IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Music Genre Classification Using CNN

Jane Crystal Rodrigues
Computer Engineering Department
Goa College of Engineering
Farmagudi, Goa

Manisha Naik Gaonkar
Computer Engineering Department
Goa College of Engineering
Farmagudi, Goa

***Abstract*— Music genre labels are useful to organize songs, albums, and artists into broader groups that share similar musical characteristics. A fundamental component of developing a powerful recommendation system is the classification of music genres. Making the selection of songs easier and quicker is the goal of automating the music classification. If one must manually do this task, it takes a lot of time and is challenging. In this research, we examine the handling of sound files in Python, compute sound and audio attributes from them, apply Deep Learning Algorithm such as – CNN and evaluate the outcomes.**

Keywords-- Deep Learning,Music Genre Classification, Python,CNN.

I. Introduction

Since music serves as a kind of entertainment, a method of bringing people together, and a means of fostering communities, music has played a significant part in society throughout history. The genre of a piece of music is a popular way to differentiate it [3].

The genre of a piece of music is a popular way to differentiate music. Songs with similar qualities, such as instrumentation, harmonic content, and rhythmic structure, are frequently used to define musical genres.

Music genre classification is an Automatic Music Classification (AMC) problem in the area of Automatic Music Retrieval (AMR) [3].

Because of the subjective character of music, it has been difficult to categorise the many genres because their lines begin to bleed into one another.Few genres of music are : Blues, Classical, Country, Pop.

There are two major challenges with this problem:

1. Musical genres are loosely defined. So much so that people often argue over the genre of a song.
2. It is a nontrivial task to extract differentiating features from audio data that could be fed into a model.

Working with audio data has been a deep learning issue that has received relatively little attention. Benchmarks for the most recent ground breaking deep learning research are frequently determined by how well it performs on text and image data. Additionally, models that use text and graphics are where deep learning has made the biggest strides. Speech and audio, which are equally significant types of data, are frequently ignored in this.

In recent years, the advancement of machine and deep learning algorithms have paved the way for Neural Network Models such as CNN – Convolution Neural Network to be among the highly sought after model when it comes to image classification, showing high accuracy compared to traditional learning algorithms in the same field.

Motivated by this, we propose a CNN – based architecture to classify music of various genres by converting audio files into images which represent the features present in the audio signals and then applying Transfer learning and analyse how training can affect the model's accuracy.

II. Literature Review

In [1], they outline a method for categorizing music genres using temporal feature learning from audio using deep neural networks. Two general categories can be made for these auditory features:

1) 1) Spectral features (frequency-based features), such as fundamental frequency and frequency components, are acquired by transforming the time-based signal into the frequency domain using the Fourier Transform. To determine the notes, pitch, rhythm, and melody, use these features.

2) Temporal features – such as the energy of the signal, the rate of zero crossing, and the maximum amplitude, are time domain features that are simple to extract and have a clear physical meaning.

The goal of this research is to use a deep neural network (DNN) to learn TFs from a low-level representation. Depending on the dataset partitioning techniques, experiments with genre classification demonstrate that the proposed temporal features produced performance equivalent to or greater than that of the spectral features.

In [2], this study compared five conventional off-the-shelf classifiers against a deep-learning convolutional neural network technique for classifying musical genres.

Logistic Regression (LR) : accuracy of 83%

K-Nearest Neighbours (KNN) : accuracy of 72.8%

Support Vector Machines (SVM):accuracies of 75.4%

Random Forest (RF): accuracy of 75.7%

Simple Multilayer Perceptron (MLP):accuracy of 75.2%

Feature selection included both spectrograms and content-based features.it worked on the Dataset : GTZAN

In [3], In this study, the pre-trained network (VGG19) parameters for the image classification task are adjusted via transfer learning. Performance research demonstrates that the optimised VGG19 architecture surpasses the other CNN and hybrid learning approach for the image classification problem, when compared to AlexNet and VGG16.

