Journal of Network Intelligence
Taiwan Ubiquitous Information
©2025 ISSN 2414-8105
Volume **10**, Number **4**, November **2025**

A Bimodal Music Genre Classification Model Based on Convolutional Attention Mechanisms

Yu-Wen Li*

Department of Music
Nanchang Institute of Technology
Nanchang 330044, P. R. China
15170235963@163.com

Yu-Han Wang

International College
Krirk University
Bangkok 10220, Thailand
zitong0825@gmail.com

*Corresponding author: Yu-Wen Li
Received July 12, 2024, revised December 29, 2024, accepted July 7, 2025.

Abstract. *Music genre classification is a key link in music information retrieval. Traditional music genre classification models need to extract music signal features for processing, leading to difficulties in global feature extraction and poor model classification performance. Aiming at the above issues, this article suggests a bimodal music genre classification model based on the convolutional attention mechanism. Firstly, for the insufficient global feature extraction ability of CNN, the original ordinary convolution operation is replaced by residual structure, and the network model is deepened to avoid model degradation problems such as gradient disappearance and ensure the effectiveness of the model. Secondly, the sound of the music is transformed by Fourier transform to form the acoustic spectrogram, and the spectral energy is obtained by preprocessing the spectrum of the music, and the local features of the acoustic spectrogram are extracted through the multi-channel convolutional attention mechanism to connect with the audio timing data before and after, and then use the mutual attention mechanism to fuse the feature data of the two modalities and fully learn the feature correlation between different modalities, and the fused features are inputted into the Softplus classifier, to output the classification outcome of music genres. Experimental outcome on the GTZAN dataset implies that the classification accuracy of the designed model is 4% to 12% higher than the other three deep learning models.*
Keywords: music genre classification; convolutional neural network; attention mechanism; residual structure; feature extraction

1. **Introduction.** Music genres are different music style categories formed in different time, different regions and different cultures, which form personalised factions through different musical elements such as melodies, rhythms, tunes, harmonies, etc. in a unique organisational form, for example, jazz, classical, and rock are different music genres [1]. As the Internet technology rapidly growing, almost all music has been digitized and uploaded to the Internet. People's demand for music appreciation is also gradually increasing, and multimedia technology is facing a brand new innovation, showing the trend of massive, fast and intelligent. Early manual classification and annotation is far from being able to keep up with the update speed of network data, which consumes a lot of time, manpower and

money [2, 3]. The increasingly large digital music databases need intelligent classification management and storage, so the Music Information Retrieval (MIR) system and the music genre classification system have received more and more extensive attention from the society and the academia, and the music genre classification has been an emerging research topic [4].

1.1. **Related work.** At the beginning of MIR research, researchers used traditional machine learning methods to solve the music genre classification problem. Nasridinov and Park [5] combined pitch, timbre and rhythm attributes into a feature set, and classified them after selecting the feature set by Gaussian mixture model. Meng et al. [6] extracted the audio features by processing the mean and autocorrelation coefficients of the original audio signals, and then used the KNN algorithm for the genre attribute discrimination. Baniya and Lee [7] introduced audio features such as spectral variance, combined with principal component analysis and sparse coding theory for dimensionality reduction, and analyzed and investigated sparse representation for model classification. Mutiara et al. [8] used scale-invariant feature transform for extracting music features, and combined with support vector machine for genre classification. Nanni et al. [9] explored the relationship between timbre and rhythmic features and the classification effect, and combined multiple features for SVM classification, which achieved good accuracy. Traditional machine learning has proved the effect of automatic genre classification in the field of music genre classification, but there are problems such as easy overfitting and difficulty in handling large-scale samples [10]. With the development of deep learning, neural network-based genre classification methods can effectively solve the above problems. Liu et al. [11] applied Deep Belief Networks (DBNs) to discriminate music authors, differentiate music genres and other music retrieval related tasks. Patil et al. [12] used BP neural networks to learn and classify Mel frequency cepstrum coefficients of music in an end-to-end manner. Hongdan et al. [13] used LSTM for music genre classification and combined them with SVM classifiers to enhance their performance. Yu et al. [14] proposed the use of RNN to extract the features while augmenting the features with attention mechanism for classification, but the underlying characteristics of music are ignored, leading to inefficient classification. Singh and Biswas [15] extracted multiple features of audio content as feature data and classified the features with the help of BP neural network. content with multiple features as feature data and used RNN algorithm for music genre classification. Compared to neural networks such as BP, LSTM, DBN, CNNs have an advantage in classification algorithms by reducing the number of parameters required for training through sensory wildcard and weight sharing. Reeja and Yaseen [16] learned and classified the Mel's inverted spectral coefficients of audio signals through CNNs. Oramas et al. [17] designed a convolutional neural network based on 1D convolution and 1D max-pooling, and verified the effectiveness of Mel's spectrum in feature representation in the feature preprocessing stage. Yang et al. [18] incorporated an attention mechanism into the CNN, which enables it to better adapt to the data that is closely connected with the data before and after the audio signals, etc. Mangolin et al. [19] used a two-channel CNN to fuse audio and lyrics semantic modalities for music genre classification.

