# Comparative Analysis of Machine Learning Methods for Bird Call Recognition: Evaluating Feature Processing and Model Performance

**Caroline Zhang**
University of California, San Diego
La Jolla, California
caz020@ucsd.edu

**Yuan Zhang**
University of California, San Diego
La Jolla, California
yuz318@ucsd.edu

**Zhoutianning Pan**
University of California, San Diego
La Jolla, California
z6pan@ucsd.edu

**Shuyu Wang**
University of California, San Diego
La Jolla, California
shw043@ucsd.edu

## 1 Introduction

Automated bird call classification is an important task in bioacoustics, enabling researchers and conservationists to monitor avian populations efficiently. Traditional methods of identifying bird species through manual audio analysis are time-consuming and require expert knowledge. Machine learning offers a scalable solution by automatically recognizing bird species from audio recordings.

This study evaluates the performance of different machine learning models for bird call classification, focusing on five species: *Barn Swallow (barswa), Common Sandpiper (comsan), Western Yellow Wagtail (eaywag1), Willow Warbler (wlwwar), and Thrush Nightingale (thrnig1).* The dataset consists of audio recordings from *xeno-canto.org*, preprocessed into structured numerical representations. Feature extraction techniques, including Mel-Frequency Cepstral Coefficients (MFCCs), spectral features, and rhythm-based attributes, are applied to capture distinct vocal patterns. Multiple classification models, including XGBoost, Random Forest, SVM, Decision Tree, and Logistic Regression, are trained and evaluated based on their accuracy and robustness.

In addition to traditional models, this study also explores the use of deep learning approaches, particularly Convolutional Neural Networks (CNNs), which utilize Mel spectrograms as input to learn temporal and spectral patterns directly from raw audio data. By comparing traditional feature-based methods with end-to-end deep learning pipelines, the study aims to identify not only the most accurate but also the most scalable and computationally efficient approach for bird call classification. The goal is to evaluate and compare the effectiveness of both paradigms—traditional machine learning and deep learning—in the context of automated bird sound recognition.

## 2 Literature Survey

Bardeli et al. (2010) [2] addressed challenges in detecting bird sounds in noisy environments, proposing a method using temporal patterns and frequency analysis to improve recognition. Their study highlighted bioacoustic monitoring's role in tracking bird populations, though background noise and overlapping sounds remained obstacles. The 2023 BirdCLEF competition [5] on Kaggle tasked researchers with developing machine learning models for bird call detection in continuous soundscapes with limited training data.

Preprint. Under review.

Processing audio data is essential for training machine learning models to classify bird species. Raw audio contains noise and redundant information that can reduce accuracy. Preprocessing techniques such as noise reduction, segmentation, and feature extraction refine input data for better model interpretability. Feature extraction converts raw waveforms into structured representations, capturing meaningful patterns for classification. Here, we mainly use python package *librosa* [6] to processing audio files into images.

Traditional methods include Mel-Frequency Cepstral Coefficients (MFCCs) [3], Perceptual Linear Prediction (PLP) [4], and chroma features for tonal analysis [7]. These serve as strong baselines for bird call recognition. More recent approaches, such as optimized MFCC parameters [1] and multi-feature selection with ensemble learning [10], further enhance classification accuracy.

Sudha et al. (2018) used Random Forest (RF) with 2D-supervised segmentation to address overlapping bird calls, achieving 83.96% accuracy [8]. Tang et al. (2024) compared RF, SVM, and XGBoost, finding XGBoost the most accurate (83.65%), though RF performed better in varying conditions [9]. Zhang et al. (2023) integrated XGBoost with deep transfer learning (VGG16) for feature extraction, enhancing bird call classification [13].

# 3 Methodology

Effective bird species classification requires a balanced approach to data selection, feature extraction, and model optimization. Given the computational challenges of processing a large dataset, we first analyzed its distribution and implemented a pre-selection strategy to focus on the most representative data. We aim to enhance bird species classification accuracy in real-world, noisy environments by combining traditional feature engineering, ensemble learning methods, and deep learning techniques. The advanced approach employed Mel Spectrograms with Convolutional Neural Networks (CNNs) and EfficientNet. The following sections provide a detailed description of the extracted features and the models used for classification.

