Music Genre Classification

Jonathan Neimann, Matt Maslow

December 6, 2024

Abstract

This project aims to develop a music genre classification system. It will compare a K-Nearest Neighbors (KNN) approach with a Convolutional Neural Network (CNN) to explore the strengths and limitations of traditional versus deep learning methods. By leveraging audio feature extraction and spectrogram-based representations, we will assess the effectiveness of each approach in genre recognition, evaluating their accuracy and adaptability to diverse musical genres.

1 Introduction

Music genre classification is a pivotal task in music information retrieval, enabling applications like personalized music recommendations, automatic playlist generation, and efficient cataloging of large music libraries. Despite its importance, genre classification remains challenging due to the subjective and overlapping nature of genre definitions and the variability in audio characteristics across different songs and styles.

This project investigates two approaches to genre classification: a traditional K-Nearest Neighbors (KNN) model and a Convolutional Neural Network (CNN). The KNN model relies on handcrafted audio features such as Mel Frequency Cepstral Coefficients (MFCCs), chroma features, and spectral contrast, which capture specific aspects of an audio file's timbre, harmony, and dynamics. In contrast, the CNN model operates directly on mel-spectrogram images, which uses the audio files to create a visual representation using these same features, leveraging its ability to learn hierarchical features automatically.

By comparing these two approaches on the GTZAN dataset—a benchmark dataset of 10 music genres—this project aims to highlight the strengths and limitations of traditional machine learning versus deep learning in music classification tasks. Evaluation metrics such as training and validation accuracy and loss are used to assess each model's performance.

2 Related Work

This section provides an overview of related research in music classification. Choi et al. [1] proposed a model using convolutional recurrent neural networks for music classification, achieving impressive results by combining the strengths of convolutional and recurrent neural networks. Their approach focused on enhancing the accuracy of genre classification tasks,

which was demonstrated through extensive experiments on various music datasets. However, despite their success, they did not release the code or models used in their research, presenting an opportunity for future work to reproduce their results and potentially improve upon them.

Tzanetakis and Cook [2] introduced one of the earliest influential systems for musical genre classification based on audio signals. Their 2002 paper explored the use of audio features such as spectral texture and rhythm, laying the groundwork for many modern genre classification systems. While the system was groundbreaking at the time, the authors did not release their models, and the dataset used for their experiments has been widely circulated in the research community. This raises the possibility of further exploration into their approach with updated techniques and more recent datasets.

3 Datasets

This project utilizes the GTZAN dataset, a benchmark dataset for music genre classification tasks. The dataset consists of audio files distributed across ten genres: blues, classical, country, disco, hip-hop, jazz, metal, pop, reggae, and rock. Each genre contains `.wav` files in two formats:

- **30-second audio files:** These are the original full-length tracks, with each audio file sampled at 22,050 Hz. They provide the basis for generating spectrograms and extracting features for model training and evaluation.
- **3-second audio files:** These are shorter segments derived from the 30-second tracks. These segments allow for increased dataset size, enabling finer analysis and better training for models, particularly for deep learning approaches like CNN's.

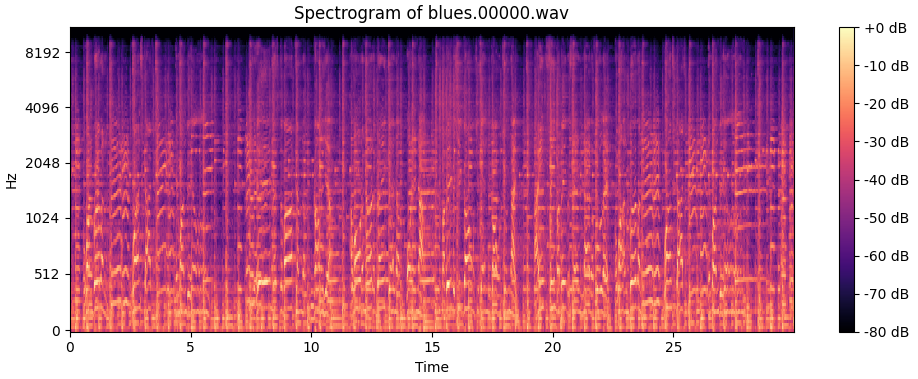


Figure 1: Example of a Mel-Spectrogram from the GTZAN Dataset.