1.2. **Contribution.** Audio features are very different from image classification in computer vision tasks. Previous models on audio feature extraction are difficult to automatically design appropriate features for a specific task; and methods based on acoustic spectrograms tend to ignore the underlying features of audio. To address these issues, this article suggests a bimodal music genre classification model based on convolutional attention mechanism. The CNN is first optimized (OCNN) using residual structure and Leaky ReLU to ensure the effectiveness of the model while enhancing the highly abstract

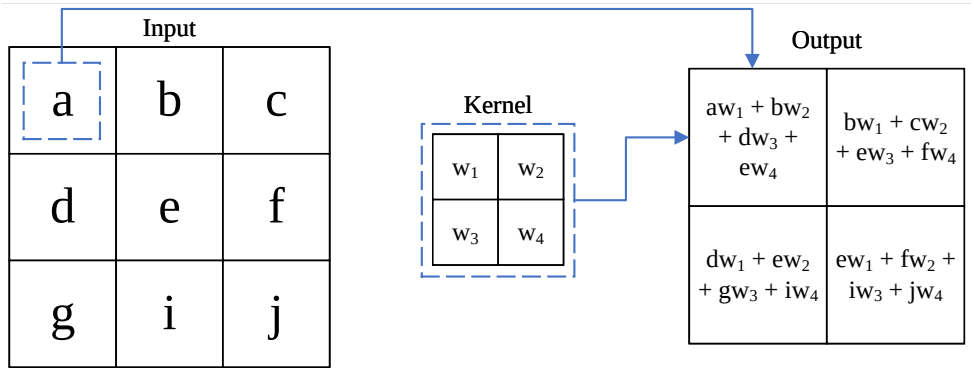

Figure 1. Convolution operation computational procedure

extraction of genre features. Then the sound and spectrum of the music are preprocessed to obtain the spectrogram and audio energy, and the features of the two modes are extracted by the multi-channel convolutional attention mechanism and OCNN, which take into account the multiple characteristics of the audio data. Using the mutual attention mechanism to fuse the feature data of the two modalities and input the fused features into the Softplus classifier to output the categories of music genres. The experimental outcome indicates that the designed model exhibits excellent classification performance with high classification accuracy and low classification time.

2. Related theoretical analysis.

2.1. **Convolutional neural network.** The CNN network structure is sparsely connected and makes the network layer translation invariant through parameter sharing [20]. In addition, CNNs can autonomously acquire key information from the learning samples; have better fault tolerance, allowing for distorted or missing samples [21]. The main components of CNNs are implied as bellow. (1) Input level, which is the sample data fed into the network for training and testing. (2) Convolutional level, i.e. feature extractor, which only focuses on the local features of the input object, the weights of the convolution kernel will not change when it is convolved with different regions of the input feature vector of the level, which can effectively reduce the parameters in the network. Figure 1 represents the process of convolution operation on a two-dimensional tensor with a single convolution kernel.

(3) Pooling level: compressing the feature representation input to this level can firstly make the dimension of the feature representation smaller and reduce the complexity of the network; secondly, compressing the features learnt by the network, so as to be able to extract the significant features from the graph. (4) Fully connected level: all the features are connected and the results obtained from this level will be output to the classifier for classification.

2.2. **Attention mechanism.** The fully-connected layer in CNN needs to reallocate the weights when dealing with input sequences with long time series, which will result in difficult and time-consuming computation. AM is able to "dynamically" allocate the weights of different connections [22], and usually adopts the QKV (Query-Key-Value) mode. The calculation process is indicated in Figure 2.

Assuming that the input sequence is $X = [x_1, \ldots, x_N] \in \mathbb{R}^{D_x \times N}$ and the output sequence is $H = [h_1, \ldots, h_N] \in \mathbb{R}^{D_v \times N}$, each input is mapped to three different spaces to obtain three vectors: the query vector $q_i \in \mathbb{R}^{D_k}$, the key vector $k_i \in \mathbb{R}^{D_k}$, and the value vector $v_i \in \mathbb{R}^{D_k}$. The inputs are mapped to three different spaces.

