A Novel Approach To Music Genre Classification Through Deep Learning

1st Pradeep Shet
dept.Computer Science and Engineering
KLE Technological University
Hubballi, Karnataka, India
shetp727@gmail.com

2nd TaranpreetSingh H Rababi
dept.Computer Science and Engineering
KLE Technological University
Hubballi, Karnataka, India
taranpreetsinghrababi@gmail.com

3rd Shantala Giraddi
dept.Computer Science and Engineering
KLE Technological University
Hubballi, Karnataka, India
shantala@kletech.ac.in

***Abstract*—All kinds of music genres are classified into some common categories such as pop, electronic dance music, metal, jazz, blues, coun- try, rock, and hip-hop. The most demanding task in the music informa- tion retrieval (MIR) process is organizing different music files based on these genres. Automatic music genre classification is a very important study for music platforms such as Spotify, Wynk, and Saavn that have a great impact in terms of personalized recommendations and offerings. Today's deep learning techniques like Convolutional Neural Networks (CNNs) are very successfully applied to a large variety of data by learning and inferring patterns and trends. Different techniques are reviewed for music genre classification using CNN. Further fine-tuning and additional development of VGG16 would help improve classification accuracy by learning complex feature representations of audio spectrograms and being trained on initial datasets. It is indicated that VGG16 is efficient for learning genre-specific patterns and so can serve as a flagship framework for automatic music genre classification. To improve automatic classifi- cation of music genres, ResNet-18 with the residual learning architecture was also tried for boosting classification performance in automatic mu- sic genre classification. The efficiency with which ResNet-18 can avoid the vanishing gradient problem because of the skip connections further improves the lend of deep architectures and advances automatic classifi- cation of music genres.**

***Index Terms*—Music Genre Classification, Convolutional Neural Network, VGG16 Model, ResNet-18, Sound Spectrogram Representations.**

I. Introduction

Music classification involves the classification of specific music based on its audio features, in other words, the automatic classification of music. This has become one of the most exciting research areas due to the large stock of music libraries in digital formats and the need for a smarter way to organize e music in applications such as music streaming services, music retrieval systems, etc. Customized playlist. Although these techniques are useful, they have several limitations in their ability to capture and reveal nonlinear relationships in music data. Therefore, the next development in deep learning has brought about a significant change in the field by automating the process of feature extraction to create more efficient and effective solutions. Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) show some promise in the field of learning music classification. This method learns the hierarchical representation of music data directly from music equipment, creates space to eliminate or reduce the architectural process, and allows the existing structure to adapt to all the subtleties of musical nuances in different genres.. This article discusses in detail the in depth study of music distribution popular skills and trends in this field, as well as challenges and opportunities. This article will examine the impact of different neural network models and performance metrics on research in order to provide an overview of current work while highlighting future fixes that could improve the accuracy and power of music distribution.

II. Related Work

Wang Hongdan et al. proposed an intelligent model for music classification via deep learning [7]. The model is designed to predict and classify the genre of a song. Feature extraction from spectral noise is done using the Bi-directional Long- Term Memory (BiLSTM) model. Respectively, their feature vector classifications are compared with the VGG-16 method to classify text. The best part of the proposed model is that it can classify songs better than existing algorithms. However, the proposed model has a weak hull without detailed information, which results in poor prediction and classification.

Also, Sunil Kumar Prabhakar and Seong-Whan Lee [8] proposed a comprehensive music classification based on adaptive and deep learning. The samples were classified according to sound types using five different models among them; BAG deep learning model was proven to be better than other models as it provided better classification results for GTZAN dataset. However, the deep learning process requires the use of multiple neural networks, which requires the use of many parameters in the model, which makes the work difficult and ultimately reduces the work.

