Targeted Feature Extraction for Bird Sound Classification Systems

Technical Report

Ifeanyi Oguamanam, Lucas Machowski, Marija Simic, Sri Krishnan
Signal Analysis Research (SAR) Group, Dept. Electrical, Computer, and Biomedical Engineering
Toronto Metropolitan University (TMU)
350 Victoria Street, Toronto, Ontario M5B 2K3, Canada
(ioguamanam)(lucas.machowski)(marija.simic)(krishnan)@torontomu.ca

1. ABSTRACT

This report will describe the attempt made by the Signal Analysis Research (SAR) group in Toronto Metropolitan University (TMU) to translate the familiar challenge of bird sound classification to a resource-constrained microcontroller. In order to describe the algorithms in place, we will detail our analysis of the design problem, and its performance against benchmarks of inference time, model size, and memory usage.

In order to describe the rationale for the feature extraction and classification models created, this team will describe the design process that led to our solutions. Our methodology focuses on Mel-frequency cepstral coefficients (MFCC), Gammatone frequency cepstral coefficients (GFCC), and spectral flux features. Different machine learning models were explored such as linear Support Vector Machines (SVMs) and shallow Convolutional Neural Network (CNN) architectures, quantifying their performance with their corresponding feature by testing accuracy and efficiency. The three systems achieved testing accuracies from 89.1% to 92.65%, with on-device storage ranging from 24 B to 18.36 KB of code, RAM usage between 780 B and 34.5 KB, and end-to-end inference time of 196 us to under 42ms. These results demonstrate that efficient, low computational models can achieve strong performance, supporting practical deployment of automated acoustic monitoring in resource limited environments.

Index Terms— *Emberiza citrinella, TinyML, Feature Extraction, Time-Frequency Analyses, Spectral Flux*.

2. INTRODUCTION

Automated acoustics monitoring enables large-scale and long term assessments as it uses technology to detect, analyze, and identify sound rather than relying on human observation. Although traditional artificial intelligence methodologies have been proven to be effective in predictive classification systems [1], there is a growing need that these systems are not only accurate but efficient enough to run on compact and lower power devices - the TinyML paradigm. The DCASE 2025 Task 3 challenged the creation of a robust detection system which was required to correctly detect the Emberiza citrinella (Yellowhammer Bird) in the presence of background noise, overlapping sounds, and varying recording conditions [2]. The model needed to be lightweight with low computational and memory requirements, making it ideal for edge hardware.

The technical approach to this challenge involved three main stages: audio preprocessing, feature extraction, and model training and evaluation. While deep learning models may require less preprocessing, they require significant computational resources from the device, leading to incompatibility with applications on many edge devices.

This work instead explores a number of carefully chosen audio features to determine if strong performance can be achieved with minimal computational cost. This paper will describe the performance of: Mel Spectrograms, Gammatonne Frequency Cepstral Coefficients (GFCCs), and spectral flux statistics.

2.1 Feature Selection

2.1.1 Mel Spectrogram

One prevalent time-frequency (TF) characteristic capable of event-detection is the mel spectrogram. The mel spectrogram combines windows of overlapping Short Time Fourier Transforms (STFTs) with logarithmically scaled frequency bins to produce a representation of the spectral distribution of energy in a signal over time.

These images contain audio signatures that have been proven to be processible by traditional and lightweight AI implementations for the purpose of audio classification [3, 4, 5].

2.1.2 Gammatone Frequency Cepstral coefficient (GFCC)

GFCCs are a compressed version of a spectrogram inspired by how the human auditory system processes sound. They are computed by taking the short-time fourier transform (STFT) of the input signal and applying a gammatone filterbank to obtain the gammatone spectrogram. Then, the discrete cosine transform (DCT) is computed to obtain a set of coefficients for each window of the signal. These coefficients describe distinct aspects of the spectral shape of the input signal. For example, the first coefficient represents the log-energy, or overall loudness of the signal window, lower-order coefficients capture the broad spectral envelope, while higher-order coefficients describe fine spectral details. By analyzing these coefficients in time, we can obtain a compressed, yet highly informative feature vector for machine learning (ML) classification.

