Proceedings of CONF-CIAP 2026 Symposium: Applied Mathematics and Statistics
DOI: 10.54254/2753-8818/2026.CH31729

Comparing the Effectiveness of Feature Analysis and Visual Analysis Machine Learning Approaches to Classifying Music Genres

Michael Sun

Seattle Pacific University, Seattle, USA
msun17@hotmail.com

Abstract. Music genre classification remains a fundamental challenge in music information retrieval, with applications spanning automated music recommendation and content organization on streaming platforms. This study compares two prevalent machine learning approaches: feature-based analysis using Random Forest (RF) classifiers and visual analysis using Convolutional Neural Networks (CNNs). Using the GTZAN dataset containing 1,000 audio samples across 10 genres, we evaluate both methodologies through comprehensive performance metrics and cross-validation. The feature-based approach employs manually extracted audio features including Mel-Frequency Cepstral Coefficients (MFCCs), spectral centroid, chroma, and temporal characteristics. The visual approach processes mel-spectrograms through a CNN architecture optimized for small datasets via Global Average Pooling, reducing parameters from 5.3 million to 392,000. Results demonstrate that the optimized CNN achieves superior performance with 67.67% mean accuracy and 5.46% standard deviation in 5-fold cross-validation, compared to the RF model's 54.50% accuracy with 20% variance. While both approaches struggle with genres like disco and rock, the CNN approach shows more consistent classification across all genres. These findings suggest that visual analysis through properly configured CNNs outperforms feature-based methods for music genre classification, particularly when architectural adjustments account for limited training data.

Keywords: Mel-Frequency Cepstral Coefficients (MFCCs), Spectral Contrast, Chroma Features, Random Forest (RF), Convolutional Neural Networks (CNNs)

1. Introduction

Music genre classification has become an important task in the field of Music Information Retrieval. Modern streaming platforms not only host hundreds of millions of songs, but also serve as multifaceted services that allow users to categorize, browse, and discover new music. This has led to a much-increased reliance on automated systems. Genre labels are a key feature for improving user experience by improving critical functions such as automated music recommendation and accurate music browsing [1,2]. As these platforms host more songs, and as genres continue to evolve

DOI: 10.54254/2753-8818/2026.CH31729

throughout time, it becomes decreasingly feasible for manual labeling to be an effective solution to this need, opening the door for machine learning models.

Supervised learning is the machine learning approach we chose to train models on labeled data to learn the relationship between input features and known output categories [3]. In this study, it is employed to classify music genres based on audio features, as the genre labels provide the necessary guidance for the model to identify patterns and make accurate predictions.

We began this study by analyzing ways that experts have already tried to approach the problem. The main obstacles others have run into can be divided into two main categories: dataset issues and feature extraction challenges. The main dataset issues are genre ambiguity, non-standard labeling, domain shift, and the lack of data for rare genres of music [4]. Genre ambiguity refers to the vagueness of some music genre definitions, which goes hand-in-hand with the issue of non-standard labeling. Many genres of music have definitions that are nebulous at best, with very subjective, non-standard criteria [5]. Domain shift refers to the limitation of models to the cultural domain of the dataset on which they were trained. Some feature extraction challenges were also made apparent in our preliminary research: representation and performance variance [5]. Representation refers to the form of the data that is being fed to the model. Some studies may choose to work with the raw audio data, while others may use the audio spectrograms or engineered features like chroma instead. Performance variance refers to the human stylistic nuances that actual performers may exhibit in their music, which can lead to a wide range of sounds within a genre [6].

We also surveyed the popular datasets that other teams have employed for this problem. GTZAN, MuMu, and Extended Ballroom seem to be among the top 3 most popular music datasets. GTZAN has representation across 10 genres of music and 100 audio file samples for each genre [5]. MuMu is a much larger dataset, at 31000 entire albums and more than 250 genre classifications [7]. Extended Ballroom has over 4000 samples but they are all under the umbrella of ballroom music [8].

We would end up using the GTZAN dataset for its readily available pre-extracted features and mel-spectrograms.

The main four methodologies we saw were traditional machine learning using manually features, convolutional neural networks (CNNs) on spectrograms made from audio data, hybrid CNN and recurrent neural networks (RNNs), weakly or self-supervised neural networks, and using wavelet transforms to preprocess the data [1-3,9,10].

Across the studies that we analyzed using these different methodologies, signal preprocessing of the GTZN dataset and then manually focusing on Mel Frequency Cepstral Coefficients (MFCC) and Short-Time Fourier Transform (STFT) features gave the best results, at 95.2 % and 95.7 % accuracy [5].

2. Methodology

2.1. Random Forest

Random Forest (RF) is our baseline machine learning method, and is applied to both the visual analysis and feature analysis methodologies. Within the "forest" of RF lies its "trees", which in this case, are decision trees [6]. Decision trees can be understood as binary flowcharts that categorize data. Each tree is created from a random sampling of a greater dataset, for example using ⅕ of the total audio samples. After the "forest" of random decision trees has been created, RF then makes its predictions by democratically heeding the "wisdom of the crowds". For example, an audio sample that is fed through 100 established decision trees and receives the label of "hip-hop" by 65 trees and

DOI: 10.54254/2753-8818/2026.CH31729

"rap" by 35 of them would be considered most likely to be "hip-hop". Oftentimes, this result would be interpreted as the sample having a 65% chance of being "hip-hop" and a 35% chance of being "rap". – no Grid)

2.2. Feature analysis mock data

We saw high test accuracy in the models trained on replicated features of the GTZAN dataset. This replicated testing dataset was for preliminary testing of our code and deciding on visualizations to include for the final implementation.

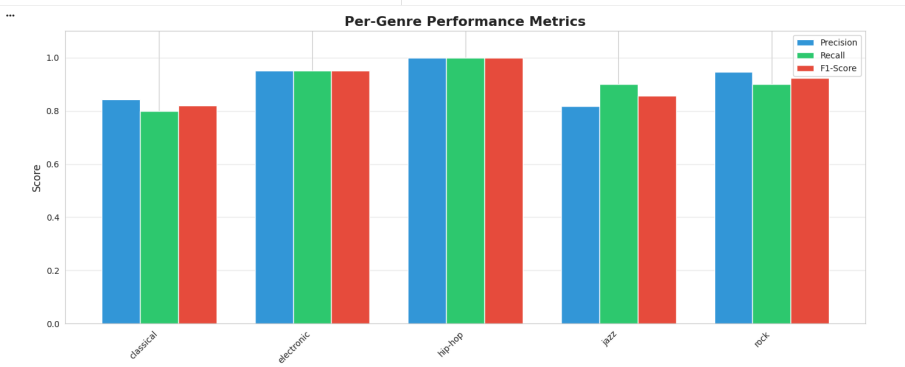

Figure 1. RF per-genre performance metrics graph

Table 1. RF mock data classification report

	precision	recall	f1-score	support
classical	0.8421	0.8000	0.8205	20
electronic	0.9500	0.9500	0.9500	20
hip-hop	1.0000	1.0000	1.0000	20
jazz	0.8182	0.9000	0.8571	20
rock	0.9474	0.9000	0.9231	20
accuracy			0.9100	100
macro avg	0.9115	0.9100	0.9101	100
weighted avg	0.9115	0.9100	0.9101	100

Figure 1 and Table 1 show the precision, recall, and F1 score performance metrics of our feature extraction model broken down into five main music genres. Precision is the percentage of true positives among the total positive predictions produced by the model. This metric describes the predictive accuracy of a model [6].