A hybrid model for music classification is used by Kalyan Kumar Jena et al. [9]. DL-based models focus on CNN, multiple training, transfer learning, hybrid model classification and other methods. According to this study, the main advantage of the proposed model is that the pooling process in CNN reduces the limitations, thus optimizing the efficiency of the learning model and preventing over- learning. However, errors in the survey may affect the classification accuracy and hence the performance of the model in real time.

This paper investigates the performance of three distinguishable types of deep learning models, CNN, VGG16, and ResNet-18, on the GTZAN music genre classification dataset. Each model has been trained and tested on the dataset and compared in terms of accuracies and efficiencies. The objective is to understand how architectures work for the classification task with the associated advantages and disadvantages in classifying the ten different music genres into which the dataset is divided. The result of each model is presented and compared with other performance metrics before the author analyses how each model's architecture impacts the classification accuracy.

III. Methodology

The model includes dataset - GTZAN dataset, feature extraction, preprocessing and classification in addition to performance evaluation in the context of music genre identification and stratification.

A. Dataset

This part of the project is specifically about collecting data for classification of music genres. This system makes use of the GTZAN Genres Classification Dataset. GTZAN genres classification dataset available at Kaggle, which consist of 1000 audio tracks having each of 30-sec duration. It contains total of 100 tracks from 10 genres. It has mean and variance, thus, visual representation is available for each file. The 10 genres may include classical, country, disco, hip-hop, jazz, metal, pop, reggae, and rock.

B. Feature Extraction

Primary objective of feature extraction process is to represent a part of music briefly and descriptively. The characteristic evaluation of each short time is obtained by the short-time Fourier transform (STFT).
Less deterministic terms, short-term features and time and frequently written features are used to describe different types of features used to control the sound characters. STFT, Mel Spectrogram and Mel Frequency Cepstrum Coefficients (MFCC) are the control tools used to analyze music. The explanation of the extraction of special characteristics of the audio signal for classification is as follows:

- By adding STFT to the music signal, the recorded time of the signal is transformed into a frequency representation.
- Secondly, the STFT is converted to a mel- spectrogram, which contains a map of the frequencies obtained from the STFT to the Mel scale.
- Finally, the MFCC is calculated from the mel- spectrogram, which involves using the logarithm of the mel-spectrogram to estimate the human perception of noise. The extracted features are then pre-classified and then classified into different types.

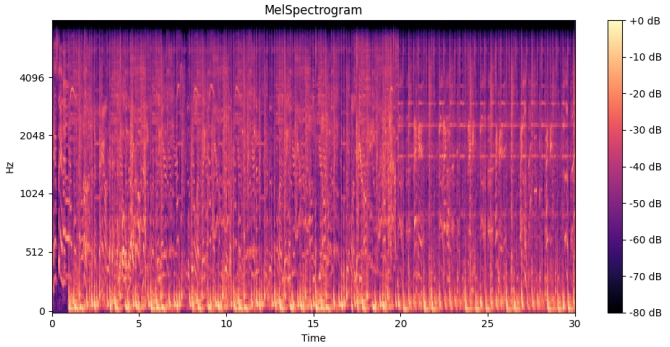

Fig. 1. Visualizing Audio as MelSpectrogram

Fig 1 Represents the MelSpectrogram of a audio file used in the training purpose, it represents the frequency vs time spectrogram of audio file.

C. Preprocessing

The features thus extracted from the audio were pre processed at a sample rate of 22,050 Hz which were then fed to give 30 seconds of audio into segments of 3 seconds each. Whenever humans need to hear audio, they usually hear it alone without any connection to other audio. In such cases, it would be possible to say that the segment division includes complete isolation of one individual clip from any other. Thus, to have some connectivity between the different segments, the 50% of the previous data duration is added to join the subsequent 50% of data duration. This would probably help in making the computer understand that the first 50% duration of data has been taken from some earlier audio clip, while the next 50% time of data is taken from that particular signal. This normalized musical signal is then passed to the proposed classifier technique for the classification of its genres.

