitable for real-world bird sound detection scenarios.
- **Operating Point Selection:** The curve enables the selection of optimal decision thresholds based on specific requirements for sensitivity versus specificity in deployment scenarios.

The high AUC score indicates that our model effectively separates bird sounds from background noise and other environmental sounds, making it reliable for real-world applications in bird sound detection.

3.3 Working Mechanism for Bird Species Classification

3.3.0.1 Working Mechanism

The working mechanism for bird identification from audio utilizes the dataset compiled from Xeno Canto, housing a diverse collection of avian vocalizations. Our methodology revolves around the utilization of EfficientNet, a state-of-the-art convolutional neural network architecture known for its balance between accuracy and efficiency.

The primary objective of this study is to achieve high levels of accuracy in identifying a broad spectrum of 41 distinct bird species. Here lies the detailed explanation of the methodology, from the working mechanism to model training.

3.3.1 Data Collection

For the bird species classification model, we scraped audio data from Xeno-canto (*xeno-canto.org*), a global repository of bird vocalizations contributed by ornithologists and birding enthusiasts. Since the dataset was highly imbalanced, we applied a series of pre-processing and augmentation techniques to ensure a more uniform distribution across bird species.

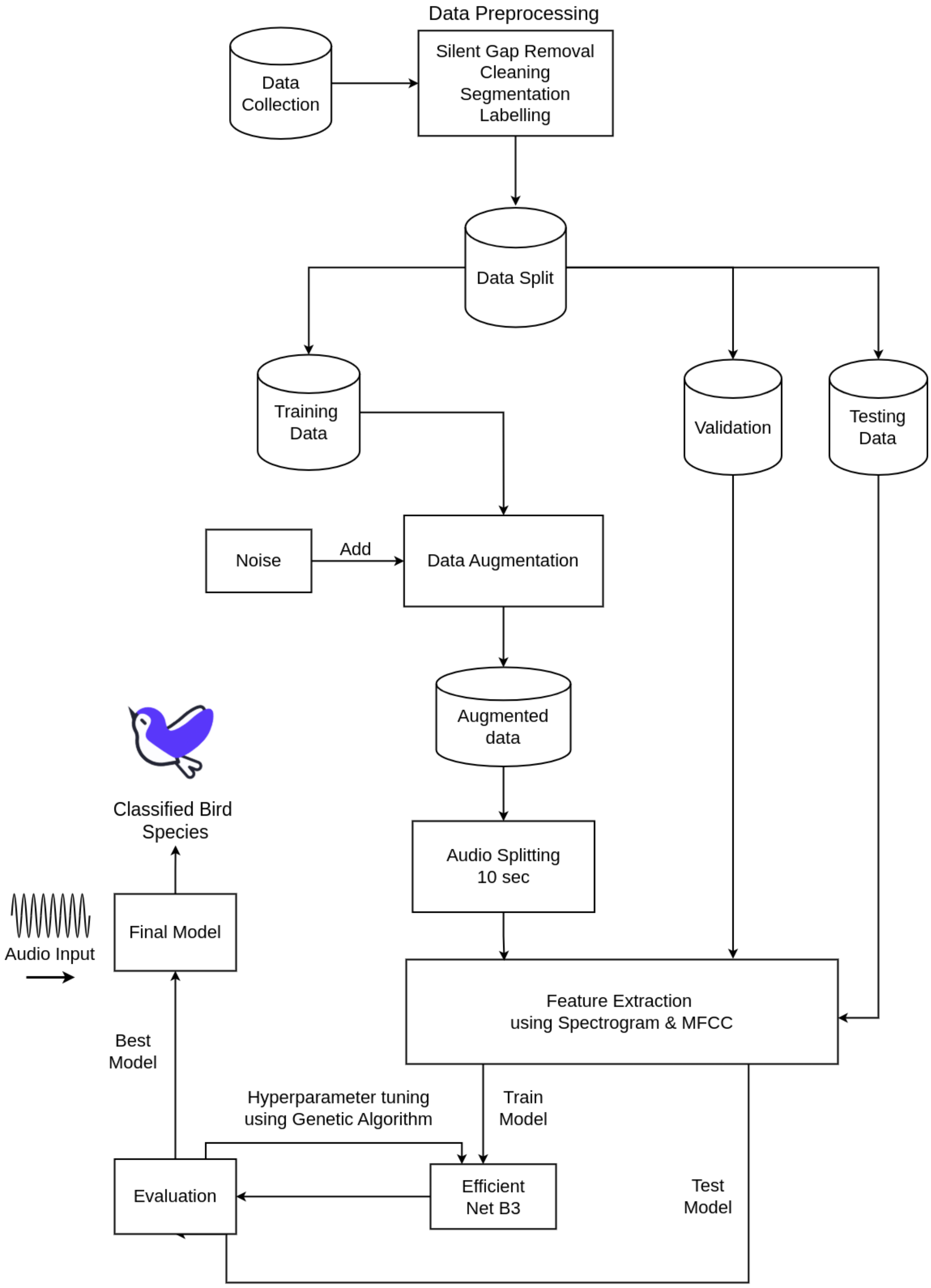

Figure 3.7: Block diagram for the working mechanism for Bird Species Identification