Recall, often also called sensitivity, is calculated by instead dividing true positives by the sum of true positives and false negatives [10]. A low sensitivity value would indicate the presence of many false negatives where the model failed to evaluate data accordingly.

F1 score serves to balance the results of precision and recall by taking the harmonic mean of the two values, making it useful for evaluating models whose precision and accuracy are in tradeoff [6].

Out of the five genres, hip-hop is the clear leader in all three of these performance metrics when analyzed by a feature driven model, with a precision, sensitivity, and F1 score of 1.0. Electronic music has the second highest metrics and also has three identical values for its performance metrics. The genre with the lowest precision was jazz, with a score of 0.8182, which means that it was the

DOI: 10.54254/2753-8818/2026.CH31729

most difficult to correctly identify. Although the model had the lowest precision score for identifying jazz, classical music ended up with the lowest recall and F1 scores at 0.8000 and 0.8205 respectively.

These results could suggest that within the genres included in the GTZAN dataset, hip-hop and electronic music have the most standout musical features, while classical and jazz features are the most muddled by other genres in this study. The confusion matrix in Figure 2 below sheds some more light on the topic of genre obfuscation. 3 classical songs were incorrectly identified as jazz, while 1 was classified as rock music.

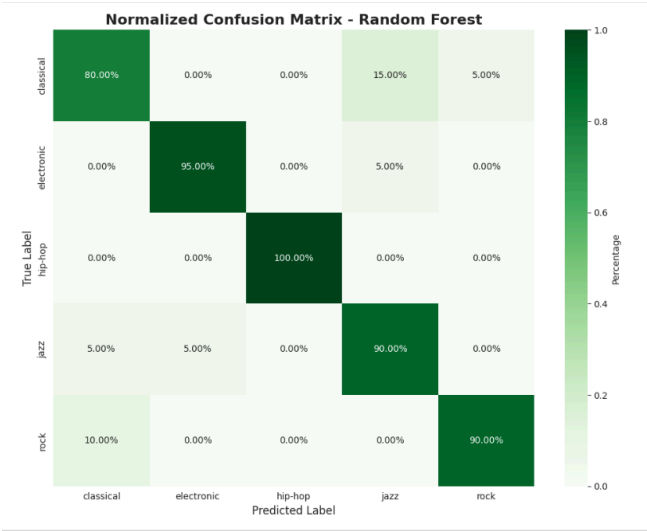

Figure 2. RF mock data normalized confusion matrix

To analyze the weight of the many features used from this dataset, we used the built-in Gini Importance functionality in pandas, which resulted in the top 20 most important features shown in Figure 3. Of these top 20 features, 15 of them involve mel-frequency cepstral coefficients. For this study we used a set of 13 from 1-13, increasing in granularity and detail of the spectrogram with higher orders. All 13 of the mean values appear in the top 14 most important features, with the second order MFCC being the most important, surprisingly enough. The box plot comparison of other important features shows the clear superiority in genre partitioning described by MFCC 2's mean , with little visual overlap between the boxes of the different genres.

