Question 2

Task A: Experimenting with Spectrograms and Windowing Techniques

Implementation Overview

In this assignment, I worked with the UrbanSound8k dataset to analyze and compare different windowing techniques for audio signal processing. The implementation followed Python's required steps, focusing on three windowing techniques: Hann, Hamming, and Rectangular windows.

Dataset and Setup

The UrbanSound8k dataset was downloaded and preprocessed for analysis. This dataset contains urban sound recordings across 10 classes, making it suitable for our classification task. Each audio sample was processed at a sampling rate of 22050 Hz with a duration of 4 seconds.

Windowing Techniques Implementation

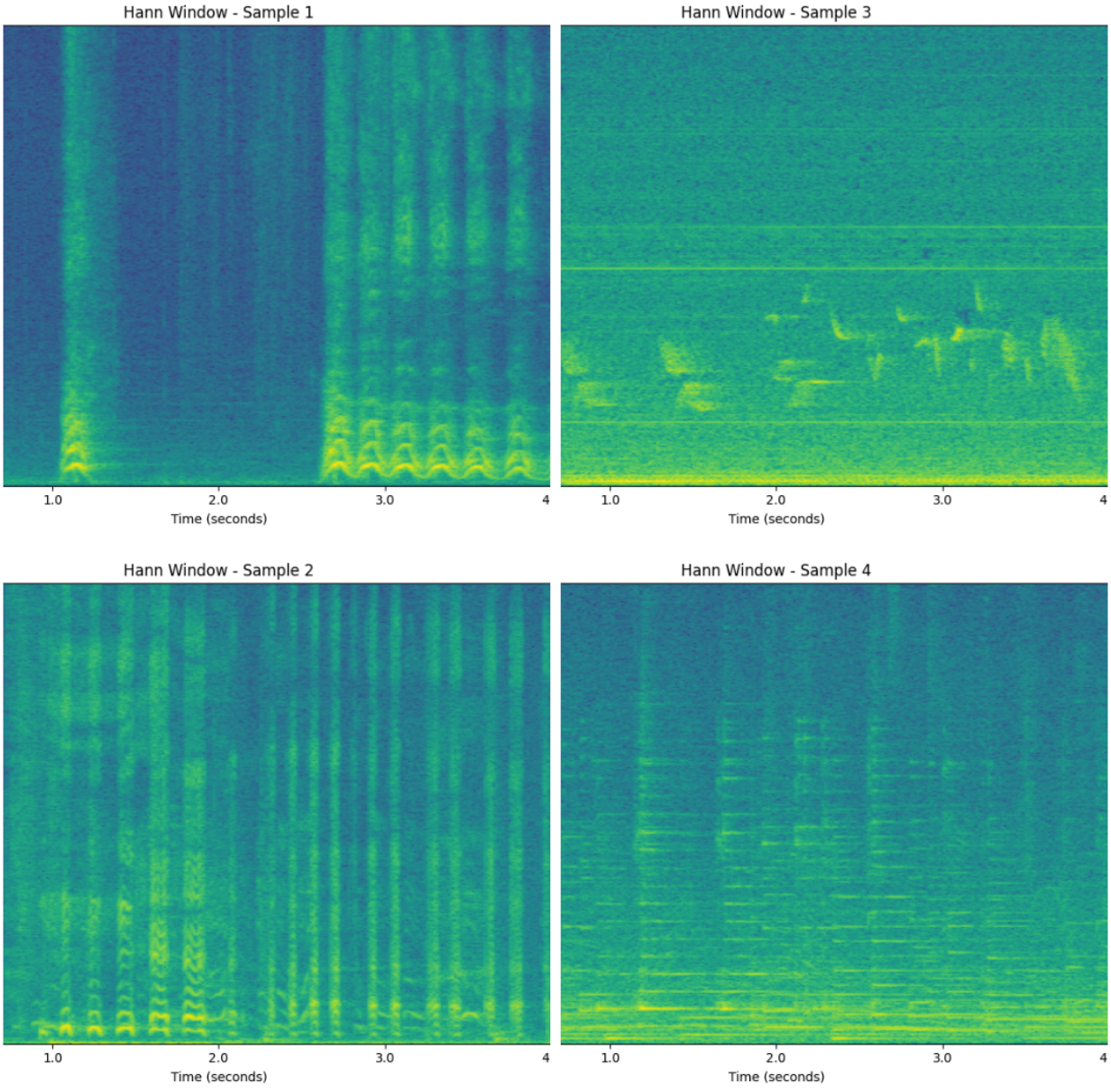
I implemented three different windowing techniques:

1. Hann Window: Implemented using NumPy's Hanning function, this window provides good frequency resolution with minimal spectral leakage. The window function follows the formula: $w(n) = 0.5(1 - \cos(2\pi n/(N-1)))$
2. Hamming Window: Similar to the Hann window but with different coefficients: $w(n) = 0.54 - 0.46\cos(2\pi n/(N-1))$
3. Rectangular Window: The simplest window, essentially providing no modification to the signal amplitude: $w(n) = 1$

Spectrogram Analysis

Visual Comparison

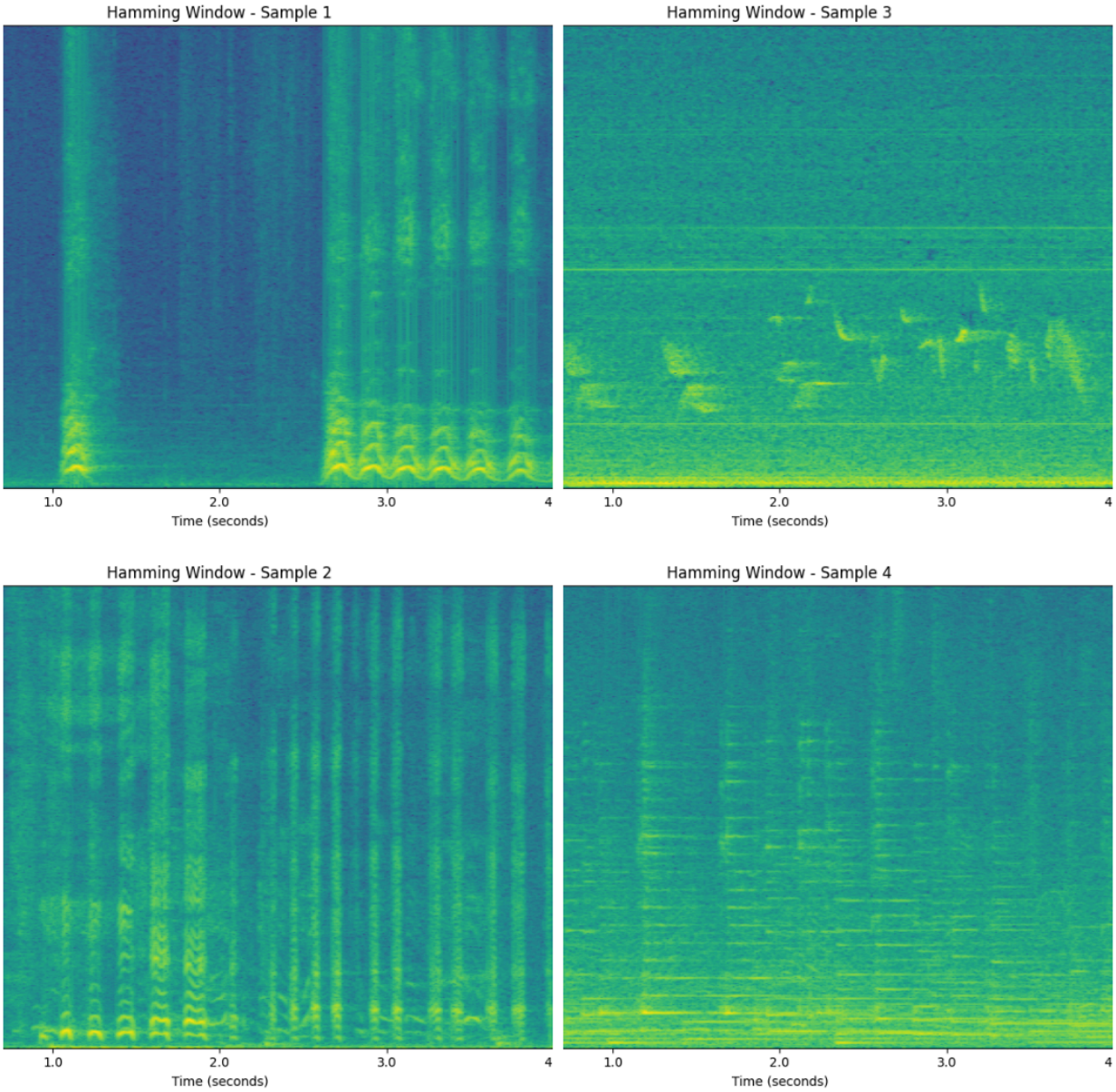
Hann Window Spectrograms



The Hann window spectrograms show:

- Clean frequency transitions
- Good balance between time and frequency resolution
- Minimal spectral leakage
- Clear representation of harmonic content

