Multimedia Tools and Applications
https://doi.org/10.1007/s11042-024-19243-3

Music genre classification and recognition using convolutional neural network

Nandkishor Narkhede[1] · Sumit Mathur[2] · Anand Bhaskar[2] · Mukesh Kalla[3]

Received: 3 December 2022 / Revised: 9 April 2024 / Accepted: 12 April 2024

Abstract
The music genre classification system is crucial to users in the digital music business since it allows them to be more effective. Music suggestion and availability to consumers is one of the most successful uses of genre classification. Songs may be easily accessible by users when the genre of the song is recognized, and music recommendations to users are made simple with an accurate categorization system in place. Furthermore, automated genre categorization is necessary to tackle difficulties such as finding similar songs, identifying cultures that would enjoy certain music, and conducting surveys. Machine learning approaches have recently been shown to be useful in a variety of classification tasks, including music genre categorization. As a result, this research investigates the use of Convolutional Neural Networks (CNN) for music genre categorization. For this study, a fresh dataset of 1000 traditional music from ten genres was employed. Content-based features, were retrieved from the songs in the dataset and used as input into the classifier, as feature extraction is critical to audio analysis. We got the results of the accuracy level of the system is 98.9% with a precision of 98.7%, recall of 98.5%, and f1 score of 97.5%.

Keywords Convolution Neural Network(CNN) · Music Information Retrieval (MIR) · Power Spectrum · Spectrogram · GTZAN · Short-term Fourier Transform (STFT) · Mel-spectrogram · Mel-frequency cepstrum coefficient (MFCC)

1 Introduction

With the widespread usage of numerous music platforms, a rising amount of music is extensively disseminated, confusing audiences and platforms trying to manage this music. Furthermore, manually organizing and distinguishing such a large amount of music is

✉ Nandkishor Narkhede
nandkishor.narkhede@sakec.ac.in

[1] Shah and Anchor Kutchhi Engineering College, Chembur, Mumbai, India

[2] Department of Electronics and Communication Engineering, Sir Padampat Singhania University, Udaipur, Rajasthan, India

[3] Department of Computer Science Engineering, Sir Padampat Singhania University, Udaipur, Rajasthan, India

Published online: 17 April 2024

impractical. As a result, figuring out how to develop a practical solution to this challenge is necessary, but difficult. The majority of current approaches try to classify the music genre, which is a top-level label on music that helps audiences define and explain different types of music. Meanwhile, for music platforms to categorize music into distinct groups, exact music genre classification is crucial. As a result, the discipline of music information retrieval has become quite interested in music genre categorization (MIR) [1, 2].

Feature extraction and classifier learning, as two critical components for music genre classification, may have a significant impact on the performance of most classification systems [3]. The goal of feature extraction is to find appropriate representations of data that are anticipated to be categorized using feature vectors or pairwise similarities [4]. Features and representations of music are given to a classifier after feature extraction, to map feature vectors into distinct music genres. To describe music signals, Baniya et al. [6] use timbral texture characteristics (i.e. Mel-frequency Cepstral Coefficient) and rhythm content information such as beat histogram (BH) [1]. Then, as a classifier, they mix the Extreme Learning Machine (ELM) [7] with bagging [8]. Arabi et al. [9] incorporate chord characteristics as well as chord progression data into feature extraction. In addition, by utilizing a Support Vector Machine (SVM), they proved chord features in conjunction with low-level features [5] can provide higher classification accuracy. Sarkar et al. [10] describe the state-of-the-art accomplishment, which uses Empirical Mode Decomposition (EMD) for signal component extraction and relies only on pitch-based characteristics. Even if all of the solutions listed above operate well in some cases, these hand-crafted features are unable to overcome some catastrophic flaws. The extraction of hand-crafted characteristics from music signals is a complicated procedure that necessitates researchers' knowledge of the musical domain. Furthermore, characteristics derived from one job may not be universal since they may perform poorly in other tasks.

Machine learning, particularly Convolutional Neural Networks (CNNs), has been effectively used in numerous picture classifications in recent years [11, 12]. Meanwhile, Sander et al. [13] show that, when compared to conventional pictures, spectrograms of music audio may also be used with CNNs to produce good results. In this situation, there is a rising trend toward using CNNs to build strong feature representations from spectrograms of music [14, 15]. Unlike previous approaches, CNNs offer an end-to-end training architecture that combines feature extraction and music categorization in a single stage. Furthermore, several publications based on CNN have demonstrated their superiority in the categorization of music genres.

The remainder of this document is arranged as follows. In Section 2, we look back on previous work on music genre categorization and evaluate its contribution as well as its limitations. The building of our suggested hybrid architecture CNN for music genre categorization and the dataset is described in Section 3. In Sections 4 and 5 we do a series of tests on a variety of datasets to prove the correctness of our suggested architecture CNN. Finally, in Section 6 results and Section 7, we conclude and outline some future work.

A Motivation of Research Work.

1. By effectively tackling the previously mentioned obstacles, it becomes possible to automatically classify music genres.
2. Introducing a diverse range of CNN as a means to achieve this goal.
3. Assessing the performance of the classifiers and suggesting the most suitable option for enhancing classifier accuracy.

B Purpose.