For KNN Input

Audio features were extracted using the `librosa` library from the 30-second `.wav` files. The key features include:

- **MFCCs (Mel Frequency Cepstral Coefficients):** Capturing timbral characteristics.
- **Chroma Features:** Representing harmonic content.
- **Spectral Contrast:** Highlighting amplitude differences across frequency bands.

The features were averaged across the duration of each file to create a single feature vector per track.

For CNN Input

Spectrograms were generated from both the 30-second and 3-second `.wav` files. Each audio file was converted into a mel-spectrogram, log-scaled to enhance visual patterns, and resized to 128x128 pixels. These spectrograms provide a visual representation of the frequency and temporal dynamics of the audio data.

The dataset was split into 80% for training and 20% for validation for both the KNN and CNN models. The shorter 3-second files were primarily used to augment training data for the CNN model, improving its ability to generalize across diverse audio patterns.

Future extensions could incorporate larger datasets, such as the Free Music Archive (FMA) or the Million Song Dataset, to further enhance the robustness of the CNN model.

4 Approach

4.1 Feature Extraction for KNN

For the KNN model, we extracted structured features from audio files using the `librosa` library:

- **MFCCs (Mel Frequency Cepstral Coefficients):** Representing timbral texture:

$$C(n) = \sum_{k=1}^{K} \log(E_k) \cdot \cos\left(n \cdot (k - 0.5) \cdot \frac{\pi}{K}\right)$$

- **Chroma Features:** Representing harmonic content by mapping audio to the chromatic scale:

$$\text{Chroma}[p] = \sum_{f \in B(p)} \text{Energy}(f)$$

- **Spectral Contrast:** Measuring amplitude differences across frequency bands:

$$\text{Contrast}[b] = \log\left(\frac{\text{Peak}(b)}{\text{Valley}(b)}\right)$$

4.2 Spectrogram Generation for CNN

For the CNN, mel-spectrograms were generated from raw audio using:

- **Short-Time Fourier Transform (STFT):** Converting time-domain signals to the frequency domain:
$$X(f,t) = \sum_n x[n] \cdot w[n-t] \cdot e^{-j2\pi fn}$$
- **Mel Scale:** Mapping frequencies to the Mel scale:
$$m(f) = 2595 \cdot \log_{10}(1 + f/700)$$
- **Log Transformation:** Enhancing visual representation:
$$S_{\text{dB}} = 10 \cdot \log_{10}(S_{\text{power}})$$

The resulting spectrograms were resized to 128x128 pixels and saved as images.

4.3 CNN Architecture

The CNN model processes spectrogram images through:

- **Convolution Layers:** Extracting spatial features using:
$$z = \text{ReLU}(W * x + b),$$
where W is the convolution kernel, x is the input, and b is the bias term.
- **Max-Pooling Layers:** Reducing dimensions by taking the maximum value in a pooling window.
- **Dense Layer:** Transforming features into a final probability distribution using:
$$P(y = c \mid x) = \frac{\exp(z_c)}{\sum_{k=1}^{K} \exp(z_k)},$$
where z_c is the logit for class c, and K is the total number of classes.

The model was trained using the Adam optimizer and `categorical_cross-entropy` loss for 10 epochs.

5 Evaluation Results

5.1 Convolutional Neural Network (CNN)

The CNN model achieved strong performance metrics during both the training and validation phases.