Dataset :

GHIM10K : 20 classes, 500 images/class

CalTech256 : 256 categories, each class having min 8o img

In[4], They give a music dataset with ten different genres in this article. The system's convolution neural network is trained and classified using a Deep Learning methodology. For sound samples, the Mel Frequency Cepstral Coefficient (MFCC) is employed as a feature vector. The findings indicate that our system's accuracy level is roughly 76%.

Dataset : Million Song Dataset (MSD) : consists of audio tracks totally of 280 GB.

In[5], This study offers a novel solution to the automatic music genre classification issue. According to space and temporal decomposition approaches, the suggested approach makes use of various feature vectors and a pattern recognition ensemble approach. We solve the multi-class problem of music genre classification using a set of binary classifiers, whose outputs are combined to give the final music genre label (space decomposition). Additionally, music pieces are divided into time segments based on the beginning, middle, and end of the original musical signal (time-decomposition). According to a combination technique, the final classification is determined from the collection of individual data.

The use of Naive-Bayes, Decision Trees, k Nearest Neighbors, Support Vector Machines, and MultiLayer Perceptron Neural Nets is a classic example of machine learning. On a cutting-edge dataset called the Latin Music Database, experiments were conducted. Experimental findings demonstrate that the proposed ensemble technique yields superior outcomes.

In[6], This shows how to create deep belief neural networks using the restricted Boltzmann machine method. The intention was to compare it to the performance of the standard neural networks by using it to complete a multi-class classification task of labelling musical genres.

In[7], The range to reach a conclusion may be seconds, although many distinct short-time features with time windows between 10 and 30 ms in size have been offered for genre classification. In order to classify music genres, this research examines various techniques for feature integration and late information fusion. The proposed AR model appears to perform better than the often employed mean-variance features.

In[8], The outcomes of using a feature selection approach to classify music genres automatically are shown in this study.

According to time and space decomposition methodologies, the classification system is based on the usage of several feature vectors and an ensemble approach. The objective of this work is to apply a feature selection procedure based on Genetic Algorithms (GA) to multiple feature vectors extracted from various parts of the music signal, analyse the discriminative power of the features according to the part of the music signal from which they were extracted, and determine how the feature selection affects the classification of music genres. However, for MLP and SVM, the feature selection approach does not boost classification accuracy. The results obtained with the feature selection procedure demonstrate that this method is effective for J48, k-NN, and Nave-Bayes classifiers.

In[9], In this study, we concentrate on obtaining representative features from an unique deep neural network model. Time, amplitude, phase, and frequency are distinguishing acoustic characteristics that are used in music recommendation algorithms. The proposed MusicRecNet enables a recommendation engine to suggest new music to a listener based on that listener's musical preferences. Three layers are the intended structure of MusicRecNet. Each layer comprises of a dropout operation, a two-dimensional maximum pooling operation, an activation function (rectified linear unit), and a two-dimensional convolution. The categorization accuracy significantly improved, going from 81% to over 90%.

In [10], By learning from training data, support vector machines are used to categorize music into vocal music and pure music. In order to describe the musical content, a variety of features are extracted. A clustering method is used to organize the music content based on determined features. The clustering results and domain information pertaining to pure and vocal music are then used to construct a music summary.

In[11], In this study, CNN and SVM are utilised as a baseline for comparing the recognition effects for the issues of picture recognition and voice emotion recognition. The best accuracy for picture recognition using SVM is 94.17 percent. However, 95.5 percent of the time CNN is accurate. The accuracy of CNN's voice recognition software is 97.6%, which compares to a baseline model's accuracy of 55.5%. This study demonstrates how well CNN extracts features and how strong its modelling for two-dimensional data is.

III. Analysis of Model

A. Deep Learning

Deep learning is a machine learning method that instructs computers to learn by doing what comes naturally to people. A computer model learns to carry out categorization tasks directly from images, text, or sound using deep learning. Modern precision can be attained by deep learning models, sometimes even outperforming human ability. A sizable collection of labelled data and multi-layered neural network architectures are used to train models.

Deep learning models are sometimes referred to as deep neural networks because the majority of deep learning techniques use neural network topologies.