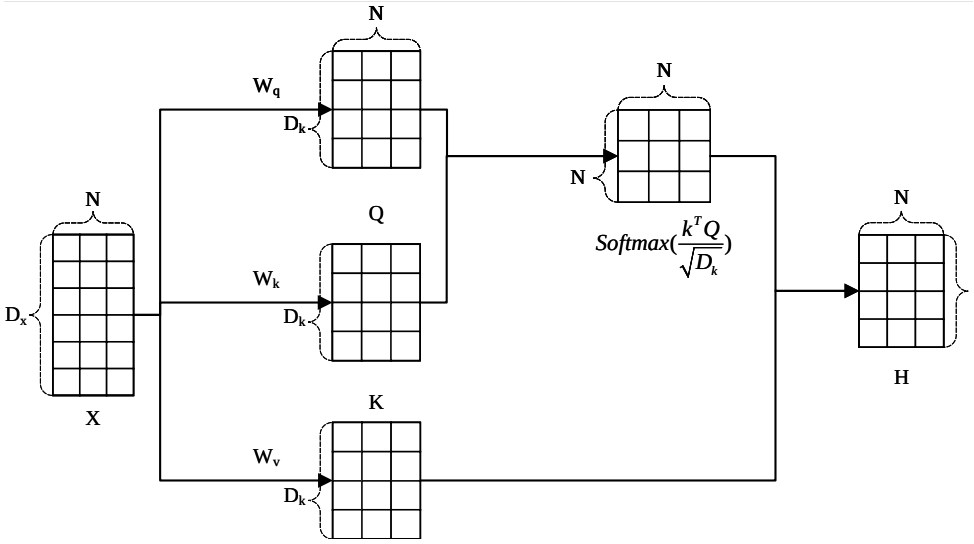

Figure 2. The computational process of the attentional mechanism

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^{\mathsf{T}}}{\sqrt{d_k}}\right) \tag{1}$$

where $Q = W_q X,\ K = W_k X,\ V = W_v X$, and W denotes the weights.

3. **Optimization of convolutional neural networks incorporating residual structure and LeakyReLU.** CNN carries out repeated convolution operations by stacking convolution kernels, but it has been found that too much increase in the number of network layers will bring problems such as gradient vanishing; for such problems, the current solution is to add batch normalization [23], but it will directly lead to a lower accuracy rate on the music training set. The residual structure can avoid the network degradation problem while enhancing the feature extraction capability of audio spectrogram. The residual structure allows the model to skip these network layers and directly accept the audio spectrogram signal transmission from the upper layers [24], preserving integrity of the information, which can be expressed as follows.

$$x_{k+1} = x_k + F\big(x_k, V_k^l\big) \tag{2}$$

where x_{k+1} denotes the input of level $k+1$ of the network, which can also be regarded as the output of level x_k of the network, and $F(x_k, V_k^l)$ denotes the convolution operation of the CNN.

The characteristics of the deep cell K can be expressed recursively as bellow.

$$x_k = x_k + \sum_{i=k}^{K-1} F\big(x_k, V_k^l\big) \tag{3}$$

The above residual structure is applied to the convolutional layer, and the neurons with larger feedback are relatively enlarged, and the neurons with smaller response are suppressed, so as to improve the generalization ability of image classification and recognition. The calculation equation is as bellow.

$$b_{x,y}^i = \frac{a_{x,y}^i}{\left(k + \alpha \sum_{j=\max(0,\, i-n/2)}^{\min(N-1,\, i+n/2)} \left(a_{x,y}^j\right)^2\right)^{\beta}} \tag{4}$$

where $b^i_{x,y}$ represents the neurons obtained by local response normalization; $a^i_{x,y}$ represents the i-th convolution kernel in position (x, y) to get the element in the feature graph; n is the number of adjacent convolution; N represents the total number of convolution nuclei in a certain level. k, n, α is a hyperparameter.

Subsequently, the activation function ReLU function of the convolutional layer is replaced with the Leaky ReLU function [25], and the equation is as bellow.

$$f(x) = \begin{cases} 0.01x, & x < 0, \\ x, & x \geq 0 \end{cases} = \max(0.01x, x) \tag{5}$$

When Leaky ReLU takes the value in the negative interval, the output value is not 0, which avoids the problem of neurons not learning after the ReLU function enters the negative interval, and thus effectively solves the necrosis problem of neurons brought by the ReLU function. Therefore, in the process of feature extraction, Leaky ReLU function can effectively improve the integrity and utilization of feature information.

To avoid overfitting during the training process, a random regularization (Dropout) [26] is applied between the fully connected layers, where neurons in the hidden layer of the network are randomly deleted with probability p during forward propagation. Then the back propagation is performed to update the parameters and restore the deleted neurons, keeping the number of input and output neurons constant and reducing the interactions between neurons.

$$\tilde{y}^{(k)} = r^{(k)} \times y^{(k)} \tag{6}$$

$$z_i^{(k+1)} = W_i^{(k+1)} \tilde{y}^{(k)} + b_i^{(k+1)} \tag{7}$$

where k denotes the number of hidden layers, Bernoulli(p) generates the probability vector r, $\tilde{y}$ denotes the intermediate result after stopping some of the neurons with probability r, z denotes the vector of inputs into level k. w and b denotes the weights and biases in level k.

The Softplus [27] classifier is added after the fully connected layer of extracted features for feature classification. Softplus is calculated as bellow.

$$\text{Softplus}(x) = \frac{1}{\beta} * \log(1 + \exp(\beta * x)) \tag{8}$$

where β is the hyperparameter.

Compared with Softmax, Softplus has the advantages of unilateral inhibition and relatively wide excitation boundary. Its continuous conductivity in the definition domain allows the gradient to propagate over the entire definition domain, which improves the classification performance of the model.

