Machine Learning with Applications 23 (2026) 100807

Contents lists available at ScienceDirect

Machine Learning with Applications

journal homepage: www.elsevier.com/locate/mlwa

# Spectrogram-Based Deep Learning Models for Acoustic Identification of Honey Bees in Complex Environmental Noises

Muhammad Anus Khan [a], Bilal Hassan Khan [a], Shafiq ur Rehman Khan [a], Ali Raza [b], Asif Raza [b], Shehzad Ashraf Chaudhry [c,d,*]

[a] Department of Computer Science, Namal University, Namal Knowledge City, Mianwali, 42250, Punjab, Pakistan
[b] Department of Software Engineering, University of Mianwali, University Road, Mianwali, 42200, Punjab, Pakistan
[c] Department of Computer Science and Information Technology, College of Engineering, Abu Dhabi University, Abu Dhabi, UAE
[d] Department of Software Engineering, Faculty of Engineering and Architecture, Nisantasi University, 34398 Istanbul, Turkey

ARTICLE INFO

*Keywords:*
Deep learning
bio acoustics
honey bee monitoring
environmental sound classification
spectrogram analysis

ABSTRACT

The rapid decline of honey bee populations presents an urgent ecological and agricultural concern, demanding innovative and scalable monitoring solutions. This study proposes a deep learning-based system for non-invasive classification of honey bee buzzing sounds to distinguish bee activity from complex environmental noise—a fundamental challenge for real-world acoustic monitoring. Traditional machine learning models using features like Mel Frequency Cepstral Coefficients (MFCCs) and spectral statistics performed well on curated datasets but failed under natural conditions due to overlapping acoustic signatures and inconsistent recordings.

To address this gap, we built a diverse dataset combining public bee audio with recordings from the Honeybee Research Center at the National Agricultural Research Centre (NARC), Pakistan, capturing various devices and natural environments. Audio signals were converted into mel spectrograms and chromograms, enabling pattern learning via pre-trained convolutional neural networks. Among tested architectures—EfficientNetB0, ResNet50, and MobileNetV2—MobileNetV2 achieved the highest generalization, with 95.29% accuracy on spectrograms and over 90% on chromograms under an 80% confidence threshold.

Data augmentation improved robustness to noise, while transfer learning enhanced adaptability. This work forms part of a broader project to develop a mobile application for real-time hive health monitoring in natural environments, where distinguishing bee buzzing from other sounds is the crucial first step. Beyond binary classification, the proposed approach offers potential for detecting hive health issues through acoustic patterns, supporting early interventions and contributing to global bee conservation efforts.

## 1. Introduction

Bees are among the most crucial pollinators in our ecosystem, responsible for the reproduction of numerous plant species and contributing significantly to agricultural productivity Hung et al. (2018), Thakur (2012), Patel et al. (2021), Aslan et al. (2016). However, the alarming global decline in bee populations poses a severe threat to biodiversity, food security, and ecosystem stability. Monitoring bee activity, particularly in both natural and managed hives, is essential for understanding the factors contributing to this decline and for developing effective conservation strategies Guzman-Novoa et al. (2016), Ellis (2012), Genersch (2010). Considering practical deployment, it is crucial that monitoring approaches remain low-cost and easily scalable to support beekeepers in resource constrained environments.

Traditional bee monitoring methods, such as visual inspections and hive mounted sensors, are often invasive, labor-intensive, and limited in their scalability. Recently, audio-based monitoring has emerged as a promising non-invasive alternative by leveraging the distinctive buzzing sound produced by bees. However, accurately classifying bee sounds remains challenging due to the presence of various environmental noises, such as birdsong, insect activity, wind, and human-generated sounds. These overlapping acoustic patterns make it difficult to distinguish bee buzzes using simple heuristics, necessitating robust preprocessing, effective feature extraction, and the application of advanced

* Correspondence. Department of Software Engineering Faculty of Engineering and Architecture Nisantasi University, Istanbul, 3400, Turkey.
*E-mail addresses:* anus2021@namal.edu.pk (M.A. Khan), bilal.hasan2021@namal.edu.pk (B.H. Khan), Shafiq.rehman@namal.edu.pk (S.R. Khan), ali.raza@umw.edu.pk (A. Raza), asifraza@umw.edu.pk (A. Raza), ashraf.shehzad.ch@gmail.com (S.A. Chaudhry).

https://doi.org/10.1016/j.mlwa.2025.100807
Received 4 August 2025; Received in revised form 18 November 2025; Accepted 30 November 2025
Available online 12 December 2025

machine learning and deep learning techniques.

While publicly available datasets such as those from California apiaries serve as a valuable starting point, they often lack the diversity required for real world deployment. To address this limitation, we developed a custom dataset by merging such datasets with new audio recordings collected from the Honey- bee Research Center at the National Agricultural Research Center (NARC), Pakistan. Recordings were captured using widely available mobile phones and external microphones in different field conditions, times of day, and background noise levels to ensure diversity in both environmental and recording contexts. This enhances accessibility and supports large scale adoption in field scenarios.

We explored three primary approaches to classify bee and non-bee sounds: traditional feature-based models using numerical descriptors such as Mel Frequency Cepstral Coefficients (MFCCs), spectral centroid, and zero-crossing rate; spectrogram-based classification using Mel-scaled visual representations of audio; and chromogram-based classification to capture harmonic and tonal patterns. While traditional machine learning models such as Random Forest performed well in controlled settings, they lacked generalization under real-world noise conditions.

To overcome these limitations, we employed pre-trained deep convolutional neural networks including EfficientNetB0, ResNet50, and MobileNetV2, using transfer learning techniques. MobileNetV2 outperformed other architectures, achieving high accuracy and robustness on both validation and real-world data. The use of spectrogram and chromogram representations significantly enhanced the model's ability to differentiate between bee and non-bee audio patterns, particularly in noisy environments.

This study presents a scalable and non-invasive framework for real-time honey bee sound classification, demonstrating that accurate audio based monitoring is feasible using low cost, mobile recording devices. Furthermore, this work contributes toward enabling early hive health diagnostics, supporting early detection of colony stressors and long-term conservation strategies.

The remainder of this paper is structured as follows: Section 2 defines the problem statement, while Section 3 reviews related work. Section 4 describes the proposed methodology in detail, and Section 5 presents the experimental results. Finally, Section 6 concludes the study and outlines potential directions for future research.

## 2. Problem Statement

Non-invasive audio-based monitoring has emerged as a promising approach for assessing honey bee activity, offering a scalable alternative to traditional methods that are often intrusive, labor-intensive, and limited in scope. However, a fundamental challenge persists: accurately distinguishing bee buzzing from a wide array of environmental sounds—including birdsong, insect noises, wind, and human activity—under real-world conditions. The acoustic overlap between bee and non-bee sounds, especially in uncontrolled outdoor environments, significantly reduces the effectiveness of existing classification models.

Misclassification of audio signals can result in misleading assessments of hive status, obstructing the early detection of colony stressors such as disease, queen loss, or impending collapse. Moreover, many existing models demonstrate high performance on curated datasets but fail to generalize across diverse environmental scenarios due to limited variability in training data and reliance on high-fidelity recording equipment.

This gap highlights the urgent need for robust, generalizable, and efficient classification systems capable of operating accurately in heterogeneous acoustic conditions using low-cost, accessible recording devices. Addressing this problem is critical for developing practical, real-time bee monitoring systems that can be deployed in resource-constrained settings.

In this study, we investigate the use of deep learning techniques—particularly spectrogram and chromogram based classification—combined with a custom, environmentally diverse dataset collected using mobile devices. Our objective is to design a scalable and effective framework for bee sound classification that supports early hive health diagnostics and contributes to pollinator conservation efforts in both research and agricultural domains.

## 3. Related Work

Monitoring honey bee colonies through acoustic analysis has gained attention as a non-invasive and scalable method to assess hive health and behavior. One of the critical challenges in beekeeping is detecting queen bee presence, a task vital to colony sustainability. A study by Soares (2022) proposed the use of Mel-Frequency Cepstral Coefficients (MFCCs) along with time and frequency domain features to create a compact and efficient descriptor. Their model achieved 99% classification accuracy, demonstrating its potential for lowcomputational, real-time deployment in field environments.

Similarly, sound-based classification approaches have been explored for broader hive activity monitoring. In Terenzi et al. (2022), the authors used Convolutional Neural Networks (CNNs) to classify acoustic signals collected from orphaned colonies. Their experiments revealed the capacity of sound features to reflect behavioral patterns, particularly under stress, highlighting the utility of audio monitoring as a complementary diagnostic tool.The work in Orlowska et al. (2021) introduced a simplified supervised method based on downsampled spectrograms for queen detection. By leveraging deep CNNs and reducing dimensionality, their model achieved 96% accuracy, outperforming traditional MFCC and STFT-based techniques, especially in noisy and previously unseen hives.

Expanding on behavioral prediction, Iqbal et al. (2024) evaluated several machine learning and deep learning models, including CNNs, LSTM, and Transformers, to anticipate swarming behavior. CNNs using MFCCs attained the highest accuracy (99%), reinforcing the effectiveness of frequency-based audio features.To improve classification granularity, Rustam et al. (2024) investigated selective acoustic features—including spectral centroid, zero-crossing rate, MFCCs, chromagrams, and constant-Q transforms—to differentiate between bee presence, absence, and queenlessness. Their findings highlight the importance of feature selection for efficient hive monitoring. Hybrid techniques have also proven useful. The approach in Phan (2024) combined MFCC and STFT features with a Random Forest classifier to enhance bee state classification accuracy to 87.2%, emphasizing the benefit of integrating spectral and temporal descriptors.