1. To develop a machine learning model for the classification of music into its genres based on various features.
2. To validate the accuracies with the existing model to developed model for classifying its genre correctly.

C Academic research value.

As a result, we can effectively annotate and index the material in order to get access to it. The difficulty in dissecting and categorizing sound signals arises from non-stationarity and intermittence within the sound signal can eliminated. We can choose the optimal features in the acoustic stream. The narrator identification, gender determination, musical genre categorization, natural sound classification, and other areas of the usage of sound categorization and reclamation framework are included. It helps to reduce the piracy.

2 Research related work

For identifying and characterizing a huge volume of music, music genre categorization is a well-explored field in Music Information Retrieval [1]. Various studies show that collecting representative characteristics from music signals may significantly enhance classification performance. As a result, the majority of current research focuses on extracting strong characteristics to characterize music to increase music genre categorization accuracy. CNNs have gotten a lot of interest in the field of music genre classification. [18]. CNNs have great powers to represent varied music with higher-level properties by training an end-to-end architecture. Furthermore, CNNs need less technical effort and a prior understanding of a certain topic. The fluctuations of musical patterns with a given transformation, such as the Fast Fourier Transform (FFT) and Mel-frequency Campestral Coefficient (MFCC), are comparable to pictures that operate well with CNNs in image classifications [12], according to Li et al. [14]. Furthermore, they demonstrate that CNNs are viable alternatives for automatically extracting musical pattern characteristics. Zhang et al. [15] presented two networks to improve music genre categorization performance using CNNs. In one of the networks, max- and average-pooling are used in tandem over the full-time axis to provide extra statistical information to the subsequent layers. They use shortcut connections inspired by residual learning [19] in another network to improve the accuracy of increased depth.

In contrast to prior results based on the GTZAN [1] dataset, the performances of two CNNs are both shown to be improved. Musical patterns, on the other hand, have some temporal links that are important for music genre classifications but will be dropped in CNNs, as described in the preceding section. Choi et al. [20] innovate a hybrid model known as the convolutional recurrent neural network (CRNN. However, this hybrid approach has limitations that impede music categorization performance. Despite the fact that CRNN uses RNNs as the temporal summarizer, it can only summarize temporal data from CNN output.

Based on signal processing and a CNN model named MusicRecNet, Elbir et al. [21] implemented Music genre classification It is capable of checking the plagiarism of songs

[25]. GTZAN dataset was used for this work. The images generated served as the input and were applied to the MusicRecNet for training. The model was used for genre classification [25]. The main performance metric was accuracy.

Gelowitz & Pelchat [22] which got a test accuracy of 67%. with Rectified Linear Unit (ReLu) activation function used with CNN.

Classification of the Million Song Dataset (MSD) is done by Vishnupriya and Meenakshi [23, 25] into different genres. Python librosa package used to carry feature vector extraction. The package is specifically used for audio analysis [25]. The extracted feature vector was the Mel-frequency Coefficient (MFCC) [25].

3 Proposed architecture and data used

3.1 Proposed architecture

With much focus on spectrogram features with 2D Convolutional Neural Networks as the classifier over the years for music genre classification, this paper focuses on content-based features and 1D Convolutional Neural Network for music genre classification to achieve an excellent accuracy result.

We carefully build the hybrid model, which includes paralleling CNN blocks, to maintain both spatial properties and temporal frame ordering of original music signals.

Figure 1 be depicted concerning the final stage The method is illustrate in Timbre Features as When the input is an audio clip STFT, the proposed network model manifests Audio-1DCNN pattern. When the input is MEL-spectrogram, the network model manifests MEL-2DCNN pattern. When the input is MFCC, it manifests MFCC-3DCNN.

By merging these three methods, you can capture a wide range of characteristics from the audio signal. The STFT provides a time–frequency representation, the Mel Spectrogram provides a perceptually weighted time–frequency representation, and the MFCCs provide a compact, decorrelated representation of the spectral envelope. This combination can be particularly useful for tasks like speech recognition and music analysis, where it's important to capture both the temporal and spectral characteristics of the audio signal.

Figure 1 describes our proposed architecture and Fig. 2 describes the conventional network architecture of CNN.

3.2 Data set

From the Marsyas site, we used the GTZAN dataset as the dataset of input audio signals. In which each genre has 100 music files. There is a total of ten genres, resulting in a total of 1000 music files. Blues, Classical, Country, Disco, Hip-hop, Jazz, Metal, Pop, Reggae, and Rock are among the ten genres. It includes a 30-s audio clip with a sampling rate of 22050 Hz and a 16-bit resolution [9].

Indian Music Genres:180 Music Signals (Genres Like Asavari, Bageshree, Bhairavi, Bhoopali, Darbari, Dkanada, Malkauns, Sarang and Yaman are also tested).