4. **Bimodal music genre classification model based on convolutional attention mechanism.**

4.1. **Acoustic spectrograms of music and music signal pre-processing.**

Based on the above optimized CNN, this article suggests a bimodal music genre classification model based on the convolutional attention mechanism. In this model, the sound of music is Fourier transformed to form the acoustic spectrogram, and the spectrum of music is preprocessed to obtain the spectral energy, after which the data of the two modalities are feature extracted by the multi-channel convolutional attention mechanism and optimized CNN respectively, which comprehensively consider the multiple characteristics of the audio data, and then the feature data of the two modalities are fused by using the inter-attention mechanism and the correlation of the features between the different modalities is fully learnt, and finally the fused features are input to the Softplus classifier

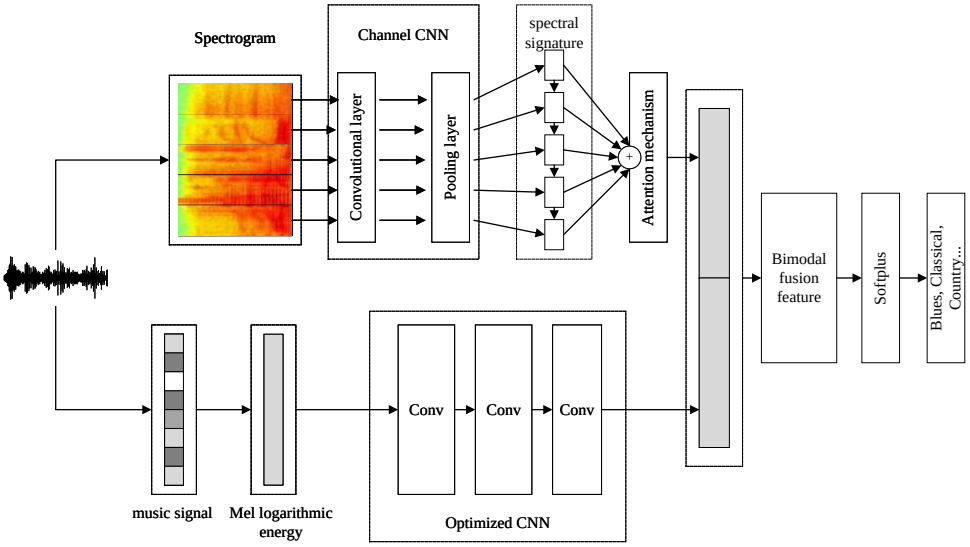

Figure 3. The designed music genre classification model

to output the category of the music genre. The suggested music genre classification model is implied in Figure 3

Music is both a sound and an art form, containing both sound and music characteristics, and this article pre-processes these two attributes separately.

(1) Sound pre-processing. The sound of music is processed by Fourier transform to form a sound spectrogram in the following steps.

Step1: Pre-Emphasis. The high frequency part of the signal is amplified by the pre-emphasis filter, and the processed digital audio signal can make the signal-to-noise ratio become balanced while highlighting the high frequency resonance peaks.

Step2: Split frame. Split the speech into frames from the beginning, intercept N samples as a frame, move $N-M$ samples, starting from the $N-M$-th sample point as the header of the next frame, each move to form a frame.

Step3: Add Window. Suppose the signal after frame splitting is $S(n)$, $n = 0, 1, \ldots, N-1$, where N is the size of the frame. After multiplying the Hamming window, $w(n)$ is as bellow.

$$\omega(n) = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) \tag{9}$$

Step4: Perform a Fast Fourier transform (FFT) on $S(n)$. By FFT the signal will be converted from time domain to frequency domain and then logarithmically transformed to Fourier inverse transform with the following equation.

$$S(m,\omega) = \sum_{n=-\infty}^{\infty} x[n] \cdot \omega[n-m] \cdot \exp(-j\omega n) \tag{10}$$

After the above four steps of processing, each sound clip is transformed into an acoustic spectrogram characterized by aspects such as chroma, pitch and intensity.

(2) Music data preprocessing. Firstly, the spectrum of the music signal is squared to obtain the spectral energy, and the energy spectrum is passed through a set of triangular band-pass Mel-frequency filters to obtain the logarithmic energy of the output of each triangular filter, a total of K.

$$E(k) = \ln\left(\sum_{n=0}^{N-1} |X(n)|^2 H(n)\right), \quad k = 1, 2, 3, \ldots, K \tag{11}$$

The filter output energy is logarithmic and its discrete cosine transformation is calculated to obtain the pre-processed data.

$$C_m = \sum_{k=1}^{N} \cos\left(\frac{m\,(k-0.5)\,\pi}{N}\right) E_k \tag{12}$$

where E_k denotes the value of the inner product of the frequency energy and the delta Mel frequency filter, and N is the number of delta filters.

4.2. **Bimodal feature extraction for music based on convolutional attention mechanism.** On the basis of the above pre-processing of sound and music signals, this paper extracts the two modal features of the acoustic spectrograms and music signals using channel CNN fused with self-attention mechanism and OCNN respectively.