2.1.3 Spectral flux

Spectral flux is a TF analysis, where the frequency content/power spectrum of the signal is examined between consecutive audio

spectral frames, measuring its rate of change [6]. The measurement quantifies how quickly the spectrum involves over time; higher values indicate rapid spectral changes which are typically seen in transient or non-stationary sounds [6]. It is particularly effective in distinguishing between background noise and biologically relevant signals, such as bird vocalizations, as well as onset detection, speech and music discrimination, and bioacoustic identification [7]. Spectral flux requires only a frame-wise spectral magnitude comparison making it computationally effective [7].

Three different signal classification methods were created to compare accuracy and resource efficiency:

- Method A - Mel Spectroscopy as an image, input into a shallow Convolutional Neural Network (CNN)
- Method B - GFCC-based model using a simple linear Support Vector Machine (SVM).
- Method C - Spectral Flux features to train a simple linear SVM.

By evaluating these approaches, this work aims to assess the tradeoffs between accuracy and efficiency, and help provide insights on a practical deployment of an automated acoustics detection system where resources are limited.

3. METHODOLOGY

3.1 Preprocessing

Audio was sampled at a rate of 14 kHz, as the team found most differential spectral information between the Yellowhammer bird (YH) and non-target bird samples visible below the ranges of 7 kHz.

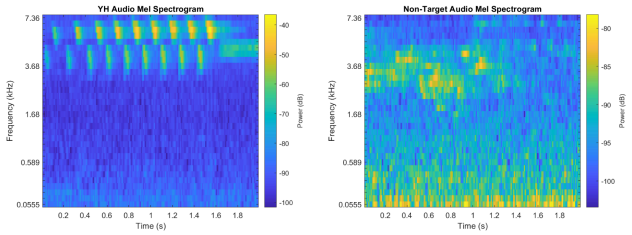

Figure 1: Typical Audio Spectrogram behaviour of YH and Non-Target Bird Sounds

It was also determined that a sufficient classification could be determined by a 1.5 second sample, leading to a total of 21,000 samples. No digital filtering was performed on the signal

3.2 Feature Extraction

3.2.1 Method A

Noting the consistent ridges present in the YH audio samples, the team chose to create 40 frequency bands between 3 kHz and 6.750 kHz, and a mel spectrogram, creating a (40 by 25 image) capable of being passed to the model input. This way, only contrasting regions of the TF spectrogram are passed to the next phase of the model. For this pipeline, the BioDCASE starter framework was incorporated [8].

3.2.2 Method B

For each set of data, the first and second GFCCs were computed. Plotting the GFCCs with respect to time gave a clear distinction between yellowhammer cries relative to other birds, shown in the plots below.

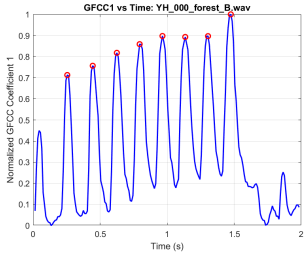

Figure 2: Yellowhammer Bird First GFCC vs Time Plot

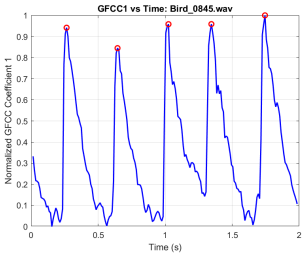

Figure 3: Negative Sample First GFCC vs Time Plot

Through the plots above, it can be seen that the yellowhammer cry GFCCs follow a consistent, and different pattern than other birds. Several statistical features were considered using this information. The set of features that resulted in the best results were autocorrelation and periodogram related features for the first GFCC, and form factor and zero crossing rate features for the second GFCC. Autocorrelation measures how similar the GFCC1 signal is to a delayed version of itself. A reference window is taken and slid across the GFCC vs time signal, for each frame, the dot product between the static and moving frames are taken.