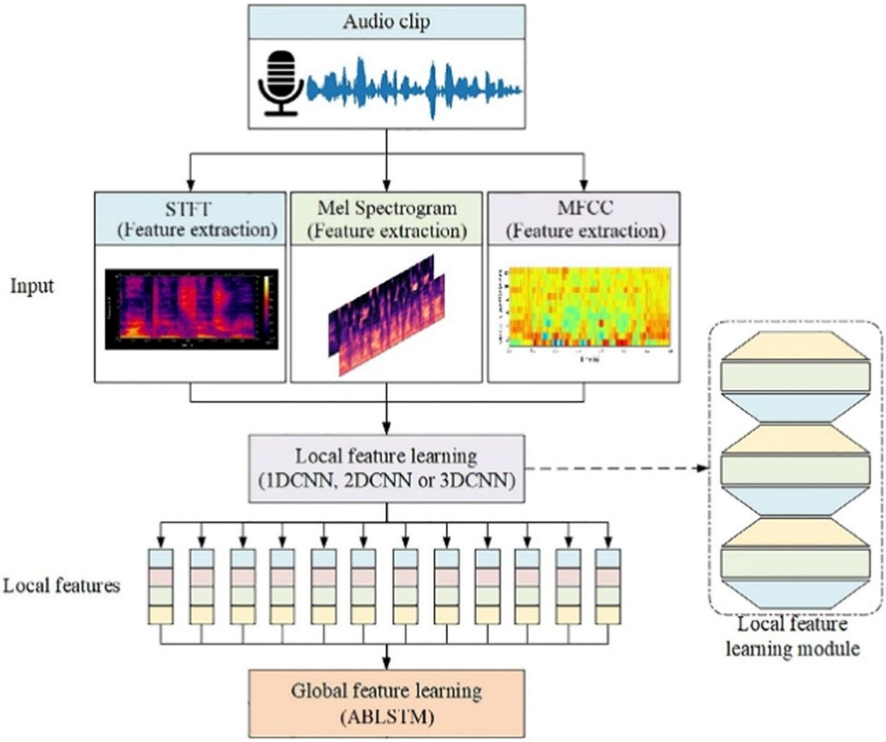


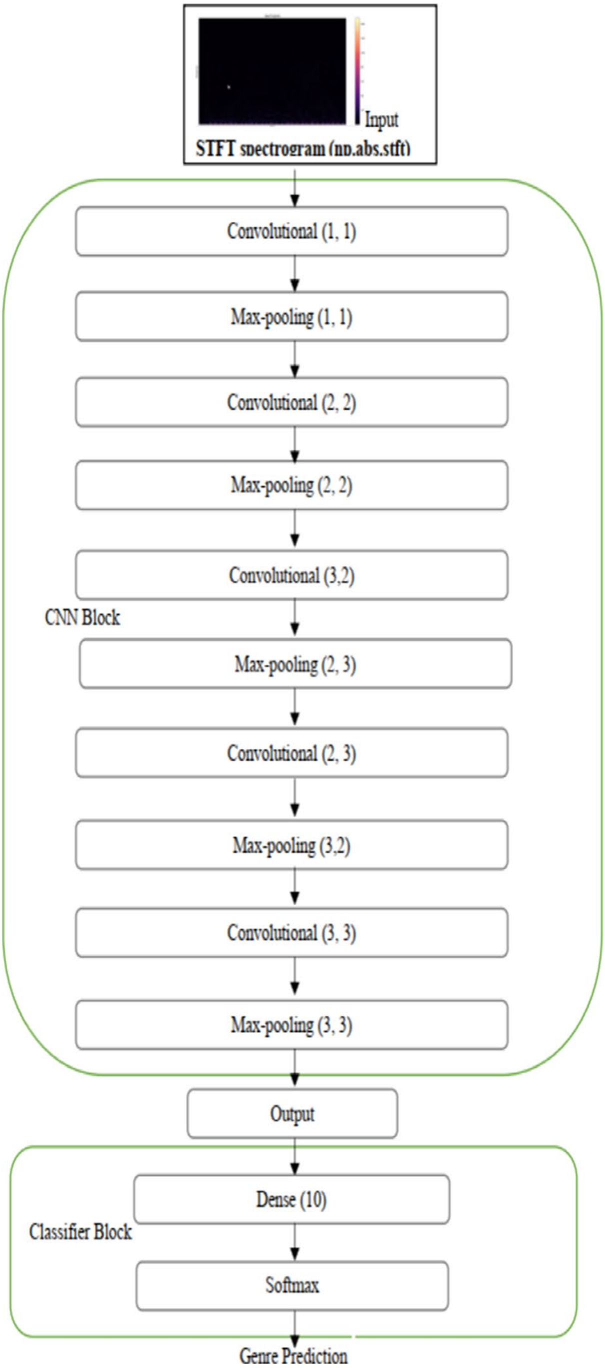
Fig. 1 Proposed architecture

4 Research methodology

Figure 3 shows the developed methodology framework for this study, which contains two major phases: feature extraction and classification. These two phases can further be broken down into six stages:

> ***Stage One:*** This stage entailed gathering the audio files from which the features that would be used to create our dataset were extracted.
> ***Stage Two:*** Thirty seconds of the beginning, middle, and end of each of the songs were extracted from the audio files
> gathered in stage one.
> ***Stage Three:*** The low-level features were extracted from the
> audio files gathered in stage two and were stored in a CSV file.
> ***Stage Four:*** Preprocessing of the dataset was achieved at this stage.
> ***Stage Five:*** The training and testing of the model with the dataset was achieved at this stage.
> ***Stage Six:*** The model's accuracy was determined at this point
> to demonstrate how well the model identified the songs properly.