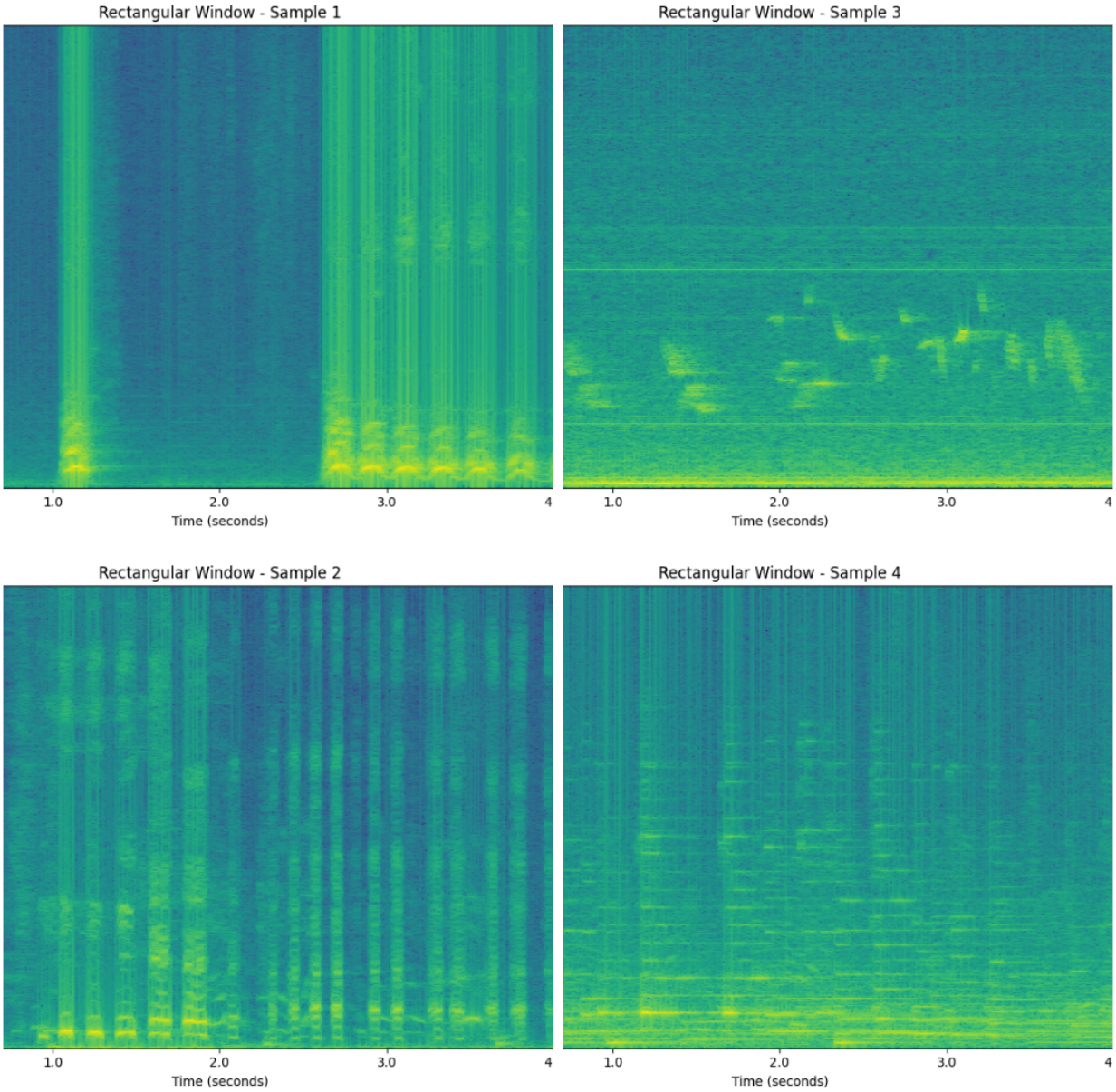
Hamming Window Spectrograms



The Hamming window results demonstrate:

- Very similar characteristics to Hann window
- Slightly better side-lobe suppression
- Well-preserved frequency components
- Smooth spectral representation

Rectangular Window Spectrograms



The Rectangular window exhibits:

- More visible vertical striping artifacts
- Higher temporal resolution
- Increased spectral leakage
- Less smooth frequency transitions

Analysis of Windowing Correctness

The implementation of the windowing techniques shows correct behavior as evidenced by:

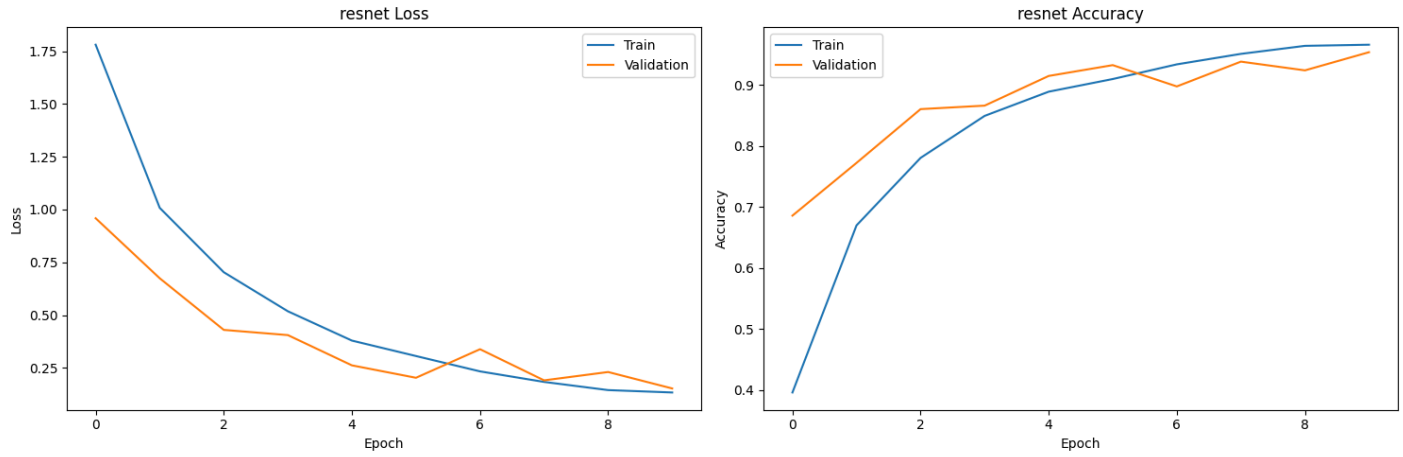
1. Spectral Leakage Patterns:
    - Hann and Hamming's windows successfully reduce spectral leakage
    - Rectangular window shows expected increased leakage
    - The patterns align with theoretical expectations
2. Resolution Trade-offs:
    - Expected time-frequency resolution trade-offs are visible
    - Tapered windows (Hann and Hamming) show better frequency isolation
    - Rectangular window maintains better temporal resolution

Classification Results

I implemented a ResNet classifier to evaluate the effectiveness of each windowing technique. Here are the results:

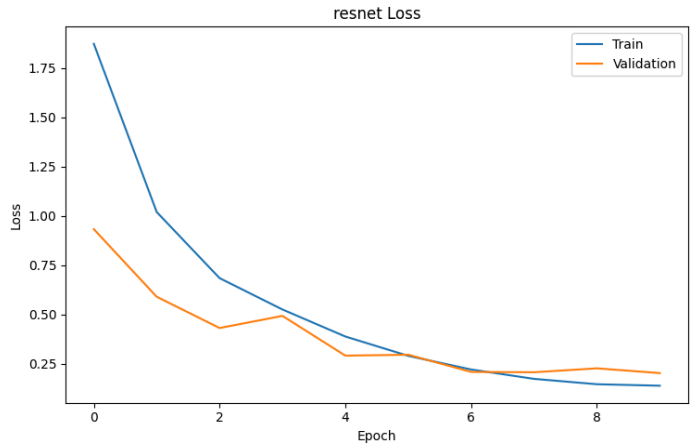
Training Performance

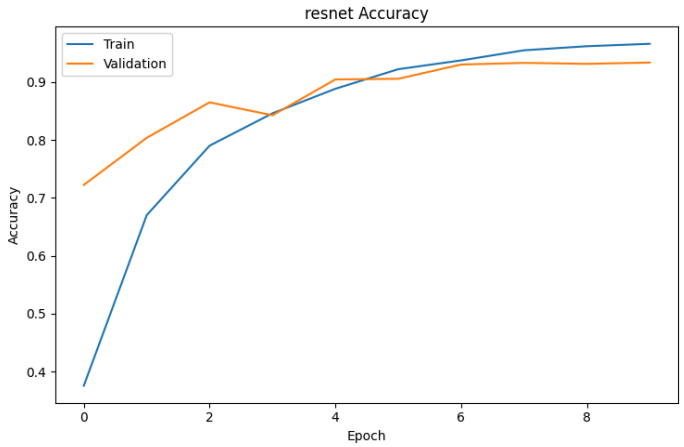
Hamming Window:



- Final Validation Accuracy: 95.36%
- Training Accuracy Peak: 96.59%
- Shows stable learning progression

