IEEE Open Journal of the Computer Society

Received 13 May 2024; accepted 12 July 2024. Date of publication 19 July 2024;
date of current version 26 September 2024. The review of this article was arranged by Associate Editor Peng Li.

*Digital Object Identifier 10.1109/OJCS.2024.3431229*

# Musical Genre Classification Using Advanced Audio Analysis and Deep Learning Techniques

**MUMTAHINA AHMED[1], ULAND ROZARIO[1], MD MOHSHIN KABIR[2], ZEYAR AUNG[3] (Senior Member, IEEE), JUNGPIL SHIN[4], AND M. F. MRIDHA[5] (Senior Member, IEEE)**

[1]Department of Computer Science and Engineering, Bangladesh University of Business and Technology, Dhaka 1216, Bangladesh
[2]Faculty of Informatics, Eötvös Loránd University, H-1117 Budapest, Hungary
[3]Department of Computer Science, Khalifa University, Abu Dhabi 127788, UAE
[4]School of Computer Science and Engineering, The University of Aizu, Aizu-wakamatsu 965-8580, Japan
[5]Department of Computer Science and Engineering, American International University-Bangladesh, Dhaka 1229, Bangladesh

CORRESPONDING AUTHORS: M. F. MRIDHA; ZEYAR AUNG (e-mail: firoz.mridha@aiub.edu; zeyar.aung@ku.ac.ae)

**ABSTRACT** Classifying music genres has been a significant problem in the decade of seamless music streaming platforms and countless content creations. An accurate music genre classification is a fundamental task with applications in music recommendation, content organization, and understanding musical trends. This study presents a comprehensive approach to music genre classification using deep learning and advanced audio analysis techniques. In this study, a deep learning method was used to tackle the task of music genre classification. For this study, the GTZAN dataset was chosen for music genre classification. This study examines the challenge of music genre categorization using Convolutional Neural Networks (CNN), Feedforward Neural Networks (FNN), Support Vector Machine (SVM), k-nearest Neighbors (kNN), and Long Short-term Memory (LSTM) models on the dataset. This study precisely cross-validates the model's output following feature extraction from pre-processed audio data and then evaluates its performance. The modified CNN model performs better than conventional NN models by using its capacity to capture complex spectrogram patterns. These results highlight how deep learning algorithms may improve systems for categorizing music genres, with implications for various music-related applications and user interfaces. Up to this point, 92.7% of the GTZAN dataset's correctness has been achieved on the GTZAN dataset and 91.6% on the ISMIR2004 Ballroom dataset.

**INDEX TERMS** Convolutional neural networks, long short-term memory, support vector machine, k-nearest neighbors, genre classification.

## I. INTRODUCTION

Humans classify music into genres based on what they think of the music, how comfortable they are with the style, and their capacity to make decisions between genres that are unclear. This makes classifying music genres in the field of Music Information Retrieval (MIR) difficult. Sound processing, audio synthesis, audio effect creation, and music information retrieval depend on the extraction of audio features. David et al. [19] proposed the method to assess current audio feature extraction toolboxes and libraries by thoroughly examining their coverage, effort, presentation, and latency. There are several toolboxes for extracting audio features, such as "Essentia," which is recommended by Bogdanov et al. [3], "Librosa," which can be found in McFee et al. [18], which describes a Python package for audio and music signal processing, and so on. Purwins et al. [25] discussed convolutional neural networks, long short-term memory architectural variations among the deep learning models studied, and other popular feature representations such as log-mel spectra and raw waveform. Previous research also indicated that participants listened to music more often than any of the other activities in Pachet et al. [22] (i.e., watching television, reading books, and watching movies). Pachet et al. [22] study examines the connections between various factors, including individual characteristics, self-perceptions, cognitive capacities, and musical preferences. Music is essentially subjective because many people experience the same song differently. This study focuses on the model's ability to

identify music based on objective audio elements but does not examine how well these categories match human perceptions of genre. However, the current problem is to organize and manage the millions of music titles produced by society, as suggested by Rentfrow et al. [26]. In the context of artists and content creators, genre classification helps the production process to be more efficient and enables artists to target targeted audiences and refine their craft. Moreover, the online music streaming platforms Prey et al. [24] and the film and video game industries benefit significantly from an accurate genre classification. More significantly, it makes it difficult for us to fully understand and simulate people's musical tastes. The fingerprinting (FP) approach was implemented by Unal et al. [31] to solve the difficult problem of resilient data extraction in query-by-humming (QBH) systems under unpredictable conditions. The use of Mel Frequency Cepstral Coefficients in the classification of musical instruments is discussed by Loughran et al. [17]. MFCCs represent the spectral characteristics of audio signals, providing a compact yet highly informative feature set for music analysis. Previous research by Jensen et al. [10] summarizes how the MFCC is being calculated. They capture key attributes of sound, such as timbre and pitch, which are instrumental in discerning musical genres.

However, to improve the model's generalization over a greater range of musical styles, bigger and more diverse music datasets should be used for testing and training. Improving the model's ability to capture the complex musical features that characterize genres may require developing more advanced feature extraction techniques. The model improves its knowledge of musical differences by exploring audio data and identifying hidden components. Experimenting with various machine learning models and hyperparameter optimization techniques can help identify the best architecture for a specific genre classification problem, optimizing performance on the selected dataset and genres. Beyond the obvious classifications, this work explores the subtle characteristics that give each genre its persona.

The contributions to this article are as follows:

- This study has extracted the features of the WAV-formatted 30-second music files that were provided in the data set so that one can re-use those extracted features to train the proposed model and test/evaluate it.
- This survey attempted CNN, LSTM, SVM, kNN, and FNN to classify the genres of music and then proposed the modified Convolutional Neural Network model, which gives the best accuracy on both the GTZAN data set for music genre classification and the ISMIR2004 Ballroom dataset among all the trained models.
- The inquiry has evaluated the proposed Convolutional Neural Network model and compared it to the other models and to some well-known published papers.

The structural organization of this study describes a thorough overview of relevant studies in the realm of music genre classification, presented in Section II. Section III methodology thoroughly explains the suggested data set, algorithm, and model architectures and designs. Section IV contains the analysis results and a description of the data set, the experimental environment, and the experiment results. Section V also outlines the arguments made for the suggested system and recommends possible directions for further study. Section VI brings this article to a close.

## II. RELATED WORK

This study explores music genre classification using a combination of machine learning and deep learning methods. Inspired by the traditional approach of analyzing instruments in music, the study turns to automated methods due to the advancements in machine learning.