Generalization to new hive conditions is addressed in Bricout et al. (2024), where the authors introduced the BeeTogether dataset and employed contrastive learning to improve model adaptability. Their method achieved over 99% accuracy on unseen hives, showing the value of standardized, labeled data. Broader applications of AI in apiculture are reviewed in Astuti et al. (2024), where models such as SVMs and CNNs were employed in systems like BeePi and DeepBee© for monitoring temperature, humidity, and audio data. Though effective, the study identified a need for improved algorithms and larger, more diverse datasets.

The behavioral effects of queenlessness were analyzed in Kanelis et al. (2023), where changes in acoustic patterns were observed within hours of queen removal. This study underlines the significance of real-time monitoring for early detection and intervention.In addition to the above, Harano et al. Harano et al. (2013) demonstrated the effectiveness of queen detection through acoustic profiling in Japanese hives, while Ramírez et al. Ramírez et al. (2015) applied machine learning models for classifying bee sounds with notable success. More recently, Kim et al. Kim et al. (2021) proposed an edge-AI-based acoustic monitoring system, offering an efficient, real-time solution for in-field hive surveillance.

These studies collectively establish a foundation for the present

research, which addresses the need for robust, generalizable, and accessible audio-based classification of bee activity under real-world, noisy conditions. The inclusion of a summary comparison in Table 1 streamlines the discussion and ensures clarity regarding methodological gaps and motivations for the proposed approach.

Despite significant advancements in audio-based hive monitoring, existing studies often rely on controlled datasets, specialized recording equipment, or limited environmental diversity, which restricts model generalizability in realworld conditions. The current methods largely overlook integration strategies that combine low-cost data acquisition with high-performing deep learning models tailored for noisy, field-level acoustic environments. This research addresses these limitations by introducing a scalable, mobile-compatible framework supported by a custom, environmentally diverse dataset and optimized deep learning architectures.

## 4. Proposed Methodology

This section outlines the methodology comprised of information related to dataset, feature extraction, and usage of Machine learning and Deep Learning approaches for classification presented in Fig. 1.

### 4.1. Dataset Development

This research utilizes a composite audio dataset consisting of two distinct sources: (1) A publicly available dataset titled *"Bee Audio Object Detection"* AndrewL (2023) from Kaggle, and (2) A custom dataset collected under realworld conditions using various mobile phones and external microphones. The custom recordings were obtained from the Honey Bee Research Center at the National Agricultural Research Center (NARC), Pakistan's largest agricultural research institution.

**Table 1**
Comparison of existing bee sound classification studies based on recording setup, feature representation, learning model, and real-world applicability.

| Study | Recording Setup | Features Set | Learning Models | Real-world Application |
|---|---|---|---|---|
| Harano et al. (2013) Harano et al. (2013) | Controlled lab | MFCC, Frequency bands | Statistical model | Limited |
| Ramírez et al. (2015), Ramírez et al. (2015) | Controlled hive | MFCC, Time-domain features | SVM, ANN | No |
| Kim et al. (2021) Kim et al. (2021) | IoT/Edge device | MFCC, Temp, Humidity | Edge AI (CNN) | Moderate |
| Terenzi et al. (2020), Terenzi et al. (2022) | Field recordings | MFCC | CNN | Moderate |
| Orlowska (2020) Orlowska et al. (2021) | Controlled, downsampled | Summarized, Spectrogram | CNN | Good |
| Phan (2023) Phan (2024) | Controlled | MFCC, STFT | Random Forest | Low |
| Bricout et al. (2024), Bricout et al. (2024) | Multi-hive datasets | Contrastive features | Contrastive, Learning | High |
| **Proposed Work** | **Field recordings using mobile mics** | MFCC, Spectrogram, Chromogram | MobileNetV2 (DL) | **High (95.29%)** |

The final dataset, compiled by merging both sources, comprises a total of 2840 audio samples. Table 2 provides a detailed overview. The Kaggle dataset was curated for bee detection using acoustic signals recorded at hive entrances. The custom dataset complements this by introducing greater environmental and device variability, improving the dataset's generalizability.

#### 4.1.1. Recording Conditions

The custom dataset was recorded under diverse real-world conditions to reflect varying ambient noise levels and acoustic backgrounds. Recordings were captured using different mobile devices and microphones in outdoor and semi-urban environments. Active beehives at NARC served as primary recording sites, ensuring biological authenticity and operational realism. realism. To minimize device induced bias, recordings were intentionally collected using multiple mobile phone models and microphones, enabling the model to learn from varied acoustic hardware responses and ambient environments.

#### 4.1.2. Audio Chunking and Dataset Structuring

To ensure consistency across samples, a series of preprocessing steps were applied, including duration standardization and noise filtering. Audio files exceeding the target length of 10 seconds were segmented into overlapping 10-second chunks to retain important temporal patterns and maximize coverage.

Let $L$ denote the original length of an audio file (in seconds). The number of overlapping chunks is computed as Equation 1:

$$\text{Number of chunks} = \left[\frac{L-10}{10-overlap}\right] + 1 \tag{1}$$

For example, a 15-second audio file with a 5-second overlap yields: Chunk 1 = [0,10], Chunk 2 = [5,15]

This strategy helps preserve signal continuity while enabling standardized model input lengths.

The dataset was organized into two primary classes:

1. **Bee Sounds:** Audio clips containing clearly identifiable bee buzzing patterns.
2. **Not Bee Sounds:** Audio clips dominated by environmental or anthropogenic noise (e.g., birds, wind, vehicles, human activity).

For efficient model training and evaluation, the audio files were stored in separate directories: /Bee-Audios/ for bee sounds and /No-Bee-Audios/ for non-bee sounds.

### 4.2. Numerical Feature Extraction (Numerical Approach)

Numerical features were extracted from the time-domain and frequencydomain representations of the audio signals. These features include:

**Mel Frequency Cepstral Coefficients (MFCCs)**: MFCCs mimic human auditory perception. The process involves a Fourier Transform to get the power spectrum, mapping to the mel scale with triangular filters, and applying the Discrete Cosine Transform (DCT) to the log energies to derive MFCCs. The $i$-th coefficient $c_i$ is given by Equation 2:

$$c_i = \sum_{n=1}^{N_f} S_n \cos\left(i(n-0.5)\frac{\pi}{N_f}\right), \; i = 1, \ldots, L, \tag{2}$$

where $c_i$ is the $i$-th MFCC coefficient, $N_f$ is the number of filters, $S_n$ is the log energy of the $n$-th filter, and $L$ is the number of coefficients.

**Zero Crossing Rate (ZCR)** ZCR measures the rate at which the audio waveform crosses the zero amplitude axis, defined as Equation 3:

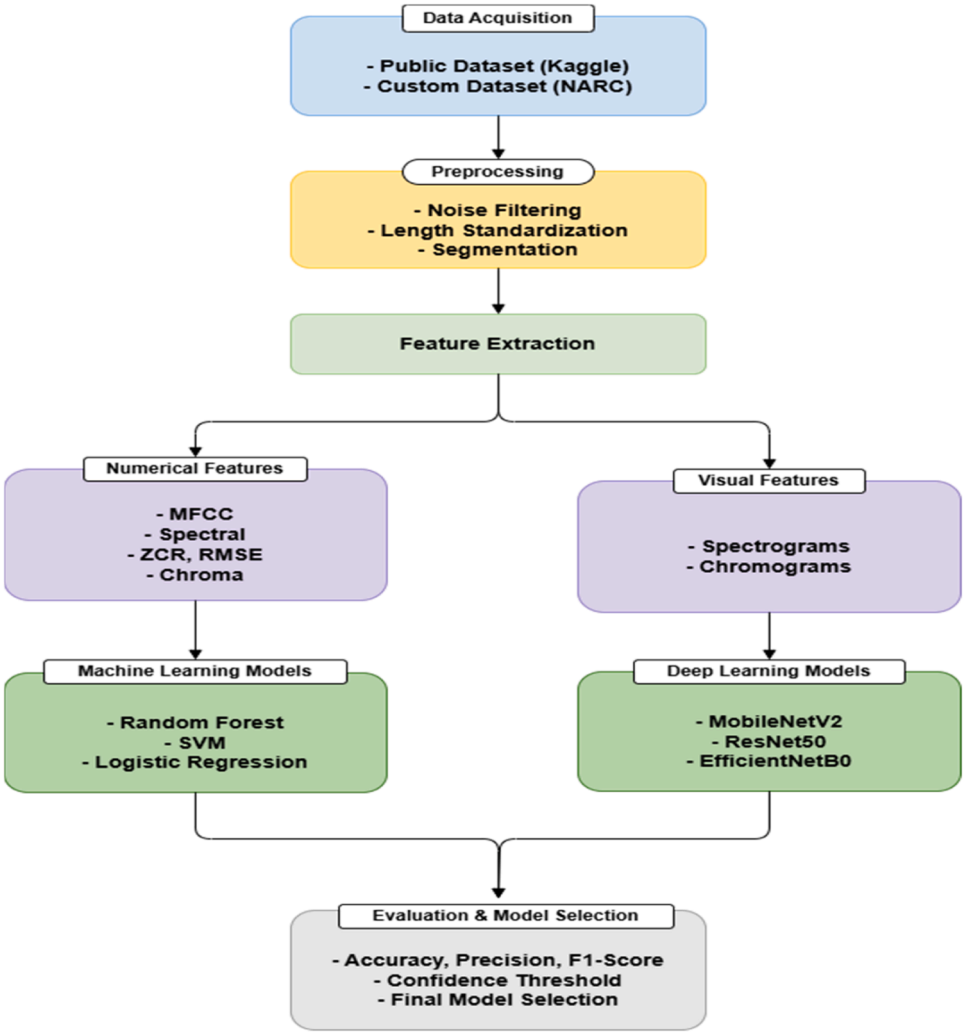



**Fig. 1.** Overview of the proposed acoustic bee identification framework, including dataset acquisition, preprocessing, feature extraction, and classification stages for both traditional machine learning and deep learning models.