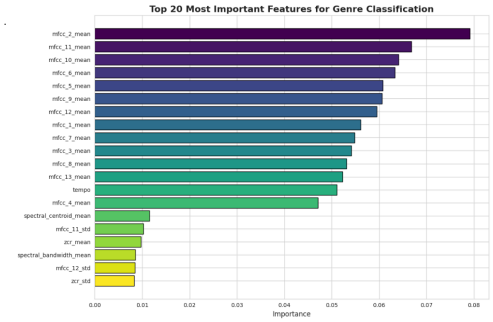

Figure 3. RF top 20 important features

DOI: 10.54254/2753-8818/2026.CH31729

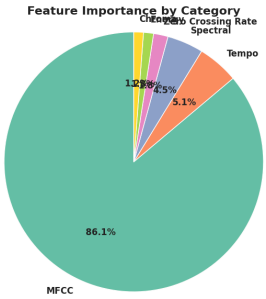

Figure 4. RF feature importance by category

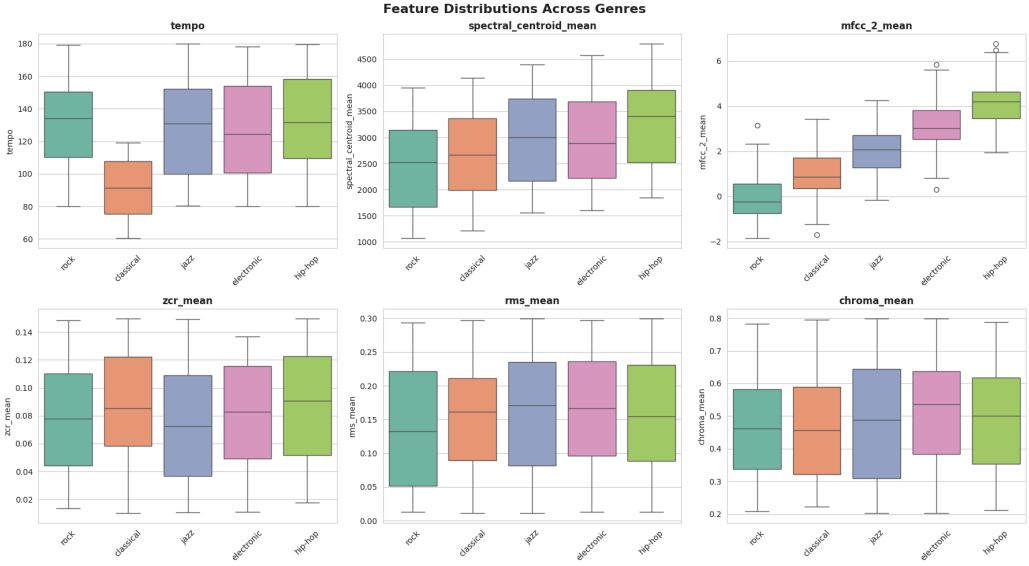

Figure 5. RF multi-graph spread

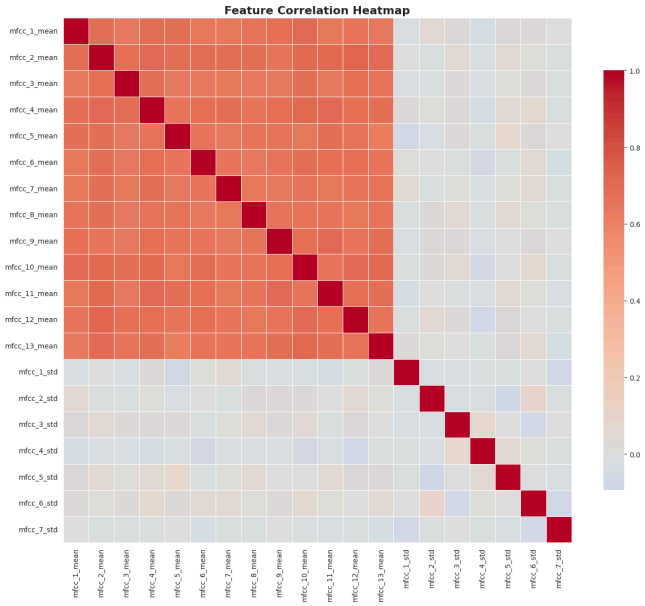

Figure 6. Feature correlation heatmap

DOI: 10.54254/2753-8818/2026.CH31729

Figure 4 gives a general idea of the influence of MFCCs in terms of feature importance. One must keep in mind the quantity of different MFCCs when compared to other features.

Another observation from the box plot comparisons in Figure 5 is the clear visual standout of classical music's tempo among the other genres, with its mean around 90bpm while the other 4 genres have means within the 120 to high 130bpm range.

The spectral aspect of a song is often described as its "brightness", and it should not be too surprising that the average of this brightness, the spectral centroid mean of these song genres, results in a very similar hierarchy to the MFCC 2 mean. The slight dip that electronic music has in its spectral centroid mean compared to its MFCC 2 mean could be explained by a bottom heavy range found in electronic music that is accounted for in a spectral centroid model.

Figure 6 shows a relatively expected, streamlined correlation heatmap on our mock data. This visualization may be useful in understanding feature-target relationships on our real data.

Going back to the performance of RF given these features, the 5-fold cross-validation results shown in Figure 7 display a modest mean of 0.8820. Given the fold CVs of 0.85, 0.90, 0.91, 0.90 and 0.85, we end up with a nice, low standard deviation of 2.64%, demonstrating low variance.

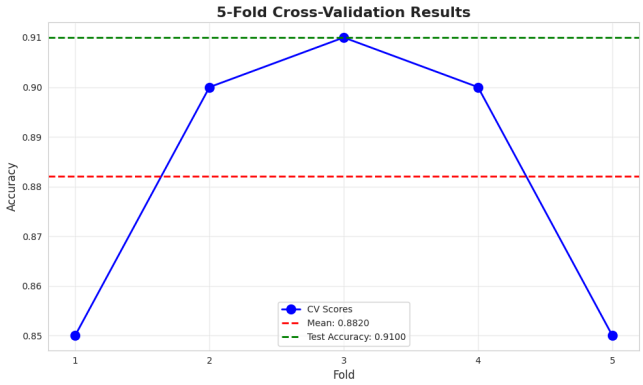

Figure 7. RF mock data cross-validation

A final analysis of the feature-trained RF model we created looks at the model performance when given different numbers of decision trees. As seen in Figure 8, the accuracy of the model on test data increases sharply before plateauing at 50 to 75 trees before decreasing and then plateauing once more for 100, 150, and 200 trees. The largest forest with 300 decision trees actually had the second lowest accuracy of all of the RFs created. We intuitively expected accuracy of the model to increase with its number of decision trees before plateauing, but our data showed otherwise. Perhaps this outcome could be explained by the relative small size of our mock dataset.

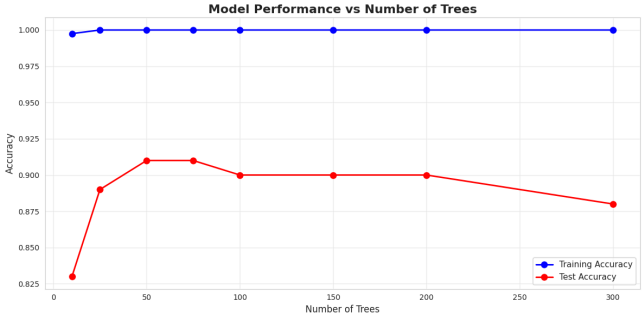

Figure 8. RF mock data training visualization

DOI: 10.54254/2753-8818/2026.CH31729

2.2.1. Feature analysis GTZAN dataset

The actual GTZAN dataset is much more diverse and thorough than the mock dataset that we established our feature analysis methodologies with. GTZAN features the following 10 music genres: blues, classical, country, disco, hiphop, jazz, metal, pop, reggae, and rock. Each of these genres is represented with 100 samples shown in Figure 9.