Cheng et al. [4] used Convolutional Neural Networks with five convolutional layers to classify the genres of music. The accuracy they got was 83.3%. Their hop size was set to 256 with a fast Fourier transform on 1024 frames. There was another approach by Ndou et al. [21] using traditional machine learning and deep learning. They presented a thoroughly reviewed paper on those approaches. They concluded their study with an accuracy of 92.69% by k-Nearest Neighbors. Their Convolutional Neural Network provided an accuracy of 72.40%. Sugianto et al. studied voting-based music genre classification [28]. They have also used the GTZAN dataset for music genre classification. They obtained a voting scheme accuracy of 71.87% and a single scheme accuracy of 63.49%, proving that the voting scheme offers greater accuracy than the single scheme. Prabhakar et al. [23] mentioned five different approaches such as WVG-ELNSC, SDA, RA-TSM, TSVM, and BAG proposed for music genre classification in their study. They obtained 93.51% accuracy using the proposed deep learning BAG model on GTZAN, ISMIR 2004, and MagnaTagATune data sets. Another work was featured by Ashraf et al. [1], where a hybrid CNN and RNN variant model was implemented. They achieved an accuracy of 89.30% by using a hybrid approach that combines CNN and Bi-GRU when the features were Mel-Spectrogram. Whereas with MFCC, they got 73.69% on the same hybrid model. They also enriched their study with the use of hybrid models such as CNN-GRU, CNN-BiLSTM, and CNN-LSTM. They used both MFCC and Mel-Spectrogram features on the hybrid models. Kostrzewa et al. [12] studied the classification of music genres, and in their study, the FMA tiny dataset was used in several tests to examine the effectiveness and performance of various models. The CNN and 1-D CRNN models produced the best outcomes, but the 2-D CRNN and LSTM models considerably underperformed. A growing need for sophisticated Music Information Retrieval (MIR) approaches to categorize digital music files into various genres was discussed in the paper by Mutiara et al. [20], the study emphasizes the necessity for an automatic genre tagging system and manual genre labeling. The optimal audio feature mixture, combining musical surface, Mel-Frequency Cepstrum Coefficients, tonality, and LPC, achieved a remarkable classification accuracy of 76.6%.

Another study by Farajzadeh et al. [7] presents PMG-Net, a customized deep neural network-based technique for

automatically classifying Persian music genres. The researchers built a dataset called PMG-Data, which included 500 songs from a variety of musical genres, including pop, rap, traditional, rock, and monody, to assess PMG-Net's performance. Other researchers can access this freely available dataset, which has been labeled for genre classification. The method's performance in classifying Persian music genres is satisfactory, as evidenced by PMG-Net's reported 86% accuracy on the PMG-Data. This study fills a research gap by utilizing deep neural network-based methods to categorize Persian music genres, a gap previously addressed by previous work focusing on western music.

Li et al. [13] utilized convolutional neural network techniques to extract musical highlights from sound music. They found that CNN can identify relevant information from the tangents of musical examples, improving over time. This approach is adaptable and validates the innate characteristics of audiovisual data collection, revealing the optimal parameter set for sound music arrangement. Fulzele et al. [8] highlighted the necessity of automatic music genre classification in the digital age, where many music files are readily available online. They used a hybrid model for classifying music genres that combines a Long Short-Term Memory (LSTM) and a Support Vector Machine (SVM) classifier. Compared to the individual accuracies of LSTM (69%) and SVM (84%) classifiers, the hybrid model comprising LSTM and SVM classifiers produced an 89% success rate in classification for musical genres. Schindler et al. [27] compared the effectiveness of several neural network topologies for automatically classifying music genres.

In conclusion, recent advancements in music genre classification, employing machine learning and deep learning, showcase significant progress. Notable research, such as Ashraf et al. [1] achieving 89.30% accuracy with a hybrid CNN and RNN variant and Ndou et al. [21] reaching 92.69% using k-Nearest Neighbors, exemplifies the capabilities of various models. Schindler et al. [27] study underscores the superiority of CNN-based approaches over manual feature creation. These developments have profound implications for the digital age, enhancing music streaming services with more accurate recommendations and personalized playlists and enriching user exploration. The future of music genre classification appears promising, with ongoing advancements in models and methodologies fueled by the dynamic intersection of machine learning and music. This study anticipates even more robust genre classification algorithms in the years ahead.

## III. METHODOLOGIES

The methodology used for the research will be discussed in this part, encompassing aspects such as data collection, data preprocessing procedures, and the foundational architectures constituting the baseline for the proposed musical genre classification system. A complete diagram of this study's workflow is presented in Fig. 3.

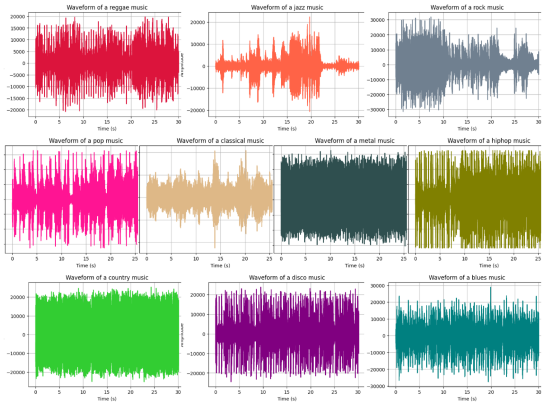


**FIGURE 1.** Waveform of a music file from each genre from the GTZAN's genre classification data set.

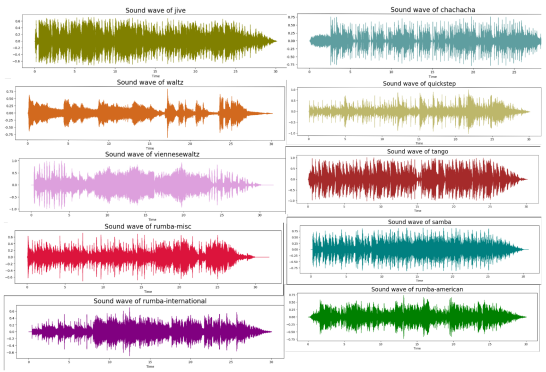


**FIGURE 2.** Waveform of a music file from each genre from the ISMIR 2004 genre classification data set.

### A. DATA COLLECTION

This study began by assembling a dataset, combining the well-known GTZAN Dataset and the Ballroom dataset, designed for ISMIR 2004's rhythm description contest. Following Tzanetakis et al. [30] methodology, the GTZAN dataset provides a diverse representation of ten musical genres, each with precisely 100 music files in WAV format, lasting 30 seconds. The Music Technology Group and Pompeu Fabra University created ISMIR2004, a genre classification dataset for a music data analysis contest. It consists of ten different genres: quickstep, jive,rumba-international, rumba-misc, chachacha, vinnsewaltz, samba, and waltz.The quality and diversity of this dataset significantly influence the performance of the proposed models. The standardized approach ensures a robust foundation for developing accurate models in music genre classification. Wave plots of samples from the GTZAN dataset and the ISMIR2004 dataset are shown in Figs. 1 and 2, respectively.