**Table 2**
Summary of the Combined Dataset.

| | |
|---|---|
| Total Number of Files | 2840 |
| Data Sources | Kaggle + Custom (NARC, Pakistan) |
| Recording Locations (Kaggle) | 25 beehives in San Jose, Cupertino, and Gilroy, California, USA |
| Recording Locations (Custom) | Research centers and field sites in Pakistan |
| Class Distribution | Bee: 1417 files, Not Bee: 1423 files |
| Recording Devices | Mobile phones and external microphones |
| Recording Duration | Standardized to 10 seconds |
| Audio Format | .wav files with variable sampling rates |

$$zcr = \frac{1}{T-1}\sum_{t=1}^{T-1} 1_{R<0(s_t s_{t-1}} \tag{3}$$

where $zcr$ is the zero crossing rate, $T$ is the number of samples, and $1_{R<0}$ is the indicator function for zero crossings.

**Root Mean Square Energy (RMSE)**: RMSE represents the average power of the signal, computed as Equation 4:

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N} x[i]^2} \tag{4}$$

Where $x[i]$ is the amplitude of the $i$-th sample.

**Spectral Features**: Several spectral features were calculated:

*1. Spectral Centroid* The spectral centroid represents the weighted mean of frequencies in a signal, computed using a Fourier Transform (Equation 5):

$$centriod = \frac{\sum_{n=0}^{N-1} f(n)x(n)}{\sum_{n=0}^{N-1} x(n)} \tag{5}$$

where $x(n)$ is the magnitude of bin $n$, and $f(n)$ is the center frequency of that bin.

*2. Spectral Bandwidth*: Spectral bandwidth measures the spread of frequencies around the centroid (Equation 6):

$$Bandwidth = \left(\sum_k S(k)(f(k) - f_c)^p\right)^{\frac{1}{p}} \tag{6}$$

*3. Spectral Rolloff* Spectral rolloff represents the frequency below which a specified percentage (e.g., 85%) of the total spectral energy is concentrated.

*4. Chroma Features*: Chroma features summarize the energy distribution across 12 pitch classes, representing the harmonic content of the signal.

#### 4.2.1. Feature Selection and Importance Analysis

A Random Forest classifier was used for its ability to capture non-linear relationships and provide insights into feature importance. The dataset was split into 80% for training and 20% for testing. The model showed near-perfect accuracy on the training set but struggled with generalization to new data.

The model identified the following features as the most influential for classification. Table 3 shows the features with the highest importance scores:

These features were deemed essential for classification because they provide valuable information about the sound's timbre, rhythm, and texture, which are key for differentiating between various audio classes.

Based on the Random Forest feature importance, the most relevant features were selected for further analysis. These include spectral centroid, spectral rolloff, zero-crossing rate (ZCR), and the first two MFCC coefficients.

#### 4.2.2. Exploratory Data Analysis (EDA) on Numerical Data

Exploratory Data Analysis (EDA) is a crucial step in understanding the underlying patterns, distributions, and relationships in the data. In this section, we analyze the features related to "bee" and "no bee" categories to understand their statistical properties, distributions, and correlations. The key aspects of this analysis include statistical summaries, comparison of feature distributions, and correlation analysis.

#### 4.2.3. Statistical Analysis of Features

We begin by performing a statistical analysis of the selected features, which include spectral centroid, spectral bandwidth, spectral rolloff, MFCC coefficients, and Zero Crossing Rate (ZCR). The statistical analysis includes key metrics such as mean, standard deviation, minimum, maximum, and quartiles (25%, 50%, 75%).

The following table shows the statistical summary of the selected features for the "bee" category:

The following table shows the statistical summary of the selected features for the "No Bee" category:

The above Tables 4 and 5 present the statistical characteristics for the features in both "bee" and "no bee" categories. These include the count of data points, the mean, standard deviation, minimum, and maximum values. We can observe significant differences in several features between the two categories.

To further understand how the "bee" and "no bee" categories differ, we calculated the mean differences (see Table 6) and range differences (see Table 7) for each feature. These values highlight the magnitude of the differences between the two categories.

The mean differences and range differences demonstrate the extent to which the two categories differ for each feature. For example, the spectral centroid and spectral roll-off have substantial differences, indicating that these features may be highly discriminative between the two categories.

#### 4.2.4. Correlation Analysis

A set of numerical acoustic features, including MFCCs, spectral centroid, zero-crossing rate, and chroma energy, was analyzed to assess their relevance for hive sound classification. Correlation analysis (presented in Figs. 2 and 3) and feature distribution plots (Figs. 4 and 5) were used to evaluate redundancy, noise sensitivity, and class- specific variability across diverse environments. To establish a baseline, ma- chine learning models such as SVM and Random Forest were trained on these extracted features (Tables 89). While these models achieved reasonable performance in controlled scenarios, their accuracy declined significantly in noisy, real- world field recordings using mobile devices. This outcome highlights the limited robustness of traditional feature-based approaches and strongly motivated the transition toward deep learning techniques using spectrogram and chromogram representations, which better capture temporal frequency patterns essential for reliable hive monitoring in varied outdoor conditions.

**Table 3**
Top Features Identified by the Model and Their Importance Scores.

| Feature | Importance |
|---|---|
| Spectral Centroid | 0.190025 |
| Spectral Bandwidth | 0.189648 |
| Spectral Rolloff | 0.137424 |
| MFCC 1 | 0.118606 |
| MFCC 2 | 0.106430 |
| Zero Crossing Rate (ZCR) | 0.060858 |

**Table 4**
Statistical Analysis for 'Bee' Category.

| Feature | Count | Mean | Std Dev | Min | Max |
|---|---|---|---|---|---|
| Spectral Centroid | 1415 | 2064.77 | 690.49 | 511.10 | 3673.86 |
| Spectral Bandwidth | 1415 | 2468.91 | 497.02 | 1232.11 | 3220.21 |
| Spectral Roll-off | 1415 | 4705.33 | 1802.09 | 764.65 | 7656.53 |
| MFCC 1 | 1415 | -416.02 | 76.48 | -752.75 | -242.65 |
| MFCC 2 | 1415 | 107.32 | 37.03 | 35.29 | 188.89 |
| ZCR | 1415 | 0.078 | 0.034 | 0.011 | 0.193 |

**Table 5**
Statistical Analysis for 'No Bee' Category.

| Feature | Count | Mean | Std Dev | Min | Max |
|---|---|---|---|---|---|
| Spectral Centroid | 1422 | 686.82 | 205.58 | 257.11 | 1976.89 |
| Spectral Bandwidth | 1422 | 1217.69 | 214.54 | 709.75 | 2297.43 |
| Spectral Roll-off | 1422 | 1214.27 | 466.91 | 287.78 | 4404.99 |
| MFCC 1 | 1422 | -605.74 | 33.83 | -687.55 | -431.67 |
| MFCC 2 | 1422 | 190.31 | 21.76 | 104.89 | 236.24 |
| ZCR | 1422 | 0.025 | 0.012 | 0.005 | 0.101 |

**Table 6**
Mean Differences Between 'Bee' and 'No Bee' Categories.

| Feature | Mean Difference |
|---|---|
| Spectral Centroid | 1455.73 |
| Spectral Bandwidth | 1319.19 |
| Spectral Roll-off | 3649.43 |
| MFCC 1 | 207.59 |
| MFCC 2 | 89.83 |
| ZCR | 0.06 |

**Table 7**
Range Differences Between 'Bee' and 'No Bee' Categories.

| Feature | Range Difference |
|---|---|
| Spectral Centroid | 2394.95 |
| Spectral Bandwidth | 1193.05 |
| Spectral Roll-off | 5513.52 |
| MFCC 1 | 172.19 |
| MFCC 2 | 86.53 |
| ZCR | 0.14 |

### 4.3. Deep Learning Modeling

The use of both spectrograms and chromograms was adopted to enhance the robustness of the classification model by leveraging complementary time frequency representations of audio signals. Spectrograms capture the distribution of energy across frequencies over time, while chromograms emphasize harmonic and tonal characteristics. By combining these features, the model gains improved discrimination capabilities, particularly in noisy environments.