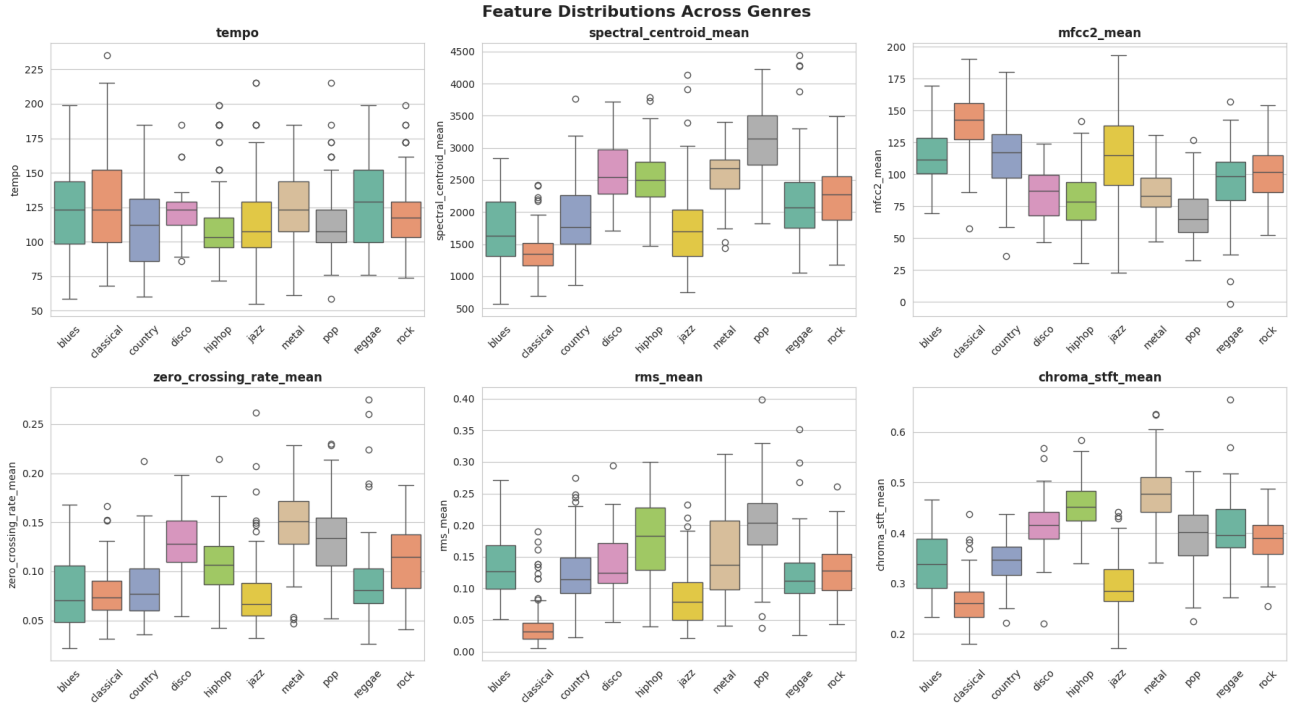

Figure 9. RF feature distribution multi-graph spread

Before training our models, we conducted preliminary data analysis on feature distribution across our genres with outcomes seen in the figure above. The tempo graph in Figure 9 qualitatively appears to be the least distinguishing, as the means of all 10 genres' samples hovers around 100 to slightly above 125bpm.

The spectral centroid of audio data describes the "center of mass" of its frequency spectrum. This often makes it a good way to classify the "brightness" or conversely the "darkness" or "warmth" of a song which correlates with high and low spectral centroids respectively. This metric could be useful in identifying genres that have defining vocal styles or instrumentation choices. For example, pop singers who often use their chest voices may create songs with much higher spectral centroids than classically trained opera singers whose technique emphasizes lower larynx projection instead.

As expected, the zero crossing rate of music genres that place more emphasis on percussion are generally higher. Metal, which is known for being a vocal genre with emphatic beats, unsurprisingly boasts the highest IQR of zero crossing rate means out of the 10 studied genres. Blues, classical, country, jazz, and reggae, all subjectively more mellow genres, have the lowest zero crossing rate means.

The root mean square graph describes the average "loudness" of a genre by quantifying the average amplitude of the sound data rather than by using the value of its highest spike.

The sixth graph in the spread above compares the Short-Time Fourier Transform means of each genre's samples. In Western music, there are 12 unique tones in an octave with 7 natural notes ,

DOI: 10.54254/2753-8818/2026.CH31729

represented by the letters A-G, and five alterations represented by sharps and flats to these natural notes. While a C is clearly in a different pitch in a higher octave than in a lower one, it has the same chroma. The purpose of chroma STFT means is to describe such chromatic hotspots of these audio samples.

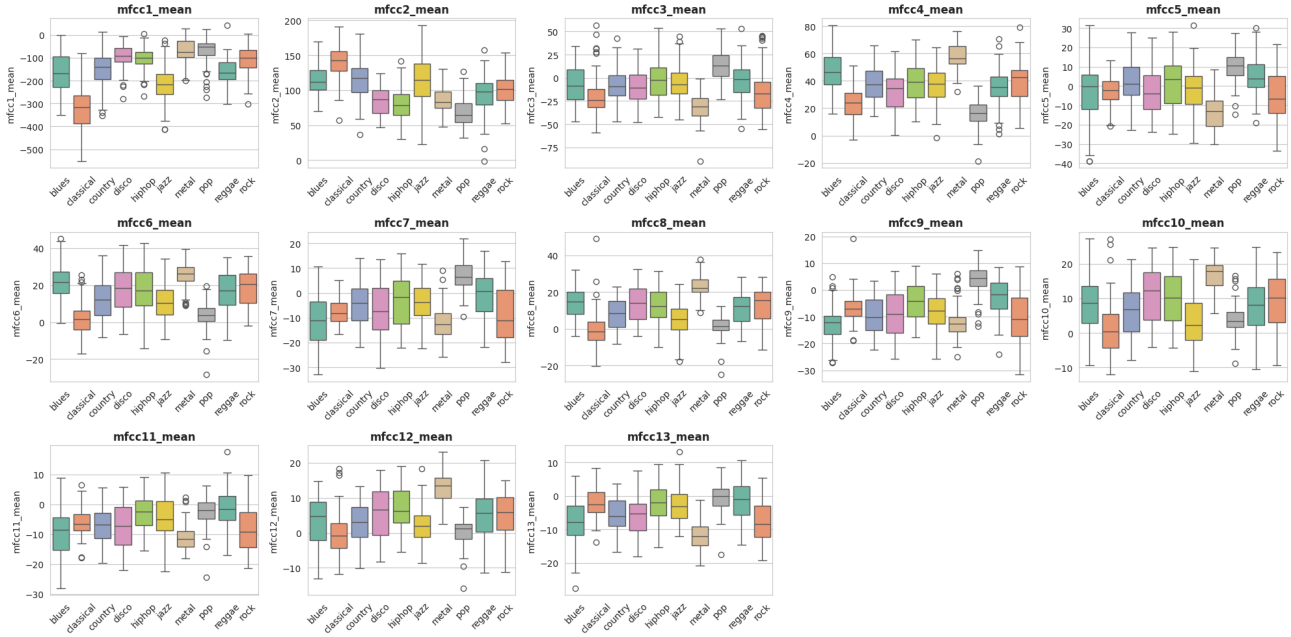

Figure 10. RF MFCC graph spread

Figure 10 is a graph spread of all 13 of the MFCCs produced for our study. Although the first MFCC looks the most broadly at the "shape" of an audio sample's song, it is interestingly more homogenized than some of the higher order visualizations such as that of the second coefficient.

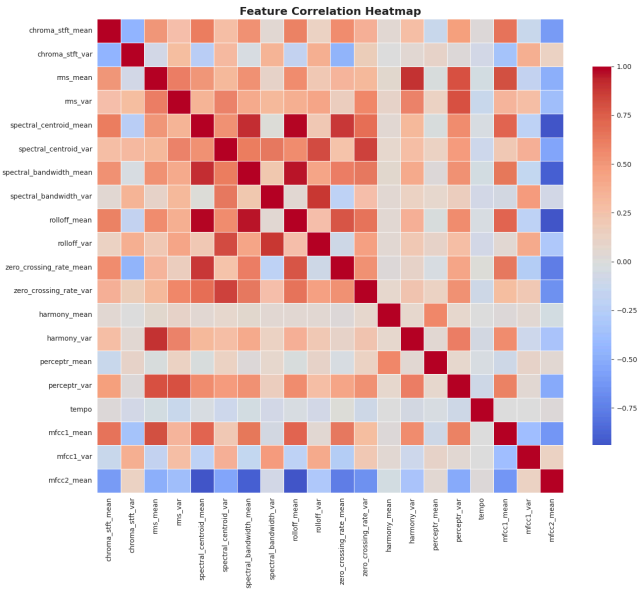

Figure 11. RF feature correlation heatmap

DOI: 10.54254/2753-8818/2026.CH31729

Figure 11 is a heatmap showing some superficial correlations between some of the main features we analyzed. There are some expected relatively strong correlations such as zero crossing rate and spectral centroid mean, as higher energy songs that have higher zero crossing rates could be expected to also have brighter overall sounds.