Fig. 2 The network architecture of CNN



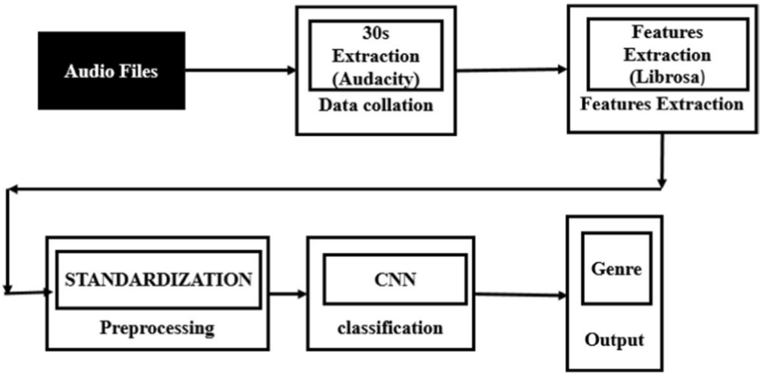


Fig. 3 Methodology framework

4.1 Features extraction

Feature extraction [15] is the process of creating a compact or succinct numerical representation that is used to characterize a segment of audio. The goal of feature extraction is to portray a piece of music or a portion of music in a concise and descriptive manner [16]. Using the relevant characteristics retrieved, suitable machine learning or deep learning algorithms are utilized to categorize the audio signals into the desired outputs (such as genre). The most important technique in pattern recognition systems is feature extraction [17].

These qualities have little or no meaning for people, but computer systems make good use of them [20]. The estimated characteristics are derived for each short-time frame of sound using the short-time Fourier transform (STFT) [15]. Low-level features, time and frequency domain features, and short-term features are all terms used to describe these features. The absolute or squared values of the amplitudes are added to determine the time domain characteristics [21]. The frequency domain characteristics are based on a preliminary Fast Fourier Transformation (FFT) that is used to produce a frequency domain representation of the audio signal. These methods are based on the assumption that the signal is periodic [21, 22]. To represent timbral texture, 14 standard features proposed for music-speech discrimination and speech recognition are used.

The timbre features extracted for this project are:

- ***Chroma STFT:*** The musical octave has 12 different semitones in audio [22, 23]. Each of the 12 semitones or pitches has chroma characteristics that correspond to the overall energy of the signal. The pitch class is determined by the chroma feature. After that, the chroma vectors are combined across the frames to produce a representative mean and standard deviation [10].

4.2 Timbre features

4.2.1 Merging STFT, Mel spectrogram, and MFCC features

The Short-Time Fourier Transform (STFT), Mel Spectrogram, and Mel-Frequency Cepstral Coefficients (MFCC) are all powerful tools for audio analysis. They each provide a different representation of the audio signal, capturing different aspects of its characteristics. Here's how you can merge them for a comprehensive audio analysis.

STFT: Start by applying the STFT to your audio signal. This will convert the time-domain signal into a frequency-domain representation, providing a 2D representation where the x-axis represents time, the y-axis represents frequency, and the intensity of each point represents the amplitude of a particular frequency at a particular time.

Mel Spectrogram: Next, convert the STFT into a Mel Spectrogram. This involves mapping the frequencies obtained from the STFT onto the Mel scale, which approximates the human ear's response to different frequencies. The result is a 2D representation similar to the STFT, but with the frequency axis warped according to the Mel scale.

MFCC: Finally, compute the MFCCs from the Mel Spectrogram. This involves applying a logarithm to the Mel Spectrogram to approximate the human perception of loudness, and then applying the Discrete Cosine Transform (DCT) to decorrelate the Mel Spectrogram coefficients. The result is a set of coefficients that capture the spectral characteristics of the audio signal.

4.2.2 Performance metrics

The evaluation metrics for the model built are:

Accuracy, Precision, Recall, and F1-score.

- Accuracy: The percentage of the properly anticipated result to the entire sum of all forecasts is calculated.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

- Precision: This statistic indicates whether the model was correct or incorrect when it predicted positive.

$$\text{Precision} = \frac{\text{Number of true positives}}{\text{Number of positive predictions}}$$

- Recall: This indicator indicates the number of positives recognized by the model out of all potential positives.

$$\text{Recall} = \frac{\text{Number of true positives}}{\text{Number of actual positives}}$$

$$\text{F1Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

5 Performance evaluation

The GTZAN dataset [1] and the CNN Audio Description Contest dataset [24] were utilized in this work to assess performance. We will present brief summaries of these two datasets in this part, followed by the experimental findings.

5.1 The GTZAN dataset

The GTZAN dataset includes 1000 audio songs from ten different musical genres: Blues, Classical, Country, Disco, Hip Hop, Jazz, Metal, Popular, Reggae, and Rock. There are exactly 100 tracks in each music genre, all of which were recorded at 22,050 Hz in mono 16-bit WAV format. To assess classification performance on the GTZAN dataset, we utilized tenfold cross-validation, as in previous studies. 900 songs were chosen at random as the training set for each fold, while the remaining 100 tracks were utilized for testing. By averaging the classification results of these 10 folds, the performance will be calculated.