D. Proposed Classifications

1) CNN
The pre-processed signal is currently fed to the classifier instead of into a deep convolutional network, CNN, The model extracts the main features from each signal and puts them in a classifier which correctly classifies the music.
The Fig 2 shows the proposed CNN Model for the classification of genres The main layers employed to design this model for genre classification are as follows:

- **Input Layer:**It consumes input of shape input_shape, which is typically (height, width, channels).
- **Convolution Layer:** The convolution layer is responsible for applying specific filters to the input image and extracting features such as edges or textures from it.
  - **First Convolution block:**

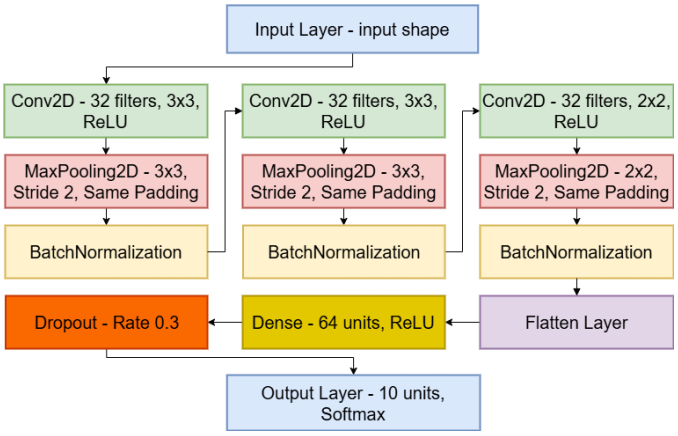

Fig. 2. Convolutional Neural Network architecture

  - ∗ Conv2D: 3x3 size filter 32 extraction low-level features, such as edges and blobs.
  - ∗ MaxPooling: The pooling layer also handles this task of reducing the spatial dimensions of the feature maps, retaining crucial information and simplifying the computational process. Space-size reduction applies a filter of 3x3 with a stride of 2.
  - ∗ BatchNormalization: It is the normalization of the activations for the learning speed.
- **Second Convolution block:**
  - ∗ Similarly, it works in the following way: Conv2D (3x3), MaxPooling (3x3), BatchNormalization.
  - ∗ This is where the network will become deeper and learn more abstract features.
- **Third Convolution block:**
  - ∗ Slightly smaller kernel (2x2) for more detailed features.
  - ∗ With pooling and normalization following this again.

- **Flatten Layer:**The 2D features extracted during convolution layers are tranferred into 1D vector in this layer.
- **Dense layer:**
  - And a Fully Connected layer with 64 ReLU activated units.
  - Followed by a Drop-out with a percentage of 0.3 in order to avoid overfitting.
- **Output Layer:** When the model produces outputs related to the presence or absence of various class probabilities, the final predictions are made. The system integrates feature extraction and classification so that input data is treated as efficiently as possible.
- uses 10 neurons for the 10 classifications.
- uses softmax activation for the classification.

2) VGG-16

Fig 3 shows the 16 layer visualization of the VGG-16 model from input to output and properly showcasing each layer. VGG16 is a highly complex in deep-convolutional-neural-network exclusively designed in image-classification-problems. It has 16-layers incorporating weight parameters (13-convolutional-layers and 3-fully connected) and applies max-pooling layers with activation functions of ReLU. Though originally built for an end image, CNNs such as VGG16 can also be modified, with image data used for other sources such as audio, in which audio signals being treated like images in their equivalent spectrogram representation.VGG16 is a highly complex deep-convolutional-neural network exclusively designed for image classification problems.

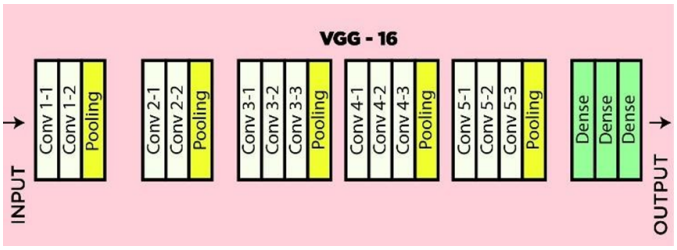

Fig. 3. VGG 16 Architecture Representation.