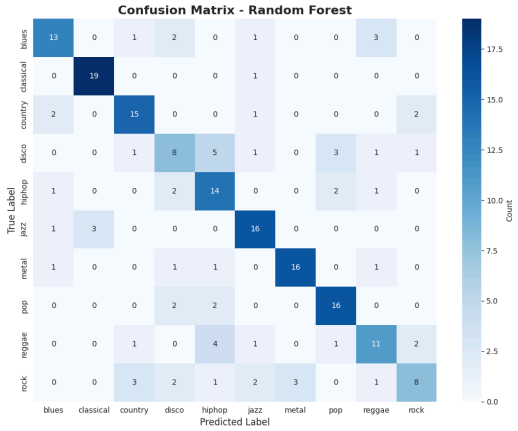

Figure 12. RF confusion matrix

Table 2. RF classification report

Genre	Precision	Recall	F1-Score	Support
blues	0.7222	0.6500	0.6842	20
classical	0.8636	0.9500	0.9048	20
country	0.7143	0.7500	0.7317	20
disco	0.4706	0.4000	0.4324	20
hiphop	0.5185	0.7000	0.5957	20
jazz	0.6957	0.8000	0.7442	20
metal	0.8421	0.8000	0.8205	20
pop	0.7273	0.8000	0.7619	20
reggae	0.6111	0.5500	0.5789	20
rock	0.6154	0.4000	0.4848	20
Accuracy			0.6800	200
Macro Avg	0.6781	0.6800	0.6739	200
Weighted Avg	0.6781	0.6800	0.6739	200

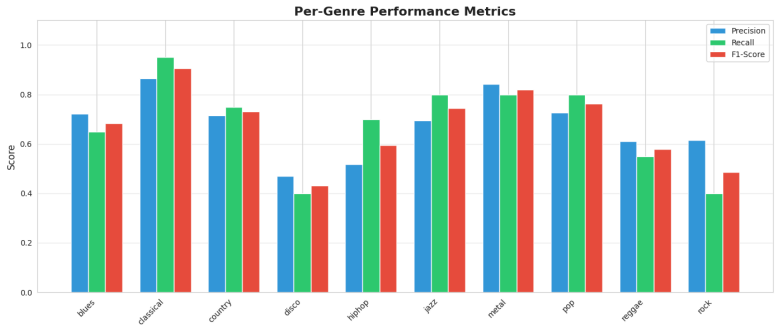

Figure 13. RF per-genre performance metrics graph

DOI: 10.54254/2753-8818/2026.CH31729

After training RFs on the actual GTZAN dataset, we ended up with a 68.0% accuracy on correctly identifying test data as shown in Table 2. Looking deeper at our classification report results shown in Figure 12 and Figure 13, our model was most successful at classifying classical music, with the highest performance in accuracy, sensitivity, and F1. Disco and rock were the genres where our model saw the worst performance where both had the same recall value at 0.4000. This low recall value could indicate the presence of an abundance of false negatives, a lack of true positives, or both. Disco, rock, and to a lesser degree, reggae, have been shown to be relatively difficult to identify in the presence of the other 7 genres, whether due to inherent nebulousness of their genre definitions or perhaps due to the conflation of common traits among the other present genres.

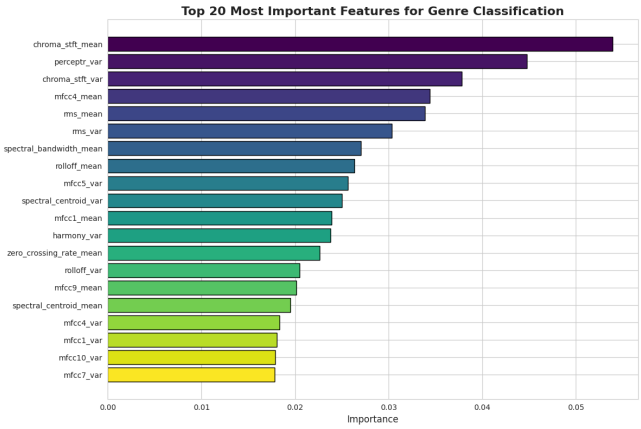

Figure 14. RF top 20 important features graph

Table 3. RF top 15 important features table

Feature	Importance
chroma_stft_mean	0.053948
perceptr_var	0.044740
chroma_stft_var	0.037812
mfcc4_mean	0.034417
rms_mean	0.033875
rms_var	0.030346
spectral_bandwidth_mean	0.027022
rolloff_mean	0.026365
mfcc5_var	0.025638
spectral_centroid_var	0.025031
mfcc1_mean	0.023919
harmony_var	0.023795
zero_crossing_rate_mean	0.022611
rolloff_var	0.020475
mfcc9_mean	0.020118

According to the information in Figure 14 and Table 3, the most important features data corroborates the eye-test importance that can be discerned from the chroma STFT mean graph in the multi-display figure. MFCC 4 is also shown to have been the most important MFCC, which could also have been assumed from its relatively standout graph among the rest of the coefficients.

DOI: 10.54254/2753-8818/2026.CH31729

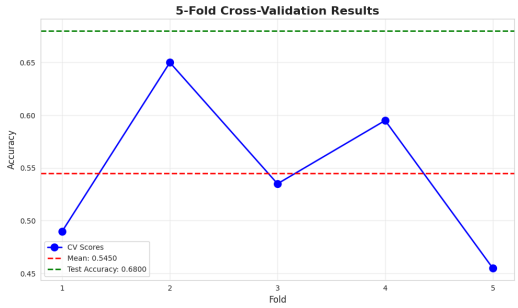

Figure 15. RF 5-fold cross-validation graph

The 5-fold cross validation displayed in Figure 15 belies a surprisingly low overall accuracy of the model at 54.50% as well as an overall variance of around 20%.