5.2 The CNN dataset

The CNN dataset contains 1458 music recordings, with 729 being utilized for training and the rest being used for testing. Blues, Classical, Country, Disco, Hip Hop, Jazz, Metal, Popular, Reggae, and Rock music tracks are divided into ten categories. 44.1 kHz, 16 bits per sample, stereo MP3 format is the audio file format. We chose 1/10 of the 729 music tracks used for training as the validation set at random in order to find the optimal parameter set for evaluating the testing data set's performance.

5.3 Data-preprocessing

A sample rate of 22,050 Hz was used to read the audio. The audio was then separated into 3-s audio snippets for a length of 30 s. The issue is that when the audio is divided, the computer assumes that each clip is unrelated to the others and is thus autonomous. To circumvent this, 50 percent of the previous data duration is collected and appended to the following 50 percent of data duration. This will aid in comprehending the computer and ensuring that the first 50 percent duration of data is taken from the prior audio clip, and from there, the following 50 percent duration of data is taken from that data.

5.4 Feature extraction used

Each clip has performed different feature extraction to see the analysis, which one has been performed better.

1. Short-Term Fourier Transform (STFT)

It employed the short-term Fourier transform cites {s1} since there was no fundamental tradeoff between time and frequency. As the name implies, "short term" indicates that the signal is divided into small blocks with a set duration and then the Fourier transform is applied to each block. Framing is the process of moving the slide to form each block of a signal. This is the frequency change with signal over time that has been calculated. Each block is multiplied with a windowing function before calculating to improve the Fourier Transform's ability to extract spectral data from signals. It made use of Hann windows, which have a bell-shaped curving form.

The outcome of determining the Fourier transform of each signal is a complex number. It evaluates the absolute value of the complex number to convert it to a real number. The resulting is termed frequency representation after applying (Fast Fourier

Transform) FFT to each block. This signal's whole frequency representation with time will now be considered its features, which is also known as a spectrogram. As a result, it used the Short Term Fourier Transform (STFT) on each clip, with the FFT window size (frame size) set to 1024, the hop length (frame increment) set to 512, and the windowing function set to Hann. Each clip has 513 frequency bins and 129 frames in time (513, 129) dimensions (Fig. 4).

2. Mel-Frequency Cepstrum Coefficient (MFCC)

It's an expansion of the Mel spectrogram. After compressing the frequency, the Mel-Frequency Cepstral Coefficient [19] is another approach to describe the spectrum of the audio sample. We take the log of the power from the mel frequencies generated in the Mel spectrogram and then choose the first 13–20 coefficient after Discrete Cosine Transformation (DCT). As the number of coefficients grows, it represents incremental changes in predicted energy, and as a result, they have less data. Of course, the majority of the information was lost, which is why it was stated that this approach was utilized for audio compression. Then, instead of using the inverse fast Fourier transform (FFT), use DCT because it is expected to perform similarly to FFT and is also simple to compute and implement. Figs. 5 and 6 shows the evaluation procedure of MFCC.

5.5 Learning algorithm

Training data, validation data, and test data are the three portions of the data. These data are divided into three categories: 80 percent training data, 10% validation data, and 10% testing data. Clips may be shuffled without causing problems or losing series since each clip retains 50% of its prior data, allowing them to be related to other clips created during the data pretreatment step. After that, the data is put into a Hybrid Convolution Neural Network, which reshapes it. Once an epoch is completed, it evaluates the performance of validation data to assess how well the unknown data can be generalized after