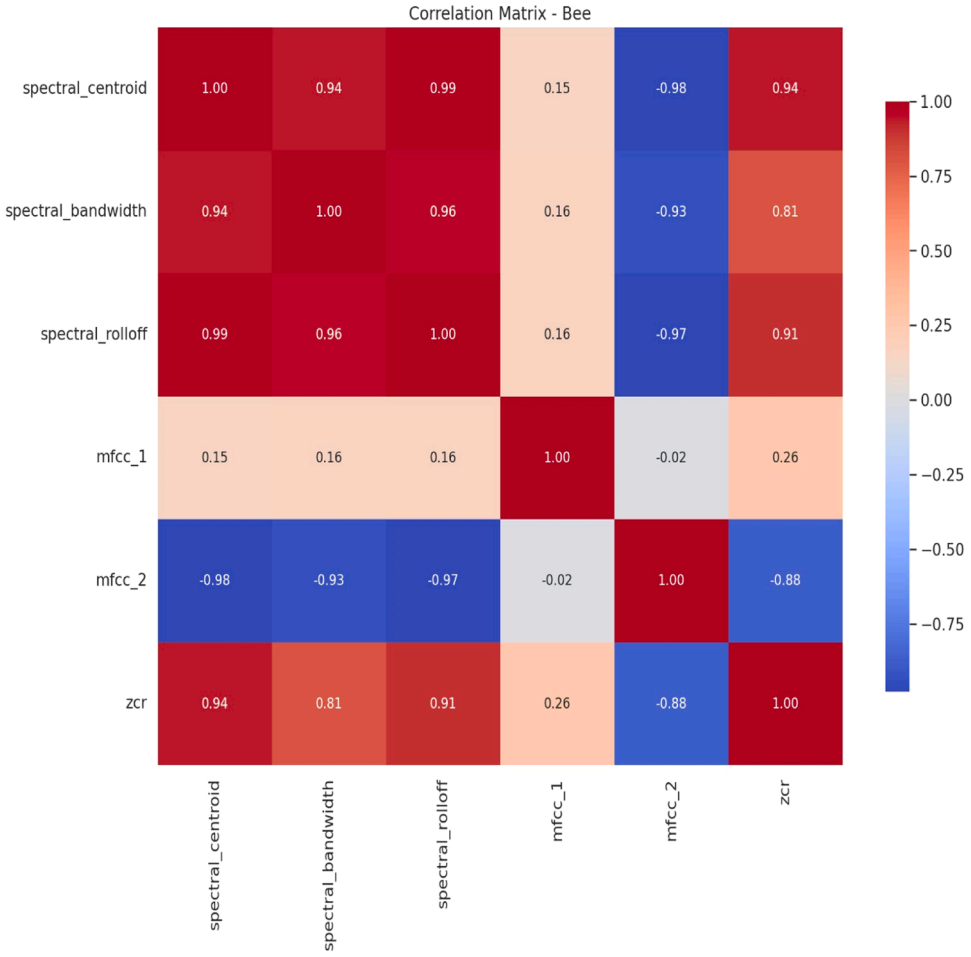



**Fig. 2.** Correlation matrix of numerical acoustic features for the Bee class, illustrating inter-feature dependencies and identifying dominant spectral and cepstral contributors to bee sound discrimination.

#### 4.3.1. *Visual Feature Preparation*

For visual feature extraction, two types of representations were generated from each audio file: spectrograms and chromograms. Spectrograms capture the distribution of sound energy over time and frequency, while chromograms highlight tonal and harmonic structures based on pitch classes. To create spectrograms, audio signals were loaded using the Librosa library McFee et al. (2015), converted into mel spectrograms with 128 mel bands, scaled into decibel units, and saved as PNG images resized to $224 \times 224$ pixels to match the input requirements of MobileNetV2 Sandler et al. (2018). Chromograms were similarly produced using Librosa's chroma functions, computed with 12 chroma bins and an FFT window size of 4096, and visualized and saved as images. Spectrograms provide a rich time-frequency representation aligned with human auditory perception, helping deep learning models learn discriminative patterns even in noisy environments. Chromograms, on the other hand, emphasize harmonic content, making them robust to background noise and useful for identifying pitch-invariant patterns.

Data augmentation was applied by adding real environmental noise (wind, hive-surrounding insects, distant human activity), along with pitch shift and time stretch transformations, where each original audio sample generated approximately three augmented variations. This improves model robustness and reproducibility under real world acoustic conditions.

#### 4.3.2. *Data Organization*

All generated spectrogram and chromogram images were systematically organized into labeled directories to facilitate efficient training. Separate folders were created for bee sounds and non-bee sounds, ensuring that images derived from each audio file were consistently named and easy to trace. This structured organization supported clear data management and streamlined the training process for deep learning models.

### 4.4. *Model Architecture*

After preparing and organizing the spectrograms and chromograms, the next step was to train and evaluate deep learning models capable of distinguishing between "bee" and "not bee" sounds. A confidence threshold of 80% was chosen to ensure the model's predictions were highly reliable.

#### 4.4.1. *Spectrograms*

For the classification of spectrograms, multiple pre-trained architectures were tested, including MobileNetV2 Aslan et al. (2016), EfficientNetB0 Tan and Le (2019), and ResNet50 He et al. (2016). Each of these models was utilized as a feature extractor, with their pre-trained weights (trained on ImageNet) frozen to preserve learned features. Custom layers were added on top of each architecture to enable predictions based on the features extracted from the spectrograms. This approach allowed for a fair comparison among the models.

- *MobileNetV2* is a lightweight convolutional neural network architecture optimized for mobile and embedded devices. It employs

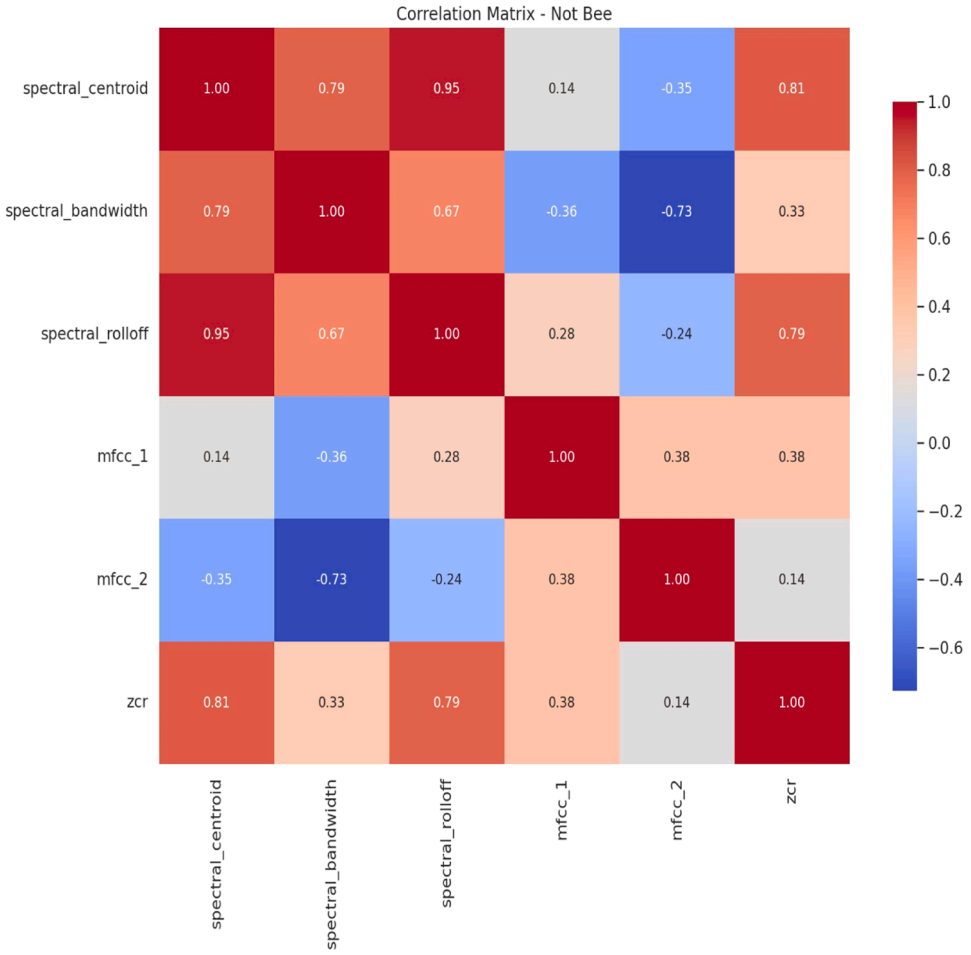



**Fig. 3.** Correlation matrix of numerical acoustic features for the Non-Bee class, highlighting differing acoustic structures and reduced harmonic complexity relative to no bee generated signals.

inverted residual blocks and linear bottlenecks to reduce computational cost while maintaining accuracy. In this study, MobileNetV2 was used with pretrained weights from ImageNet. Spectrogram and chromogram images were resized to 224 × 224 pixels for model input. The model was fine-tuned using the Adam optimizer and categorical cross-entropy loss. Early stopping was applied to mitigate overfitting, and predictions were filtered using an 80% confidence threshold.