2.2.2. CNN analysis mel-spectrogram

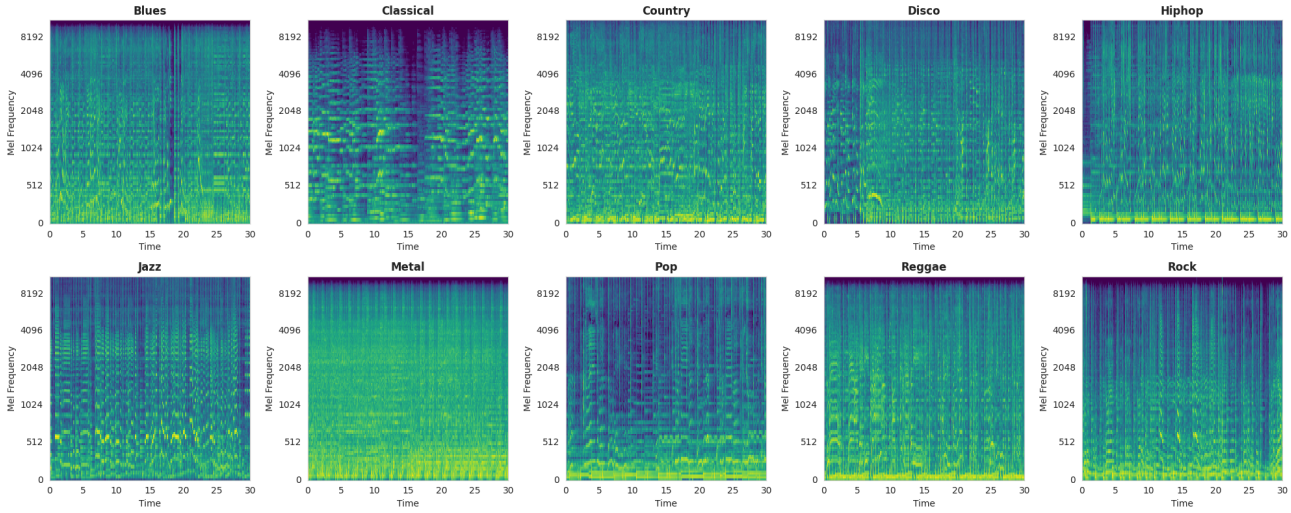

Figure 16. Genre mel-spectrogram visualization spread

We compared the RF feature model above to the performance of CNNs trained on GTZAN's mel-spectrograms represented in Figure 16. Out of the 1000 samples provided by the GTZAN database, 1 jazz sample file by the name of "jazz.00054.wav" seems to be corrupted and could not be used.

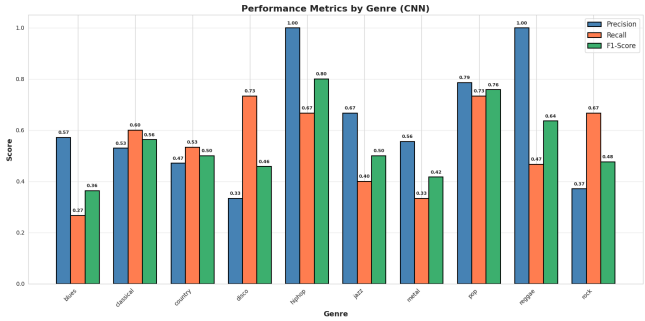

Figure 17. Initial CNN performance metrics

DOI: 10.54254/2753-8818/2026.CH31729

After validating the intact data that we would be working with, we went on to build CNN models to be trained with the above data. The first iterations of our trained models saw relatively poor performance across the board. The graph of performance metrics by genres in Figure 17 shows extremely inconsistent performance of our CNN with genres like classical and reggae being correctly identified 100% of the time in true positives during our testing while this was true of disco and rock a measly 33% and 37% of the time. Even among the highest precision score genres, the model lacked strong recall capabilities, peaking at 73% overall for disco and weighing down reggae's F1 score from its 100% precision score with a 47% recall performance.

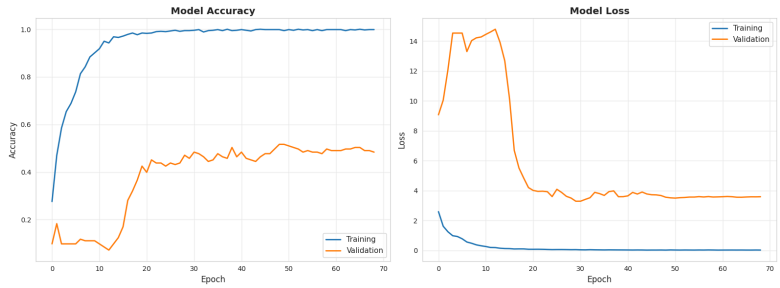

Figure 18. Initial CNN training history visualization

The training history visualization seen in Figure 18 is quite in line with the expected outcome, suggesting that our training process itself might not have been the main culprit affecting performance. As expected, the model's accuracy increased with additional epochs before plateauing and being cut due to diminishing returns. Similarly, the loss metric decreased sharply before also plateauing with additional epochs. These training outcomes are also corroborated by the plateaus reached when using the validation data alongside each epoch. Something slightly unexpected is the sharp jump in loss when validating the first 5 epochs. This could be explained by the model starting with random weights and initially overcommitting. While the validation loss is quite high, spiking to around 14.5, it appears to be relatively stable lending credence to the hypothesis that it is just overfitting early from early chaos

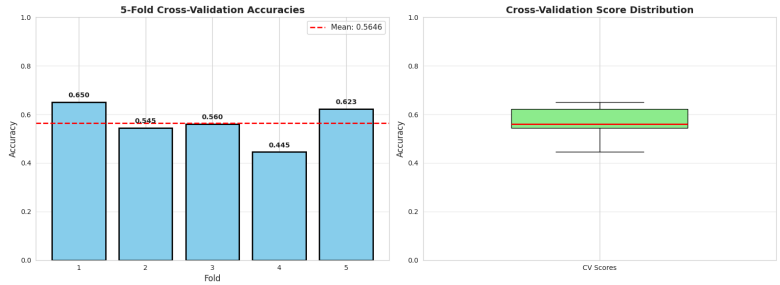

Figure 19. Initial CNN 5-fold cross-validation graph spread

A 5-Fold Cross-Validation analysis shown in Figure 19 revealed a low mean accuracy of 56.46% with a moderate standard deviation of 7.13%. While it was significantly better than randomly guessing, which would have been 10%, there was clearly a lot of improvement to be had. Learning from our early attempts at creating and training CNN models for this purpose, we made a few important changes, as shown in Table 4. Looking at the model parameters shown below, we determined that our extremely high parameter count of more than 5 million from just 700 samples could have been contributing to a lot of our overfitting woes. In order to remedy this, we

DOI: 10.54254/2753-8818/2026.CH31729

implemented global average pooling (GAP), averaging the values of the feature map, to reduce the parameter count to around 400k.