### B. DATA PREPROCESSING

In this experiment, a 30-second audio segment undergoes a comprehensive preprocessing stage. The audio is transformed into its corresponding Mel spectrums, quantized into audio

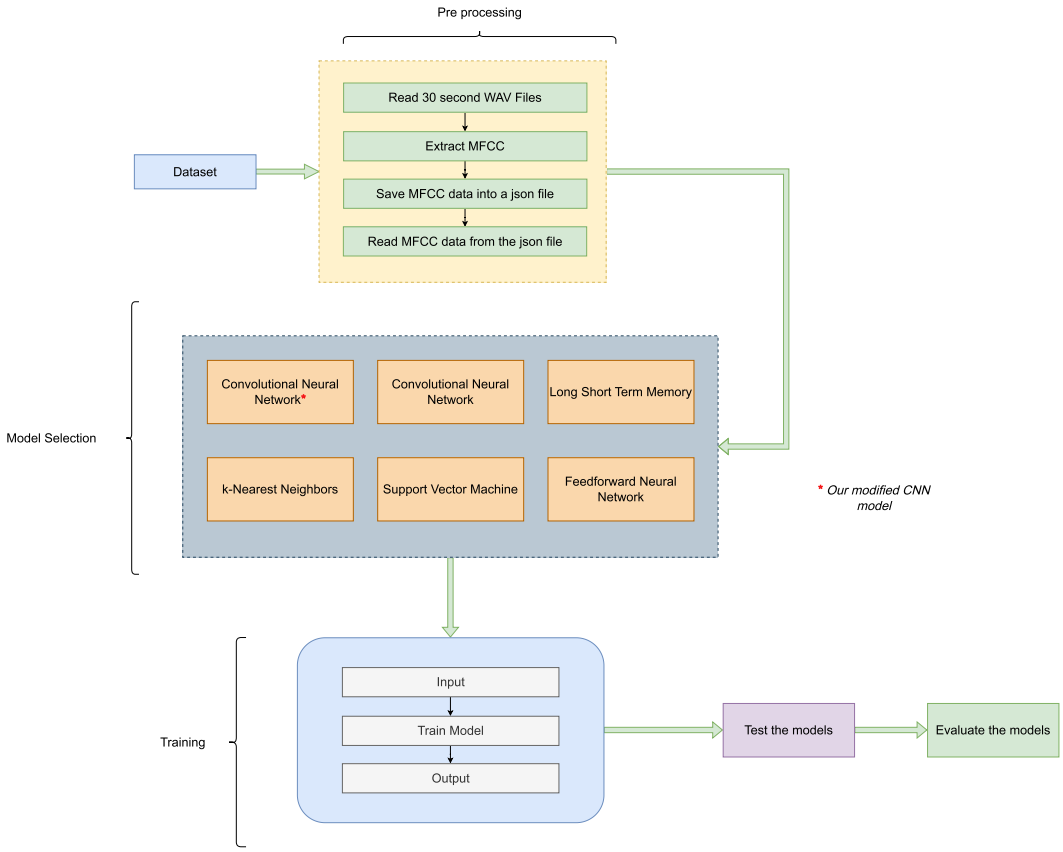


**FIGURE 3.** Workflow diagram of the proposed framework.

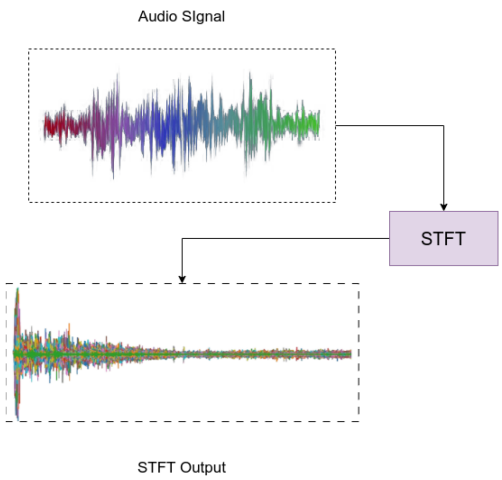


**FIGURE 4.** Extraction of Short-Time Fourier Transform from the WAV-formatted audio data.

signals at a sampling rate of 22050, and subjected to a Fast Fourier Transform (FFT) process applied to 2048 frames (visually represented in Fig. 4). This study initializes the preprocessing by extracting the Mel-Frequency Cepstral Coefficients from audio files from each genre. MFCC focuses on the perceptually important parts of the audio using a mel scale, similar to human hearing. This compressed data is less affected by noise and allows machine learning to efficiently recognize genres based on their unique sound characteristics. Feature selection plays an important role in machine learning for two key reasons. First, it improves model performance by removing irrelevant or redundant data. This prevents the model from making noise-based decisions and leads to more accurate predictions. Second, it reduces computational complexity. Less data means faster training times and lower resource demands, making the entire machine-learning process more efficient. The challenge in feature selection is the large dimensionality of audio data. Extracting several characteristics can result in a complicated feature space, making it difficult to determine which ones are most useful. Furthermore, finding a balance between informative and redundant features is critical. Removing too much data may result in the loss of important information for genre separation, while maintaining duplicate attributes may hamper performance and increase calculation time. Bergstra et al. [2] used an ensemble learner called ADABOOST to select from a set of audio features that have been extracted from segmented audio and then aggregated. Fig. 5 shows the spectrogram of a wav file from each genre; the spectrogram was plotted by extracting a short-time Fourier Transform and then visualizing by Librosa's spaceshow, and Figs. 6 and 7 show the amplitude vs frequency graph after the Fast Fourier Transform on audio data in both datasets, respectively.

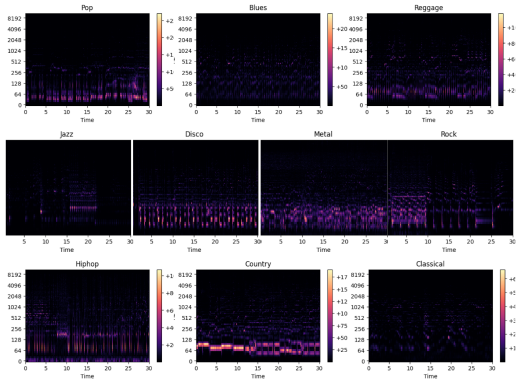


**FIGURE 5.** Time vs. Frequency (Hz) Spectrograms of wav file from each genre.

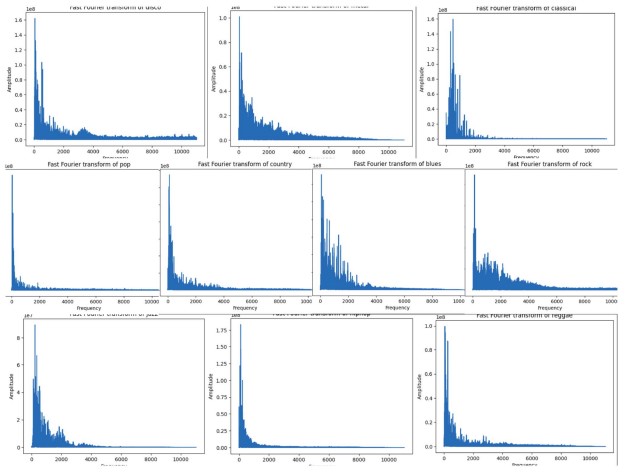


**FIGURE 6.** Amplitude vs. Frequency graph after Fast Fourier Transform on audio data on GTZAN dataset.

The STFT calculation is represented by the formula X(n$\Delta t$,m$\Delta f$), where n and m represent time and frequency, respectively. The $\Delta t$ signifies the time resolution, $\Delta f$ denotes the frequency resolution, and the summation considers windowed signal values along with a complex exponential term to account for frequency components. This method offers information on how the frequency content of a signal varies over time, which is helpful for tasks like spectrogram synthesis and audio analysis.

$$X(n\Delta_t, m\Delta_f) = \sum_{p=n-Q}^{n+Q} w((n-p)\Delta_t)x(p\Delta_t)e^{-j2\pi pm\Delta_t\Delta_f}\Delta_t \tag{1}$$

$$X_k = \sum_{n=0}^{N-1} x_n.e^{\frac{-i2\pi kn}{N}}$$

$x_k$ represents the k-th coefficient in the frequency domain. $n$ is the index variable used for the summation. $x_n$ represents the n-th data point in the original signal. $i$ is the imaginary unit. It's used to create the complex exponential term. $N$ represents