- *EfficientNet-B0* is a convolutional neural network architecture designed to achieve high performance with fewer parameters through a compound scaling method that uniformly scales network depth, width, and resolution. In this study, EfficientNet-B0 was implemented using the Keras API with ImageNet-pretrained weights. The input spectrogram images were resized to 224 × 224 pixels to match the network's input requirements. During training, categorical cross-entropy was used as the loss function, and the Adam optimizer was employed for parameter updates. Early stopping was applied to prevent overfitting, and a confidence threshold of 80% was used for class predictions.
- *ResNet50* is a deep residual network architecture featuring 50 convolutional layers with identity shortcut connections, enabling efficient gradient flow and mitigating the vanishing gradient problem. In this work, ResNet50 was utilized with ImageNet-pretrained weights, adapting the final classification layer to predict two classes: bee and no bee. Input spectrogram images were resized to 224 × 224 pixels. Training was performed using categorical cross-entropy as the loss function and the Adam optimizer. Early stopping was applied based on validation loss, and an 80% confidence threshold was set for final predictions.

The results guided the selection of MobileNetV2 as the final model for spectrogram classification due to its adaptability and efficiency in processing audio data.

#### 4.4.2. Chromograms

For the chromograms, only MobileNetV2 was employed, reflecting its effectiveness in capturing relevant features for accurate classification. The model was trained twice with different learning rates to explore the impact of learning rate on performance. The training sessions were conducted as follows:

- **First Training Session**: The learning rate was set to 0.001. This learning rate aimed to accelerate convergence and allow the model to adapt quickly to the features present in the chromograms, resulting in relatively good performance across classes.
- **Second Training Session**: The learning rate was reduced to 0.000001. This lower learning rate was intended to provide finer adjustments to the model's weights, potentially leading to better generalization on unseen data. However, this reduction biased the model toward the "no bee" class, and it did not yield as good results as the higher learning rate, particularly in detecting the presence of bees.

Both training sessions maintained the same custom layers and threshold criteria used for the spectrograms. The decision to utilize only

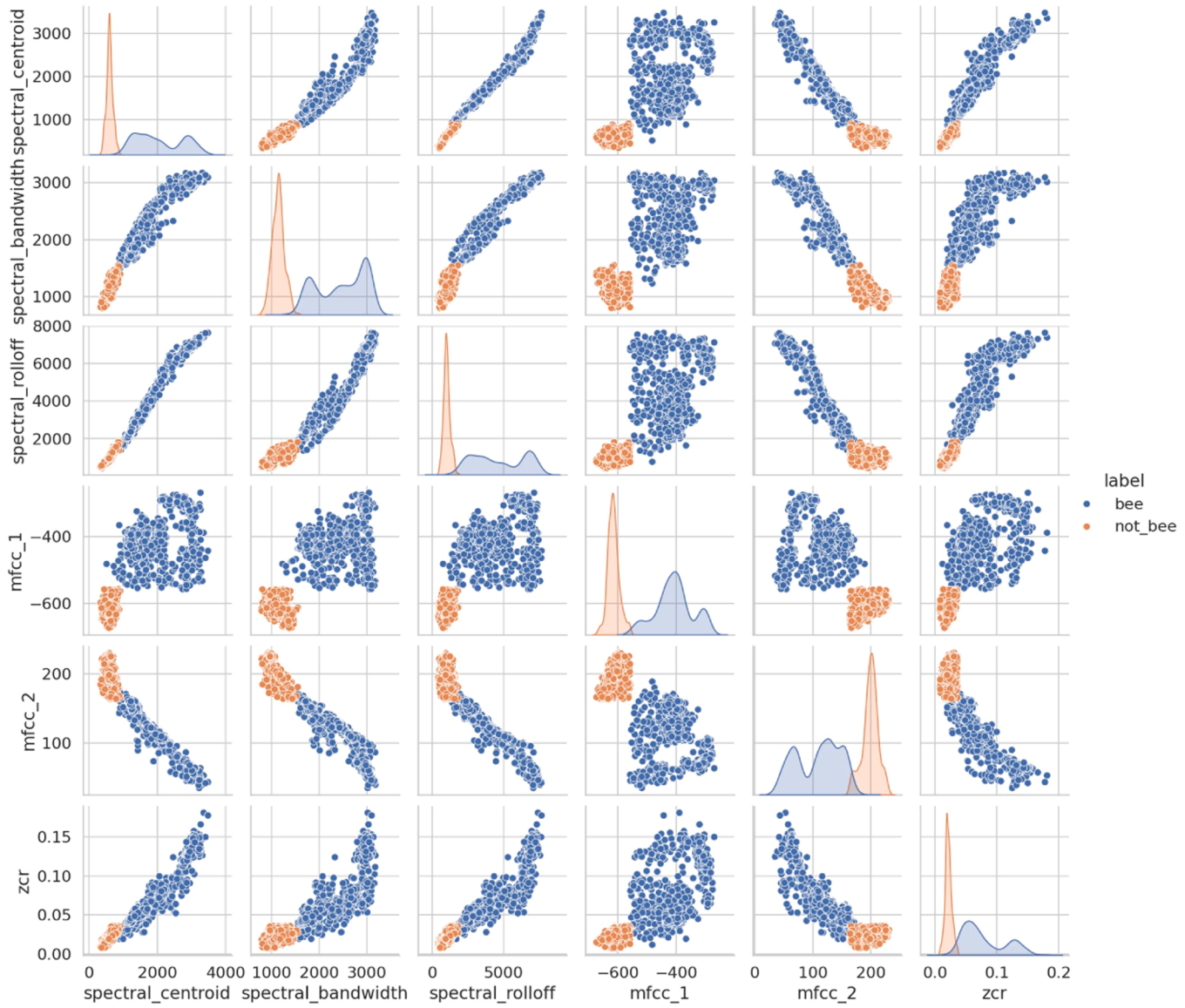



**Fig. 4.** Pair plot of Features for 'Bee' and 'No Bee' Categories.

MobileNetV2 for chromograms was based on its demonstrated capability to effectively classify the relevant features, ensuring consistent performance across different data types.

#### 4.4.3. Threshold-Based Classification

The chosen confidence threshold of 80 percent required the model to classify both spectrograms and chromograms into three categories:

- **Bee**: If the model's confidence for the "bee" class exceeds 80%.
- **Not Bee**: If the model's confidence for the "not bee" class exceeds 80%.

This approach ensures that the predictions made by the models are of high confidence.

#### 4.5.4. Custom Layers & Transfer Learning

To tailor the pre-trained model for the binary classification of both spectrogram and chromogram images, the architecture was fine-tuned by adding custom layers. These modifications ensured the models could effectively handle the unique characteristics of the data.

**Global Average Pooling (GAP)**: The fully connected layers in the original pre-trained architecture were replaced with a Global Average Pooling layer. GAP reduces the spatial dimensions of feature maps by averaging their values, resulting in a single value per feature map. This layer prevents overfitting and reduces the number of trainable parameters while retaining spatial information. The operation is mathematically expressed as Equation 7:

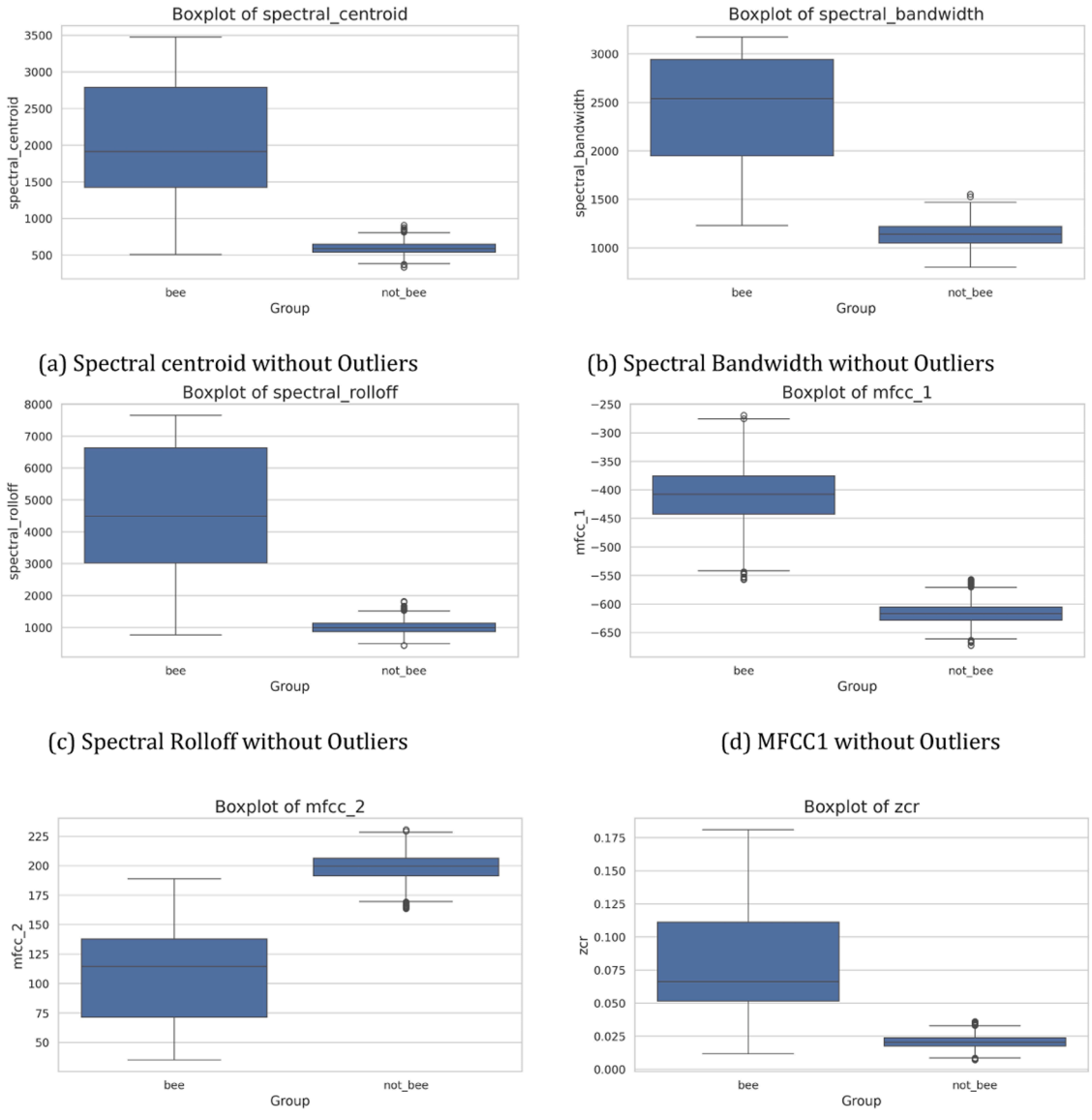
$$GAP(x) = \frac{1}{HXW} \sum_{i=1}^{H} \sum_{j=1}^{W} x_{i,j} \tag{7}$$

where $H$ and $W$ are the height and width of the feature map, and $x_{i,j}$ represents the activation at position $(i,j)$.