3) ResNet-18

Fig 4 shows the visualization diagram of ResNet-18 model which showcases what each layer contains and how it skips certain layers for classification. ResNet-18, or residual network 18, refers to a deep-convolutional-neural-network, architecture that solves the problem of vanishing-gradients in deep-networks, introduced as part of the resnet family by kaiming he et al. In the paper of their work ”deep residual learning for image recognition” in 2015.

Some characteristics of resnet-18:

- **Residual blocks:** Resnet-18 employs skip connections, which enable networks to bypass certain layers and directly incorporate the input of a layer into their output. It aids in establishing identity mappings and reduces the degradation issue when the network depth expands.
- **18 levels:** The resnet-18 model consists of 18 trainable layers, which are organized into convolution and fully connected layers grouped into residual blocks.
- **Architecture:** An initial convolutional layer is followed by 4 stages of residual blocks, each with an increasing number of feature map sizes and decreasing spatial dimensions, and closed with a global-average-pooling and a, fully-connected-layer for classification.

- **Lightweight:** Resnet-18 is more computationally efficient than deeper models like resnet-50 and resnet-101, making it suitable for use in environments with limited resources.
- **Performance:** Resnet-18 is an excellent choice for image recognition tasks, consistently delivering high accuracy levels while maintaining simplicity.

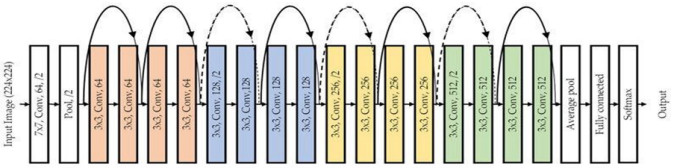

Fig. 4. Resnet 18 Architecture Representation

IV. Results and Discussion

The performance measures used for experimental evaluation are Accuracy Score, which is given by (1)

$$Accuracy = \frac{TP + TN}{FP + FN + TP + TN} \quad (1)$$

Where,
TP - True Positive
TN - True Negative
FP - False Positive
FN - False Negative

A. Comparative Analysis

The data in the below table is as follows:

TABLE I
Comparative Analysis.

Sl No	Model	Accuracy Score
1	Proposed CNN	74.23%
2	VGG16	46.15%
3	ResNet	95.33%

The Table I represents the comparative analysis between the Proposed CNN, VGG-16 model and ResNet-18 model compared with Accuracy Score of the models.

B. Training Details

TABLE II
Training Configuration for CNN

Parameter	Details
Model	CNN
optimizer	Adam
Loss Function	Sparse Categorical Cross- Entropy
Epochs	Max 30
Batch Size	32
Early Stopping	Monitors val_loss with patience of 5 and restores best weights
Metrics	Accuracy

TABLE III
Training Configuration for VGG16

Parameter	Details
Model	VGG16 (base) + Flatten + Dense (256 units) + Dropout (0.5) + Dense (10 units)
Pre-trained model	VGG16 (weights="imagenet", include_top=False)
Base Layer	Frozen
Optimizer	Adam
Loss function	Sparse Categorical Cross-Entropy
Activation function	Relu, Softmax
Early stopping	Monitors 'val_loss', patience=3, restores best weights
Epochs	Maximum number of epochs: 30; however, early stopping might end sooner.
Metric	Accuracy

TABLE IV
Training Configuration for ResNet-18

Parameter	Details
Model	ResNet-18
Dataset	GTZAN Dataset
Number of Classes	10
Input Image Size	224 × 224
Batch Size	32
Number of Epochs	25
Optimizer	Adam
Learning Rate	0.001
Loss Function	Cross-Entropy Loss
Metrics	Accuracy