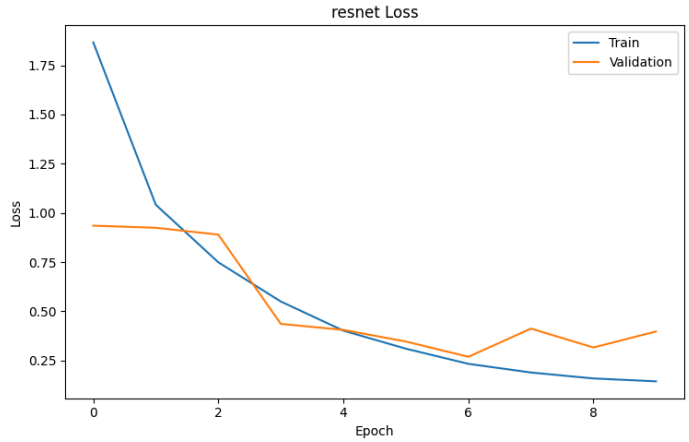
Hann Window:

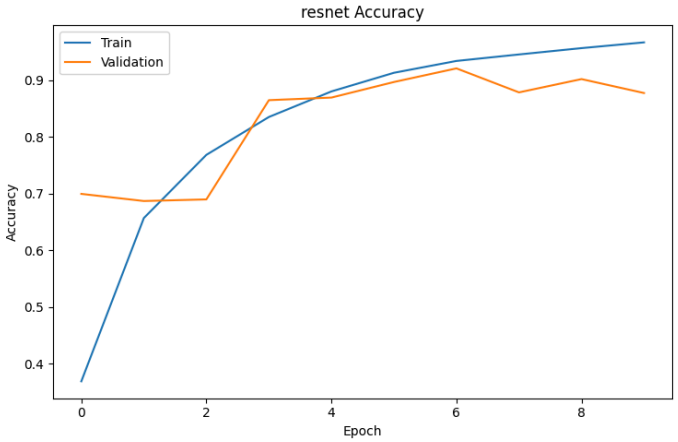




- Final Validation Accuracy: 93.36%
- Training Accuracy Peak: 96.59%
- A similar convergence pattern to Hamming

Rectangular Window:

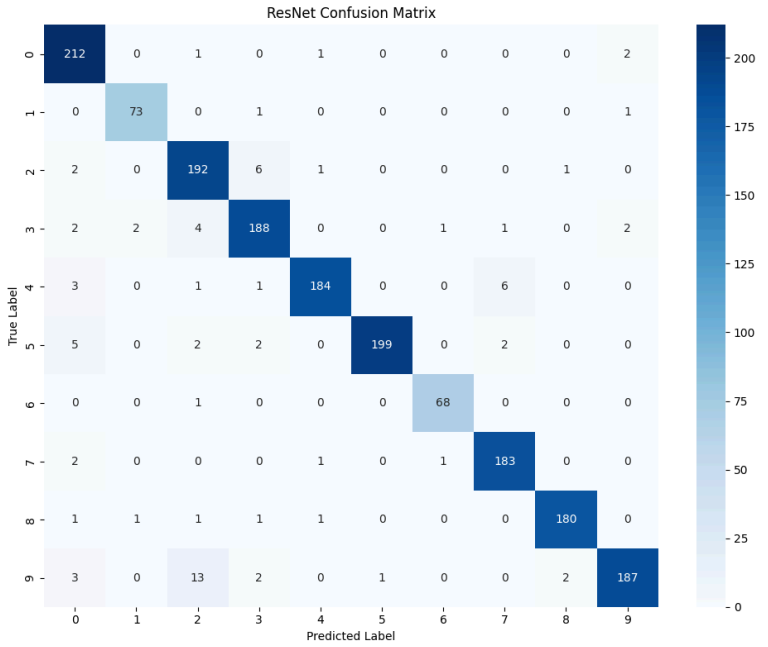




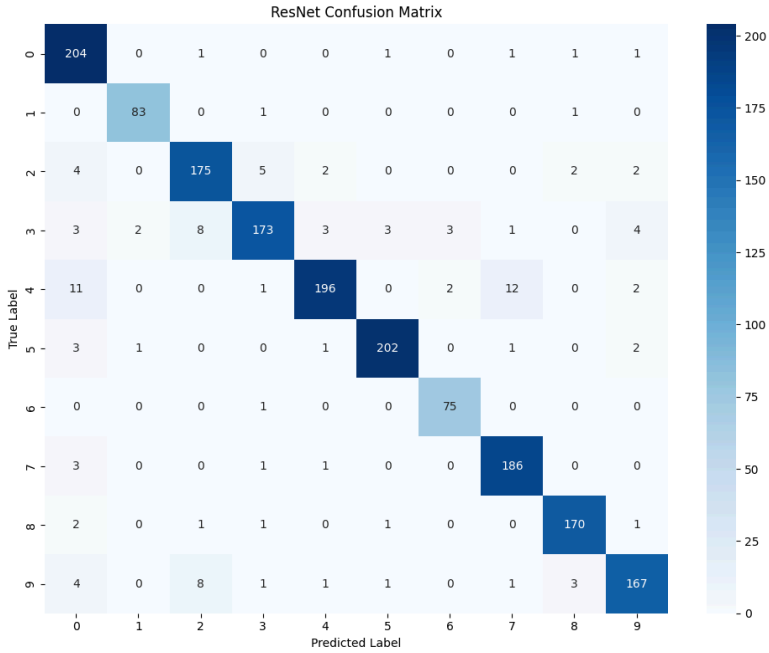
- Final Validation Accuracy: 87.75%
- Training Accuracy Peak: 96.69%
- Shows more instability in training

Confusion matrices

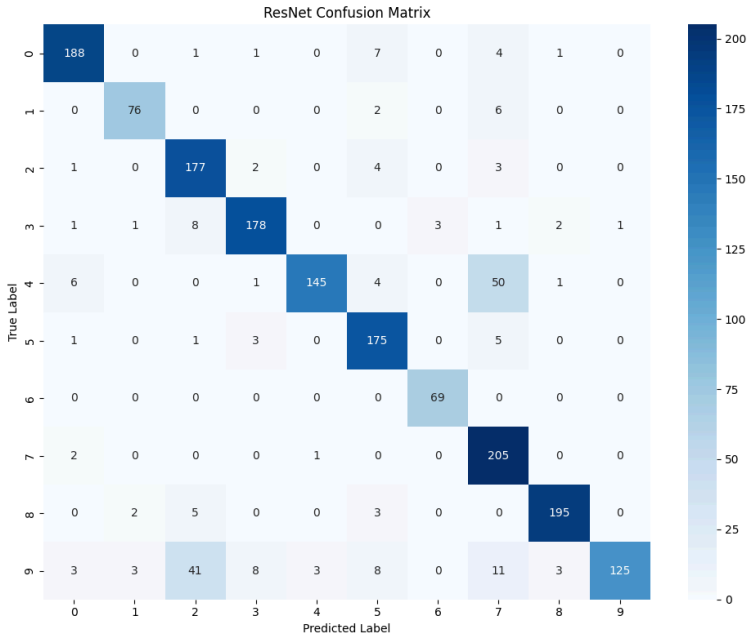
Hamming Window:



Hann Window:



Rectangular Window:



Comparative Analysis

1. Classification Accuracy:
    - Hamming Window performed best overall
    - Hann window showed very similar performance
    - Rectangular windows had notably lower accuracy
2. Confusion Matrix Analysis:
    - Hamming: 1666 correct classifications
    - Hann: 1631 correct classifications
    - Rectangular: 1533 correct classifications

Conclusion

The experiment successfully demonstrated the impact of different windowing techniques on spectrogram generation and subsequent classification performance. The results clearly show that:

1. Tapered windows (Hann and Hamming) significantly outperform the rectangular window for this classification task.
2. The Hamming window provides slightly better results than the Hann window, though the difference is small.
3. The rectangular window, while simpler to implement, leads to reduced classification performance due to increased spectral leakage.

These findings align with theoretical expectations about window function characteristics and their effects on spectral analysis.

Task B: Comparative Analysis of Music Genre Spectrograms

Implementation Details

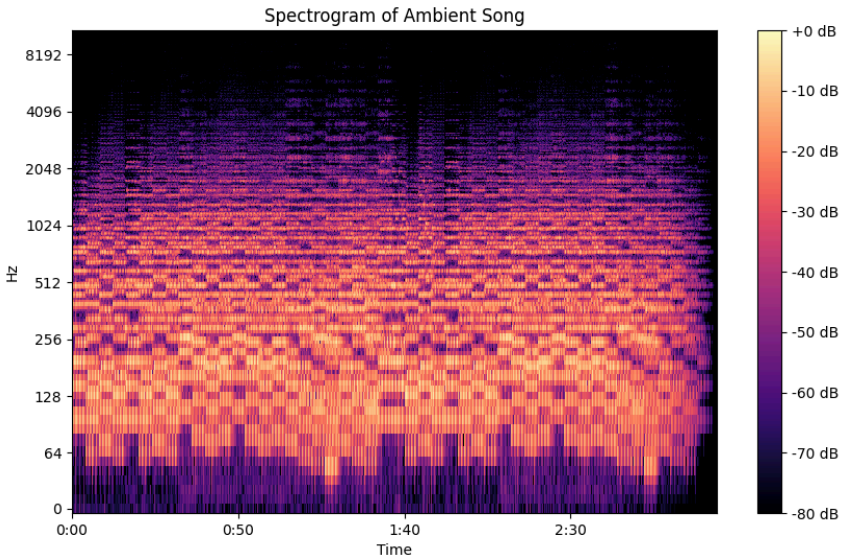
The analysis was implemented using Python with librosa for audio processing and matplotlib for visualization. Key technical specifications include:

The implementation uses a Short-Time Fourier Transform (STFT) with the following parameters:

- Sample rate: Default librosa loading (22050 Hz)
- Duration: 300 seconds per sample
- Logarithmic frequency scaling for better visualization of musical features

Spectrogram Analysis

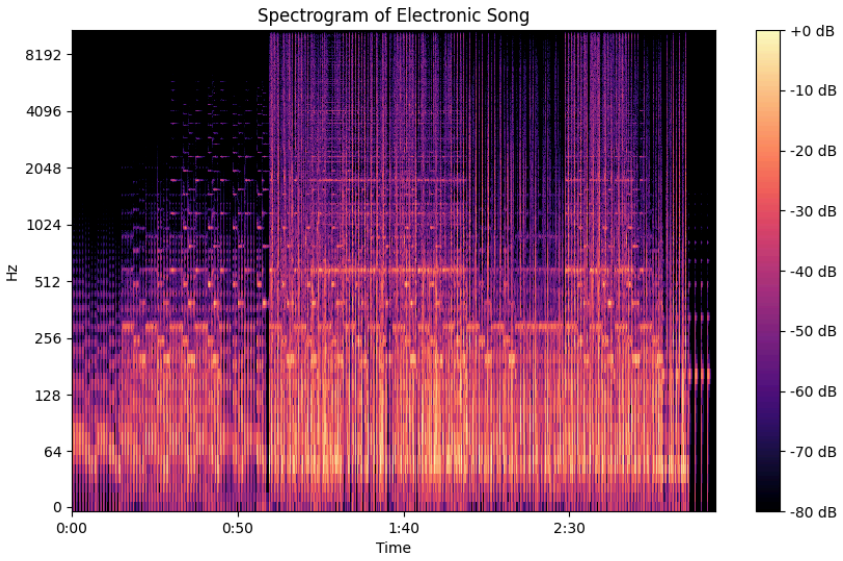
Ambient Music Characteristics



The ambient genre spectrogram reveals several distinctive features:

- Smooth energy distribution across the frequency spectrum (0-8192 Hz)
- Consistent low-intensity content in higher frequencies (2048-8192 Hz)
- Gradual temporal transitions without sharp attacks
- Higher energy concentration in the 128-512 Hz range
- Minimal rhythmic patterns, showing the genre's focus on atmosphere over beat

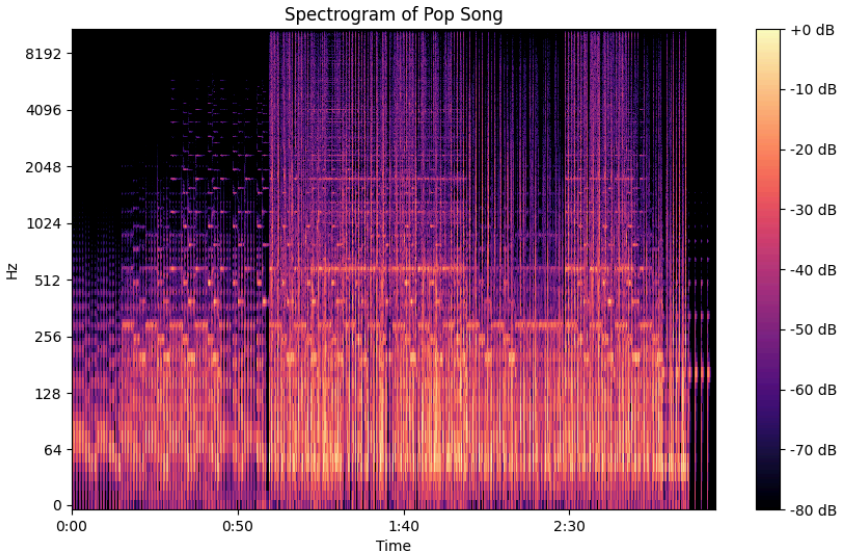
Electronic Music Features



The electronic genre demonstrates clear structural elements:

- Strong vertical striping patterns indicating rhythmic content
- Concentrated energy in the sub-bass region (below 64 Hz)
- Regular patterns in the mid-frequency range (512-2048 Hz)
- Sharp transitions between frequency bands
- Clear separation between rhythmic elements and sustained sounds