(1) Feature extraction of acoustic spectrograms The number of CNN channels represents the size of the convolutional kernel in each convolutional layer. Each channel is an abstraction of the original image. As the number of channels increases, the loss of the acoustic spectrogram decreases. In the first three layers of the CNN, we need to obtain more local features from the acoustic spectrogram, so we choose 32, 64, and 128 channels, respectively. In the last two layers, we use the number of channels of 256 and 128 to accelerate the size reduction of the feature maps so that the network can capture the global features.

The model generates each frame of the acoustic spectrogram X from the sample through a channel CNN for feature representation and splices it in the timing, and the size of the output vectors H depends on W, F, P, and S. The equation is as bellow.

$$H = \frac{W - F + 2P}{S} + 1 \tag{13}$$

where W denotes the input size, F denotes the convolution kernel size, P denotes the filling size, and S denotes the step size.

During feature extraction, each local feature of the input is first computed using the residual convolution kernel of a single channel CNN as shown in Equation (14). The computed features are then vertically concatenated as shown in Equation (15). Finally, the results are nonlinearly computed by the Leaky ReLU activation function to obtain the final residual convolutional features as shown in Equation (16), where W_F denotes the residual convolutional kernel with height F, b indicates deviation, H is the size of the output vectors, and h denotes the nonlinear computation.

$$h_F(i) = f\big(W_F \cdot X(i : i + F - 1) + b\big) \tag{14}$$

$$h_F = [\, h_F(1);\ h_F(2);\ \ldots;\ h_F(H)\,] \tag{15}$$

$$h_r = \mathrm{LeakyReLU}[\, h_F\,] \tag{16}$$

To enhance the key features of each channel, the self-attention mechanism is used to compute an attention score corresponding to it for each feature extracted on the CNN of each channel using the following formula.

$$e_i = v_i^{\mathsf{T}} h_i = \frac{\exp\big(f(r_i)\big)}{\sum_j \exp\big(f(r_j)\big)} \tag{17}$$

where e_i denotes the attention score assigned to the i-th feature vector, r_i, r_j denotes the attention weight value, $E = \{e_1, e_2, \ldots, e_r\}$, h_i are the i-th feature vectors of h_r.

After obtaining the feature weights for each channel, the key feature information obtained for each channel is aggregated using two fully connected layers to obtain the final key features as follows.

$$h^* = e_i \cdot r_i \tag{18}$$

(2) Music data feature extraction. After preprocessing, the music data X' is firstly subjected to convolution operation, and the convolution feature F^C can be denoted as $F^C = X' * K + b$, where K is the residual convolution kernel, b is the bias, and $*$ denotes the convolution operation. After that, the pooling operation is performed on the convolutional feature F^C, and the pooled feature can be denoted as $F^P = pool(F^C)$. At the same time, the nonlinear feature F^N can be obtained by Leaky ReLU as $F^N = \text{LeakyReLU}(F^P)$. Therefore, the feature node Z can be denoted as follows.

$$Z = \text{LeakyReLU}(F^N W + \beta) \tag{19}$$

where W and β are the weights and biases of the fully connected layer. Assuming that there are n groups of audio feature nodes embedded by CNN, the i-th feature node is named as Z_i. All feature nodes can be denoted as $Z^n = [Z_1, Z_2, \ldots, Z_n]$, so that the m groups of feature nodes can be expressed as in Equation (20), where W_{h_i} and β_{h_i} are randomly generated, ξ denotes hyperparameter

$$h_m = \xi\left(Z'' W_{h_i} + \beta_{h_i}\right) \tag{20}$$

4.3. **Bimodal feature fusion and genre classification in music.** In general, the feature fusion is a direct splicing of the eigenvectors of different modes, which cannot consider the differences between different modes and the different weights of modes in decision making well. Therefore, this article adopts the mutual attention mechanism of feature fusion, and the specific formulas are as bellow.

$$F_m^* = \text{softmax}\left(\frac{h^*(h_m)^{\mathsf{T}}}{\sqrt{d_k}}\right) h^* \tag{21}$$

$$F_*^m = \text{softmax}\left(\frac{h_m(h^*)^{\mathsf{T}}}{\sqrt{d_k}}\right) h_m \tag{22}$$

$$F_{m^*} = \text{Concat}(F^*, F_m) \tag{23}$$

where h^* is the feature of the spectrogram and h_m is the audio feature of the music. The mutual attention feature F_m^* of the spectrogram on the music and the mutual attention feature F_*^m of the music on the spectrogram are obtained through the computation, and finally the spectrogram–music mutual attention feature F_{m^*} is obtained through the cascade of vectors.

F_{m^*} is used as the input to Softplus's classifier to classify music genres. Assuming a total of K labels and V^{label} denoting the parameter matrix of Softplus, the distribution of classes can be predicted based on the following equation.

$$\mathbf{y} = \text{Softplus}\left(V^{\text{label}} F_{m^*}\right) \tag{24}$$

where $\mathbf{y}$ is a k-dimensional output level representation and each dimension represents the predicted probability value for each label.