**Dense Layer**: A fully connected layer with 128 neurons and ReLU activation was added after the GAP layer. This dense layer captured high-level patterns in the data, enabling the model to better differentiate between the "bee" and "not bee" classes.

**Dropout Layer**: To address overfitting, a dropout layer with a rate of 0.4 was included. This layer randomly deactivates 40% of the neurons during training, ensuring the model does not rely too heavily on specific features.

**Output Layer**: The final layer consisted of a single neuron with




(a) Spectral centroid without Outliers

(b) Spectral Bandwidth without Outliers

(c) Spectral Rolloff without Outliers

(d) MFCC1 without Outliers

(e) MFCC2 without Outliers

(f) ZCR without Outliers

**Fig. 5.** Feature distribution of bee and non-bee audio samples. Enlarged boxplots show clearer separation across spectral and MFCC-based features after removing outliers. Axis labels are improved for interpretability.

sigmoid activation, producing a probability value between 0 and 1. This value represented the likelihood of the input belonging to the "bee" class.

#### 4.4.5. Training strategy

The spectrogram and chromogram data were preprocessed and fed into the customized pre-trained models for training. The key steps in the training pipeline are described below:

**Input Data Preparation**: Each spectrogram and chromogram was resized to 224 × 224 pixels, matching the input dimensions required by MobileNetV2. The pixel values were normalized to a range of 0 to 1, ensuring numerical stability during training.

**Optimizer**: The Adam optimizer was used for its adaptive learning rate and efficiency in handling sparse gradients. The learning rate was set to 0.001 for spectrograms and varied for the two MobileNetV2 training sessions on chromograms, balancing convergence speed with model stability.

**Loss Function**: The binary cross-entropy loss function was employed to optimize the model for the binary classification task. The loss function is defined as (Equation 8):

$$L = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p_i) + (1-y_i)\log(1-p_i)] \tag{8}$$

where $y_i$ is the true label (0 or 1) for the $i$-th sample, and $p_i$ is the predicted probability for the same sample.

**Batch Size and Epochs**: The models were trained in mini-batches to balance computational efficiency and performance. A batch size of 32 was used for both spectrogram and chromogram training, and the training process was conducted over 20 epochs, ensuring adequate time for convergence.

#### 4.4.6. Evaluation Metrics

The performance of all models was assessed using several evaluation metrics, including accuracy, precision, recall, and F1-score. Accuracy represents the proportion of correctly classified instances among all predictions. Precision measures the proportion of true positives among

all instances predicted as positive. Recall reflects the proportion of true positives identified among all actual positive cases. The F1-score is the harmonic mean of precision and recall, providing a balanced measure between false positives and false negatives.

In addition, a confidence threshold of 80% was applied during prediction, meaning that only predictions with softmax probabilities above 80% were considered confident enough for classification. This threshold was chosen to ensure reliable model decisions, particularly under noisy real-world conditions.

#### 4.4.7. Model Selection and Rationale

MobileNetV2 was selected due to its advanced design, which combines efficiency with strong performance, making it ideal for resource-constrained tasks like spectrogram and chromogram classification.

**Lightweight Architecture**: MobileNetV2 is optimized for low-resource environments, ensuring faster training and inference without sacrificing accuracy. It uses depthwise separable convolutions to significantly reduce the computational cost compared to standard convolutional layers, allowing the model to process high-dimensional spectrogram and chromogram images efficiently.

**Inverted Residual Blocks**: A key innovation of MobileNetV2 is its use of inverted residual blocks, where the input and output are connected via shortcut connections, enabling the model to focus on critical features while minimizing redundant computations. This structure enhances feature extraction, making it particularly effective for complex tasks like audio spectrogram and chromogram classification.

**Pre-trained Weights**: MobileNetV2 comes pre-trained on the ImageNet dataset, a large-scale image database. These pre-trained weights provide a strong initialization for the model, allowing it to leverage general visual feature representations and adapt them to the specific tasks of spectrogram and chromogram classification. This transfer learning approach accelerates convergence and improves overall performance.

Fig. 6 represents the model architecture which tells about layers and which part of the model use for feature extraction and which part is used for classification.

## 5. Results

### 5.1. Numerical Approach Results

The performance of the numerical features-based model was evaluated using several machine learning algorithms, including Random Forest (RF), Support Vector Machine (SVM), and Logistic Regression. These models exhibited exceptionally strong results on the publicly available dataset, achieving near-perfect scores across key performance metrics such as test accuracy, precision, recall, and F1-score. However, despite this impressive performance on the training data, the models encountered significant challenges when applied to real-world scenarios. They failed to generalize effectively to unseen, noisy data, resulting in a notable decline in their predictive capabilities. This limitation was partly attributed to feature overlapping within the dataset, where the numerical features exhibited high similarity or redundancy, reducing the models' ability to distinguish between classes under varying conditions. This inconsistency, combined with their over-reliance on the specific characteristics of the public dataset—which did not adequately represent the variability and complexity of practical environments—rendered these numerical models unsuitable for use in a production setting.

### 5.2. Spectrogram-Based Approach Results

The spectrogram-based approach utilized several deep learning models for classification. The results for each model are shown in Table 8.

**EfficientNetB0**: Achieved 50.18% accuracy but struggled with generalization, overfitting the spectrogram dataset towards classifying everything as "no bee." **ResNet50**: Also achieved 50.18% accuracy but faced moderate generalization issues, similarly overfitting towards "no bee."

**MobileNetV2**: Achieved 95.29% accuracy with exceptional generalization. MobileNetV2 demonstrated better robustness to high frequency background noise and retained discriminative spectral patterns more eectively than heavier models such as ResNet50 and E-cientNetB0.

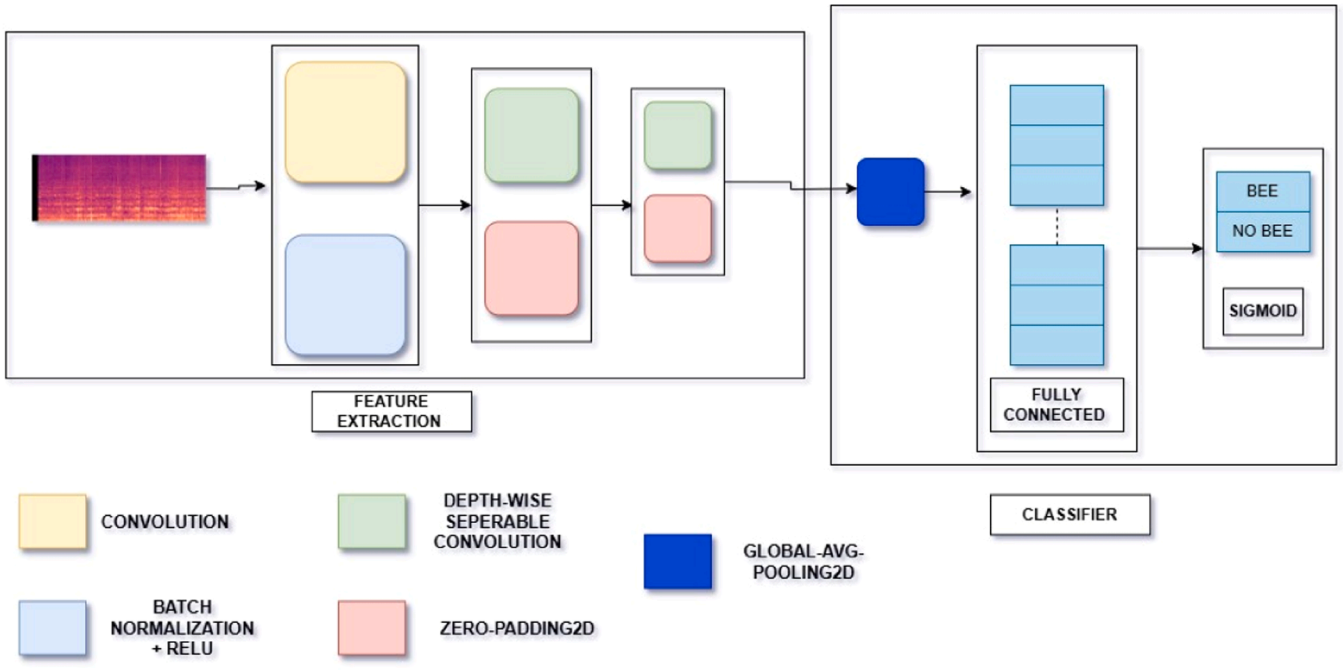



**Fig. 6.** Model Architecture Diagram.

**Table 8**
Performance and Threshold Adherence of Evaluated Models (Spectrogram Approach).

| Metric | EfficientNet-B0 | ResNet50 | MobileNet-V2 | Reason for Selection/ Exclusion |
|---|---|---|---|---|
| Accuracy, (Test Set) | 50.18% | 50.18% | 95.29% | MobileNetV2, performed best under the 80% threshold. |
| Precision | 25% | 25% | 95% | EfficientNet-B0 and ResNet50, failed under, strict thresholds. |
| Recall | 50% | 50% | 96% | MobileNetV2 achieved the most reliable predictions. |
| F1-Score | 33% | 33% | 95% | Other models were overfitted and biased to one class. |
| Threshold Adherence | Failed on testing and real-world data | Inconsistent due to strict thresholds | Consistent and generalized on real data | EfficientNet-B0 and ResNet50 missed the 80% threshold. |