C. Results

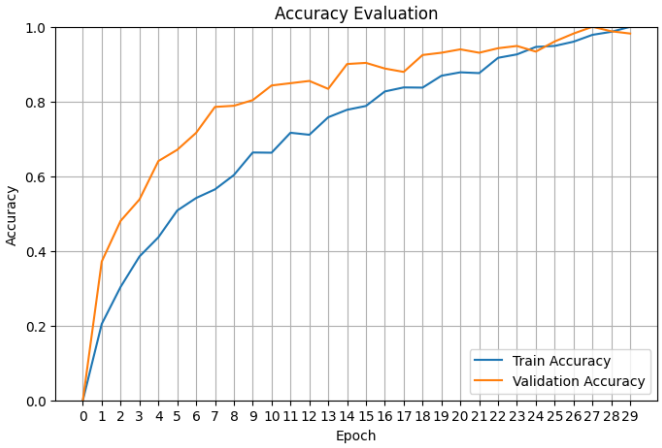

Fig. 5. Training and Validation Accuracy Graph of Proposed CNN

Fig 5 Displays the Training and Validation Data Sets Graphically, accuracy was achieved through an accuracy vs. epochs graph, where the blue line represents the training data set accuracy and the orange line indicates the accuracy of the validation data set.

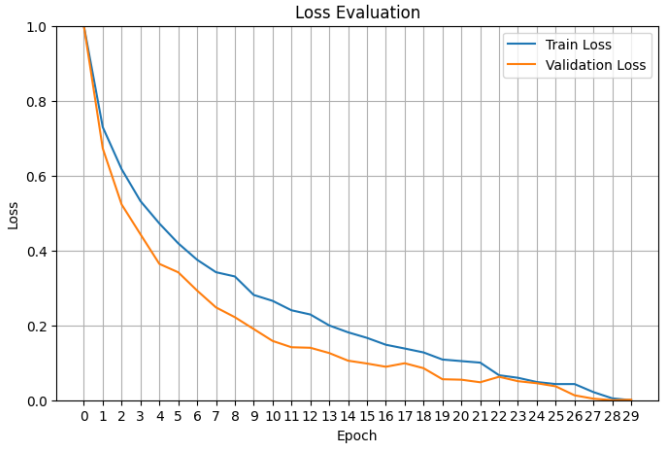

Fig. 6. Training and Validation Loss Graph of Proposed CNN

Fig 6 Provides graphical representation training and validations itself rather loss versus epochs curve by making it apparent how the blue line indicates training loss and the orange line validation loss.

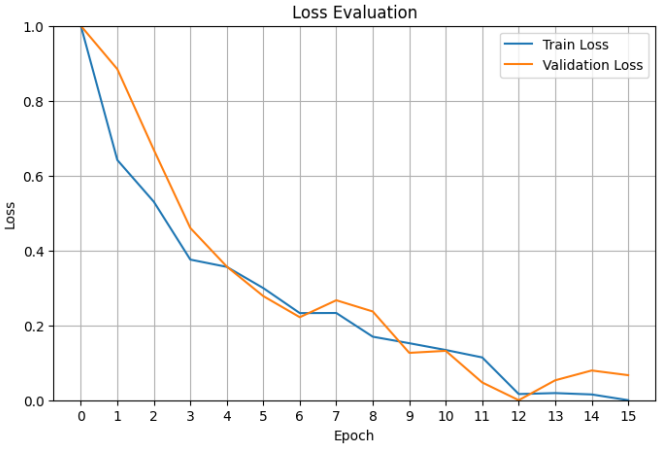

Fig. 8. Training and Validation Loss Graph of VGG16 Model

Figure 8 shows the Training and Validation loss of the VGG16 model graphically. This is the loss versus epoch findings where the blue line represents training loss, and the orange line represents validation loss.