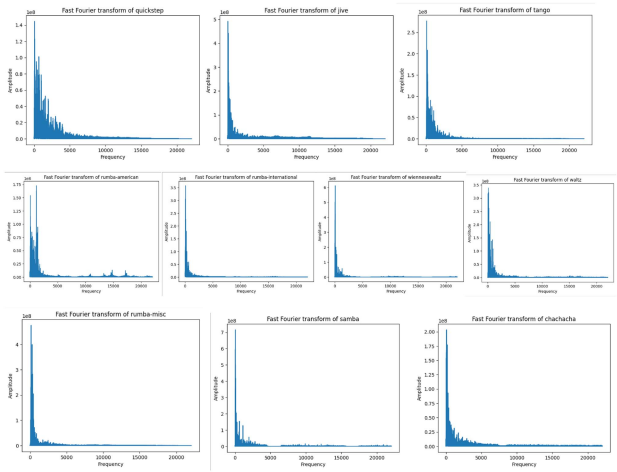


**FIGURE 7.** Amplitude vs. Frequency graph after Fast Fourier Transform on audio data on ISMIR2004 dataset.

the total number of data points in the original signal. $e$ is the base of the natural logarithm (approximately 2.71). $e^{\frac{-i2\pi kn}{N}}$ is a complex exponential term that plays a crucial role in converting the time-domain signal to the frequency domain. Now, the equations below show the separated forms into simpler compounds based on the operations on $x_k$ and the sum of operations on the even and odd samples of $x_k$.

$$= \sum_{m=0}^{\frac{N}{2}-1} x_{2\,m}.e^{\frac{-i2\pi k2m}{N}} + \sum_{m=0}^{\frac{N}{2}-1} x_{(2m+1)}.e^{\frac{-i2\pi k(2m+1)}{N}}$$

$$= \sum_{m=0}^{\frac{N}{2}-1} x_{2\,m}.e^{\frac{-i2\pi km}{\left(\frac{N}{2}\right)}} + e^{\frac{-i2\pi k}{N}} + \sum_{m=0}^{\frac{N}{2}-1} x_{(2m+1)}.e^{\frac{-i2\pi km}{\left(\frac{N}{2}\right)}} \tag{2}$$

The Hop size, set at 512 samples, is vital in controlling the analysis frequency. The preprocessing is facilitated through the utilization of the Librosa library, a powerful tool in the field of audio analysis. The preprocessing pipeline outlined in this experiment encompasses a series of carefully designed steps that transform raw audio data into a structured and informative representation.

$$C(x(t)) = F^{-1}[log(F[x(t)])] \tag{3}$$

$x(t)$ represents a signal, likely in the time domain, where $t$ denotes time. Signals in music are audio waveforms, representing sound over time. $F$ is a Fourier Transform, which is used in signal processing to convert signals from the time domain to the frequency domain. $log(F[x(t)])$ is done for compressing the dynamic range of the signal. We used the inverse transformation $F^{-1}$ for feature extraction. So, after calculating the logarithm of the modified signal, we use an inverse transformation to return to a new domain.

Fig. 8 shows a visual representation of a rock audio file's spectrogram and MFCC. These Mel Frequency Cepstral Coefficients serve as the basis for subsequent analysis and

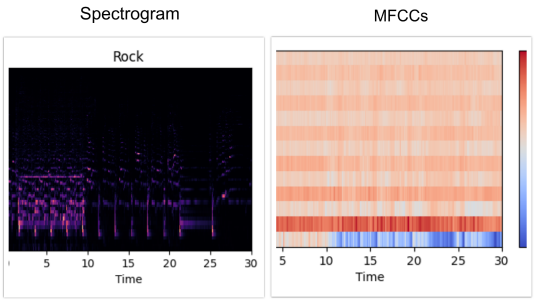


**FIGURE 8.** 'Rock' audio file's spectrogram and MFCC visualization.

classification tasks, ultimately contributing to the understanding and categorization of musical genres.

### C. ARCHITECTURE OF FNN MODEL

Feedforward Neural Networks are the foundation of deep learning architectures. It is also known as Artificial Neural Networks (ANN). FNNs are a class of neural networks where the information flows in one direction. The direction is from the input layer to the output layer, with the help of concealed intermediary layers. Unlike CNN, which is used for image classification, object detection [15], etc., FNN is widely used for many machine learning tasks, including classification, regression, and function approximation.

$$\frac{\partial L}{\partial w_{ik}^{[L]}} = \delta_i^{[L](j)} a_k^{[L-1](j)} \tag{4}$$

The paper presents an advanced neural network model designed for categorizing music genres using the Keras Sequential API. The model starts with an input layer flattening the data and adds densely connected hidden layers, each with 2048 to 64 neurons. These layers record low-level and high-level data representations, capturing complex relationships between traits and musical genres. Dropout layers prevent overfitting and follows each dense layer, while Softmax activation is used in the output layer. Adam, an optimizer, helps modify the model's weights throughout training to minimize loss. The model also undergoes L2 regularization to manage complexity. The goal is to accurately classify music genres based on complex feature correlations in audio data.

### D. ARCHITECTURE OF CNN MODEL

In a Convolutional Neural Network architecture, the network typically comprises a stack of convolutional layers, followed by optional pooling layers, fully connected layers, and an output layer. The 2D convolution formula is given below.

$$G[i, j] = \sum_{u=-k}^{k} \sum_{u=-k}^{k} H[u, v] F[i - u, j - v] \tag{5}$$

The $G[i, j]$ represents the output of the convolution operation at position $(i, j)$. $H[u, v]$ is the kernel or filter being applied to the input signal $F$ during convolution. It represents the weights or coefficients of the filter. $F[i - u, j - v]$ represents the input signal $F$ being shifted by $(u, v)$ during the convolution operation.

Convolutional layers are designed to detect spatial patterns and features in the input data. These layers use learnable filters to perform convolution operations over the input, capturing local features. However, CNN models are getting improved [9] by modification in many recent works. Dong [6] used CNNs to extract musical pattern features from the mel-scale spectrogram of audio signals. Multiple convolutional layers in Convolutional Neural Networks identify hierarchical features, and pooling layers then do spatial reduction and downsampling. For high-level feature extraction and classification, the output is flattened and fed into fully connected layers; for multi-class problems, softmax activation is frequently used. Dropout and batch normalization layers are examples of extra elements that can be included for regularization and stability during training. The CNN used in this work has an initial layer of 128 filters ($3 \times 3$ kernel) with ReLU activation, which was then added. Max-pooling, batch normalization, and another set of convolutional layers were then added. Feature extraction is improved with a final convolutional layer with 64 filters ($2 \times 2$ kernel). After that, the architecture switches to fully linked layers, which include a dense layer for high-level feature processing that has 64 units, ReLU activation, and dropout regularization.

$$f(x) = \max(0, x) \tag{6}$$

Finally, the output layer, equipped with softmax activation, provides predictions across 10 classes for the classification task. The softmax function is described below,

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{7}$$

Here, the softmax function $P(y_i)$ converts class scores to probabilities, showing the probabilities that the input belongs to each class $i$. It exponentiates the scores $z_i$ to ensure positivity before normalizing them to produce a probability distribution over all classes.