3.3.2 Data Augmentation

Initial analysis of the collected data revealed significant class imbalance, with some species having over 400 recordings while others had fewer than 50. Additionally, the audio recordings varied in length. To address these issues:

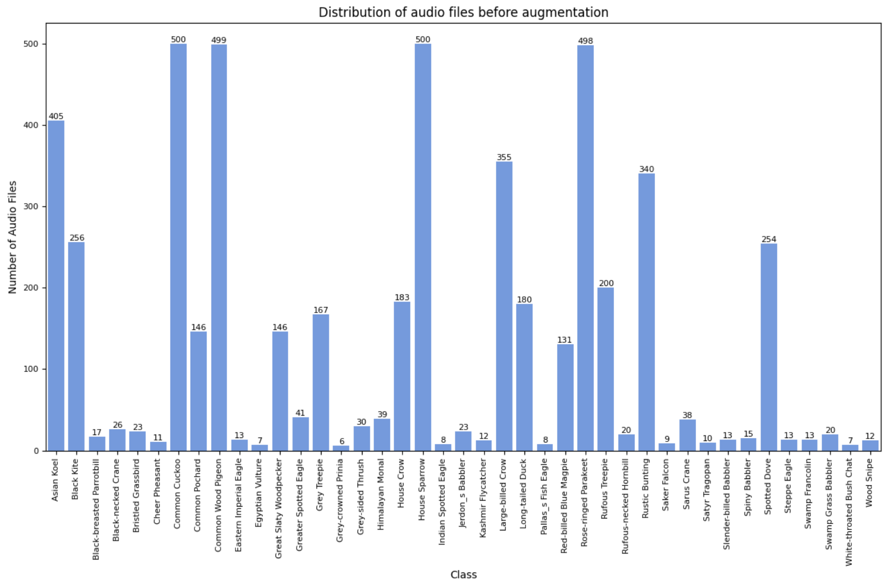

Figure 3.8: Dataset before performing augmentation

- **Audio Clipping:**
  - Recordings were clipped into segments of 10 seconds each.
  - Clips shorter than 5 seconds were discarded to avoid blank audio segments and insufficient data.
  - Clips between 5 and 10 seconds were padded with silence at the end to standardize their length to 10 seconds.
- **Augmentation Techniques:** To balance the dataset, augmentation techniques such as time stretching, phase shifting, and noise addition were applied. The parameters were varied to ensure diversity in the augmented data.
  - Each bird class was augmented to contain exactly 500 audio clips.

3.3.3 Data Preprocessing

The audio recordings were transformed into Mel Spectrograms for further analysis.

- **Conversion Details:**
  - Audio files were converted using a sample rate of 32,000 Hz.
  - The spectrograms were generated with a Hanning window and 48 Mel bands.
- **Dataset Expansion:**
  - Each audio file was converted into corresponding Mel Spectrogram images

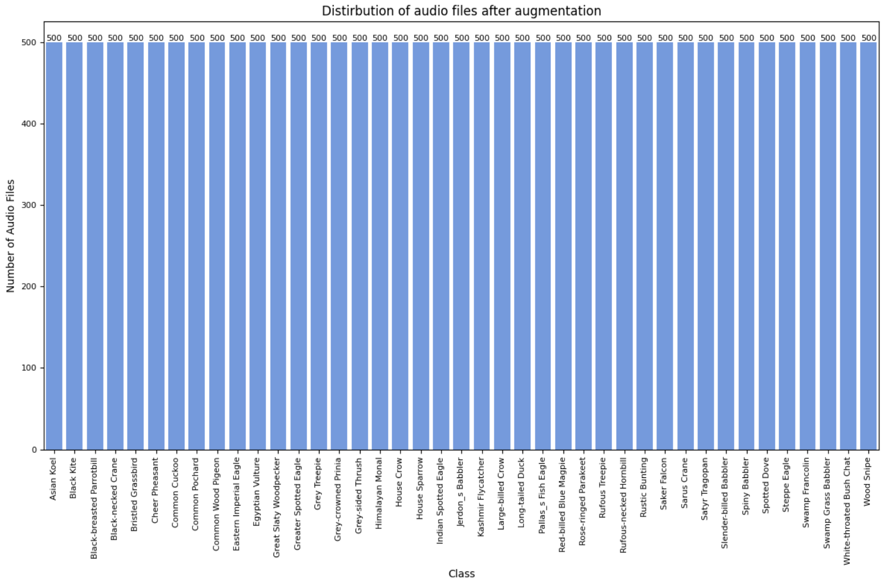

Figure 3.9: Dataset after performing augmentation

for CNN based models and MFCC vectors for LSTM based model.

– The resulting dataset was divided into training, validation, and testing sets for model development.

3.3.4 Feature Extraction

The Mel Spectrogram was employed to transform raw audio recordings into a two-dimensional time-frequency representation, effectively converting sound into an image-like format for processing by convolutional neural networks. This method captures both the temporal and spectral nuances of bird calls, allowing the classifier to discern distinguishing features among various species. As demonstrated in the work of Sevilla et al.[5], such time-frequency representations enhance classification accuracy by emphasizing key frequency components inherent in bird vocalizations.

For a more detailed discussion on the Mel Spectrograms, please refer to the **??** in the Feature Extraction section.