The fusion feature F_{m^*} obtained above consists of randomly selected anchors g_a^i with positive intraclass samples g_p^i and negative interclass samples g_n^i. Therefore, a large number of sample pairs $D(g_a^i, g_p^i) - D(g_a^i, g_n^i)$ with different spacing distances $\{g_a^i, g_p^i, g_n^i\}$ are constructed in each iteration for robust embedding optimisation. In order to achieve intra-class tightness and inter-class separation, all retained sample pairs have to meet the final optimization objective.

$$D(g_a^i, g_p^i) + \alpha < D(g_a^i, g_n^i) \tag{25}$$

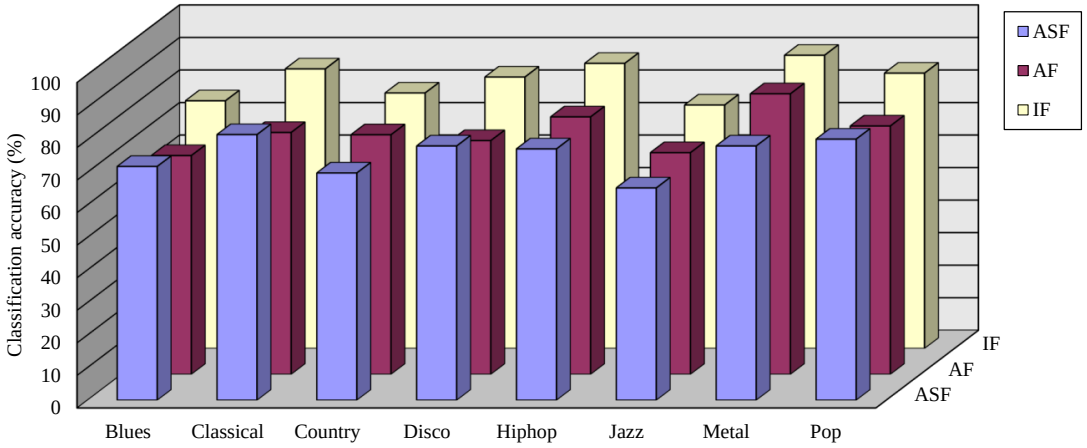

Figure 4. Histogram comparing the classification accuracy of different features

According to the idea of Attentional Comparative Learning (ACL), attention weights are generated for the edge distances, and higher weights should be given to difficult sample pairs with larger distances if $D(g_a^i, g_p^i) \gg D(g_a^i, g_n^i)$, in order to gain more attention. Therefore, under the guidance of ACL, the learning loss of the designed model is as bellow.

$$L_{\text{ACL}} = \sum_{i=1}^{M} \alpha_1 \max\left(D(g_a^i, g_p^i) - D(g_a^i, g_n^i) + \alpha,\, 0\right) \tag{26}$$

5. **Performance testing and analysis.**

5.1. **Analysis of classification results for different features.** The experiments in this article are mainly conducted on a server configured with Intel Core i7 2.9 GHz Central Processing Unit (CPU) + Nvidia GeForce GTX 2080 Ti GPU, and in addition, the TensorFlow deep learning framework based on the Python language is used to build the models for the comparison experiments. The experiments use the open-source GTZAN dataset [28], which is a public dataset commonly used in the field of music genre recognition and contains 800 music data, including Blues, Classical, Country, Disco, Hiphop, Jazz, Metal, Pop, and a total of eight music genres. The dataset is divided into training set, validation set and test set, of which 70% is training set, 20% is validation set and 10% is test set. The Adam Optimizer was used for model training, with a learning rate of 0.0003, Epochs set to 300, BatchSize set to 128, and GPU accelerated training.

This article compares the classification performance of acoustic spectrogram features, audio features, and their fusion features applied to GTZAN music database through simulation experiments. Figure 4 implies the histogram of the classification accuracy of music genres under the three features. The fusion of the two modalities of sound spectrogram and audio was generally higher than that of the single-modality approach, and the classification accuracy was better in each genre, especially in hip-hop, metal, and pop, but not in blues and jazz. The reason is that the vocal part of blues is low and the frequency is concentrated in the low frequency, which is not easy to be extracted, and the characteristics of the whole time-frequency plane are not prominent enough, while the frequency components of jazz have small harmonic gaps and short time, and there are frequent pitch changes, which makes it easier to distort the timbres in the feature extraction.

Table 1. Overall comparison of the classification effects of different features

Features	Maximum (%)	Minimum (%)	Average (%)
ASF	84.2	65.2	81.5
AF	80.3	67.2	74.7
IF	92.4	74.8	88.9

A comparison of the highest, lowest, and average classification accuracies for the acoustic spectrogram feature (ASF), the audio feature (AF), and the proposed fusion feature (IF) is given in Table 1.

5.2. **Comparative analysis of classification performance.** To evaluate the classification performance of the proposed model MG-CAM, this paper conducts comparative experiments with the NMM-DN model [12], the DA-MGC model [14] and the MAL-GC model [19] with respect to classification accuracy, training time, classification time, and ROC curve [29]. The comparison of classification performance metrics of different models is given in Table 2. As can be seen from the table, MG-CAM has the shortest training time and classification time, and the highest classification accuracy of 89.3%, which is an improvement of 11.2%, 8.8%, and 4.6% compared to the NMM-DN, DA-MGC, and MAL-GC models, respectively.