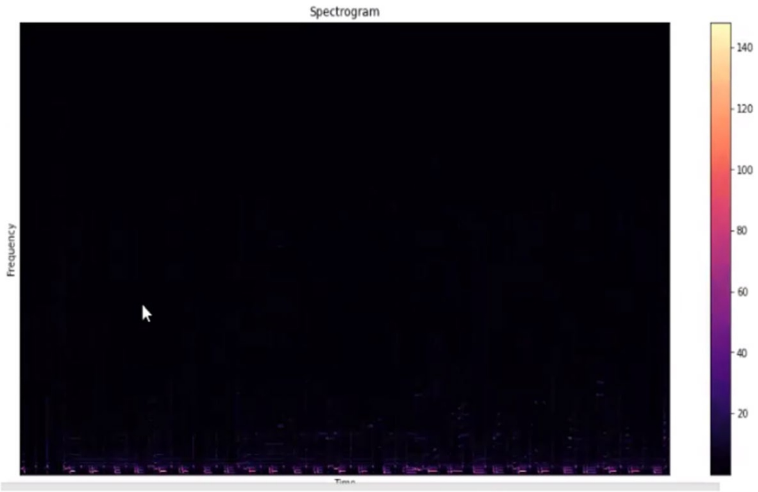


Fig. 4 STFT Spectrum for a blue class audio with 3 s duration

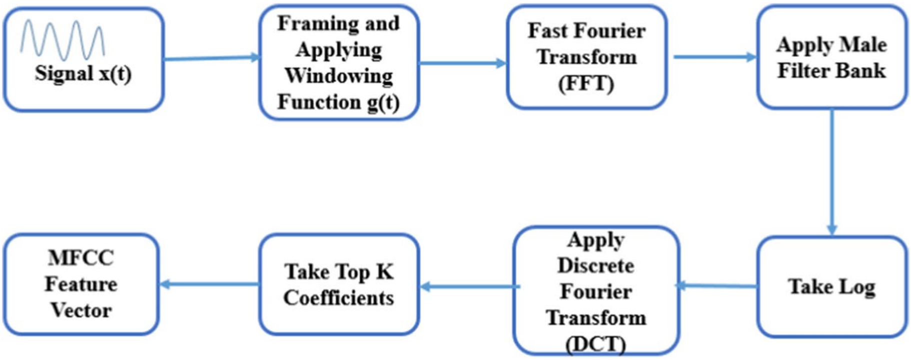

Signal x(t)
Framing and Applying Windowing Function g(t)
Fast Fourier Transform (FFT)
Apply Male Filter Bank
Take Log
Apply Discrete Fourier Transform (DCT)
Take Top K Coefficients
MFCC Feature Vector

Fig. 5 Procedure for evaluating MFCC

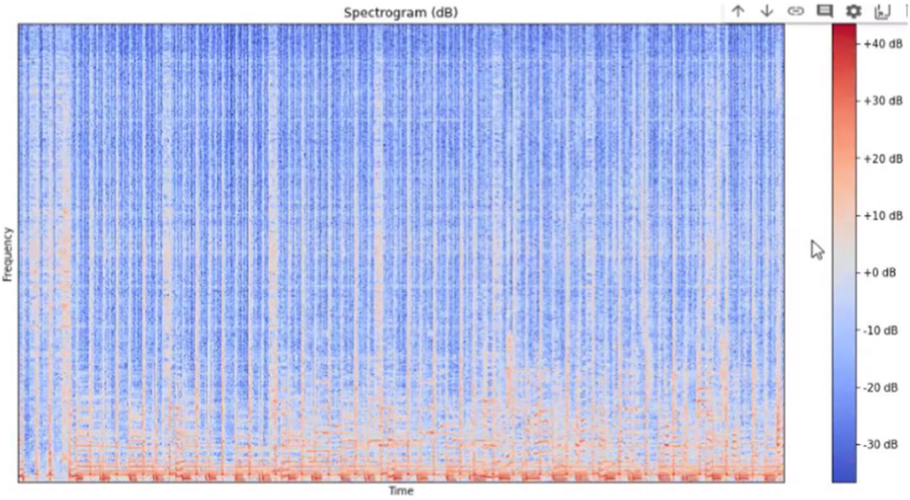


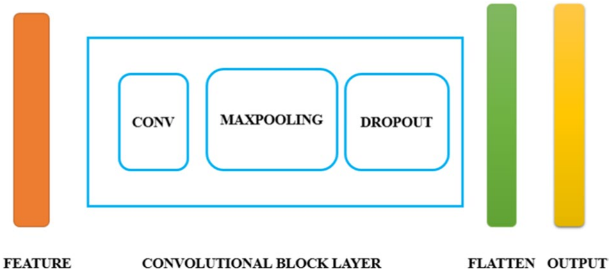
Fig. 6 MFCC for a blue class audio with 3 s duration

learning from the training data. Following the completion of all trained networks, the test data performance is evaluated after requesting training and validation data.

The use of Convolution Neural Networks (CNN) has shown promising and improved picture categorization and recognition results. So, all features are treated as image features to recognize the pattern using the CNN model, which will give better performance results, CNN is also considered as a self-feature learning, i.e., it gets various features with different convolution kernel filters while performing convolution to those images. Because of the restricted resources in our system, VGG16 [12], which comprises 16 layers of CNN, has only been trained on Mel-Spectrogram feature extraction data. In several areas, this network has outscored CNN. In the next step, you'll view the results in tabular format.

The VGG 16 model [12] is characterized by its simplicity, using small 3×3 convolution filters throughout the architecture, and it includes 16 layers that have weights. It's widely used in the computer vision community for a variety of tasks due to its excellent performance.