3.3.5 Model Architecture

The proposed model is based on EfficientNetB3, a deep convolutional neural network (CNN) known for its optimized accuracy and efficiency. EfficientNetB3 utilizes the compound scaling method, which uniformly scales network depth, width, and input resolution to achieve better performance with fewer parameters. The backbone of the model consists of Mobile Inverted Bottleneck Convolution (MBConv) blocks, squeeze-and-excitation (SE) modules for adaptive feature recalibration, and Swish activation for smooth gradient propagation.

To leverage pre-trained knowledge, transfer learning was applied by using EfficientNetB3 pre-trained on ImageNet as a feature extractor while fine-tuning specific layers for task-specific learning. The include_top=False parameter was used to remove the original fully connected layers, allowing for a custom classification head to be added.

After feature extraction, the output was flattened using a Flatten layer, converting the multi-dimensional feature maps into a 1D vector. A Dropout layer (0.3) was then applied to randomly deactivate neurons during training, reducing overfitting and improving generalization. Finally, a Dense layer with L2 regularization (λ = 1e-4) was used for classification, ensuring weight penalization to prevent overfitting. A softmax activation function was employed in the final layer to generate probability distributions across the target classes.

This architecture effectively balances feature extraction, regularization, and classification accuracy, making it well-suited for image-based tasks, including fine-grained classification problems.

3.3.6 Training Process

The training process utilizes the **categorical cross-entropy loss function**, which is suitable for multi-class classification problems. The Adam optimizer with a learning rate of 0.0001** was used to update model weights, and the model was trained for **30 epochs** with callbacks to enhance performance and prevent overfitting.

A learning rate scheduler was implemented to adjust the learning rate dynamically. For the first five epochs, the learning rate remains constant, after which it follows an **exponential decay function**. Additionally, **early stopping** was employed, monitoring the validation loss with a patience of **3 epochs**, ensuring that the best model weights are restored. A **model checkpoint** callback saves the best model based on validation loss.

The key hyperparameters used during training are summarized in Table 3.1.

Table 3.1: Training Hyperparameters

Hyperparameter	**Value**
Batch Size	4
Image Size	128×128
Learning Rate	0.0001
Loss Function	Categorical Cross-Entropy
Optimizer	Adam
Epochs	30

3.3.7 Evaluation Metrics

To assess the performance of the proposed model, the following evaluation metrics were used. These metrics provide a comprehensive understanding of the model's classification ability by analyzing both correct predictions and class-wise performance.

- **Accuracy:** Accuracy measures the overall correctness of the model by calculating the ratio of correctly classified samples to the total number of samples.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.11}$$

- **Recall:** Recall (also known as sensitivity or true positive rate) measures the model's ability to correctly identify positive instances.

$$Recall = \frac{TP}{TP + FN} \tag{3.12}$$

- **Precision:** Precision quantifies the proportion of correctly predicted positive instances out of all predicted positive instances.

$$Precision = \frac{TP}{TP + FP} \tag{3.13}$$

- **F1-score:** The F1-score is the harmonic mean of precision and recall, providing a balanced measure when there is an uneven class distribution.

$$F1\text{-}score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.14}$$

3.4 Feature Extraction

Feature extraction was a critical step where the audio clips were transformed into a format that were fed into machine learning models. We used two primary techniques for feature extraction: Mel Spectrograms and Mel-Frequency Cepstral Coefficients (MFCC).

3.4.1 Mel Spectrogram

A spectrogram is a visual representation of the spectrum of frequencies in a sound signal as they vary with time. It is generated by applying the Short-Time Fourier Transform (STFT) to the audio signal, followed by mapping the frequencies to the Mel scale, which better represents human auditory perception. This transformation provides insight into how the frequency content of the signal changes over time.

1. **Short-Time Fourier Transform (STFT):** The STFT is computed by dividing the signal into overlapping frames, applying a window function, and then performing the Fourier Transform on each frame.

$$X(n) = \sum_{m=0}^{N-1} x(m) \cdot w(n - m) \tag{3.15}$$

   where $x(m)$ is the audio signal and $w(n)$ is the window function.
2. **Mel Scale Transformation:** The frequency axis of the spectrogram is then mapped

to the Mel scale using triangular filter banks.

$$M(f) = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{3.16}$$

where $M(f)$ is the Mel frequency, and f is the frequency in Hz.

3. **Log Scaling and Normalization:** Convert Mel spectrogram to a logarithmic scale:

$$S_{\text{log}}(m,n) = \log\left(S_{\text{mel}}(m,n)\right) \tag{3.17}$$

where $S_{\text{log}}(m,n)$ is the log-scaled Mel spectrogram, and $S_{\text{mel}}(m,n)$ is the Mel spectrogram before log transformation.