Table 4. Initial CNN model parameters

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 128, 1293, 32)	320
batch_normalization_4	(None, 128, 1293, 32)	128
max_pooling2d_3 (MaxPooling2D)	(None, 32, 323, 32)	0
dropout_4 (Dropout)	(None, 32, 323, 32)	0
conv2d_4 (Conv2D)	(None, 32, 323, 64)	18,496
batch_normalization_5	(None, 32, 323, 64)	256
max_pooling2d_4 (MaxPooling2D)	(None, 8, 80, 64)	0
dropout_5 (Dropout)	(None, 8, 80, 64)	0
conv2d_5 (Conv2D)	(None, 8, 80, 128)	73,856
batch_normalization_6	(None, 8, 80, 128)	512
max_pooling2d_5 (MaxPooling2D)	(None, 4, 40, 128)	0
dropout_6 (Dropout)	(None, 4, 40, 128)	0
flatten_1 (Flatten)	(None, 20480)	0
dense_2 (Dense)	(None, 256)	5,243,136
batch_normalization_7	(None, 256)	1,024
dropout_7 (Dropout)	(None, 256)	0
dense_3 (Dense)	(None, 10)	2,570
Total params		5,340,298 (20.37 MB)
Trainable params		5,339,338
Non-trainable params		960

Table 5. Final CNN parameters

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 128, 1293, 32)	320
batch_normalization	(None, 128, 1293, 32)	128
max_pooling2d (MaxPooling2D)	(None, 32, 323, 32)	0
dropout (Dropout)	(None, 32, 323, 32)	0
conv2d_1 (Conv2D)	(None, 32, 323, 64)	18,496
batch_normalization_1	(None, 32, 323, 64)	256
max_pooling2d_1 (MaxPooling2D)	(None, 8, 80, 64)	0
dropout_1 (Dropout)	(None, 8, 80, 64)	0
conv2d_2 (Conv2D)	(None, 8, 80, 128)	73,856
batch_normalization_2	(None, 8, 80, 128)	512
max_pooling2d_2 (MaxPooling2D)	(None, 2, 20, 128)	0
dropout_2 (Dropout)	(None, 2, 20, 128)	0
conv2d_3 (Conv2D)	(None, 2, 20, 256)	295,168
batch_normalization_3	(None, 2, 20, 256)	1,024
global_average_pooling2d	(None, 256)	0
dropout_3 (Dropout)	(None, 256)	0
dense (Dense)	(None, 10)	2,570
Total params		392,330 (1.50 MB)

DOI: 10.54254/2753-8818/2026.CH31729

Table 5. (continued)

Trainable params	391,370 (1.49 MB)
Non-trainable params	960 (3.75 KB)

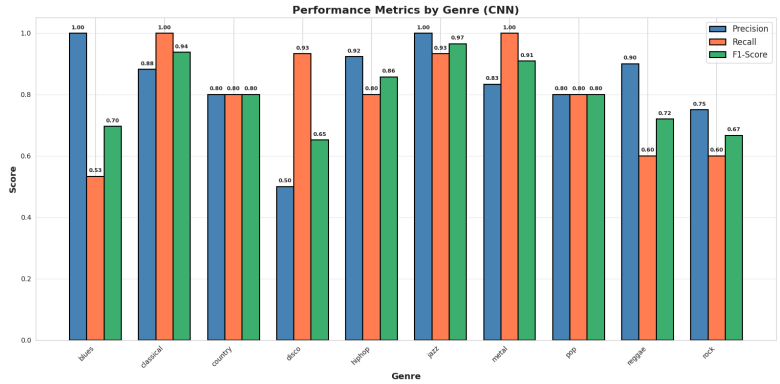

Figure 20. Final CNN performance metrics by genre graph

Table 6. Final CNN performance metrics by genre table

Genre	Precision	Recall	F1-Score	Support
blues	1.0000	0.5333	0.6957	15
classical	0.8824	1.0000	0.9375	15
country	0.8000	0.8000	0.8000	15
disco	0.5000	0.9333	0.6512	15
hiphop	0.9231	0.8000	0.8571	15
jazz	1.0000	0.9333	0.9655	15
metal	0.8333	1.0000	0.9091	15
pop	0.8000	0.8000	0.8000	15
reggae	0.9000	0.6000	0.7200	15
rock	0.7500	0.6000	0.6667	15
Mean	0.8389	0.8000	0.8003	150

Best performing genre: jazz (F1: 0.9655)

Worst performing genre: disco (F1: 0.6512)

The data in Table 5, visualized in Figure 20, shows the parameters of our final GAP CNN with greatly reduced parameters compared to its predecessors. It is immediately clear in its performance metrics graph that it has much better performance all across the board. Interestingly, Table 6's data shows that the most difficult genre to classify by F1 score is disco, with an F1 score of 0.6512. Disco also had a bottom F1 score in the previous CNN model at 0.46. Jazz is now the genre that sees the best performance with an F1 score of 0.9655, while the previous model had a middling performance with this genre and an F1 score of 0.50.

DOI: 10.54254/2753-8818/2026.CH31729

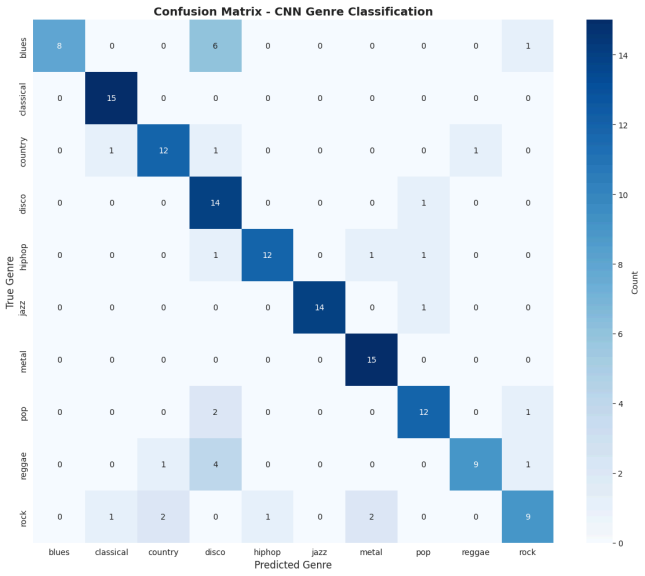

Figure 21. Final CNN confusion matrix

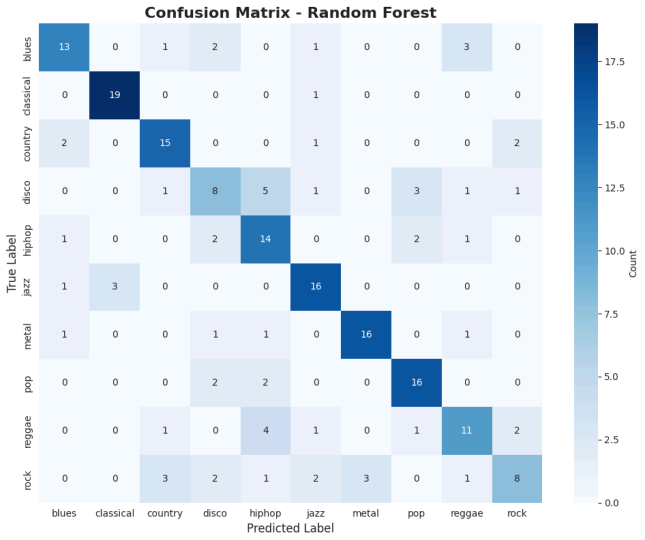

Figure 22. RF confusion matrix

A glance value comparison of the confusion matrices in Figure 21 and Figure 22 of our two final models shows a clarity in the predicted label accuracy of the CNN model compared to the RF model. There also seems to be a shared hotspot of false labeling around disco. However, while the CNN model never mislabeled rock as disco but mixed up disco and reggae 4 times, the RF model mislabeled reggae as rock 2 times while never mistaking reggae for disco. The RF model also showed confusion in labeling reggae as hiphop which the CNN model never did. Perhaps the most standout and surprising performance discrepancy is with identifying blues. While the RF model succeeded in true labeling blues 65% of the time, the CNN model was only able to do so around 53% of the time.