### 5.3. Chromogram-Based Approach Results

The chromogram-based approach employed the MobileNetV2 model with two different learning rates for classification, and the results are summarized in Table 9. During experimentation, it was observed that reducing the learning rate to a very low value introduced a noticeable bias toward the "No Bee" class. This bias likely stemmed from the model's tendency to overfit to the majority class under such conservative updates, failing to adequately learn the distinguishing features of the "Bee" class. In contrast, a higher learning rate enabled more balanced learning across both classes, mitigating this issue and improving overall classification performance.

The performance of MobileNetV2 (95.29% accuracy) compared to EffcientNetB0 and ResNet50 can be attributed to both architectural efficiency and better suitability to noise rich spectrogram and chromogram representations as shown in Figs. 7–12. MobileNetV2 employs inverted residual blocks with linear bottlenecks, which preserve critical spectral cues such as harmonic structure and energy concentration while suppressing irrelevant background noise characteristics essential in heterogeneous mobile recordings. In contrast, ResNet50, with its deeper architecture, demonstrated higher sensitivity to non-bee environmental noise due to overfitting risks in limited labeled audio datasets, while E-cientNetB0 suffered performance degradation as its compound scaling approach assumes clean and stationary feature distributions, which do not fully align with dynamic field environments. Additionally, using complementary visual features (mel spectrograms + chromograms), along with an 80% confidence threshold, enabled MobileNetV2 to exploit both time frequency energy variations and pitch-invariant harmonic patterns, improving overall robustness. These results validate the design choices made in the methodology particularly low cost multi-device data collection, optimized chunking, and diverse noise exposure ultimately demonstrating that MobileNetV2 offers a better generalization trade-off for real-world acoustic hive monitoring.

**Table 9**
Comparison of MobileNetV2 Results on Chromograms.

| Metric | Learning Rate = 0.000001 | Learning Rate = 0.001 |
|---|---|---|
| Precision (No Bee) | 54% | 85% |
| Recall (No Bee) | 99% | 95% |
| F1-Score (No Bee) | 70% | 90% |
| Precision (Bee) | 98% | 96% |
| Recall (Bee) | 34% | 87% |
| F1-Score (Bee) | 51% | 91% |
| Overall Accuracy | 62.91% | 90.58% |

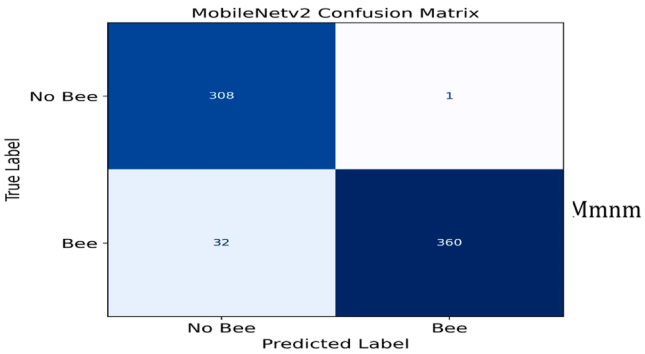



**Fig. 7.** Confusion matrix illustrating MobileNetV2 performance on spectrogram based classification using the selected confidence threshold.

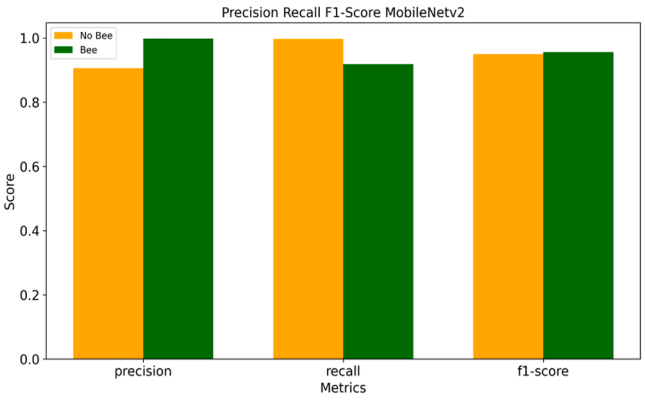



**Fig. 8.** Precision, recall, and F1-score comparison for MobileNetV2, demonstrating balanced predictive performance across both classes.

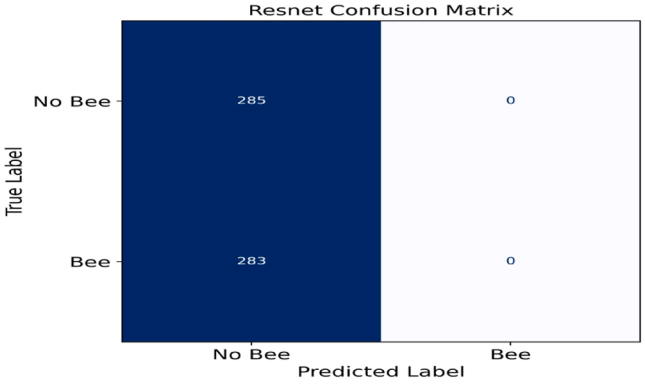



**Fig. 9.** Confusion matrix showing classification behavior of the ResNet50 model, highlighting its limited generalization on real world spectrogram data.

## 6. Conclusions and Future Work

In this study, we explored various machine learning approaches for classifying audio signals. The numerical approach models, although offering perfect accuracy on training data, failed to generalize effectively to unseen data, highlighting the need for better generalization

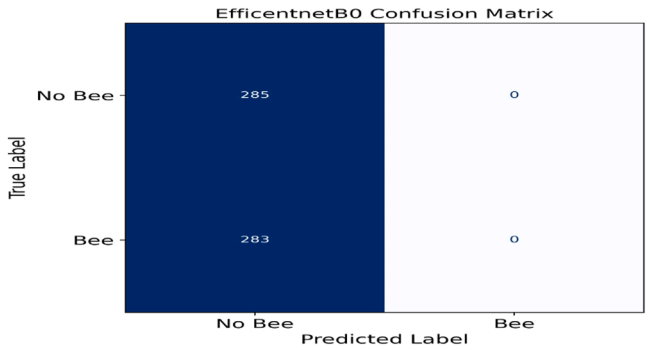



**Fig. 10.** Confusion matrix showing classification behavior of the ResNet50 model, highlighting its limited generalization on real world spectrogram Mmnm.

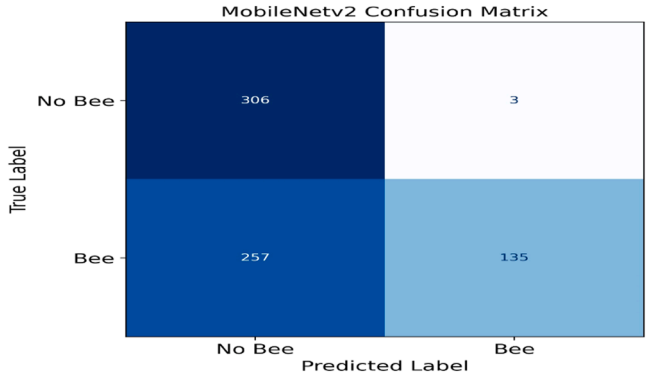



**Fig. 11.** Confusion matrix of E-cientNetB0, indicating misclassification trends and class bias under field-level noise conditions.

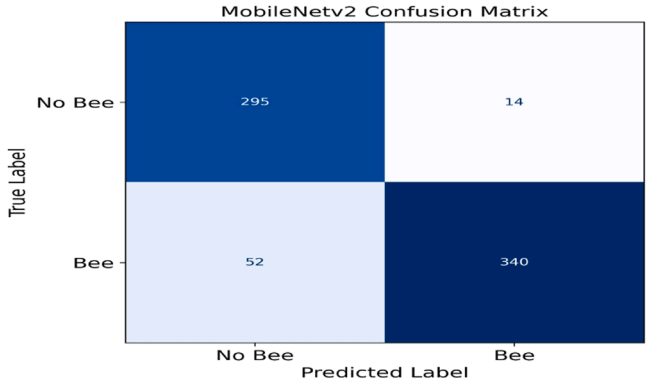



**Fig. 12.** Confusion matrix of EffcientNet B0, indicating misclassification trends and class bias under field-level noise conditions.

techniques in production environments.