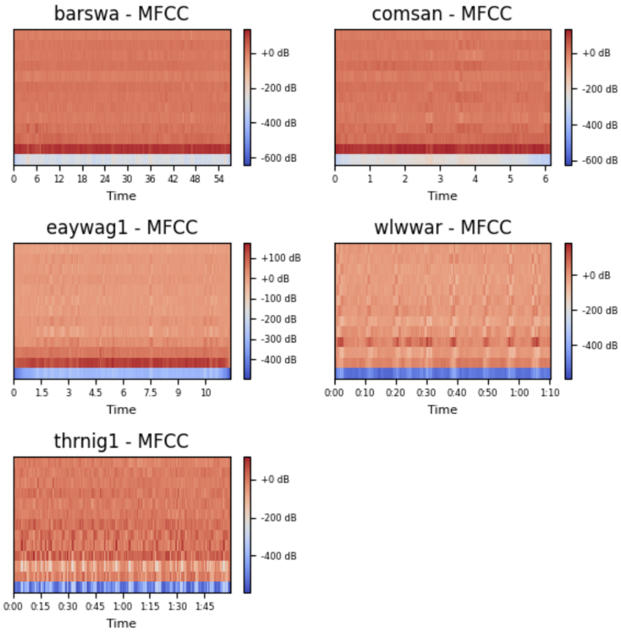

Figure 1: MFCCs graph for Top Five Species in the train_data.csv

## 3.1 Dataset Description

The BirdCLEF 2023 dataset serves as the basis for developing a machine learning model to identify bird species based on their calls. The training dataset used in this study was sourced from the Xeno-Canto [11] community and comprises short audio recordings of individual bird calls, covering

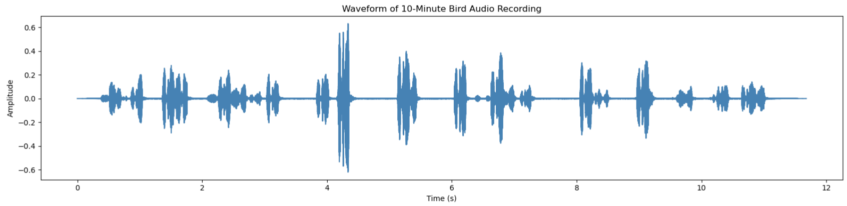

Figure 2: Waveform of audio file

multiple species relevant to the Kenyan soundscape. The dataset includes over $16,900$ recordings across $264$ species, with all audio of $0-10$ minutes long as illustrate in Figure 2.

The original dataset comprises $264$ species, but we encountered computational bottlenecks when applying a naïve feature extraction algorithm for audio processing. Without utilizing the `TensorFlow` package, the processing time for each species was significantly prolonged. To address this efficiency issue, we implemented a pre-selection strategy, sorting the dataset based on the number of available audio samples per species. We then selected the top $5$ species with the highest sample counts to evaluate and compare the performance of different machine learning models in a more controlled and efficient manner.

## 3.2 Feature Engineering + ML

We extract audio files from the dataset using the `librosa` library, loading each file from the `train_audio/` directory at a sampling rate of 16 kHz. To facilitate efficient storage and processing, we convert the extracted audio into `numpy` arrays. This enables streamlined manipulation and analysis for subsequent feature extraction and model training.

### 3.2.1 Feature Selection

To analyze bird call recordings, we extract a set of audio features from the `numpy` arrays, capturing various acoustic characteristics. Our feature set includes Rhythm Features, Pitch Features, Spectral Features, Zero-Crossing Rate (ZCR), and MFCCs. The Spectral Features subset consists of Spectral Centroid, Spectral Bandwidth, Spectral Rolloff, and Spectral Flatness, while Pitch Features include statistical measures such as mean, median, standard deviation, and percentiles of pitch values. We iterate through all extracted `numpy` arrays, compute these features, and store them in a structured DataFrame, organized by species.