The number of hidden layers in the neural network is typically indicated by the term "deep."

Convolutional neural networks are among the most often used varieties of deep neural networks (CNN or ConvNet). A CNN uses 2D convolutional layers and combines learnt features with input data, making it an excellent architecture for processing 2D data, including images.

B. CNN - Convolution Neural Network

A feed-forward neural network with biological inspiration is called a convolution neural network (CNN). The most advanced neural network architecture for image classification right now is CNN. Neurons in the CNN have biases and weights that can be learned[12].

Convolutional layers, pooling layers, fully linked layers, and normalisation layers (ReLU). are frequently seen in a CNN's hidden layers For more sophisticated models, more layers can be used[1]. The fundamental component of how data and weights are represented—including edges, bright spots, dark spots, forms, etc.—is matrix vector multiplication. The following set of layers will include recognisable elements from the image, such as the eyes, nose, and mouth.

The next layer includes of features that resemble real faces, or objects and shapes that the network can use to characterise a human face. By matching features rather than the complete image, CNN breaks down the classification process into smaller parts.

C. Layers in CNN

1) Convolution Layer

Each pixel in a patch is multiplied by the associated feature pixel one at a time. Once this is done, all the values are added up, and the result is divided by the entire number of pixels in the feature space[3].

The feature patch is then updated with the feature's final value. This procedure is repeated for the remaining feature patches, followed by applying this convolutional filter again and trying every possible match – this process is called convolution.

2) Pooling Layer

Pooling is used to reduce the spatial volume of the image data after convolution. The most well-liked pooling technique is called Max Pooling, and it deletes the items with lower values while restoring those with higher values. The window size (often 2x2/3x3 pixels) and the stride must also be supplied in order to pool a picture (e.g. usually 2 pixels). The greatest value for each window is recorded while the window is successively filtered across the image.

3) Normalization/ReLU Layer

Involves setting the filtered image's entire range of negative values to 0. Then, this process is carried out once more on each filtered image. The model's non-linear features are enhanced by the ReLU layer.

4) Fully Stacked Layer

The output of one layer becomes the input of the subsequent layer after the CNN stacks the layers (convolution, pooling). Deep stacking could happen as a result of continuous layering. The final layer in the CNN design is the fully connected layer, often known as the classifier. Each value contained within this layer is given the opportunity to vote on how the images are categorised. Fully connected layers are typically stacked on top of one another, and each intermediary layer casts a vote on a category that is "hidden."

5) Activation Function

The activation function's objective is to add non-linearity to a neuron's output. Without an activation function, a neural network is essentially just a linear regression model. The activation function transforms the input in a non-linear way, enabling it to learn and carry out more difficult tasks.

Egs : tanh(), sigmoid(), softmax()

IV. PROPOSED DESIGN

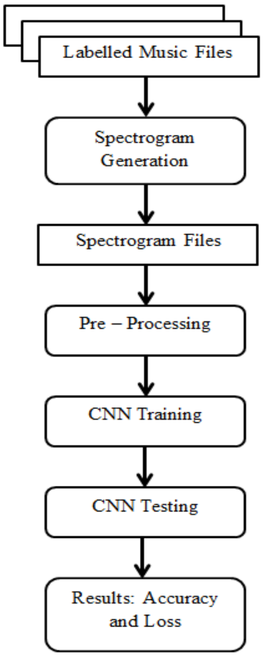


Fig. 1. Overview of Proposed methodology for the genre classification

A. Spectrogram Generation

A spectrogram is a graphic representation of the frequencies of a spectrum signal as they change over time. Every audio file is converted into a spectrogram using the librosa package.

How much crucial information is kept when the audio form is lost depends on how the audio data is converted into numeric or vector form. For instance, even the finest machine learning models would struggle to identify the genre and categorise the sample if a data format was unable to accurately capture the volume and tempo of a rock song.

From the several methods to represent audio data, some are as follows : MFCC, Spectrograms, Tempo, Wavelets etc.