NMM-DN is based on BP neural network to classify music genres, which only considers the audio features of music, resulting in inefficient classification. DA-MGC extracts single modal features and uses the attention mechanism to assign weights to the features and performs important feature enhancement, which is used as an input to the RNN to achieve the classification of music genres, which fails to consider the multimodal features of music, resulting in poorer classification performance than the MAL-GC and MG-CAM. Although MAL-GC extracts and classifies the two modal features of music through a two-channel CNN, it does not amplify the key features and does not optimize the CNN, so the classification accuracy is lower than that of MG-CAM, which makes use of the convolutional attention mechanism to extract the features of the two modes of data, and takes into account the multiple characteristics of the audio data, so as to improve the classification performance.

Table 2. Performance comparison of different music genre classification models

Model	Accuracy (%)	Training time (s)	Classification time (s)
NMM-DN	78.1	8.02	7.35
DA-MGC	80.5	6.93	6.81
MAL-GC	84.7	4.75	3.65
MG-CAM	89.3	3.17	1.98

Comparison of the ROC curves of different music genre classification models is implied in Figure 5, where the curve is located in the coordinate graph, the horizontal axis is the False Positive Rate (FPR), and the vertical axis is the True Positive Rate (TPR). The AUC value represents the size of the area below the ROC curve, and the larger the AUC value is, the better the classification effect is. The AUC values of NMM-DN, DA-MGC, MAL-GC and MG-CAM are 0.729, 0.794, 0.892 and 0.936, respectively, and it can be concluded that MG-CAM is the best in the task of categorizing the four music genres, with an AUC value closest to 1. The classification ability of NMM-DN and DA-MGC is weaker than that of MAL-GC and MG-CAM. Overall, the values of the ROC curves of

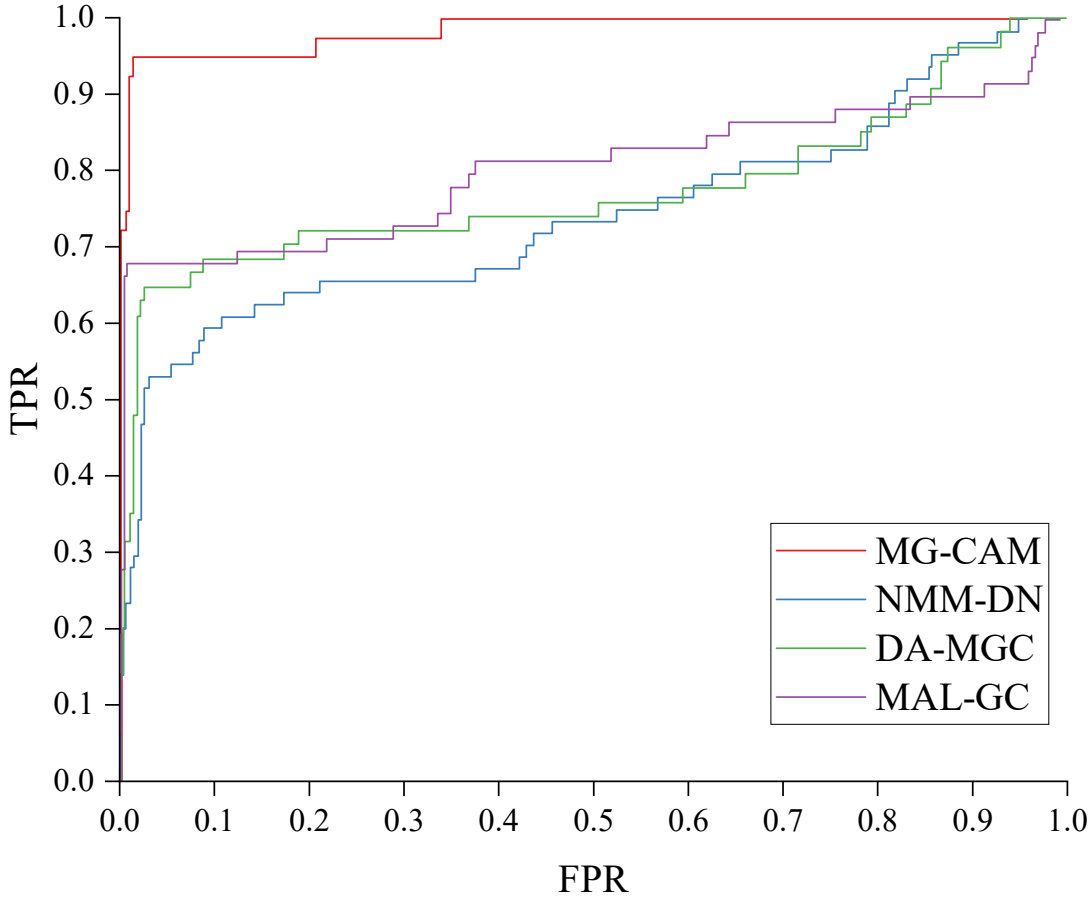

Figure 5. Comparison of ROC curves for different models

each model are above the invalid ROC curves, which proves the validity and some degree of superiority of the various models.