$$Corr[i] = \sum_{m=0}^{L-1} x[m] \cdot x[n] \quad (1)$$

A high correlation indicates that the signal contains periodic patterns, a feature which is consistent with the yellowhammer bird cries unlike other birds. The periodogram ratio feature quantifies how much energy is concentrated in the dominant frequency within the signal. To obtain this feature, the power spectral density (PSD) is computed at each frequency, providing the amount of power at that frequency. Then, the ratio between the maximum PSD and the total PSD is taken. A high ratio indicates that there is a dominant frequency within the signal, revealing periodicity. We expect that the yellowhammer cries

should have a higher power ratio than other bird cries. The zero-crossing rate computes how frequently the 0.5 threshold is crossed for a 0 to 1 normalized GFCC2 vs time signal. Finally, form factor *(FF*) measures the complexity of the GFCC2 signal by comparing the standard deviation and variance of the signal and its derivatives:

$$FF = \frac{\sigma''\sigma}{\sigma^{2'}} \quad (2)$$

3.2.3 Method C

For each preprocessed audio file, the spectral flux was computed using a STFT. The audio was divided into overlapping frames with a length of 50ms and a 25ms hop (overlap), and the magnitude spectrum was computed for each. The Yellowhammer uses frequencies between 2 kHz to 8 kHz with its most prominent frequencies being in the range of 5 kHz-8 kHz. To focus on the prominent frequency range of the Yellowhammer, only the spectral bins between 5 kHz-8 kHz were kept and analyzed. Spectral flux was calculated as the squared difference in magnitude spectrum between each successive frames within the frequency band:

$$Flux(t) = \sqrt{\sum_k (X_t(k) - X_{t-1}(k))^2} \quad (3)$$

Where $X_t(k)$ is the magnitude of the *k*-th frequency bin at frame *t*, and the summation is over all frequency bins within the range of 5 kHz-8 kHz. The mean and variance of the spectral flux values across all the frames in each signal were used as the final features.

Table 1. Examples of Spectral Flux Mean and Variance for Yellowhammer, background noise, and other birds

Audio	Mean	Variance (Standard Deviation)
Yellowhammer	334.1239	417.0241
Yellowhammer	349.1590	412.5898
Bird	289.3187	215.0526
Bird	346.0958	287.4858
Background	356.2084	150.8736
Background	360.1533	55.3285

3.3 Machine Learning Implementation

3.3.1 Method A

For the purposes of this classification, we found a shallow CNN to be more than sufficient. Classification accuracies of up to 93% could be achieved with just one convolution layer, one max pooling layer, and a dropout rate of 40%.

3.3.2 Methods B and C

A linear SVM classifier was trained using the extracted spectral flux features. The model was fit using two dimensional inputs, the extracted mean and variance of spectrum flux, for all training samples which had positive class labels for Yellowhammer and negative labels for the other audio files. The model was validated with a separate set of validation files.

4. RESULTS

Table 2: Resource Efficiency for all Models

	Method A	Method B	Method C
Training Model			
Model Size (Bytes) TF model	3.6 kB	40 bytes	24 bytes
Model Parameters	922	10	6
Input shape	(40, 25)	(4, 1)	(2, 1)
Tiny Model			
TFLite Micro Size (kB)	18.364 kB	N.E	1848 bytes
Memory Usage on Target Device (kB)	34.46 kB	N.E	N.E
Inference Time			
Feature Extraction (μs)	34520 μs	N.E	N.E
Model Inference (μs)	7397 μs	N.E	196 μs
Classification Performance			
Testing Accuracy (%)	92.65%	91.18%	89.10%
False Positive (%)	0.535%	4.14%	21.52%
False Negative (%)	19.9%	17.4%	8.08%
N.E. = Not experimented			

4.1 Accuracy Rates

4.1.1 Method A

The spectral image performs well as a CNN model input, generating an average accuracy of 92.65%. With the False Positive rate at 0.53% and the False Negative rate at 20%, one could determine that the system does a significantly better job at filtering non-target bird sounds at the cost of under-identifying positive samples.