3.4.2 Mel-Frequency Cepstral Coefficients (MFCC)

MFCCs are coefficients that collectively describe the short-term power spectrum of a sound signal. The process of obtaining MFCCs involves several steps:

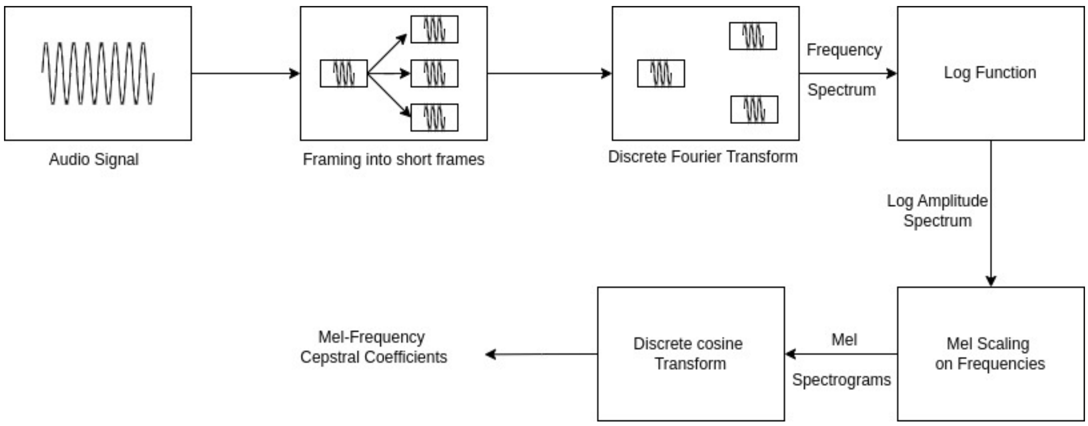

Figure 3.10: Feature Extraction Using Spectrogram and MFCC

1. **Framing:** Divide the audio signal into short frames.
2. **Discrete Fourier Transform (DFT):** Convert each frame to the frequency domain.

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j\frac{2\pi}{N}kn} \tag{3.18}$$

3. **Log Function:** Apply a logarithm to the amplitude spectrum.

$$S_{\text{log}}(k) = \log(|X(k)|) \tag{3.19}$$

4. **Mel-Scaling:** Map the frequencies to the Mel scale, which better represents how humans perceive sound.

$$f_{\text{mel}} = 2595 \cdot \log_{10}(1 + \frac{f}{700}) \tag{3.20}$$

5. **Discrete Cosine Transform (DCT):** Convert the Mel spectrum to the cepstral domain, yielding the MFCC features.

$$C(n) = \sum_{k=0}^{K-1} S_{\text{mel}}(k) \cdot \cos\left(\frac{\pi n(k + 0.5)}{K}\right) \tag{3.21}$$

3.5 Hyperparameter Optimization using Genetic Algorithm

Genetic algorithms are a powerful method for optimizing hyperparameters in machine learning models. Genetic Algorithm have proven to significantly improve the performance metrics of the CNN model instead of using hand tuned approach for hyperparameters. This section outlines the steps involved in using GAs for hyperparameter optimization[12].

- **Encoding the Hyperparameters**
  - Hyperparameters are represented as a chromosome, where each hyperparameter is a gene in the chromosome.
  - For example, in a neural network, a chromosome might include genes for the learning rate, number of layers, number of neurons per layer, and activation functions.
- **Initial Population**
  - An initial population of chromosomes is generated randomly, with each chromosome representing a different set of hyperparameters.
- **Fitness Function**
  - A fitness function is defined to evaluate the performance of each set of hy-

perparameters.

  - This typically involves training the model with the given hyperparameters and measuring its performance on a validation set.

- **Selection**
  - Selection involves choosing the best-performing chromosomes to serve as parents for the next generation.
  - Various selection methods can be employed, such as tournament selection, roulette wheel selection, or rank-based selection.

- **Crossover (Recombination)**
  - Crossover combines pairs of parent chromosomes to produce offspring for the next generation.
  - This is done by swapping segments of parent chromosomes to create new chromosomes, thereby combining features of both parents.
- **Mutation**
  - Mutation introduces random changes to some of the genes in the offspring chromosomes.
  - This helps maintain genetic diversity in the population and allows the algorithm to explore a broader search space.
- **Replacement**
  - The current population is partially or entirely replaced with the new generation of chromosomes, ensuring that better solutions are carried forward while allowing for exploration of new possibilities.
- **Termination**
  - The process of selection, crossover, mutation, and replacement is repeated until a termination criterion is met.
  - This could be a set number of generations, convergence of fitness scores, or achieving a satisfactory performance level.
- **Best Solution**
  - The best chromosome at the end of the process represents the optimal or near-optimal set of hyperparameters for the model.

CHAPTER 4
RESULTS AND ANALYSIS

In the course of our project on Bird Species Classification from Audio, we have accomplished several key tasks to progress our analysis and model development:

4.1 Bird sound Detection

The FeatherFind project has made significant progress in developing an advanced bird species identification system using audio recordings. Datasets used for the identification of bird sound i.e the presence and absence of bird sounds in the audio are ("freefield1010") dataset as shown in figure() for training and ("warblrb10k") dataset as shown in figure ()

The combined dataset leverages the diverse environmental conditions and noise variations from both sources, enhancing the model's robustness and generalization capabilities. After the model was subjected to the testing dataset, the following confusion matrix was obtained, yielding an accuracy of **87.28%**. for testing. To assess the effec-