This carefully designed CNN architecture, also represented in Fig. 9, showcases a hierarchy of feature extraction capabilities, making it well-suited for a range of image classification tasks.

### E. ARCHITECTURE OF MODIFIED CNN MODEL

This study uses a Convolutional Neural Network architecture to classify music genres. The model uses Keras' sequential technique. Our model starts with three convolutional layers, both with a kernel size of (3, 3) and ReLU activation.These layers generate local features from input Mel-frequency cepstral coefficients (MFCCs), whose dimensionality (input shape) is determined by training data. The first layer utilizes 256 filters, while the second layer has 128 filters. For additional feature extraction, an optional third layer of 64 filters can be added. For each convolutional layer, MaxPooling2D layers with pool sizes (3, 3) and strides (2, 2) have been used to reduce the feature maps. Batch normalization layers

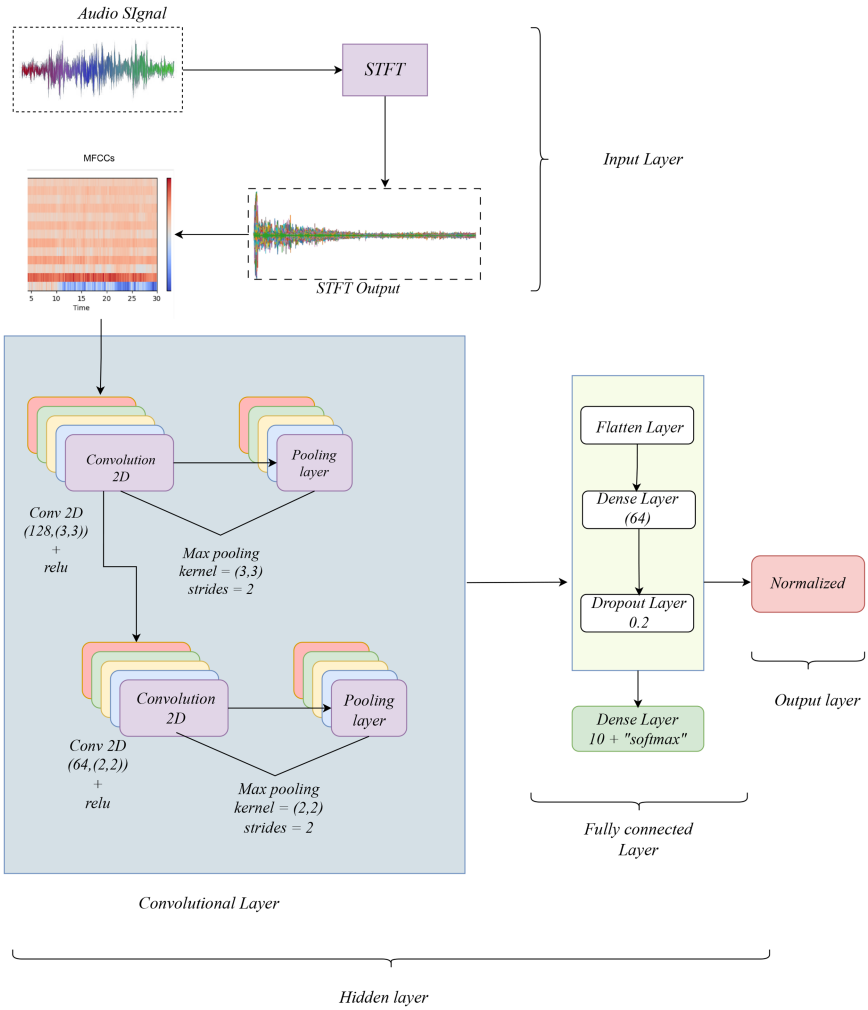


**FIGURE 9.** CNN Model to classify the genres of music on GTZAN's data set.

are introduced after each pooling layer to improve training stability. Two more convolutional layers have been added, applying kernel sizes of (3, 3) and (2, 2) with ReLU activation. These layers refine higher-level features extracted from the music data. Downsampling and training stability are obtained using MaxPooling2D and batch normalization layers, as in the previous sections. A flattening layer converts the pooled feature maps into one-dimensional vectors that may be fed into fully connected layers. A dense layer with 64 units and ReLU activation is employed for further processing. A dropout layer with a rate of 0.2 helps to prevent overfitting. The last layer is dense, consisting of 10 units, and activated with softmax. This layer predicts the probability distribution of music falling into one of the 10 genre classifications. Our model is compiled using the Adam optimizer with a learning rate of 0.0001.

Both traditional CNN and modified CNN are CNN structures for music genre categorization; however, their complexity varies. Traditional CNN has a simpler structure with two "Conv2D" layers, each with 128 filters and a kernel size of (3, 3). This process pulls characteristics from the MFCC input. In comparison, updated cnn utilizes a possibly more powerful

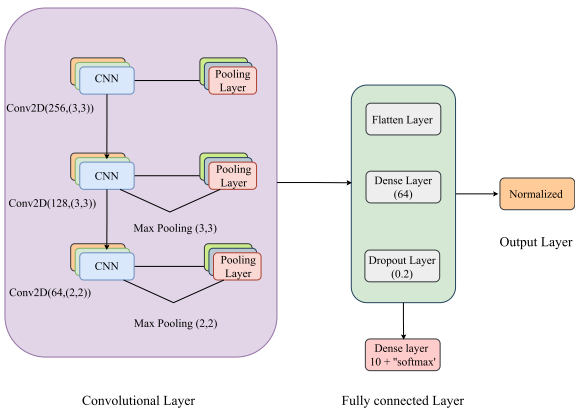


**FIGURE 10.** Proposed Modified CNN Architecture to Classify the Genre of Music.

design. It begins with three "Conv2D" layers, the first having 256 filters, the second with 128 filters, and the third with 64 filters, which is shown in Fig. 10. This augmentation is especially notable since it represents an honest attempt to

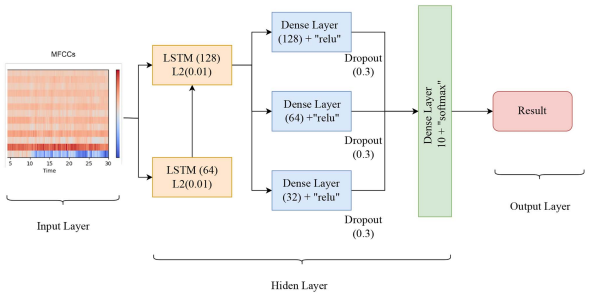


**FIGURE 11.** Proposed LSTM Model to Classify the Genre of Music.

modify higher-level characteristics collected from music data, which might lead to increased classification accuracy. This may allow for richer feature extraction. Modified CNN was updated with higher model capacity due to extra convolutional layers and filters, which may lead to greater performance, but it must be monitored for overfitting. Traditional CNN delivers computational efficiency, but its limitations may restrict its ability to capture complicated genre-specific properties.