4.1.2 Method B

Combining the four features computed from the first and second GFCC features provided an accuracy of 90.18%, with a false positive rate of 4.14% and a false negative rate of 17.40%. The higher false negative rate of this model is likely due to certain yellowhammer cries being recorded with high noise levels, which attenuated the periodicity of the signal, resulting in an altered frequency composition which was less differentiable through GFCCs.

4.1.3 Method C

The spectral flux linear SVM model achieved a testing accuracy of 94.01%, demonstrating its strong performance of detecting Yellowhammer acoustics under varied conditions with only two features, mean and variance spectral flux. The model has a false positive of 21.52% indicating that some background and other bird acoustics were incorrectly classified as Yellowhammer. However, the false negative percentage is only 8.08% meaning that the model is far less likely to misclassify Yellowhammer calls. Overall, the model has a high accuracy and relatively low false negative suggesting that the model is reliable for automated bird detection.

4.2 Efficiency

4.2.1 Method A

Resource efficiency, the combined effect of a highly condensed model and an efficient feature extraction process, is the main limiting factor pertaining to the deployment of ML models on edge devices. The current pipeline boasts an inference time of 41917 μs, with 34520 of that time centered on extracting the spectrogram image.With an entire inference in under a second, this solution presents itself as very time conservative. At the same time, it boasts a conservative arena allocation of 18 kB during inference and classification.

4.2.2 Method B

The GFCC feature model is highly compressible; considering the first two GFCCs, TF information of the bird cry signal is represented in two values per window. Using the statistical parameters discussed above as features, a 1x4 feature vector of these features is able to capture TF trends for each sample, providing a highly compressed model for classification. However, the creation of the GFCC statistics is a highly resource intensive process which should not be understated.

4.2.3 Method C

This model is lightweight and fast, allowing it to be suitable for resource-constrained devices. The TensorFlow model size is only 24 bytes, and
has a corresponding TFLite tiny model of just 1848 bytes. There is minimal memory usage on the targeted device of only 780 bytes, and the inference is completed in just 196 μs. The model has only 6 internal model parameters and 2 input features achieving detection accuracy with low computation. Due to the simplicity of the classifier, the majority of the computation effort is shifted to the feature extraction stage. Digital signal processing such as windowing, SFFT, and the spectral flux calculation takes up the RAM and timing requirements. The overall efficiency of Model C not only reflects the model inference cost, but also the memory and computational overhead of audio processing in real-time.

5. DISCUSSION

For this submission, both mel and gammatone filterbanks were used as part of two different models. In comparing the results, it is seen that using gammatone filterbanks to compute GFCCs provided higher accuracy than analyzing mel spectrograms. Since animal sound data typically contains high amounts of noise, gammatone filterbanks are the ideal choice through the use of sharp cutoff bandpass filters. Since noise has a broad frequency band, one specific noise frequency will likely only appear in one filter if a small overlap is used. However, with the mel filterbank, which uses triangular filters, a particular noise frequency can be captured by multiple filters, causing noise amplification relative to the bird sounds. The two plots below are a mel and gammatone spectrogram for the same yellowhammer bird sample. It is seen that due to the difference in filter structure, the gammatone spectrogram contains higher contrast between the bird cry and background noise.

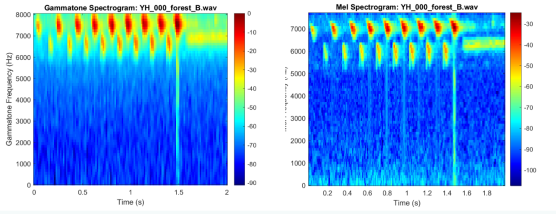

Figure 4: Gammatone Spectrogram (Left) vs Mel Spectrogram (Right) Comparison for Yellowhammer Bird Cry

It is no surprise, however, that the compression of the GFCC images to a handful of statistics, while having benefits on the model side, has significant drawbacks on the resource efficiency of the application.