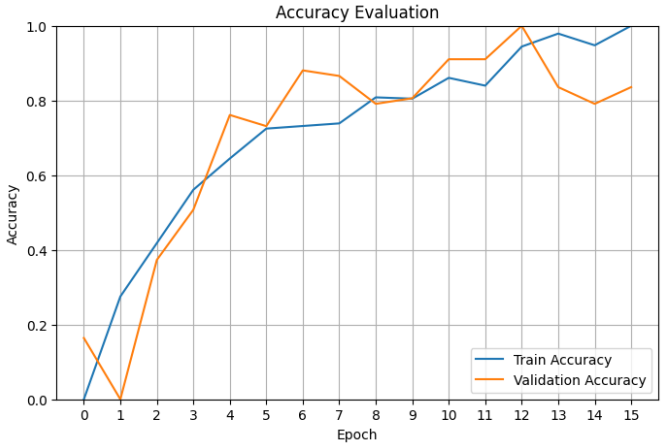

Fig. 7. Training and Validation Accuracy Graph of VGG16 Model

Fig 7 represents the training and the validation accuracy metrics obtained graphically for the VGG16 model where the blue line signifies the accuracy of training, while the orange line signifies the accuracy of validation.

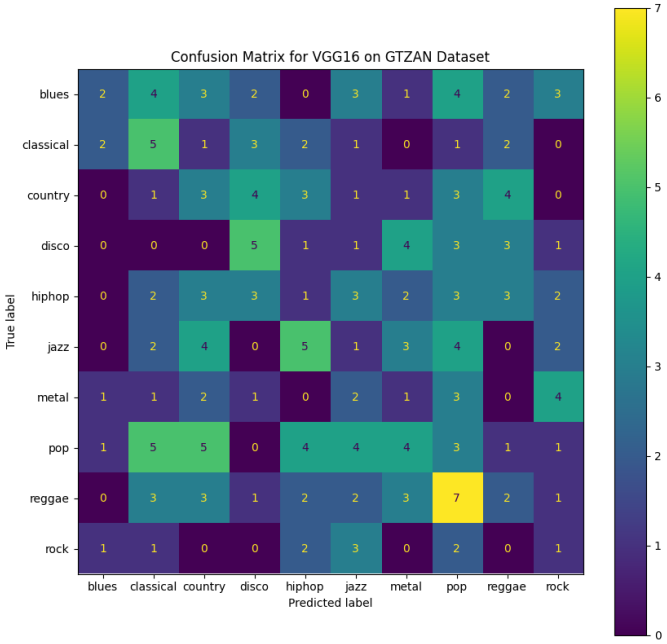

Fig. 9. Confusion Matrix of VGG16 Model

Fig 9 Training using the VGG16 model shows confusion matrices for various genres with the counts of their true-predicted labels.

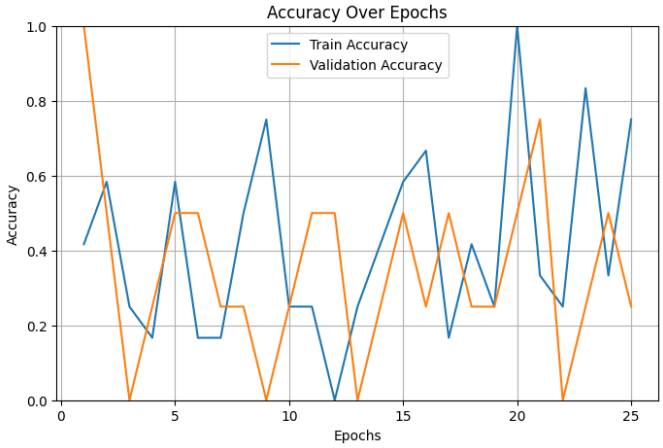

Fig. 10. Training and Validation Accuracy Graph of Proposed ResNet18 model

Training and Validation values for the ResNet18 model have been shown in Fig 10, where the graph depicts the Accuracy vs Epochs. In this case, the blue line shows the training accuracy,, and the orange line shows the validation accuracy.