We need to treat our data as image data since we are dealing with a CNN model, so we convert them into spectrograms using the librosa library.

B. Spectrogram Pre processing

After generating spectrograms we apply spectrogram image preprocessing steps to generate training and testing data. In this step we are made ascertain that all the images are of the same size, ie, they are resized to make the image suitable to be sent as the input to the CNN model.

C. CNN model training

After preparing the data, we develop our first deep learning model using the basic CNN model. With the necessary input and output units, we build a convolution neural network model.

We use only spectrogram data for the training and testing.

V. Implemenation

A. Dataset Overview

We will use GITZAN dataset, which contains 1000 music files and has the following properties :

- Dataset has ten types of genres with uniform distribution.
- Dataset has the following genres: blues, classical, country, disco, hiphop, jazz, reggae, rock, metal, and pop.
- Each music file is 30 seconds long.
- Each audio file is represented by 100 tracks.
- The tracks are all 22050 Hz monophonic 16-bit audio files in .au format.

Thus, with the GTZAN dataset, we can perform music genre classification using CNN. A challenge with this dataset is that the restricted number of samples might not be enough to get great results.

We have considered two versions of this same dataset as follows :

1) Setting A :

- Dataset including 1000 spectrogram images.
- Duration of each music sample 30 seconds.
- Dataset has ten types of genres with uniform distribution.
- Dataset has the following genres: blues, classical, country, disco, hiphop, jazz, reggae, rock, metal, and pop.

2) Setting B :

Since the amount of data present is very less to effectively train a CNN we have created more data from this data, by dividing each song into 10 segments each having duration of 3 secs each . This allows us to increase the amount of data in order to train the CNN model more effectively. The following image shows us the spectrograms generated from the 1000 original images. While generating this dataset, there was an issue generating the divided songs under the jazz genre so we have excluded that genre for this experiment. So the total spectrogram images generated are 9000 images.

- Dataset including 9000 spectrogram images.
- Duration of each music sample 3 seconds.
- Dataset has nine types of genres with uniform distribution.
- Dataset has the following genres: blues, classical, country, disco, hiphop, reggae, rock, metal, and pop.

B. Spectrogram Resizing/Preprocessing

The spectrogram generated in the previous step are resized to various sizes to perform the training and before passing it to the CNN model.

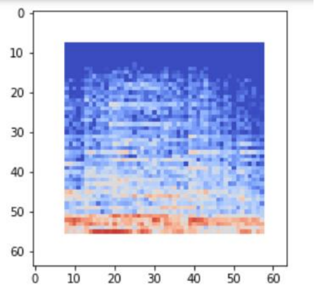


Fig. 2. Spectrogram resized to size 64 X 64 X 3

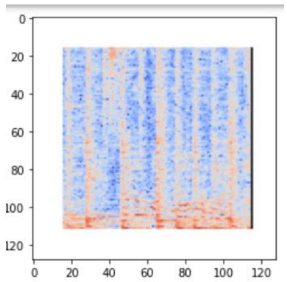


Fig. 3. Spectrogram resized to size 128 X 128 X 3

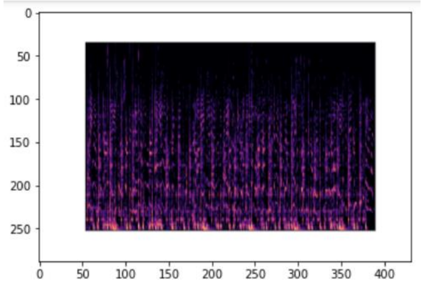


Fig. 4. Spectrogram resized to size 288 X 432 X 3

C. CNN Model Building :

1) Input Layer

Contains information in the form of an image which is the spectrogram in the following case, which is displayed as a three dimensional matrix, according to the image size.

2) Convolution Layer

As features from the spectrogram are extracted in this layer for the model to learn, it is sometimes referred to as the feature extraction layer. Convolutional layers at low levels extract shallow characteristics (such as edges, lines, and corners). Through the input of low-level features, the high-level convolutional layer further learns abstract features.