- Rhythm Features: Capture patterns in amplitude variations over time, representing the temporal structure of bird calls.
- Pitch Features: Measure the fundamental frequency characteristics, including statistical summaries like mean, median, standard deviation, and percentiles.
- Spectral Features: Describe the frequency distribution of the signal, including Spectral Centroid (center of mass of the spectrum), Spectral Bandwidth (range of frequencies present), Spectral Rolloff (frequency below which most energy is concentrated), and Spectral Flatness (tonal vs. noise-like quality).
- ZCR: Counts the number of times the signal crosses zero amplitude, indicating noisiness or percussiveness of the audio.
- MFCCs: Represent the spectral envelope of the audio signal using a perceptually motivated frequency scale, commonly used in speech and bioacoustic analysis.

### 3.2.2 Model Implementation

We first implemented a multi-class classification framework to classify bird species based on their audio feature datasets using multiple supervised learning models. The dataset was created by merging five species-specific feature datasets, with categorical species labels encoded using LabelEncoder. Preprocessing steps included handling missing values via mean imputation, standardizing numer-

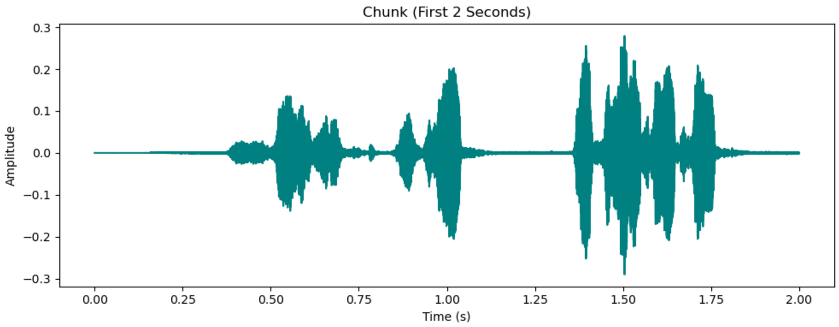

Figure 3: Segmentation of audio file

ical features using StandardScaler, and performing stratified train-test splitting to preserve class distribution.

We trained four machine learning models: Random Forest, Support Vector Classifier (SVC), Decision Tree, and Logistic Regression, with hyperparameters set for optimized performance. Each model was evaluated using accuracy, classification reports, and confusion matrices, with performance results visualized using Seaborn heatmaps and bar plots.

We also employed an XGBoost-based One-vs-All classification framework to distinguish between five bird species based on their audio features. The dataset consists of feature representations extracted from bird audio recordings, with each instance labeled by its corresponding species. We preprocessed the data by standardizing feature values using StandardScaler and ensuring proper numerical formatting of attributes such as tempo. For each species, a separate XGBoost classifier was trained in a One-vs-All setting, where the target variable was binarized with $1$ for the species of interest, $0$ for all others. The dataset was split into training ($80\%$) and testing ($20\%$) sets using stratified sampling to maintain class distribution. Each classifier was configured with $100$ estimators, a learning rate of $0.1$, and a maximum depth of $5$. Performance was evaluated using accuracy $Accuracy$ and generate a classification report, with results reported for each species.

### 3.3 Advanced Method

In addition, we developed a deep learning framework that converts raw audio files into small chunks and Mel spectrograms for improving bird species classification accuracy. The advanced method has been commonly used in BirdCLEF 2024 [12], however, many of them doesn't extend this method to the previous competition dataset.

#### 3.3.1 Data Preprocessing

The data processing process for bird audio recognition starts with organizing the dataset into structured folders to ensure that each species is correctly labeled and stored systematically. This structure facilitates efficient data processing and retrieval throughout the pre-processing and training phases. Given the nature of the raw audio recordings, which often contain noise and silent time periods, a comprehensive preprocessing approach was used to improve the quality and reliability of the dataset. Audio files are segmented into fixed-length segments to standardize the input length and make it suitable for subsequent feature extraction and model training as shown in Figure 3. During the segmentation process, efforts are made to reduce background noise, normalize amplitude levels, and resample the audio to a consistent sample rate to ensure consistency across all samples. In order to convert the audio data into a format more suitable for deep learning models, each processed segment is subjected to feature extraction via ML spectrogram transformation. This transformation captures important frequency and temporal patterns, which are crucial for distinguishing between different bird species. The Mel spectrogram was chosen because of its ability to mimic human auditory perception of pitch and frequency distribution. In addition, the spectrogram was adapted to a fixed dimension in order to maintain consistency and facilitate efficient processing by a convolutional neural network. Once the feature extraction process was complete, the species labels were encoded into numerical values that allowed the model to effectively interpret these labels. The dataset is then divided into a training subset and a test subset to evaluate the generalization and performance of the model. To account for potential class imbalances, class weighting was considered to ensure that underrepresented species were not masked during model training.