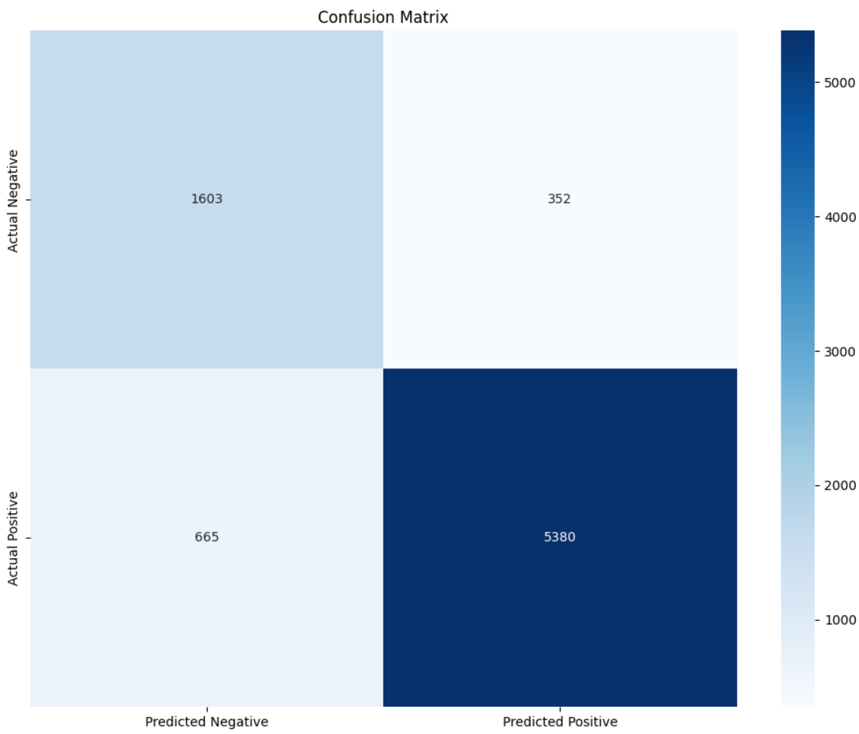

Figure 4.1: Confusion Matrix for Bird Sound Detection Model

tiveness of our model in distinguishing between the target classes, its performance using the Receiver Operating Characteristic (ROC) curve was analaysed. The ROC curve provides a comprehensive view of the trade-off between the true positive rate (TPR) and the false positive rate (FPR) at various threshold settings. The Area Under the Curve (AUC) metric, derived from the ROC curve, quantifies the overall discriminatory power of the model, with a higher AUC indicating better performance.

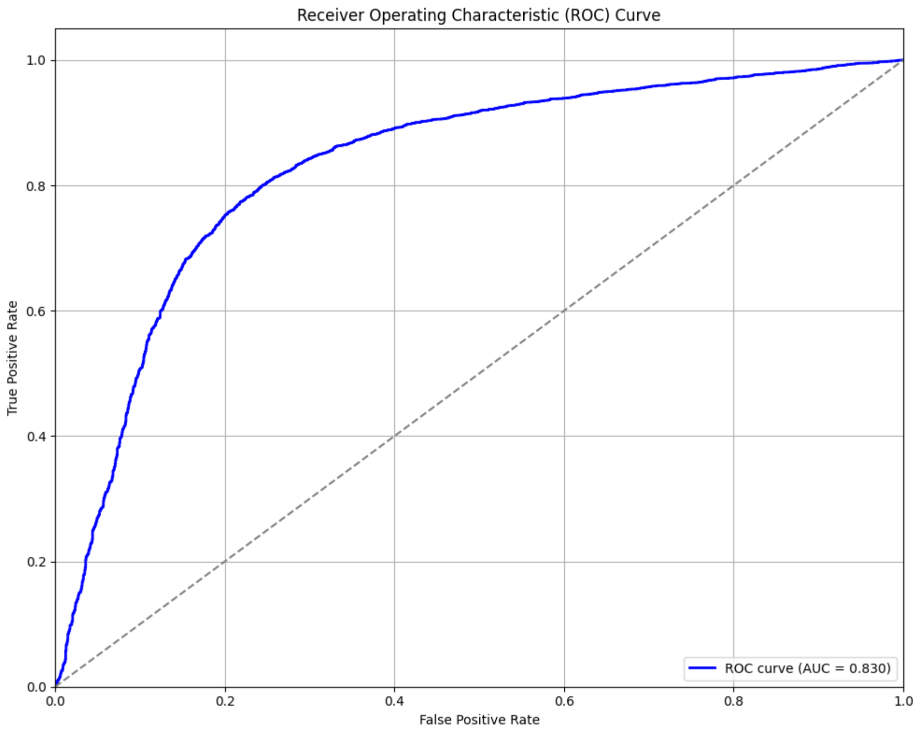

Figure 4.2: ROC Curve of the detection model

4.2 Bird Species Classification

4.3 Genetic Algorithim for CNN-LSTM and Efficient-Net

To optimize the performance of our model, we used genetic algorithm for hyperparameter tuning. This approach helped us systematically explore and identify the most effective hyperparameter settings for our model, including network architecture parameters like learning rates, batch sizes, images sizes, dropout and others. The genetic algorithm enabled a more efficient and comprehensive search for optimal configurations.