Fig. 7 Proposed CNN architecture



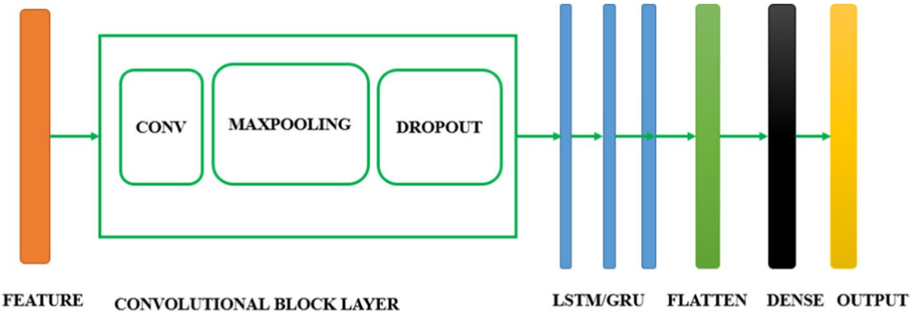


Fig. 8 Proposed CNN with LSTM (or GRU) architecture

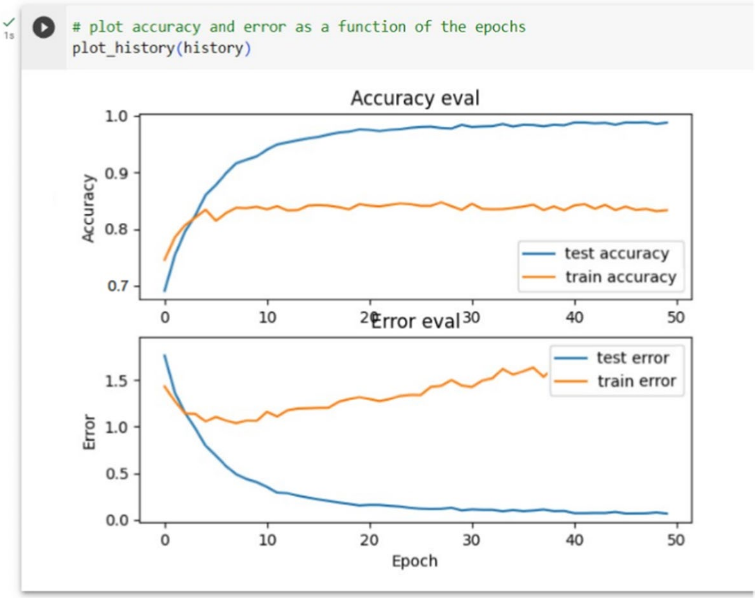
Table 1 Output shape at different layers of CNN

Layer (type)	Output Shape	Param #
conv2d_3 (Conv2D)	(None, 128, 11, 32)	320
max_pooling2d_3 (MaxPooling2D)	(None, 64, 6, 32)	0
batch_normalization_3 (BatchNormalization)	(None, 64, 6, 32)	128
conv2d_4 (Conv2D)	(None, 62, 4, 32)	9248
max_pooling2d_4 (MaxPooling2D)	(None, 31, 2, 32)	0
batch_normalization)_4 (BatchNormalization)	(None, 31, 2, 32)	128
conv2d_5 (Conv2D)	(None, 30, 1, 32)	4128
max_pooling2d_5 (MaxPooling2D)	(None, 15, 1, 32)	0
dropout_l (Dropout)	(None, 64)	0
dense_3 (Dense)	(None, 10)	650

From it we develop five Convolution Block layers for the suggested CNN with LSTM (or GRU) architecture. After that, we applied three LSTM (or GRU) layers, each with two 128 LSTM (or GRU) units and one 64 LSTM(or GRU) unit, flattening them into a 1D array and applying one dense layer, and lastly, the output layer. (As shown in Figs. 7 and 8).

Table 2 Accuracy table of CNN

Time inference/step	Loss	Testing accuracy	Validation loss	Validation accuracy
321 ms/step	0.0471	0.9863	1.3917	0.8396
319 ms/step	0.0459	0.9855	1.4913	0.8262
321 ms/step	0.0529	0.9828	1.4521	0.8276
320 ms/step	0.0477	0.9860	1.4313	0.8402
318 ms/step	0.0527	0.9810	1.4281	0.8669
320 ms/step	0.0592	0.9820	1.3544	0.8636
320 ms/step	0.0504	0.9833	1.3661	0.8683
320 ms/step	0.0519	0.9823	1.4381	0.8522
318 ms/step	0.0539	0.9806	1.4043	0.8456
320 ms/step	0.0392	0.9861	1.4511	0.8576
330 ms/step	0.0399	0.9875	1.5154	0.8596
314 ms/step	0.0460	0.9830	1.5103	0.8482
325 ms/step	0.0409	0.9865	1.5336	0.8516
317 ms/step	0.0479	0.9831	1.4706	0.8436
319 ms/step	0.0438	0.9838	1.6224	0.8569



```
# plot accuracy and error as a function of the epochs
plot_history(history)
```

Fig. 9 Graph plot for CNN-trained network of accuracy and error as a function of the epochs

6 Results

Output Shape at different layers of CNN as shown in below Tables 1 and 2. This is the core building block of CNN. This layer's parameters consist of a set of learnable filters also called kernels. A convolutional layer simply transforms the input data in order to extract features from it. The convolution slides the kernel over the input data, a procedure referred to as the shift-compute procedure. This happens in two ways: Non casual convolution and casual convolution. In non-casual convolution, the output is dependent on the future input while in casual convolution, the output is not dependent on future inputs.

The dimensionality of a given mapping that is the number of parameters is reduced while the prominent features are being highlighted. Pooling is simply employed to reduce the dimension of the convolution output, which reduces in return the computational cost. This layer also helps to avoid overfitting. The max pooling technique is the most common technique and it works by selecting the maximum value in each patch of each feature map (Fig. 9).

7 Conclusion and future scope

The research used a dataset that included 10 distinct genres: Blues, Classical, Country, Disco, hip-hop, Jazz, Metal, Pop, Reggae, and Rock. This dataset was chosen to test the algorithm with genres since certain genres have been extensively tested all over the world, leaving genres unique to specific areas unexplored.