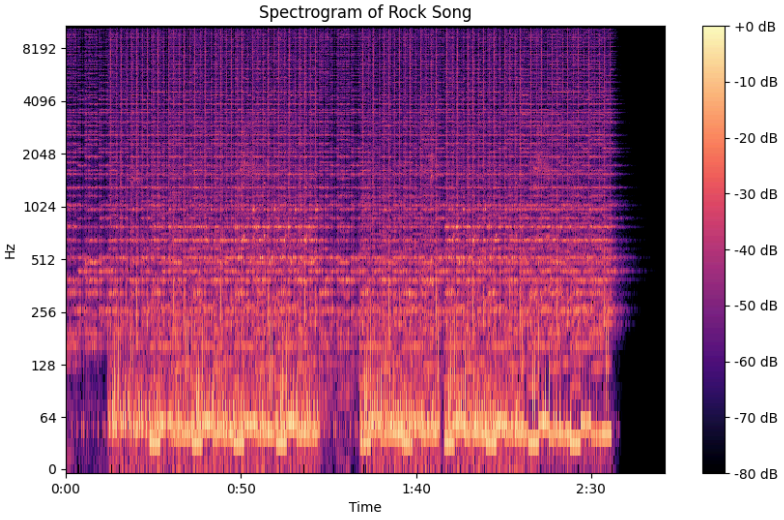
Pop Music Characteristics



The pop genre spectrogram shows balanced energy distribution:

- Prominent mid-frequency content (256-2048 Hz) typical of vocal ranges
- Regular vertical patterns indicating consistent rhythm
- Even energy distribution across frequency bands
- Clear structural sections visible in the temporal domain
- Strong presence in the vocal frequency range (300-3000 Hz)

Rock Music Analysis



The rock genre exhibits distinctive spectral characteristics:

- High energy content across the entire frequency spectrum
- Strong presence in the guitar frequency range (1000-4000 Hz)
- Dense spectral content indicating multiple simultaneous instruments
- Variable dynamic range with clear sectional changes
- Prominent low-frequency energy from bass and drums

Technical Comparison

Frequency Distribution Analysis

The genres show distinct frequency characteristics:

- Ambient: Most even distribution, emphasizing mid-frequencies
- Electronic: Focused energy in low and high frequencies
- Pop: Balanced distribution with emphasis on vocal ranges
- Rock: Wide-spectrum energy with guitar-focused mid-ranges

Temporal Patterns

Rhythmic structure varies significantly:

- Ambient: Minimal temporal structure, flowing transitions
- Electronic: Regular, machine-precise patterns
- Pop: Structured patterns with clear sectional changes
- Rock: Complex patterns with varying intensities

Dynamic Range Comparison

Each genre exhibits characteristic dynamic behavior:

- Ambient: Narrow dynamic range (approximately -60 to -20 dB)
- Electronic: Moderate range with consistent peaks
- Pop: Controlled range with regular patterns
- Rock: Wide dynamic range with frequent variations

References

1. J.-W. Jung *et al.*, “ESPnet-SPK: full pipeline speaker embedding toolkit with reproducible recipes, self-supervised front-ends, and off-the-shelf models,” *Interspeech 2022*, pp. 4278–4282, Sep. 2024, doi: 10.21437/interspeech.2024-1345.

2. D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, *X-Vectors: robust DNN embeddings for speaker recognition*. 2018, pp. 5329–5333. doi: 10.1109/icassp.2018.8461375.

3. Y. Zhang *et al.*, “MFA-Conformer: Multi-scale feature Aggregation conformer for automatic speaker verification,” *Interspeech 2022*, Sep. 2022, doi: 10.21437/interspeech.2022-563.

4. B. Desplanques, J. Thienpondt, and K. Demuynck, “ECAPA-TDNN: Emphasized channel attention, propagation and aggregation in TDNN based speaker verification,” *Interspeech 2022*, Oct. 2020, doi: 10.21437/interspeech.2020-2650.

5. J.-W. Jung, Y. Kim, H.-S. Heo, B.-J. Lee, Y. Kwon, and J. S. Chung, “Pushing the limits of raw waveform speaker recognition,” *Interspeech 2022*, Sep. 2022, doi: 10.21437/interspeech.2022-126.

6. S. H. Mun, J.-W. Jung, M. H. Han, and N. S. Kim, “Frequency and Multi-Scale Selective Kernel attention for speaker verification,” *2022 IEEE Spoken Language Technology Workshop (SLT)*, pp. 548–554, Jan. 2023, doi: 10.1109/slt54892.2023.10023305.