The final dataset, consisting of ML spectrogram representations and their corresponding labels, is the basis for training deep learning models capable of recognizing bird species based on sound.

### 3.3.2 Model Implementation

We developed the CNN architecture using sequential Conv2D layers with ReLU activation and L2 regularization to reduce overfitting. After each convolutional layer, we applied MaxPooling to reduce spatial dimensions and highlight the most relevant features.

To further prevent overfitting, we included a Dropout layer before fully connected dense layers that consolidate the extracted features. The final layer employed a softmax activation function for multi-class classification.

We compiled the CNN model with the Adam optimizer and trained it using sparse categorical cross-entropy loss. Additionally, we incorporated learning rate scheduling and early stopping to optimize training performance. Model evaluation was performed using accuracy metrics for each species.

## 4 Result

Table 1: Comparison of Model Performance on Different Species Based on Accuracy

| Model | Barswa | Comsan | Eaywag | Thrnig | Wlwwar |
|---|---|---|---|---|---|
| Random Forest | 0.50 | 0.65 | 0.48 | 0.70 | 0.55 |
| SVC | 0.55 | 0.70 | 0.44 | 0.69 | 0.58 |
| Decision Tree | 0.39 | 0.52 | 0.34 | 0.64 | 0.49 |
| Logistic Regression | 0.54 | 0.67 | 0.47 | 0.69 | 0.58 |
| CNN | 0.84 | 0.80 | 0.79 | 0.85 | 0.81 |
| Efficient Net | 0.83 | 0.81 | 0.77 | 0.82 | 0.78 |

Table 2: Experimental Results of One-vs-All XGBoost Classification for Bird Species

| Species | Accuracy | Precision | Recall | F1-Score | Macro Avg F1 | Weighted Avg F1 |
|---|---|---|---|---|---|---|
| Barswa | 0.8340 | 0.64 | 0.38 | 0.48 | 0.69 | 0.82 |
| Comsan | 0.8820 | 0.83 | 0.52 | 0.64 | 0.78 | 0.87 |
| Eaywag | 0.8240 | 0.62 | 0.31 | 0.41 | 0.65 | 0.80 |
| Wlwwar | 0.8420 | 0.66 | 0.44 | 0.53 | 0.72 | 0.83 |
| Thrnig | 0.8980 | 0.84 | 0.61 | 0.71 | 0.82 | 0.89 |

As shown in Table 2, the XGBoost classifier achieved the highest accuracy for thrnig (89.80%), followed by comsan (88.20%), while the lowest performance was observed for eaywag (82.40%). Across species, the macro-averaged F1-score ranged from 0.65 to 0.82, demonstrating the robustness of the model in classifying different bird species. However, the recall values for minority classes were relatively lower, indicating that the model struggled to correctly identify certain species when they were the positive class.

To assess the effectiveness of different classification models, we compared the performance of Random Forest, SVC, Decision Tree, and Logistic Regression across all species. As presented in Table 1, the Random Forest classifier achieved moderate accuracy across species, with its highest performance on thrnig (70%) but struggled with eaywag (48%). The SVC model outperformed Random Forest in most cases, achieving the highest classification accuracy for comsan (70%). Similarly, Logistic Regression performed competitively, particularly for comsan (67%) and thrnig (69%), demonstrating its capability as a baseline model. On the other hand, the Decision Tree classifier showed the weakest performance overall, with the lowest accuracy scores for barswa (39%) and eaywag (34%), suggesting susceptibility to overfitting and reduced generalization.