Spectral flux variance showed to be an effective differentiation factor in bird sound classification and background sounds. While the mean flux represents the overall rate of spectral change, the variance of the flux across the audio frames reflects the fluctuation and irregularities. Using the files, Yellowhammer typically exhibited a greater variance (Table 1) compared to non-target sounds, indicating a more dynamic, transient sound compared to

the background noises and more regular calls of the non-target birds. However, it should be noted that this SVM-inspired approach has limitations in application and generalizability. The simplicity of the model allows it to be interpretable and efficient, but its performance may decrease in more complex acoustic environments. This approach works best when the classifications are linearly separable based on the chosen features, spectral flux mean and variance, but in scenarios where the acoustics are more nuanced and require more modeling on temporal dependency than more features and a model capable of capturing deeper/complex patterns would be necessary [9].

6. CONCLUSION

The Mel Spectrogram provides a versatile audio fingerprint to identify audio samples, seen very obviously by its high accuracy in identifying the YH birds against not just background signals, but also other bird sounds. With localized frequency ranges tweaked through experimentation, we can create low-resolution that can still serve as a signature.

We also found the spectral flux and GFCC characteristics capable of differentiating the positive and negative bird audio samples with the added benefit of reducing model size. Further investigation may go into comparing the performance of a system with minimal feature extraction and a deep neural network to one of our lightweight model-centered systems.

7. REFERENCES

[1] Yadav, S. Gaikwad, T. Kuigade, and A. Patil, “MUSIC CHORD PREDICTION USING MACHINE LEARNING,” International Research Journal of Modernization in Engineering Technology and Science, vol. 5, no. 12, pp. 194–199, 2023. doi:10.56726/IRJMETS46945.

[2] I. Morandi, P. Linhart, M. Kwak, and T. Petrusková, BioDCASE 2025 Task 3: Bioacoustics for Tiny Hardware Development Set. Zenodo, 2025. [Online]. Available: https://doi.org/10.5281/zenodo.15228365

[3] M. E. ElAlami, S. M. K. Tobar, S. M. Khater, and Eman. A. Esmaeil, “Texture Feature and Mel-Spectrogram Analysis for Music Sound Classification,” Int. J. Adv. Comput. Sci. Appl., vol. 15, no. 9, 2024, doi: 10.14569/IJACSA.2024.0150918.

[4] A. Bawitlung and S. K. Dash, “Genre classification in music using Convolutional Neural Networks,” Lecture Notes in Computer Science, pp. 397–409, Oct. 2023. doi:10.1007/978-981-99-7339-2_33.

[5] T. Li, “Optimizing the configuration of deep learning models for music genre classification,” Heliyon, vol. 10, no. 2, Jan. 2024. doi:10.1016/j.heliyon.2024.e24892.

[6] “Spectral flux,” Spectral Flux - an overview | ScienceDirect Topics, https://www.sciencedirect.com/topics/engineering/spectral-flux. (accessed Jun. 14, 2025).

[7] Speech/Audio Signal Classification Using Spectral Flux Pattern Recognition | IEEE Conference publication | IEEE Xplore, https://ieeexplore.ieee.org/abstract/document/6363260/ (accessed Jun. 15, 2025).

[8] G. Carmantini, F. Förstner, C. Isik, and S. Kahl, BioDCASE-Tiny 2025: A Framework for Bird Species Recognition on Resource-Constrained Hardware. Software. Cornell University and Chemnitz University of Technology, 2025. [Online]. Available: https://github.com/birdnet-team/BioDCASE-Tiny-2025

[9] G. Sharma, K. Umapathy, and S. Krishnan, 2020. Trends in audio signal feature extraction methods. *Applied Acoustics*, *158*, p.107020.