4.3.1 Genetic Algorithim for CNN-LSTM

For the CNN-LSTM model, GA was applied to tune critical hyperparameters that impact both convolutional feature extraction and sequence modeling. The key hyperparameters optimized includes Batch Size, Learning Rate, Number of Epochs, Dropout Rate, LSTM Units and Dense Units. The genetic algorithm was executed over multiple generations, evolving hyperparameter configurations towards optimal performance. The best model achieved a Test Accuracy of **75.17%** and Test Loss of **1.0235**, demonstrating improved learning efficiency and reduced overfitting.

images/.png

Figure 4.3: GA CNN-LSTM.

4.3.2 Genetic Algorithim for Efficient-Net

For EfficientNet, GA was used to fine-tune hyperparameters that directly affect feature extraction and classification efficiency. The optimized parameters includes Batch Size, Learning Rate, Number of Epochs, Dropout Rate,Regularization and Dense Units Through multiple generations, the genetic algorithm selected the best-performing hyperparameters, leading to a final Test Accuracy of **93.37%** and Test Loss of **0.3056**, outperforming baseline configurations.

images/.png

Figure 4.4: GA Efficient-Net.

4.4 Mobile Application

The mobile application allows the user to record the audio, visualize it with waveforms during recording, before sending the actual audio data to the server for processing. The user can also use the application to add their current location to the map.The homepage of the application allows the user to view the birds along with the audio that are available in FeatherFind's database. The map page in FeatherFind allows the user to view all the birds that has been located using the application and view their location in the maps

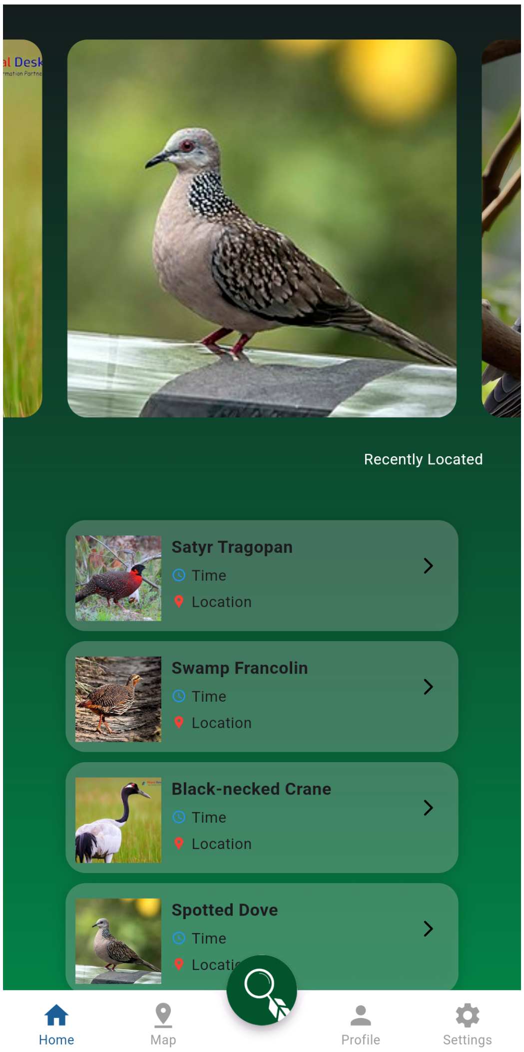

Figure 4.5: Home Page of FeatherFind.

embedded in the application alone with the detailed description of the birds.

For the audio recognition the user can , record the audio in the application and as the audio is being recorded the app visualizes the audio in appropriate waveform. When the user stops recording the audio, the user is prompted with a confirmation dialog, which can be used to process the audio and recognize it.

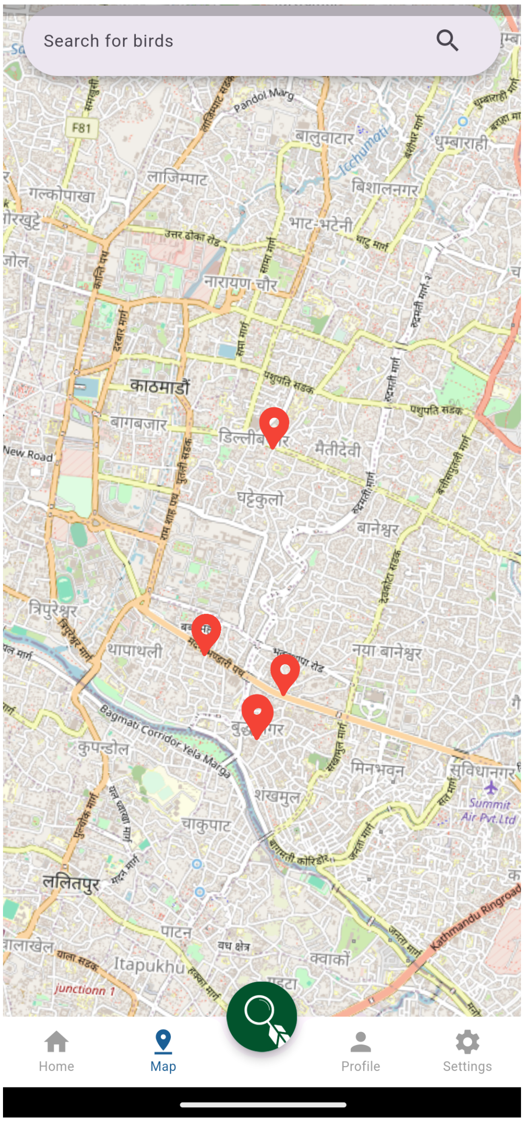

Figure 4.6: Mapped Birds in FeatherFind.