DOI: 10.54254/2753-8818/2026.CH31729

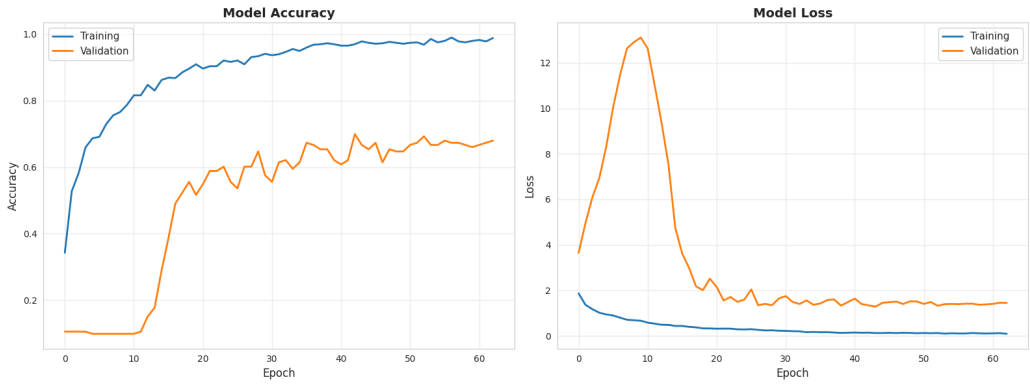

Figure 23. Final CNN training history visualization

The training history visualization in Figure 23 of the GAP CNN also aligns with expected performance. This time, there seemed to be a much smoother loss curve during initial overfitting with a very smooth path downwards when learning, and a noticeable bumpiness that normalizes into a plateau in both the accuracy and loss graphs.

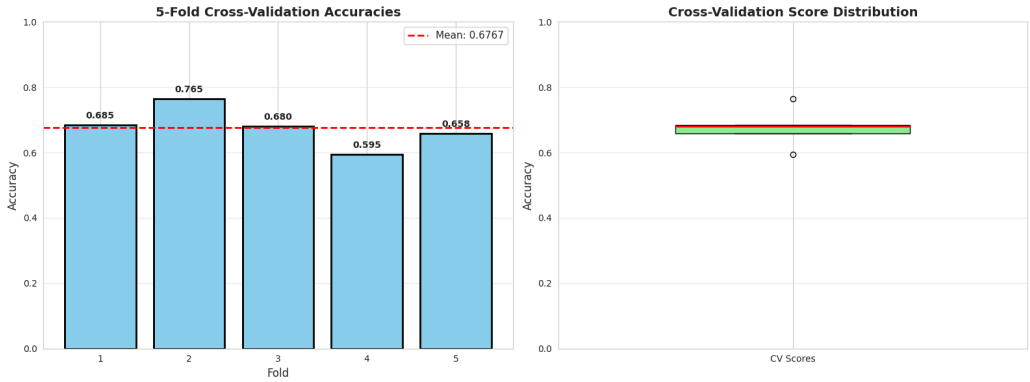

Figure 24. Final CNN 5-fold cross-validation graph spread

The 5-Fold Cross-Validation in Figure 24 shows a modest mean accuracy of 67.67% with a low variance of a 5.46% standard deviation. This is a significant improvement over the RF model's results of a 54.50% mean accuracy.

3. Conclusion

Although the exact data both of our final models were trained on were ostensibly different, there is a similarity in scale that allows us to draw a conclusion between the use

cases of the two. The data in the GTZAN dataset has many well-extracted and proven manual features such as spectral centroid chroma and MFCCs in addition to spectrogram accessibility. Although the breadth of important sample data is robust, it lacks in sheer quantity. Our findings showed that before tweaking our CNN approach to match the relatively small dataset conditions we were working with, an RF approach yielded competitive results. After making the appropriate changes, reducing the parameters considered when building our CNN, its performance quickly caught up and overshot that of the RF model. Our findings showed a clearly better performance in spectrogram analysis using properly adjusted CNNs compared to feature analysis using RFs in music genre classification.

DOI: 10.54254/2753-8818/2026.CH31729

If this study were to be replicated as shown, it would be optimal to feed the models with larger datasets. Deep learning models such as the CNNs that we used thrive in higher data environments, and we had to do a lot of compromising and manual tweaking in areas such as parameter retention in order to reach a reasonable level of performance. The performance of the RFs could also be improved in the future given a larger dataset that includes more features such as spectral entropy, which describes the randomness of an audio file's spectrum and could lend itself to analyzing, a combined feature that looks into harmony in audio data.

References

[1] Chaudhury, M., Karami, A. and Ghazanfar, M.A. (2022) Large-scale music genre analysis and classification using machine learning with Apache Spark. Electronics, 11, 2567.
[2] Oramas, S., Nieto, O., Barbieri, F. and Serra, X. (2017) Multi-label music genre classification from audio, text, and images using deep features. Proceedings of the 18th International Society for Music Information Retrieval Conference (ISMIR). Retrieved from https: //arxiv.org/abs/1707.04916
[3] Li, X., Li, F., Lu, Z.P. and Yang, Z.Y. (2025) Music genre classification: A comprehensive study on feature fusion with CNN and MLP architectures. Proceedings of the ACE Conference. Retrieved from https: //direct.ewa.pub/proceedings/ace/article/view/20632
[4] Varela, D.J. (2025) Genre or genre-less: An analysis of music classification and complexities of genres and subgenres. Honors Thesis, Southeastern University, Lakeland, FL, USA.
[5] Li, T. (2024) Optimizing the configuration of deep learning models for music genre classification. Heliyon, 10, e24892.
[6] Kumaraswamy, B. (2022) Optimized deep learning for genre classification via improved moth flame algorithm. Multimedia Tools and Applications, 81, 17071–17093.
[7] Bahuleyan, H. (2018) Music genre classification using machine learning techniques. arXiv preprint. Retrieved from https: //arxiv.org/abs/1804.01149
[8] Jaishankar, B., Anitha, R., Shadrach, F.D., Sivarathinabala, M. and Balamurugan, V. (2023) Music genre classification using African Buffalo optimization. Computer Systems Science and Engineering, 44.
[9] Kumar, A., Rajpal, A. and Rathore, D. (2018) Genre classification using feature extraction and deep learning techniques. Proceedings of the 10th International Conference on Knowledge Systems Engineering (KSE), 175–180.
[10] Ndou, N., Ajoodha, R. and Jadhav, A. (2021) Music genre classification: A review of deep-learning and traditional machine-learning approaches. Proceedings of the IEEE International IOT, Electronics and Mechatronics Conference (IEMTRONICS), 1–6.