On top of the provided spectrogram is applied a matrix filter of a specific size. The filter and overlapping image elements are multiplied element by element for each spectrogram element, and then the values are added to produce the final convoluted value.

In this research we have used 3 convolution layers as can be seen in Fig 5, which shows us the architecture of the CNN model used.

3) Pooling layer

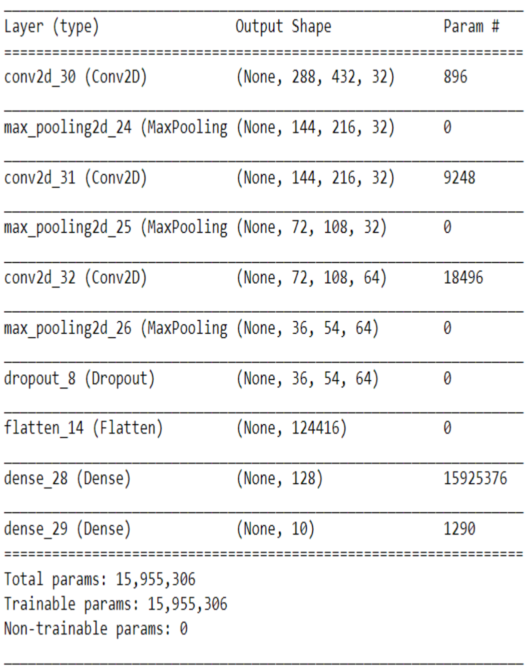
As already explained, in order to reduce the processing load on the system due to the large number of features extracted which are available through the data, we use Max Pooling techniques, keeping the most values information for each stride and deleting the rest.

In the model architecture, each convolution layer is followed by a max pooling layer.

4) Fully connected(FC) layer or Output Layer

In our problem, for the dataset distribution of Setting A we have 10 output classes, while for Setting B we have 9 classes. The corresponding class numbers are entered in the model architecture, which can we seen in the last Dense layer.

We use the following model to train the spectrograms generated of different sizes:



```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_30 (Conv2D)           (None, 288, 432, 32)      896
_________________________________________________________________
max_pooling2d_24 (MaxPooling (None, 144, 216, 32)      0
_________________________________________________________________
conv2d_31 (Conv2D)           (None, 144, 216, 32)      9248
_________________________________________________________________
max_pooling2d_25 (MaxPooling (None, 72, 108, 32)       0
_________________________________________________________________
conv2d_32 (Conv2D)           (None, 72, 108, 64)       18496
_________________________________________________________________
max_pooling2d_26 (MaxPooling (None, 36, 54, 64)        0
_________________________________________________________________
dropout_8 (Dropout)          (None, 36, 54, 64)        0
_________________________________________________________________
flatten_14 (Flatten)         (None, 124416)            0
_________________________________________________________________
dense_28 (Dense)             (None, 128)               15925376
_________________________________________________________________
dense_29 (Dense)             (None, 10)                1290
=================================================================
Total params: 15,955,306
Trainable params: 15,955,306
Non-trainable params: 0
_________________________________________________________________
```

Fig. 5. CNN model for image size 288 X 432 X 3

D. Experimental Results :

1) CNN Model Results

Model No.	Dataset Setting	Image Size	Epochs	Accuracy	Loss
Model 1	Setting A	288 X 432 X 3	20	**69.4%**	**1.5**
Model 2	Setting B	64 X 64 X 3	250	**75.6%**	**1.8**
Model 3	Setting B	128 X 128 X 3	50	**72.7%**	**1.7**