The integration of a CNN-based deep learning pipeline significantly enhanced the classification accuracy across all bird species, as evidenced in Table 1. Compared to traditional machine learning classifiers such as Random Forest, SVC, Decision Tree, and Logistic Regression, the CNN model consistently outperformed them with a substantial margin. For instance, the CNN achieved 0.84 accuracy for Barswa and 0.85 for Thrnig, representing over 20–30 percentage points improvement compared to the best-performing traditional models.

Furthermore, EfficientNet—a more advanced convolutional architecture—yielded comparable performance to the CNN, slightly outperforming it for species like Comsan and providing more balanced accuracy across species. However, EfficientNet required more runtime compared to the CNN in our experiment. These results show that deep feature extraction through convolutional layers can better capture temporal and spectral variations in Mel spectrogram representations than manually engineered features.

While XGBoost previously held the best overall performance among traditional models, its relatively lower recall and F1-scores for classes like Eaywag and Barswa indicate limitations in capturing more nuanced patterns in the audio data. In contrast, CNN and EfficientNet demonstrated improved generalization, particularly evident in higher and more balanced accuracy scores across all five bird species.

#### 4.0.1 Runtime comparison

In our experiment, one major challenge was the limited computational resources, as all models were trained and evaluated on a 16-core CPU without GPU acceleration. This constraint significantly affected the efficiency of traditional method feature extraction and deep learning methods.Both of those methods require high computational power for processing large volumes of Mel spectrogram data.

Table 3: Runtime Comparison Between Traditional and Deep Learning Models (CPU)

| **Model** | **Model Runtime (s)** | **Total Pipeline Runtime** |
|---|---|---|
| Random Forest | 12 | ~4 hours |
| SVC | 15 | |
| Decision Tree | 8 | |
| Logistic Regression | 10 | |
| CNN | 450 | ~40 minutes |
| Efficient Net | 520 | |

In addition to classification performance, we also evaluated the computational cost associated with each method. As shown in Table 3, traditional machine learning models such as Random Forest, SVC, Decision Tree, and Logistic Regression each required only a few seconds of runtime individually. However, when training and evaluating all traditional models across all five bird species, the entire pipeline took approximately 4 hours due to repeated feature extraction and model-specific preprocessing steps. In contrast, while deep learning models like CNN and EfficientNet required significantly longer runtimes per model (450–520 seconds), the total runtime for the entire advanced pipeline was approximately 40 minutes. This efficiency gain can be attributed to the streamlined workflow using Mel spectrograms and end-to-end learning. Despite the higher per-model cost, the overall deep learning pipeline was more time-efficient and scalable when processing multiple species.

## 5 Discussion

Our study aims to conduct a comparative analysis between traditional machine learning methods and advanced deep learning approaches for bird call classification. The results clearly demonstrate that deep learning models—particularly those using Mel spectrograms combined with convolutional neural networks (CNNs)—offer superior classification accuracy and robustness across various bird species.

Traditional models such as Random Forest, SVC, and Logistic Regression are lightweight, interpretable, and computationally efficient. These models serve as strong baselines, especially when computational resources are limited. However, their performance often suffers in complex or noisy audio environments due to their reliance on manually engineered features, which may not capture the rich temporal and spectral characteristics present in bird calls.

In contrast, our advanced CNN-based method significantly improved accuracy by learning deep features directly from Mel spectrograms. The CNN and EfficientNet models showed strong generalization across species, particularly for those with more distinct acoustic signatures. These deep learning models were more effective at identifying subtle differences in audio patterns, leading to higher recall and F1-scores for several species.

Despite these advantages, we encountered a major barrier in terms of computational efficiency. All training and evaluation were conducted using a 16-core CPU setup, which is hard for large dataset to proceed. Using more advanced hardware would considerably increased the runtime for deep learning models. Training CNNs and EfficientNet architectures on CPUs proved to be time-consuming and inefficient, limiting our ability to experiment with more complex architectures or perform extensive hyperparameter tuning. Access to GPU resources would likely lead to faster training, better scalability, and more thorough experimentation, allowing for further performance improvements.

