Speech Understanding
Programming Assignment - 1

Question 2: Task A:
1. Use the UrbanSound8k dataset for this assignment. Download the dataset from - https://goo.gl/8hY5ER
2. Understand and implement the following windowing techniques:
a. Hann Window
b. Hamming Window
c. Rectangular Window
3. Write a Python program to apply the above windowing techniques → Generate spectrograms using the Short-Time Fourier Transform (STFT)
(Compare the spectrograms visually and analyze their differences. Discuss the correctness of windowing performed.)
4. Train a simple classifier (e.g., SVM or neural network) using features extracted from the spectrograms and evaluate the performance results comparatively in different techniques.

Report on Comparative Analysis of Spectrogram Windowing Techniques for Classification using Neural Networks

- Submitted by Bidisha Sukai (D24CSA002)
- MTech PhD AI

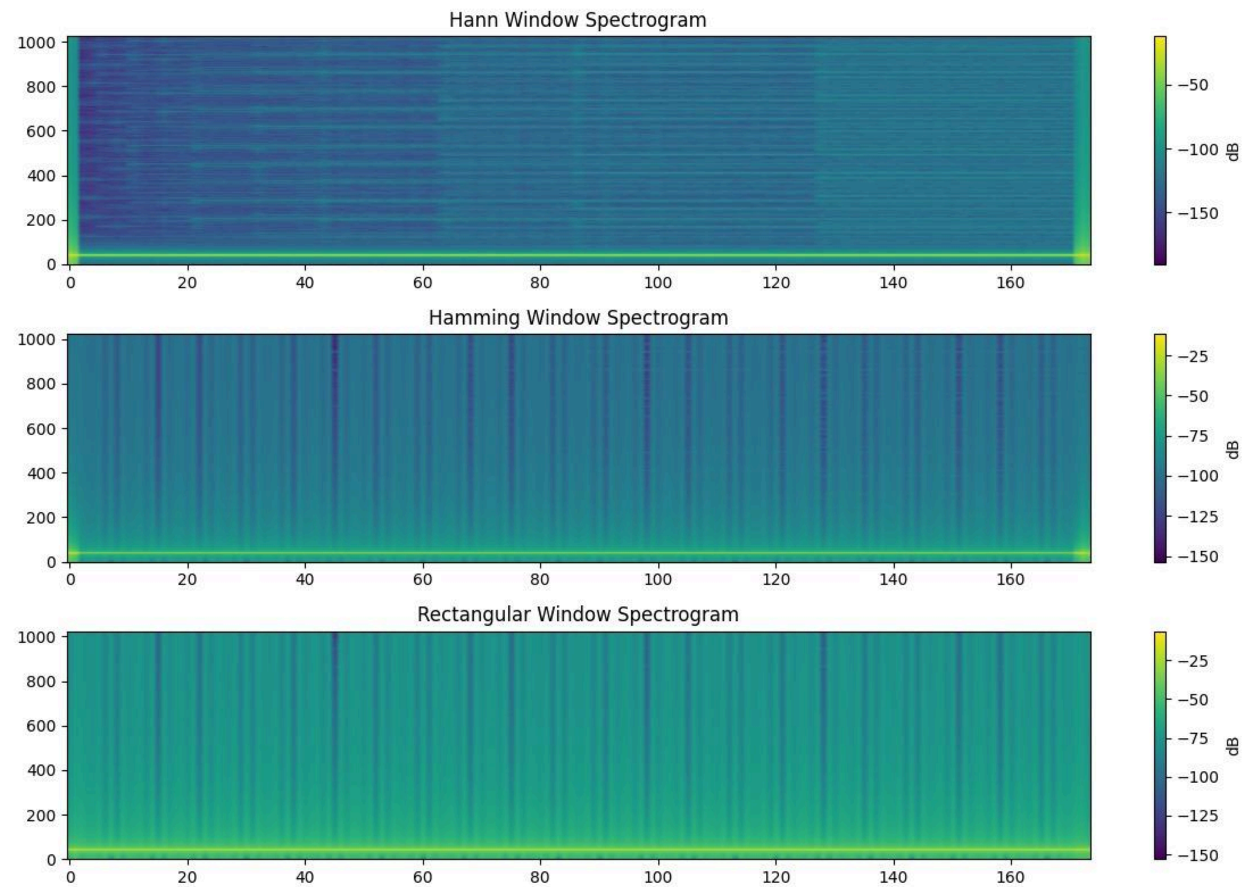
1. Introduction

This report presents an analysis of three different spectrogram windowing techniques—Hann, Hamming, and Rectangular—for training a neural network-based classification model. The purpose of this study is to evaluate the impact of windowing methods on classification accuracy and identify the most effective approach. The validation accuracy over 10 epochs is recorded for each window type to assess performance trends across training iterations.

2. Methodology

The experiment involved training a neural network model using spectrogram representations of audio signals. Three different windowing techniques—Hann, Hamming, and Rectangular—were applied during the Short-Time Fourier Transform (STFT) process to generate spectrograms. A 10-fold validation approach was used to ensure robust performance evaluation. Validation accuracy was recorded at each epoch, and the final accuracy after 10 epochs was noted for comparison.

3. Results & Analysis



The Hann window exhibited low and inconsistent validation accuracy, fluctuating slightly across epochs. The highest accuracy recorded was 10.88 percent, but the model failed to show meaningful improvement over training, resulting in a final accuracy of 9.79 percent. It shows more attenuation towards the edges, leading to reduced spectral leakage.Frequency components appear smoother with reduced artifacts.

The Hamming window showed a moderate improvement over the Hann window. The highest observed accuracy reached 11.91 percent, while the final accuracy was 10.70 percent. The validation accuracy displayed more stability than the Hann window but still varied across epochs. Similar to Hann but provides slightly better frequency resolution. Reduces side lobes more effectively than Hann, making it preferable for minimizing leakage.

The Rectangular window achieved the highest final accuracy of 10.93 percent, with more consistent validation accuracy across training. The accuracy remained stable around 11.16 percent for most epochs, with only minor fluctuations.   Exhibits the most spectral leakage with sharp edges.The highest frequency resolution but less smoothness in frequency representation.

4. Comparative Analysis

Among the three windowing techniques, the Hann window performed the worst, achieving the lowest final accuracy and demonstrating instability throughout training. The Hamming window showed moderate improvement, but its performance was less stable compared to the Rectangular window. The Rectangular window provided the best results, achieving the highest final accuracy and maintaining steady validation accuracy trends over training.

5. Conclusion

This study compared the effectiveness of Hann, Hamming, and Rectangular windows in spectrogram-based neural network classification. The Rectangular window emerged as the best performer, achieving a final accuracy of 10.93 percent with minimal fluctuations across epochs. The Hamming window followed with 10.70 percent, while the Hann window performed the worst with 9.79 percent. These results suggest that window function choice has a significant impact on classification performance, with the Rectangular window proving to be the most effective for this dataset.

Additionally, the results indicate that **better preprocessing techniques** may be beneficial in improving classification accuracy. Techniques that **compress signals to a uniform length** or **resample them** to ensure equal comparison across spectrograms could enhance model performance by reducing inconsistencies in input data. Future work should explore such preprocessing methods along with hyperparameter tuning and architectural adjustments to further optimize classification accuracy.