On the other hand, the spectrogrambased deep learning models such as MobileNetV2 demonstrated excellent performance with high accuracy, showcasing the robustness of advanced deep learning architectures for real world applications.

Previous studies on distinguishing between "bee" and "not bee" sounds, have primarily relied on spectrograms as their main audio representation technique. These studies often utilized high quality microphones in controlled environments, leading to promising results on their collected datasets. However, despite their success in a laboratory setting, these models frequently failed to generalize to real world scenarios due to variations in environmental noise, microphone quality, and different recording conditions.

A significant limitation of existing research is the lack of exploration into mobile phone recordings as a viable data source. Most prior studies relied on specialized, high fidelity recording equipment, making their approaches impractical for widespread deployment, particularly in real world agricultural and beekeeping applications. To the best of our knowledge, no study has incorporated mobile phone recordings, which could provide a more accessible and scalable solution for monitoring bee populations in local farms.

By addressing these gaps, incorporating mobile phone recordings and merging diverse datasets, we develop a more practical and scalable approach for real world deployment. Our method considers local farm environments, where variable background noise and diverse acoustic conditions exist, making it more suitable for practical use. This novel approach demonstrates that deep learning models can still achieve high accuracy even with non specialized recording equipment, ensuring broader applicability in real world monitoring scenarios.

However, performance in extremely noisy environments and variability across mobile microphone sensors remain important limitations and opportunities for further improvement.

The present study establishes a foundational framework for distinguishing bee sounds from diverse environmental noise using deep learning models and mobile device recordings. While 5-fold cross-validation was performed to assess model stability, more extensive statistical tests—such as repeated runs, variance reporting, or confidence intervals—were not feasible due to current dataset size and computational constraints. Similarly, although recordings were collected from multiple devices and locations, the limited number of samples per device prevented meaningful leave-one-device-out or cross-site validation in this initial study.

Performance in extremely noisy environments and variability across mobile microphone sensors remain important limitations and opportunities for further improvement. In future work, we will focus on improving statistical robustness by performing extensive cross-validation and testing across additional geographic and device conditions. Furthermore, we will refine noise-handling strategies to enhance performance in highly dynamic environments and explore model optimization techniques for efficient on-device inference. These improvements will help further validate scalability and strengthen real-world deployment capability..

## Authors Statement

The authors confirm that all listed individuals have made substantial contributions to the work reported in the manuscript titled:" Spectrogram-Based Deep Learning Models for Acoustic Identification of Honey Bees in Complex Environmental Noises".

Contributor Roles (CRediT Taxonomy):

Muhammad Anus Khan – Department of Computer Science, Namal University, Namal Knowledge City, Mianwali, 42250, Punjab, Pakistan

**Contributions:** Conceptualization, Data Curation, Investigation, Software Development, Writing Original Draft.

Bilal Hassan Khan – Department of Computer Science, Namal University, Namal Knowledge City, Mianwali, 42250, Punjab, Pakistan.

Contributions: Methodology, Software Development, Formal Analysis, Writing Original Draft.

Shafiq ur Rehman Khan – Department of Computer Science, Namal University, Namal Knowledge City, Mianwali, 42250, Punjab, Pakistan

**Contributions:** Supervision, Conceptualization, Methodology, Project Administration, Writing – Review & Editing.

Ali Raza – Department of Software Engineering, University of Mianwali, University Road, Mianwali, 42200, Punjab, Pakistan

**Contributions:** Validation, Proofreading, Review & Editing.

Asif Raza – Department of Software Engineering, University of Mianwali, University Road, Mianwali, 42200, Punjab, Pakistan

**Contributions:** Validation, Proofreading, Review & Editing.

**Shehzad Ashraf Chaudhry** – Department of Computer Science and

Information Technology College of Engineering Abu Dhabi University, Abu Dhabi, UAE

Department of Software Engineering Faculty of Engineering and Architecture Nisantasi University, Istanbul, 3400, Turkey. Email

**Contributions:** Experimental setup understanding, revision of manuscript, revised version based on reviewers comments.

All authors have read and approved the final version of the manuscript and agree to be accountable for the content of the work.

## Declaration of generative AI and AI-assisted technologies in the writing process

During the preparation of this work, the author(s) used the Generative AI tool GPT for grammar and language correction. After using this tool, the author(s) reviewed and edited the content as needed and take (s) full responsibility for the content of the publication.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

The author is an Editorial Board Member/Editor-in-Chief/Associate Editor/Guest Editor for this journal and was not involved in the editorial review or the decision to publish this article.

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

There is no competing interest

## Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.mlwa.2025.100807.

## Data availability

Data will be made available on request.

## References

Andrew L. (2023). Bee audio object detection. https://www.kaggle.com/dsv/5135390. 10.34740/KAGGLE/DSV/5135390.

Aslan, C. E., Liang, C. T., Galindo, B., Hill, K., & Topete, W. (2016). The role of honey bees as pollinators in natural areas. *Natural Areas Journal, 36*(4), 478–488.

Astuti, P. K., Hegedűs, B., Oleksa, A., Bagi, Z., & Kusza, S. (2024). Buzzing with intelligence: Current issues in apiculture and the role of artificial intelligence (ai) to tackle it. *Insects, 15*(6), 418.

Bricout, A., Leleux, P., Acco, P., Escriba, C., Fourniols, J.-Y., Soto-Romero, G., & Floquet, R. (2024). Bee together: Joining bee audio datasets for hive extrapolation in ai-based monitoring. *Sensors, 24*(18), 6067.

Ellis, J. (2012). The honey bee crisis. *Outlooks on Pest Management, 23*(1), 35–40.

Genersch, E. (2010). Honey bee pathology: Current threats to honey bees and beekeeping. *Applied Microbiology and Biotechnology, 87*, 87–97.

Guzman-Novoa, E., Cork, S., Hall, D., & Liljebjelke, K. (2016). Colony collapse disorder and other threats to honey bees. *One Health Case Studies: Addressing Complex Problems in a Changing World*, 204–216. pages.

Harano, K., Sasaki, M., & Sasaki, K. (2013). Acoustic monitoring of honeybee colonies to identify colony status and queen presence. *Apidologie, 44*(3), 276–290.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778). pages.

Hung, K.-L. J., Kingston, J. M., Albrecht, M., Holway, D. A., & Kohn, J. R. (2018). The worldwide importance of honey bees as pollinators in natural habitats. *Proceedings of the Royal Society B: Biological Sciences, 285*(1870), Article 20172140.

Iqbal, K., Alabdullah, B., Al Mudawi, N., Algarni, A., Jalal, A., & Park, J. (2024). Empirical analysis of honeybees acoustics as biosensors signals for swarm prediction in beehives. *Ieee Access*.

Kanelis, D., Liolios, V., Papadopoulou, F., Rodopoulou, M.-A., Kampelopoulos, D., Siozios, K., & Tananaki, C. (2023). Decoding the behavior of a queenless colony using sound signals. *Biology, 12*(11), 1392.

Kim, J., Cho, S., & Lee, H. K. (2021). Development of a sound-based monitoring system for honey bee colonies using edge ai. *Sensors, 21*(16), 5553.

McFee, B., Raffel, C., Liang, D., Ellis, D. P. W., McVicar, M., Battenberg, E., & Nieto, O. (2015). librosa: Audio and music signal analysis in python. *Proceedings of the 14th Python in Science Conference, 8*, 18–25. pages.

Orlowska, A., Fourer, D., Gavini, J.-P., & Cassou-Ribehart, D. (2021). Honey bee queen presence detection from audio field recordings using summarized spectrogram and convolutional neural networks. In *International Conference on Intelligent Systems Design and Applications* (pp. 83–92). Springer. pages.

Patel, V., Pauli, N., Biggs, E., Barbour, L., & Boruff, B. (2021). Why bees are critical for achieving sustainable development. *Ambio, 50*, 49–59.

Phan, T.-T.-H. (2024). Synergistic mel-frequency cepstral coefficients and shorttime fourier transform for enhanced bee states detection using machine learning. In *International Conference on Intelligent Systems and Data Science* (pp. 166–177). Springer. pages.

Ramírez, J., Gómez-García, J. A., et al. (2015). Classification of honeybee sound signals using machine learning techniques. *Computers and Electronics in Agriculture, 119*, 123–132.

Rustam, F., Sharif, M. Z., Aljedaani, W., Lee, E., & Ashraf, I. (2024). Bee detection in bee hives using selective features from acoustic data. *Multimedia Tools and Applications, 83*(8), 23269–23296.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L.-C. (2018). Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 4510–4520). pages.

Soares, B. S. (2022). Mfcc-based descriptor for bee queen presence detection. *Expert Systems with Applications, 201*, Article 117104.

Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the International Conference on Machine Learning* (pp. 6105–6114). PMLR. pages.

Terenzi, A., Ortolani, N., Nolasco, I., Benetos, E., & Cecchi, S. (2022). Comparison of feature extraction methods for sound-based classification of honey bee activity. *IEEE/ACM Transactions on Audio, Speech, and Language Processing, 30*, 112–122.

Thakur, M. (2012). Bees as pollinators—biodiversity and conservation. *International Research Journal of Agricultural Science and Soil Science, 2*(1), 1–7.