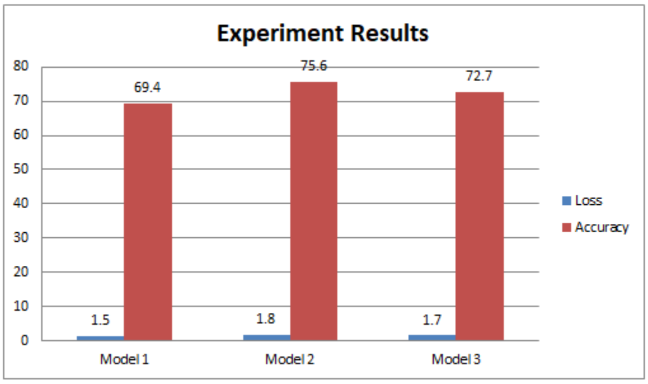


Fig. 6. CNN model Experimental Results

VI. CONCLUSION

As we have seen, the genre of a piece of music is a popular way to differentiate music. Songs with similar qualities, such as instrumentation, harmonic content, and rhythmic structure, are frequently used to define musical genres.

In this project we have aimed to use Deep Learning Methods in order to tackle the problem of Music genre classification which is an Automatic Music Classification (AMC) problem in the area of Automatic Music Retrieval (AMR).

Many factors were taken into consideration in this project, the initial model was a simple CNN model which took the original image size of 288 X 432 X 3 and the dataset size was of 1000 images giving us the accuracy of **69.4%.**

The same model was then trained on a larger dataset which consisted of 9000 images which were generated using the original dataset of 1000 images, taking an image size of 64 X 64 X 3 we obtained an accuracy of **75.6 %.** The same dataset distribution was used on the same model but the image size was taken to be 128 X 128 X 3 which obtained an accuracy of **72.7%.**

As we have noticed the deep learning techniques of CNN, produce some reasonable results for the complex problem of Music Genre Classification.

REFERENCES

[1] [1] Jeong, Il-Young, and Kyogu Lee. "Learning Temporal Features Using a Deep Neural Network and its Application to Music Genre Classification." Ismir. 2016.

[2] Lau, Dhevan S., and Ritesh Ajoodha. "Music Genre Classification: A Comparative Study Between Deep Learning and Traditional Machine Learning Approaches." Proceedings of Sixth International Congress on Information and Communication Technology. Springer, Singapore, 2022.

[3] Shaha, Manali, and Meenakshi Pawar. "Transfer learning for image classification." 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA). IEEE, 2018.

[4] Vishnupriya, S., and K. Meenakshi. "Automatic music genre classification using convolution neural network." 2018 International Conference on Computer Communication and Informatics (ICCCI). IEEE, 2018.

[5] Silla, Carlos N., Alessandro L. Koerich, and Celso AA Kaestner. "A machine learning approach to automatic music genre classification." Journal of the Brazilian Computer Society 14.3 (2008): 7-18.

[6] Feng, Tao. "Deep learning for music genre classification." private document (2014).

[7] Meng, Anders, Peter Ahrendt, and Jan Larsen. "Improving music genre classification by short time feature integration." Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.. Vol. 5. IEEE, 2005.

[8] Silla Jr, Carlos N., Alessandro L. Koerich, and Celso AA Kaestner. "Feature selection in automatic music genre classification." 2008 Tenth IEEE International Symposium on Multimedia. IEEE, 2008.

[9] Elbir, Ahmet, and Nizamettin Aydin. "Music genre classification and music recommendation by using deep learning." Electronics Letters 56.12 (2020): 627-629.

[10] Xu, Changsheng, Namunu Chinthaka Maddage, and Xi Shao. "Automatic music classification and summarization." IEEE transactions on speech and audio processing 13.3 (2005): 441-450.

[11] Zhang, Bin, Changqin Quan, and Fuji Ren. "Study on CNN in the recognition of emotion in audio and images." 2016 IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS). IEEE, 2016.

[12] https://www.analyticsvidhya.com/blog/2021/06/music-genres-classification-using-deep-learning-techniques/#h2_3

[13] MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, Andrew G. Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, Hartwig Adam

[14] K. Gadzicki, R. Khamsehashari and C. Zetzsche, "Early vs Late Fusion in Multimodal Convolutional Neural Networks," 2020 IEEE 23rd International Conference on Information Fusion (FUSION), 2020, pp. 1-6, doi: 10.23919/FUSION45008.2020.9190246.