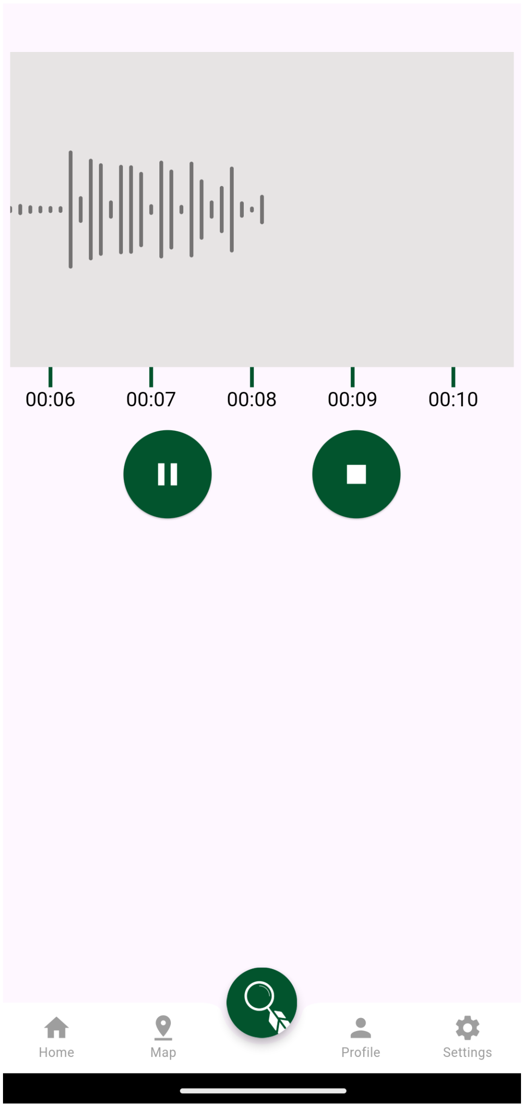

Figure 4.7: Recording using FeatherFind.

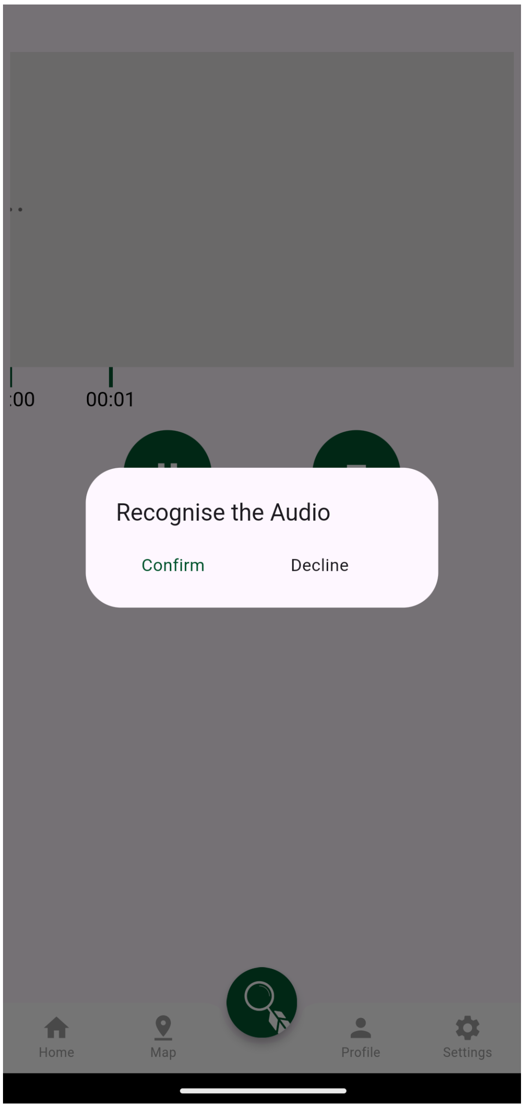

Figure 4.8: Confirmation for Audio Recognition.