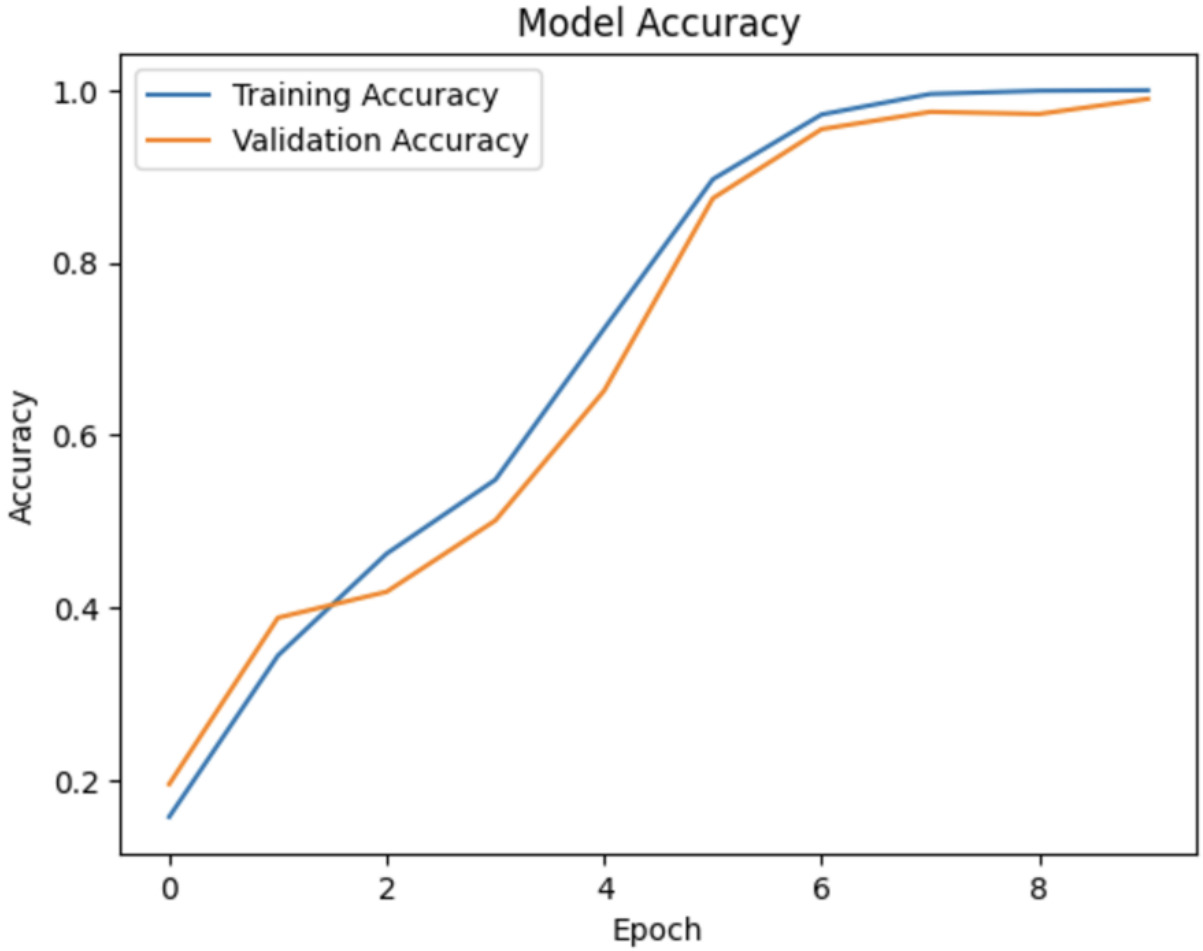
Final Training Results:

- Accuracy: 98.52%
- Loss: 0.0448

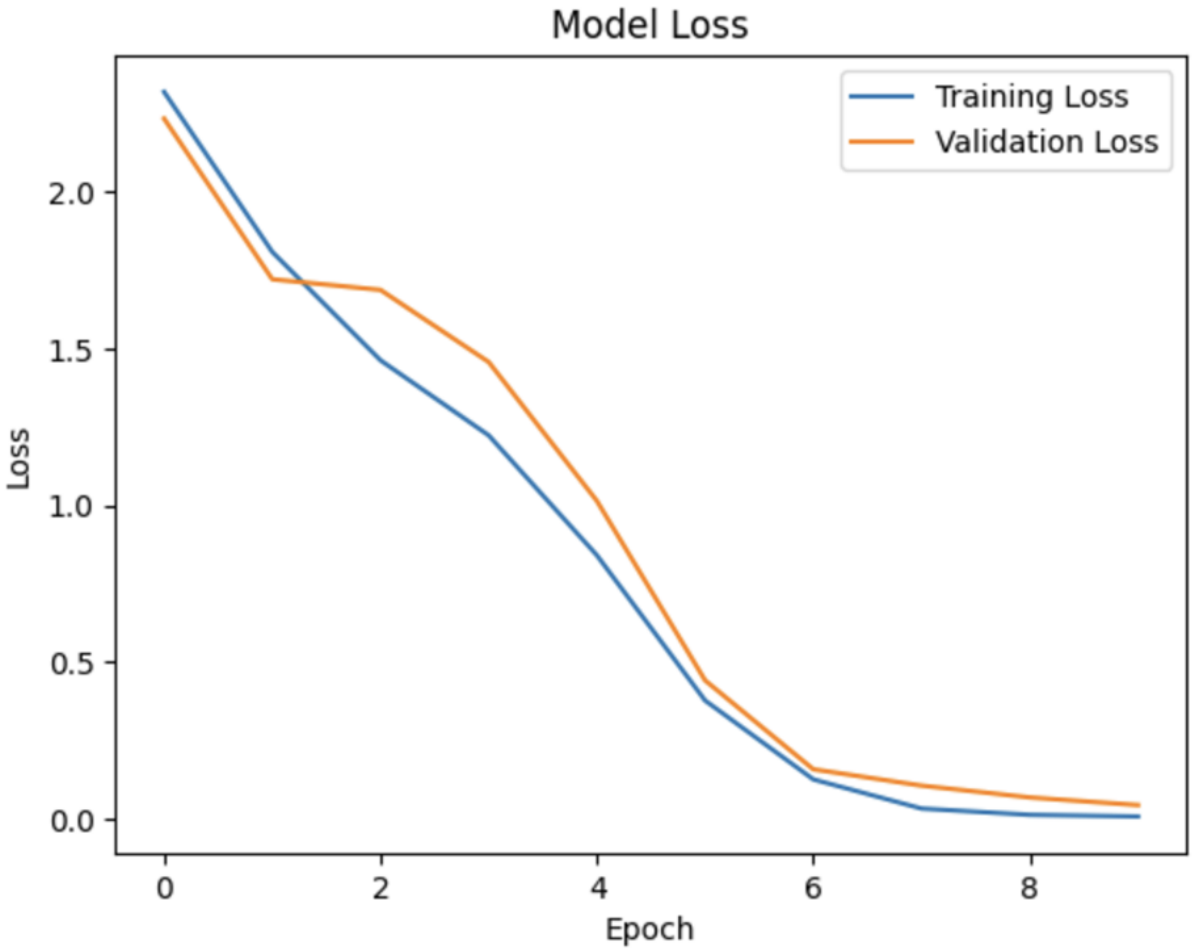
Validation Results:

- Validation Loss: 0.0434
- Validation Accuracy: 98.99%

These results indicate that the model effectively generalized to unseen data, as demonstrated by the high validation accuracy and low validation loss.



figureTraining Accuracy for CNN Model



figureTraining Loss for CNN Model

5.2 k-Nearest Neighbors (KNN)

The KNN model achieved the following performance metrics during evaluation:

Overall Accuracy:

- 90%

Precision, Recall, and F1-Score:

- Macro Average:
  - Precision: 0.90
  - Recall: 0.91
  - F1-Score: 0.90
- Weighted Average:
  - Precision: 0.90

– Recall: 0.90

– F1-Score: 0.90

The KNN classifier demonstrated strong performance, particularly in the Blues, Country, and Rock genres, but struggled slightly with genres like Jazz and Hip-Hop. Since these genres are closely related in instrumentation and rhythms, it is possible the KNN got these intermingled.

	precision	recall	f1-score	support
blues	0.90	1.00	0.95	35
classical	0.87	0.93	0.90	29
country	1.00	0.90	0.95	42
disco	0.95	0.88	0.91	48
hiphop	0.82	0.90	0.86	41
jazz	0.80	0.77	0.79	43
metal	0.96	0.93	0.95	46
pop	0.89	0.89	0.89	36
reggae	0.90	0.86	0.88	42
rock	0.93	1.00	0.96	38
accuracy			0.90	400
macro avg	0.90	0.91	0.90	400
weighted avg	0.90	0.90	0.90	400

figureClass-Level Performance Metrics for KNN

5.3 Baseline Comparison

KNN Baseline: The KNN model achieved 90% accuracy, which is comparable to the upper range of traditional machine learning methods (e.g., SVM, Random Forest) applied to the GTZAN dataset. These methods typically achieve 60-80% accuracy depending on pre-processing and feature extraction techniques. For instance, in "Musical Genre Classification of audio signals" [2], the authors reported early performance benchmarks with accuracies around 60-70% using k-Nearest Neighbors, Gaussian Mixture Models, and other feature-based classifiers.

CNN Performance Relative to Baseline: The CNN achieved a significantly higher accuracy (98.99%) compared to the KNN classifier and traditional baselines. This highlights the strength of deep learning in capturing complex patterns from spectrogram images, as

opposed to relying on hand-crafted features.

5.4 Analysis of Results

Learning Dynamics (CNN): The steady decline in loss and high validation accuracy suggest that the CNN effectively learned genre-specific patterns without overfitting.

Class-Level Performance (KNN): The class-level metrics for KNN show varying performance across genres:

- Genres with distinct audio features (e.g., Blues, Country, Metal) were classified with high accuracy.
- Overlap in feature space for certain genres (e.g., Jazz, Hip-Hop) led to lower performance.

Generalization: Both models achieved strong generalization on the GTZAN dataset, but the CNN's near-perfect validation accuracy suggests it captures the nuances of spectrogram features more effectively.

6 Conclusion

This project developed a music genre classification system by comparing a Convolutional Neural Network (CNN) and a k-Nearest Neighbors (KNN) classifier. The CNN achieved exceptional performance, with a validation accuracy of 98.99% and strong generalization capabilities. By leveraging spectrogram-based visual representations, the CNN effectively captured audio patterns and demonstrated its ability to distinguish each genre individually, making it the superior approach for this task.

The KNN classifier, trained on handcrafted features such as MFCCs and chroma, achieved an accuracy of 90%. It excelled in genres with distinct characteristics, like Blues and Country, but struggled with overlapping genres such as Jazz and Hip-Hop. This highlights the strengths and limitations of feature-based methods when compared to deep learning.

Overall, this project demonstrates the power of CNN's in audio classification tasks. Future work could involve using larger datasets with more genres, exploring ensemble approaches, or investigating other architectures for improved classification performance.

References

[1] Choi, K., et al. "Convolutional Recurrent Neural Networks for Music Classification." 2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2017, New Orleans, LA, pp. 2392-2396. doi:10.1109/ICASSP.2017.7952585.

[2] Tzanetakis, George, and Perry Cook. "Musical Genre Classification of Audio Signals." *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, 2002, pp. 293-302. `http://www.cs.cmu.edu/~gtzan/work/pubs/tsap02gtzan.pdf`.