### F. ARCHITECTURE OF RNN-LSTM MODEL

This work presents an advanced Recurrent Neural Network (RNN) architecture designed to represent sequential data and extract subtle information from audio inputs, enabling more precise genre categorization. Each layer in the architecture has been thoughtfully created to play a distinct function in the categorization process. Because of their fundamental connection with music's sequential and temporal aspects, RNNs can capture both short- and long-term dependencies, as stated in [29], and can be flexible enough to operate with a variety of input representations. The input layer of the RNN serves as the point of entry for audio data, which is presented in the form of sequential feature vectors. This study opts for Mel-frequency cepstral coefficients as the primary feature representation, encapsulating essential spectral information.

This article emphasizes the use of recurrent layers, specifically Long Short-Term Memory (LSTM) cells, to capture long-term temporal dependencies in audio data. The flattened feature vector undergoes connection to fully connected layers with Rectified Linear Units for non-linearity.

To prevent overfitting, dropout layers (0.3) are strategically placed between fully connected layers, promoting diversity in feature reliance during training. L2 regularization (coefficient: 0.01) further ensures model robustness. The final output layer, matching music genre count, employs softmax activation for probability distributions, with the highest probability determining the model's genre classification.

The proposed Recurrent Neural Network architecture shown in Fig. 11 defines a deep neural network with LSTM layers for processing sequential data. The network has numerous LSTM layers, including two with 128 and 64 units, respectively, all of which are fitted with L2 regularization to improve generalization and reduce overfitting. Following these recurrent layers are Dense layers with Rectified Linear Unit activation functions, which are supplemented with dropout layers to counteract overfitting. The model culminates in an output layer with softmax activation, allowing for multi-class categorization into 10 separate music genres. This architectural arrangement uses the temporal correlations inherent in audio data, proving its potential to improve the accuracy and performance of music genre categorization tasks.

### G. ARCHITECTURE OF SVM MODEL

In our study, the SVM architecture is described in the obtained Mel-Frequency Cepstral Coefficients (MFCC) features of audio samples for music genre classification. Initially, the audio files are sampled at a rate of 22,050 Hz, and each track is divided into numerous parts lasting 30 seconds each. MFCC features are then computed for each segment via the librosa package. The generated MFCC vectors are flattened into one-dimensional arrays and divided into training and testing sets. Changsheng et al. [32] propose effective algorithms to automatically classify and summarize music content, and SVM is used to classify music. Selecting the hyperplane in the feature space that best divides several classes is the foundation of SVMs, as opposed to neural networks. A linear SVM model is trained on the training set using the sci-kit-learn library's SVC class, with the regularization parameter (C) set to one. Subsequently, the trained model is used to predict genre labels for the test set. The accuracy-score function from sci-kit-learn is used to evaluate the SVM model's performance in classifying music genres. This architecture shows the process of feature extraction, model training, prediction, and evaluation in SVM-based music genre classification, with a focus on the audio sample rate and track duration parameters.

### H. ARCHITECTURE OF KNN MODEL

In our study, the K Nearest Neighbors (KNN) approach for music genre classification starts by modeling each music sample as a set of statistical data taken from its audio signal, namely the mean and covariance matrices of Mel-Frequency Cepstral Coefficients (MFCC). These attributes capture key audio elements, such as spectral content and timbral properties. The program then determines the distance between each pair of samples using a mathematical calculation known as the Kullback-Leibler divergence, which evaluates the difference in their feature distributions. The Kullback-Leibler divergence formula is as follows,

$$KL(\hat{y}||y) = \sum_{c=1}^{M} \hat{y}_c \log \frac{\hat{y}_c}{y_c} \tag{8}$$

In the study of the kNN algorithm, $\hat{y}$ represents the estimated or predicted probabilities of a data point belonging to each of the $M$ classes, while $y$ represents the true probabilities or the ground truth distribution. $\hat{y}_c$, $y_c$ denote the probabilities associated with category $c$ in the distributions $\hat{y}_c$ and $y_c$, respectively. It is used as a distance metric to determine related data points for classification based on class probabilities.

$$\text{Euclidean Distance} = \sqrt{(p_1 - q_1)^2 + \cdots + (p_n - q_n)^2} \tag{9}$$

**TABLE 1. Comparing Different Approaches and Their Accuracy on GTZAN Dataset**

| Method | Accuracy | F1 Score | Precision |
|---|---|---|---|
| **Modified CNN** | **92.7%** | 0.96 | 0.93 |
| Convolutional Neural Network | 85.56% | 0.85 | 0.89 |
| Support Vector Machine(SVM) | 79% | 0.80 | 0.81 |
| Feedforwarding Neural Network | 76% | 0.79 | 0.77 |
| RNN-LSTM | 73% | 0.72 | 0.73 |
| K-Nearest Neighbors (kNN) | 70% | 0.68 | 0.65 |

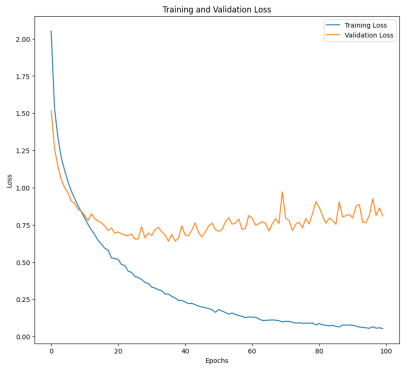


**FIGURE 12. Modified Convolutional Neural Networks model's loss after 100 epochs.**

In the topic of kNN, the distance is calculated between each pair of data points in the dataset using this formula. Based on these distances, the kNN algorithm chooses the K nearest neighbors for each test sample. The genre labels of these neighbors are then utilized to estimate the genre of the test sample via a voting process, with the most frequently occurring genre among the neighbors selected as the predicted genre label. Finally, the model's accuracy is assessed by comparing the predicted genre labels to the actual genre labels of the test samples. This approach enables the KNN algorithm to categorize music samples into distinct genres based on auditory attributes and similarities to other samples in the database.

## IV. RESULT & ANALYSIS

After the training as well as the evaluation of the models, this study found out that the modified CNN model demonstrated exceptional performance, achieving an impressive accuracy rate of 92.7% (shown in Fig. 13) on the GTZAN music genre classification dataset, where the base Convolutional Neural Network provided an accuracy of 85.56%, SVM attained an accuracy of 79%, FNN gave an accuracy of 76%, Long Short Term Memory gave 73% of accuracy, and kNN supported with an accuracy of 70%. Table 1 shows a detailed comparison of results. The result states that the proposed CNN model performs better on GTZAN's music genre classification data set.

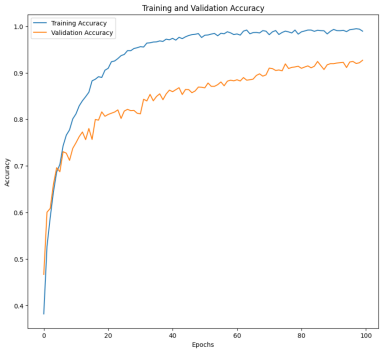


**FIGURE 13. Obtained accuracy of the modified CNN model after 100 epochs.**

**TABLE 2. Comparing Different Approaches and Their Accuracy on Ballroom Dataset**

| Method | Accuracy | F1 Score | Precision |
|---|---|---|---|
| **Modified CNN** | **91.6%** | 0.92 | 0.9 |
| Convolutional Neural Network | 90% | 0.92 | 0.94 |
| Support Vector Machine(SVM) | 76% | 0.83 | 0.88 |
| Feedforwarding Neural Network | 74.1% | 0.81 | 0.74 |
| RNN-LSTM | 71.6% | 0.68 | 0.72 |
| K-Nearest Neighbors (kNN) | 68.5% | 0.75 | 0.69 |

The study extensively tested using the BallRoom genre classification dataset, consisting of 698 occurrences of 10 genres. Mel-frequency cepstral coefficient data from corresponding WAV files was used to train models. The modified Convolutional Neural Network achieved the highest accuracy at 91.6%, followed closely by the typical CNN with 90%. The Feedforward Neural Network and Support Vector Machine models achieved accuracies of 74.1% and 76%, respectively. The k-Nearest Neighbors model had an accuracy of 68.5%, and the Recurrent Neural Network with Long Short-Term Memory (RNN-LSTM) achieved a performance of 71.6%.