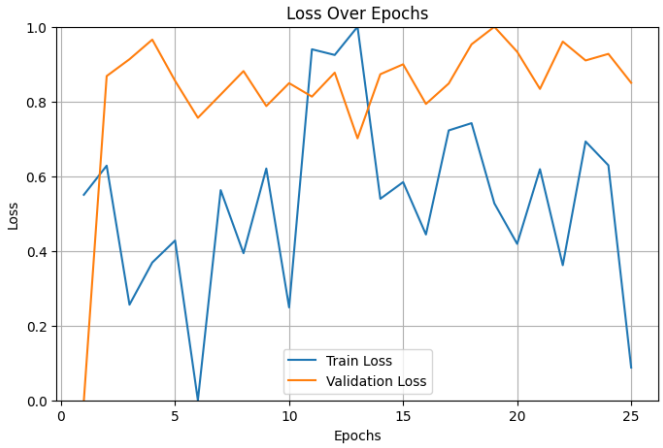

Fig. 11. Training and Validation Loss Graph of ResNet18 Model

This is the graph that illustrates the training and validation loss of the Resnet18 model. A Loss vs Epochs graph is shown here along with the blue line indicating training loss and orange representing validation loss. Fig 11 depicts the Training and Validation loss of ResNet18.

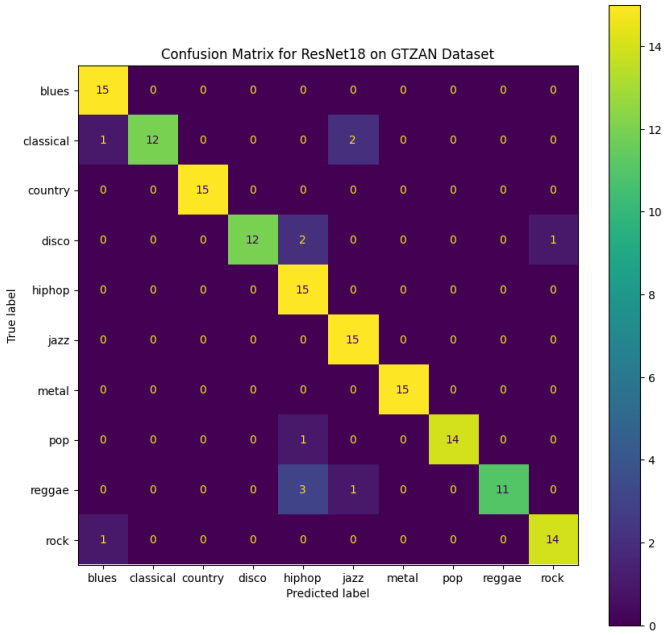

Fig. 12. Confusion Matrix of Resnet18 Model

The Confusion Matrix of the ResNet18 Model is displayed in Fig 12. The True labels on the y-axis and Predicted labels on the x-axis present the counts of true versus predicted labels for all genres of music.

V. Conclusion

Significantly, the accuracy of ResNet model was 95.33This was far ahead of the accuracy by Proposed CNN (74.23%) and VGG16 (46.15%). It thus proves the feature extraction capabilities that have been made possible by the residual connections within the neural network that reduce the problem of vanishing gradients. Performance of the Proposed CNN was modest, showing that there is more possible optimization on the model. At the same time, very low accuracy was achieved with VGG16, indicating that this model was less suited for the particular task or dataset.

VI. Future Work

In future studies, one could transfer learning for model improvement or additional data preprocessing techniques to improve VGG16 performance further and hence augment accuracy. Besides these, trying more complex models or hybrid approaches could yield better generalization for this particular problem.

References

[1] Li, J., Han, L., Li, X., Zhu, J., B. Yuan, and Z. Gou: Siv or spectrogram. Multimedia Tools and Applications, pp. 1–27 (2022).