It may be stated that the content-based features retrieved from the songs are excellent qualities for classifying songs into their genre. In addition, based on the results of this research, Convolutional Neural Network (CNN) has been demonstrated to be an effective classifier for music genre categorization. As a result, this research investigates the use of Convolutional Neural Networks (CNN) for music genre categorization. A new dataset of 1,000 traditional songs from ten genres was used in this investigation. Because feature extraction is crucial to audio analysis, 10 low-level characteristics, also known as content-based features, were extracted from the songs in the dataset and utilized as input into the classifier. Our findings revealed that the system's accuracy level is 98.9%, with precision of 98.7%, recall of 98.5 percent, and f1 score of 97.5 percent.

In order to have a solid dataset, future research should include additional untested genres from around the world in the dataset. Because of their structure, some genres may perform differently with various classifiers. As a result, it's critical to conduct studies on all untested genres throughout the world. Additionally, alternative criteria other than spectrogram and content-based variables should be investigated for application in music genre classification.

Data availability Data sharing is not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Competing interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Tzanetakis G, Cook P (2002) Musical genre classification of audio signals. IEEE Trans Speech Audio Process 10(5):293–302
2. Taylor J, Meng A (2005) An investigation of feature models for music genre classification using the support vector classifier. In: 6th International Conference on Music Information Retrieval (ISMIR 2005), London, UK, pp 604–609
3. West K, Cox S (2005) Finding an optimal segmentation for audio genre classification. In: 6th International Conference on Music Information Retrieval (ISMIR 2005), London, UK, pp 680–685
4. Duda RO, Hart PE, Stork DG (2000) Pattern Classification, 2nd edn. Wiley-Interscience
5. Fu Z, Lu G, Ting K, Zhang D (2011) A survey of audio-based music classification and annotation. IEEE Trans Multimedia 13(2):303–319
6. Baniya B, Ghimire D, Lee J (2014) A novel approach of automatic music genre classification based on timbrai texture and rhythmic content features, in Advanced Communication Technology (ICACT), 2014 16th International Conference, pp 96–102, IEEE
7. Huang G, Zhu Q, Siew C (2006) Extreme learning machine: theory and applications. Neurocomputing 70(1):489–501
8. Breiman L (1996) Bagging predictors. Mach Learn 24(2):123–140
9. Arabi A, Lu G (2009) Enhanced polyphonic music genre classification using high level features, In Signal and Image Processing Applications (ICSIPA), 2009 IEEE International Conference on, pp 101–106, IEEE
10. Sarkar R, Saha S (2015) Music genre classification using emd and pitch based feature, in Advances in Pattern Recognition (ICAPR), 2015 Eighth International Conference on, pp 1–6, IEEE
11. Wei Y, Xia W, Lin M, Huang J, Ni B, Dong J, Zhao Y, Yan S (2016) Hcp: A exible cnn framework for multi-label image classification. IEEE Trans Pattern Anal Mach Intell 38(9):1901–1907
12. Ciresan D, Meier U, Masci J, Gambardella ML, Schmidhuber J (2011) Flexible, high performance convolutional neural networks for image classification, in IJCAI Proceedings-International Joint Conference on Artificial Intelligence, 22:1237, Barcelona, Spain
13. Dieleman S, Schrauwen B (2014) End-to-end learning for music audio, in Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference, pp 6964–6968, IEEE
14. Li T, Chan A, Chun A (2010) Automatic musical pattern feature extraction using convolutional neural network. In: International Multi Conference of Engineers and Computer Scientists (IMECS 2010), vol 1, pp 546–550
15. Zhang W, Lei V, Xu X, Xing X (2016) Improved music genre classification with convolutional neural networks, in INTERSPEECH, pp 3304–3308
16. Elman J (1990) Finding structure in time. Cogn Sci 14(2):179–211
17. Pons J, Lidy T, Serra X (2016) Experimenting with musically motivated convolutional neural networks, in Content-Based Multimedia Indexing (CBMI), 2016 14th International Workshop on, pp 1–6, IEEE
18. Lawrence S, Giles C, Tsoi A, Back A (1997) Face recognition: A convolutional neural-network approach. IEEE Trans Neural Networks 8(1):98–113
19. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition, in Proceedings of the IEEE conference on computer vision and pattern recognition, pp 770–778
20. Choi K, Fazekas G, Sandler M, Cho K (2016) Convolutional recurrent neural networks for music classification, arXiv preprint arXiv:1609.04243
21. Elbir A, Aydin N (2020) Music genre classification and music recommendation by using deep learning 2020. Electron Lett 56(12):627–629
22. Pelchat N, Gelowitz C (2019) Neural network music genre classification. In: 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE), IEEE, pp 170–173
23. Vishnupriya S, Meenakshi K (2018) Automatic music genre classification using convolution neural network. 2018 International Conference on Computer Communication and Informatics (ICCCI - 2017), Coimbatore, INDIA.IEEE
24. Cano P, Gómez E, Gouyon F, Herrera P, Koppenberger M, Ong B, Serra X, Streich S, Wack N (2006) ISMIR 2004 audio description contest. Music Technology Group of the University at Pompeu Fabra, Technical Report

25. Falola P, Alabi E, Ogunajo F, Fasae O (2022) Music genre classification using machine and deep learning techniques: A review. ResearchJet J Anal Inventions- RJAI 3(03):35–50

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.