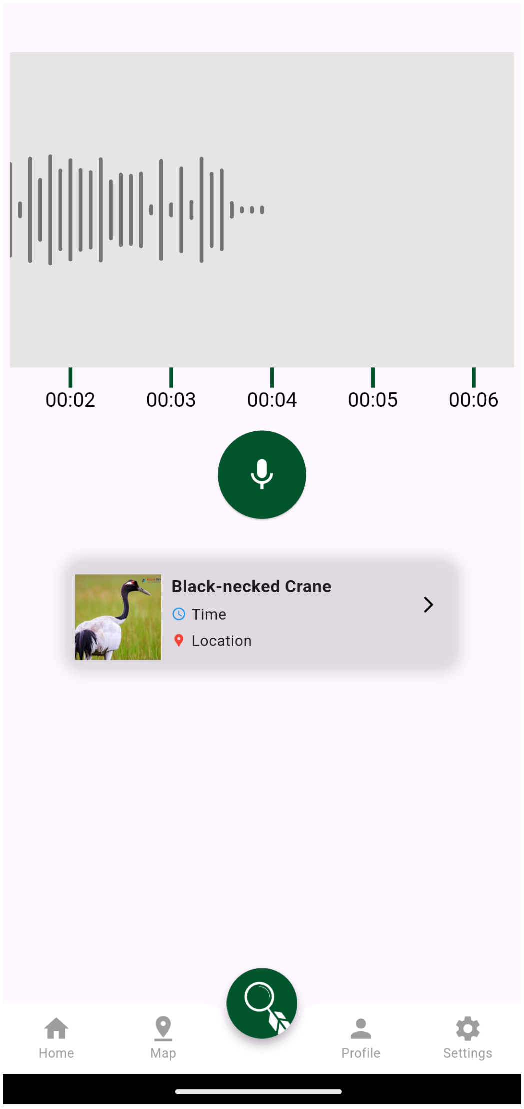

Figure 4.9: Bird Species Identification using FeatherFind.

CHAPTER 5
DISCUSSION

The FeatherFind project has made significant progress in developing an advanced species identification using audio recordings.Some the key achievements of the project include:

5.1 Achievements

- Comprehensive Data Collection: The project ssuccessfully comiled an extensive dataset of d sounds by scraping Xeno-Canto, ensuring diversity in species representaion.
- Advanced Data Processing Techniques: The project utilized advancd audio processing techniques along with data augmentaion(time stretching,phase shifting, gradient noiise addition), were implemented to balance dataset and enhance model robustness.
- Feature Extraction and Model Training: Spectrograms and Mel-Frequency Cepstral Coefficients (MFCCs) were employed for feature extraction. EfficientNetB3 architecture was utilized for bird species classification, achieving an accuracy of 89.36% on the test dataset. Avolutional Neural Network (CNN) combined with Long ShoTerm Memory (LSTM) networks was used to capture both spatial ad temporal features.
- Genetic Algorithm for Hyperparameter Optimization: The model's performance was fine-tuned using genetic algorithms, ensuring optimal parameter selection for improved accuracy.
- Bird Sound Detection Model: A preliminary model was trained to verify the presence of bird sounds before classification, using the InceptionV3 network.AUC score of 83% and an accuracy of 87.28% was achieved.
- Deployment and Mobile App Development: The classification model was deployed on Huggingface Spaces for seamless API integration.A mobile application was developed to allow users to record and analyze bird sounds in real-time while tagging locations using GPS integration.

5.2 Limitations

5.3 Future Improvements

REFERENCES

[1] I. O. Union, "Ioc world bird list," https://www.worldbirdnames.org/new/updates/, 2024, accessed: 2024-06-04.

[2] H. Nature, "National red list of nepal's birds," https://www.himalayannature.org/works/projects/national-red-list-of-nepals-birds/, 2024, accessed: 2024-06-03.

[3] C. Inskipp, H. S. Baral, T. Inskipp, A. P. Khatiwada, M. P. Khatiwada, L. P. Poudyal, and R. Amin, "Nepal's national red list of birds," *Journal of Threatened Taxa*, vol. 9, no. 1, pp. 9700–9722, 2017.

[4] R. Gautam, B. Khatiwada, B. P. Subedi, N. Duwal, and K. C. Dahal, "Audio classifier for automatic identification of endangered bird species of nepal," 2023.

[5] A. Sevilla and H. Glotin, "Audio bird classification with inception-v4 extended with time and time-frequency attention mechanisms." *CLEF (Working Notes)*, vol. 1866, pp. 1–8, 2017.

[6] B. Chandu, A. Munikoti, K. S. Murthy, G. Murthy, and C. Nagaraj, "Automated bird species identification using audio signal processing and neural networks," in *2020 International Conference on Artificial Intelligence and Signal Processing (AISP)*. IEEE, 2020, pp. 1–5.

[7] L. Nanni, G. Maguolo, S. Brahnam, and M. Paci, "An ensemble of convolutional neural networks for audio classification," *Applied Sciences*, vol. 11, no. 13, p. 5796, 2021.

[8] H. Wang, Y. Xu, Y. Yu, Y. Lin, and J. Ran, "An efficient model for a vast number of bird species identification based on acoustic features," *Animals*, vol. 12, no. 18, p. 2434, 2022.

[9] R. Pahuja and A. Kumar, "Sound-spectrogram based automatic bird species recognition using mlp classifier," *Applied Acoustics*, vol. 180, p. 108077, 2021.

[10] S. Carvalho and E. F. Gomes, "Automatic classification of bird sounds: using mfcc and mel spectrogram features with deep learning," *Vietnam Journal of Computer Science*, vol. 10, no. 01, pp. 39–54, 2023.

[11] M. Lasseck, “Acoustic bird detection with deep convolutional neural networks.” in *DCASE*, 2018, pp. 143–147.

[12] A. Reiling, W. Mitchell, S. Westberg, E. Balster, and T. Taha, “Cnn optimization with a genetic algorithm,” in *2019 IEEE National Aerospace and Electronics Conference (NAECON)*, 2019, pp. 340–344.