These findings highlight the research's efficacy in using the modified CNN model to classify genres more accurately than other examined models. The various accuracy results show the advantages and disadvantages of each model, offering insightful information for additional improvement and optimization in subsequent iterations of the classification system.

Precision in this study refers to the percentage of accurate positive predictions among all positive predictions generated by the model.

$$Precision = \frac{TP}{TP + FP} \tag{10}$$

The goal of recall is to find the most appropriate items from all the available options. It quantifies the percentage of predictions that came true out of the dataset's actual positive

**TABLE 3. Comparison With State of the Art**

| | Year | Model | Dataset | Accuracy | F1 Score | Precision |
|---|---|---|---|---|---|---|
| **This study** | 2024 | CNN | GTZAN, ISMIR | 92.7% | 0.96 | 0.93 |
| Prabhakar et al. [23] | 2023 | WVG-ELNSC, SDA, RA-TSM, TSVM, BAG | GTZAN, ISMIR2004, MagnaTagATune | 93.51% | - | - |
| Ashraf et al. [1] | 2023 | LSTM,Bi-LSTM,GRU,Bi-GRU | GTZAN | 89.3% | 0.88 | 0.85 |
| Kostrzewa et al.[12] | 2021 | Ensemble-1 Vote | FMA | 53.39% | 0.54 | 0.56 |
| Ndou et al.[21] | 2021 | SVM | GTZAN and BMD | 79.7% | 0.519 | - |
| Karunakaran et al.[11] | 2018 | Hybrid Classifier On Spark | GTZAN,FMA | 82.4% | 0.82 | 0.82 |

cases. Where TP means true positive, and FN refers to a false negative.

$$Recall = \frac{TP}{TP + FN} \tag{11}$$

The F1 score is an essential metric for evaluating the accuracy and reliability of classification models in various fields because it can balance the trade-off between precision and recall and effectively addresses imbalanced datasets. The F1 score was computed using the following formula:

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{12}$$

### A. DATASET DESCRIPTION

In this research, the data were divided into two sets. One is for training and the other is for testing, which is 70% and 30% for both GTZAN and ISMIR2004 datasets, respectively. The following papers split the dataset 70/30 (70% used to train, 30% used to test), and the total number of iterations performed in the experiment is 2,180. The batch size is set to 32, and the epochs are set to 100. De Sousa et al. [5] randomly sorted the GTZAN dataset with 1000 pieces and selected the first 667 to train and the latest 333 to test the model created. Representing two-thirds (66.7%) of the dataset was used for training, and one-third (33.3%) for testing. This process was repeated 30 times.

### B. EXPERIMENTAL ENVIRONMENT

The experiment was conducted on a Linux-based virtual computer with an NVIDIA Tesla T4 GPU, executing 2048 iterations and 100 epochs. Adam [16], an optimization technique, was used to reduce difficulties in deep neural network training, achieving a 0.0001 learning rate in the modified architecture.

### C. EXPERIMENTAL RESULT

The architecture of the proposed model has a remarkable accuracy rate of 92.7% based on the modified CNN model. Table 3 shows the result of the method compared with Cheng et al. [4], De Sousa et al. [5], Elbir et al. [33], Lidy et al. [14], and Bergstra et al. [2]. All these research studies split the GTZAN dataset into 70% for the training set and 30% for the test set, converted all the audio in the dataset into their respective MFCC, and sent them to the proposed CNN model for training. It is important to consider alternative approaches, such as N. Karunakaran and A. Arya [11] employed Principal Component Analysis (PCA) for dimensionality reduction and selected the 30 principle components, different Machine learning models are trained and tested with 10-fold stratified samples, 9 folds for training, and 1 fold for testing. Prabhakar et al. [23] explored a deep learning approach using a BAG model, achieving 93.51% accuracy across three datasets: GTZAN, ISMIR 2004, and MagnaTagATune. Figs. 12 and 13 show the loss curve and accuracy curve of the model, respectively.

## V. DISCUSSION AND FUTURE WORK

The usefulness of many machine learning models, such as K-nearest neighbors, Feedforward Neural Networks, Convolutional Neural Networks, Support Vector Machines, and Recurrent Neural Networks with Long Short-Term Memory (RNN-LSTM), for the categorization of musical genres was investigated in this study. Compared to other models, CNNs obtained better results, demonstrating their capacity to identify complex spectrogram patterns–a crucial skill for genre classification. However, the interpretability of CNNs is limited because of their black-box character, highlighting the need for more study to improve model comprehension. The choice of the dataset, traditionally GTZAN, should be reconsidered for more extensive, diverse, and real-world datasets to better represent modern music genres. Addressing dataset bias is crucial for improving model generalizability. Music genre classification models not only serve their primary purpose but also impact music recommendation systems, content organization, and staying updated with musical trends. Further research can expand these models' applications, incorporating user behavior analysis, cross-cultural music knowledge, and additional features like lyrics analysis, enhancing the overall listening experience on digital platforms. To ensure fair comparisons, continuous improvement of evaluation criteria and benchmarking standards is necessary. Defined criteria and methods will facilitate impartial model evaluations. In conclusion, this study highlights the potential of deep learning models, particularly CNNs, for music genre classification.

Future research should prioritize interpretability, diverse datasets, multimodal techniques, and practical applications to enhance precision and utility in music genre categorization systems, benefiting both the music industry and users.

## VI. CONCLUSION

This study demonstrates the efficacy of convolutional neural networks by classifying music genres with an impressive 92.7% accuracy. However, there are still difficulties since genre separation is complex and impacted by personal, cultural, and historical variables. The study recognizes the intricacy of music genre classification and the necessity for more investigation. Even with exceptional results on the GTZAN dataset, there is still a lot of new ground that has to be explored. Future research is broadening its scope and concentrating on other aspects of audio, such as spectral quality, rhythmic patterns, and lyric analysis. In order to improve accuracy and capture a variety of genre features, complex model architectures are used, which include ensemble learning and attention procedures. Classification across genres and cultures is a growing field of study that presents fascinating problems. In summary, this work represents a development in categorizing musical genres and highlights the need for more research. Investigating proposed paths can enhance our knowledge of musical genres, impacting information structure, music suggestion programs, and wider uses in the constantly changing music sector and worldwide consumer inclinations.

## ACKNOWLEDGMENT

The authors would like to sincerely thank the Advanced Machine Intelligence Research Lab (AMIR Lab) for its continuous support and instructions to fulfill what the author wanted to achieve.