[2] Jena, G., Bhavani, B.G., Naidu, S.R., Prasad, T.V.D., Sravya, T.H., Ganesh, P.V., and Sarki, H.: A web-based music distribution system for real-time song visualization and analysis. Distance Education, pp. 4061–4075 (2022).

[3] Sharma, A.K., Aggarwal, G., Bhardwaj, S., Chakrabarti, P., Chakrabarti, T., Abawajy, J.H., Bhattacharyya, S., Mishra, R., Das, A., and Mahdin, H.: IEEE Access **9**, pp. 102041–102052 (2021).
[4] Local activation gated neural network is very much needed for automatic music genre classification Liu, Z., Bian, T., and Yang, M. Applied Science **13**(8), p. 5010 (2023).
[5] Zheng, Y., and Kuo, C.N.: Lej **10**(23), p. 4427 (2022).
[6] Sharm, J., Divya, P., Vishnu, C., Reddy, C.L., Shekhar, B.H., and Mohan, J.K. See VISIGRAPP (4:VISAPP), pp. 56–64 (2023).
[7] Hongdan, W., SalmiJamali, S., Zhengping, C., Qiaojuan, S., and Le, R.: Deep Learning Techniques for Intelligent Music Analysis, Extraction, and Classification. Computers and Electrical Engineering **100**, p. 107978 (2022).
[8] Prabhakar, S.K., and Lee, S.W.: Applied Expert Systems **211** (2023).
[9] Jena, K.K., Bhoi, S.K., Mohapatra, S., and Bakshi, S. Neural Computing and Applications **35**(15), pp. 11223–11248 (2023).
[10] Xie, C., Song, H., Zhu, H., Mi, C., Li, Z., Zhang, Y., Cheng, J., Zhou, H., Li, R., and Cai, H.: Music classification based on RES gated CNN and listening technique. Multimedia Tools and Applica
[11] Rawat, Parv Dharaskar, Krushna Nandanwar, Anshula Dhawale, Krupali Rupesh. (2023). MUSIC GENRE CLASSIFICATION USING DEEP LEARNING. 2582-5208. 10.56726/IRJMETS34036.
[12] McFee, Brian Raffel, Colin Liang, Dawen Ellis, Daniel Mcvicar, Matt Battenberg, Eric Nieto, Oriol. (2015). librosa: Audio and Music Signal Analysis in Python. 18-24. 10.25080/Majora-7b98e3ed-003.
[13] Hossan, Md Memon, Sheeraz Gregory, Mark. (2011). A novel approach for MFCC feature extraction. 1 - 5. 10.1109/ICSPCS.2010.5709752.
[14] Tzanetakis, George Cook, Perry. (2002). Musical Genre Classification of Audio Signals. IEEE Transactions on Speech and Audio Processing. 10. 293 - 302. 10.1109/TSA.2002.800560.
[15] Bhatia, Jitesh Singh, Rishabh Kumar, Sanket. (2021). Music Genre Classification. 1-4. 10.1109/ISCON52037.2021.9702303.
[16] Mounika, K Deyaradevi, S Swetha, K Vanitha, V. (2021). Music Genre Classification Using Deep Learning. 1-7. 10.1109/ICAECA52838.2021.9675685.
[17] K, Purushotam Rao, B. Revathi, K. Gayathri, M. Jayasri, G.. (2025). Classification of Music Genres using Multimodal Deep Learning Technique. E3S Web of Conferences. 616. 10.1051/e3sconf/202561602012.
[18] Chettiar, Gautam Selvakumar, Kalaivani. (2021). Music Genre Classification Techniques.
[19] Kilambi, Bhargav Parankusham, Anantha Tadepalli, Satya Kiranmai. (2021). Instrument Recognition in Polyphonic Music Using Convolutional Recurrent Neural Networks. 10.1007/978-981-15-8443-5_38.
[20] Bawitlung, Andrew Dash, Sandeep. (2023). Genre Classification in Music using Convolutional Neural Networks. 10.1007/978-981-99-7339-2_33.