Looking forward, we plan to explore more optimized architectures such as ResNet or MobileNet for a better balance between performance and computational cost. Additionally, using GPU acceleration and incorporating transfer learning or attention-based mechanisms may help improve classification performance, particularly for underrepresented species. Developing efficient data preprocessing pipelines and semi-supervised labeling techniques will also be essential for scaling up our framework to larger and more diverse datasets.

## 6 Conclusion

In this study, we conducted a comparative analysis of traditional machine learning methods and advanced deep learning approaches for bird call classification. Our results show that while traditional models like Random Forest and SVC provide decent baseline performance, they are limited in handling the complex acoustic patterns present in bird vocalizations. In contrast, deep learning models—particularly CNNs applied to Mel spectrograms—demonstrated superior accuracy and generalization across species. However, our experiments also highlighted the computational challenges of training deep models on CPU-only systems, suggesting the need for GPU acceleration to fully leverage their potential. Moving forward, our work can be extended by integrating more efficient architectures, using transfer learning, and optimizing the preprocessing pipeline to improve both scalability and performance. This study contributes to the growing body of research on bioacoustic classification and offers practical insights into model selection under resource-constrained settings.

## 7 Acknowledgment

We are really grateful to Prof. Zhiting Hu, University of California, San Diego, for his continuous support, encouragement, and insightful ideas in this class. Many of ideas in this review are proposed and taught by him during the class. I also genuinely appreciate support from our TA Yi Gu, who help us throughout the quarter and answering our questions.

## References

[1] H. Afzal, M. Sheeraz, and A. Mark. "A Novel Approach for MFCC Feature Extraction". In: *Proceedings of the Conference on Advanced Signal Processing*. 2011, pp. 123–130.

[2] R. Bardeli et al. "Detecting bird sounds in a complex acoustic environment and application to bioacoustic monitoring". In: *Pattern Recognition Letters* 31.12 (2010), pp. 1524–1534.

[3] S. Davis and P. Mermelstein. "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences". In: *IEEE Transactions on Acoustics, Speech, and Signal Processing* 28.4 (1980), pp. 357–366.

[4] Hynek Hermansky. "Perceptual Linear Predictive (PLP) Analysis of Speech". In: *Journal of the Acoustical Society of America* 87.4 (1990), pp. 1738–1752.

[5] Stefan Kahl et al. "Overview of BirdCLEF 2023: Automated Bird Species Identification in Eastern Africa". In: *Conference and Labs of the Evaluation Forum*. 2023. URL: https://api.semanticscholar.org/CorpusID:264441540.

[6] Brian McFee et al. *librosa/librosa: 0.11.0*. Version 0.11.0. 2025. DOI: `10.5281/zenodo.15006942`. URL: `https://doi.org/10.5281/zenodo.15006942`.

[7] Meinard Müller. *Fundamentals of Music Processing: Audio, Analysis, Algorithms, Applications*. Springer International Publishing, 2015. ISBN: 978-3-319-21944-8.

[8] Lavanya Sudha, G. Lavanya Devi, and Naresh Nelaturi. "Random Forest Algorithm for Recognition of Bird Species using Audio Recordings". In: *IJAMTES Conference on Machine Learning for Bioacoustics*. 2018.

[9] Tang, Chenshu Liu, and Xiang Yuan. "Recognition of bird species with birdsong records using machine learning methods". In: *PLOS ONE* (2024).

[10] M. Turab et al. "Investigating Multi-Feature Selection and Ensembling for Audio Classification". In: *arXiv preprint* (2022). arXiv: `2201.12345`.

[11] Willem-Pier Vellinga and Robert Planqué. *Xeno-canto: Bird sounds from around the world*. Accessed: 2025-02-26. 2024. URL: `https://xeno-canto.org`.

[12] Emiel Witting et al. "Addressing the Challenges of Domain Shift in Bird Call Classification for BirdCLEF 2024". In: *Conference and Labs of the Evaluation Forum*. 2024. URL: `https://api.semanticscholar.org/CorpusID:271856080`.

[13] Fei Zhang et al. "Bird Song Recognition Based on Deep Transfer Learning with XGBoost". In: *2022 4th International Conference on Robotics and Computer Vision (ICRCV)*. 2022, pp. 96–102. DOI: `10.1109/ICRCV55858.2022.9953226`.