## REFERENCES

[1] M. Ashraf et al., "A hybrid CNN and RNN variant model for music classification," *Appl. Sci.*, vol. 130, no. 3, 2023, Art. no. 476.
[2] J. Bergstra, N. Casagrande, D. Erhan, D. Eck, and B. Kégl, "Aggregate features and ADABOOST for music classification," *Mach. Learn.*, vol. 65, pp. 473–484, 2006.
[3] D. Bogdanov et al., "Essentia: An audio analysis library for music information retrieval," in *Proc. 14th Conf. Int. Soc. Music Inf. Retrieval*, A. Britto, F. Gouyon, and S. Dixon, eds., Curitiba, Brazil, Nov. 2013, pp. 493–498.
[4] Y.-H. Cheng, P.-C. Chang, and C.-N. Kuo, "Convolutional neural networks approach for music genre classification," in *Proc. IEEE Int. Symp. Comput., Consum. Control*, 2020, pp. 309–403.
[5] J. M. de Sousa, E. T. Pereira, and L. R. Veloso, "A robust music genre classification approach for global and regional music datasets evaluation," in *Proc. IEEE Int. Conf. Digit. Signal Process.*, 2016, vol. 35, pp. 109–113.
[6] M. Dong, "Convolutional neural network achieves human-level accuracy in music genre classification," 2018, *arXiv:1802.09697*.
[7] N. Farajzadeh, N. Sadeghzadeh, and M. Hashemzadeh, "PMG-Net: Persian music genre classification using deep neural networks," *Entertainment Comput.*, vol. 44, 2023, Art. no. 100518.
[8] P. Fulzele, R. Singh, N. Kaushik, and K. Pandey, "A hybrid model for music genre classification using LSTM and SVM," in *Proc. IEEE 11th Int. Conf. Contemporary Comput.*, 2018, pp. 1–3.
[9] A. Ishraq, A. A. Lima, M. M. Kabir, M. S. Rahman, and M. F. Mridha, "Assessment of building damage on post-hurricane satellite imagery using improved CNN," in *Proc. IEEE Int. Conf. Decis. Aid Sci. Appl.*, 2022, pp. 665–669.
[10] J. H. Jensen, M. G. Christensen, M. N. Murthi, and S. H. Jensen, "Evaluation of MFCC estimation techniques for music similarity," in *Proc. IEEE 14th Eur. Signal Process. Conf.*, 2006, pp. 1–5.
[11] N. Karunakaran and A. Arya, "A scalable hybrid classifier for music genre classification using machine learning concepts and spark," in *Proc. IEEE Int. Conf. Intell. Auton. Syst.*, 2018, pp. 128–135.
[12] D. Kostrzewa, P. Kaminski, and R. Brzeski, "Music genre classification: Looking for the perfect network," in *Proc. Int. Conf. Comput. Sci.*, 2021, pp. 55–67.
[13] T. Li, A. B. Chan, and A. Chun, "Automatic musical pattern feature extraction using convolutional neural network," *Genre*, vol. 100, pp. 546–550, 2010.
[14] T. Lidy, A. Rauber, A. Pertusa, and J. M. Inesta, "Mirex 2007 combining audio and symbolic descriptors for music classification from audio," in *Proc. MIREX 2007-Music Inf. Retrieval Eval. eXchange*, Citeseer, 2007.
[15] A. A. Lima, M. M. Kabir, S. C. Das, M. N. Hasan, and M. Mridha, "Road sign detection using variants of YOLO and R-CNN: An analysis from the perspective of Bangladesh," in *Proc. Int. Conf. Big Data, IoT, Mach. Learn.: BIM 2021*, Springer, 2022, pp. 555–565.
[16] M. Liu, W. Zhang, F. Orabona, and T. Yang, "Adam: A stochastic method with adaptive variance reduction," 2020, *arXiv:2011.11985*.
[17] R. Loughran, J. Walker, M. O'Neill, and M. O'Farrell, "The use of mel-frequency cepstral coefficients in musical instrument identification," in *Proc. Int. Comput. Music Conf. Proc.*, 2008.
[18] B. McFee et al., "librosa: Audio and music signal analysis in python," in *Proc. 14th python Sci. Conf.*, 2015, vol. 8, pp. 18–25.
[19] D. Moffat, D. Ronan, and J. D. Reiss, "An evaluation of audio feature extraction toolboxes," 2015.
[20] A. B. Mutiara, R. Refianti, and N. R. Mukarromah, "Musical genre classification using SVM and audio features," *TELKOMNIKA Telecommunication Comput. Electron. Control*, vol. 140, no. 3, pp. 1024–1034, 2016.
[21] N. Ndou, R. Ajoodha, and A. Jadhav, "Music genre classification: A review of deep-learning and traditional machine-learning approaches," in *Proc. 2021 IEEE Int. IOT, Electron. Mechatronics Conf.*, 2021, pp. 1–6.
[22] F. Pachet and J.-J. Aucouturier, "Improving timbre similarity: How high is the sky," *J. Negative Results Speech Audio Sci.*, vol. 10, no. 1, pp. 1–13, 2004.
[23] S. K. Prabhakar and S.-W. Lee, "Holistic approaches to music genre classification using efficient transfer and deep learning techniques," *Expert Syst. Appl.*, vol. 211, 2023, Art. no. 118636.
[24] R. Prey, "Nothing personal: Algorithmic individuation on music streaming platforms. media," *Culture Soc.*, vol. 400, no. 7, pp. 1086–1100, 2018.
[25] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, and T. Sainath, "Deep learning for audio signal processing," *IEEE J. Sel. Topics Signal Process.*, vol. 13, no. 2, pp. 206–219, May 2019.
[26] P. J. Rentfrow and S. D. Gosling, "The do re mi's of everyday life: The structure and personality correlates of music preferences," *J. Pers. Social Psychol.*, vol. 840, no. 6, 2003, Art. no. 1236.
[27] A. Schindler, T. Lidy, and A. Rauber, "Comparing shallow versus deep neural network architectures for automatic music genre classification," in *Proc. 9th Forum Media Technol.*, 2016, pp. 17–21.
[28] S. Sugianto and S. Suyanto, "Voting-based music genre classification using melspectogram and convolutional neural network," in *Proc. IEEE Int. Seminar Res. Inf. Technol. Intell. Syst.*, 2019, pp. 330–333, doi: 10.1109/ISRITI48646.2019.9034644.
[29] C. P. Tang, K. L. Chui, Y. K. Yu, Z. Zeng, and K. H. Wong, "Music genre classification using a hierarchical long short term memory (lstm) model," *Proc. SPIE*, vol. 10828, pp. 334–340, 2018.
[30] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Trans. Speech Audio Process.*, vol. 10, no. 5, pp. 293–302, Jul. 2002.
[31] E. Unal, E. Chew, P. G. Georgiou, and S. S. Narayanan, "Challenging uncertainty in query by humming systems: A fingerprinting approach," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 160, no. 2, pp. 359–371, Feb. 2008.
[32] C. Xu, N. C. Maddage, and Xi. Shao, "Automatic music classification and summarization," *IEEE Trans. Speech Audio Process.*, vol. 13, no. 3, pp. 441–450, May 2005.
[33] A. E. H. B. Çam, M. E. Iyican, B. Öztürk, and N. Aydin, "Music genre classification and recommendation by using machine learning techniques," in *Proc. IEEE Innovations Intell. Syst. Appl. Conf.*, 2018, pp. 1–5.