6. **Conclusion.** In the field of music information retrieval (MIR), classification based on music genres is a challenging task. Existing music genre classification methods suffer from the problems of partial data not suitable for the model and the difficulty of global feature extraction. For this reason, this paper fuses two modalities, music spectrogram and audio feature time-series data, and proposes a bimodal music genre classification model based on the convolutional attention mechanism. The CNN is first optimized using residual structure and Leaky ReLU to enhance the extraction of highly abstracted genre features. Based on this, the sound and spectrum of the music are preprocessed, and a convolutional attention mechanism is used to extract the link between the local features of the sound spectrogram and the audio timing data before and after. The feature data of the two modalities are fused using the mutual attention mechanism and fed into a Softplus classifier to output the categories of music genres. Comparative experiments were conducted based on GTZAN dataset, and the results indicate that the suggested model has the classification accuracy and AUC value of 89.3

REFERENCES

[1] J.-J. Aucouturier, and F. Pachet, "Representing musical genre: A state of the art," *Journal of New Music Research*, vol. 32, no. 1, pp. 83–93, 2003.

[2] J. Abeßer, H. Lukashevich, and P. Bräuer, "Classification of music genres based on repetitive basslines," *Journal of New Music Research*, vol. 41, no. 3, pp. 239–257, 2012.

[3] N. Pelchat, and C. M. Gelowitz, "Neural network music genre classification," *Canadian Journal of Electrical and Computer Engineering*, vol. 43, no. 3, pp. 170–173, 2020.
[4] H. C. Ceylan, N. Hardalaç, A. C. Kara, and H. Firat, "Automatic music genre classification and its relation with music education," *World Journal of Education*, vol. 11, no. 2, pp. 36–45, 2021.
[5] A. Nasridinov, and Y.-H. Park, "A study on music genre recognition and classification techniques," *International Journal of Multimedia and Ubiquitous Engineering*, vol. 9, no. 4, pp. 31–42, 2014.
[6] A. Meng, P. Ahrendt, J. Larsen, and L. K. Hansen, "Temporal feature integration for music genre classification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 5, pp. 1654–1664, 2007.
[7] B. K. Baniya, and J. Lee, "Importance of audio feature reduction in automatic music genre classification," *Multimedia Tools and Applications*, vol. 75, pp. 3013–3026, 2016.
[8] A. B. Mutiara, R. Refianti, and N. R. Mukarromah, "Musical Genre Classification Using SVM and Audio Features," *Telecommunication Computing Electronics and Control*, vol. 14, no. 3, pp. 1024–1034, 2016.
[9] L. Nanni, Y. M. Costa, A. Lumini, M. Y. Kim, and S. R. Baek, "Combining visual and acoustic features for music genre classification," *Expert Systems with Applications*, vol. 45, pp. 108–117, 2016.
[10] J. Ramírez, and M. J. Flores, "Machine learning for music genre: multifaceted review and experimentation with audioset," *Journal of Intelligent Information Systems*, vol. 55, no. 3, pp. 469–499, 2020.
[11] C. Liu, L. Feng, G. Liu, H. Wang, and S. Liu, "Bottom-up broadcast neural network for music genre classification," *Multimedia Tools and Applications*, vol. 80, pp. 7313–7331, 2021.
[12] S. A. Patil, G. Pradeepini, and T. R. Komati, "Novel mathematical model for the classification of music and rhythmic genre using deep neural network," *Journal of Big Data*, vol. 10, no. 1, 108, 2023.
[13] W. Hongdan, S. SalmiJamali, C. Zhengping, S. Qiaojuan, and R. Le, "An intelligent music genre analysis using feature extraction and classification using deep learning techniques," *Computers and Electrical Engineering*, vol. 100, 107978, 2022.
[14] Y. Yu, S. Luo, S. Liu, H. Qiao, Y. Liu, and L. Feng, "Deep attention based music genre classification," *Neurocomputing*, vol. 372, pp. 84–91, 2020.
[15] Y. Singh, and A. Biswas, "Robustness of musical features on deep learning models for music genre classification," *Expert Systems with Applications*, vol. 199, 116879, 2022.
[16] S. Reeja, and I. Yaseen, "Intelligent Music Genre Classification using CNN," *International Journal of Science Technology & Engineering*, vol. 8, no. 10, pp. 1–12, 2022.
[17] S. Oramas, F. Barbieri, O. Nieto Caballero, and X. Serra, "Multimodal deep learning for music genre classification," *Transactions of the International Society for Music Information Retrieval*, vol. 1, no. 1, pp. 4–21, 2018.
[18] R. Yang, L. Feng, H. Wang, J. Yao, and S. Luo, "Parallel recurrent convolutional neural networks-based music genre classification method for mobile devices," *IEEE Access*, vol. 8, pp. 19629–19637, 2020.
[19] R. B. Mangolin, R. M. Pereira, A. S. Britto Jr, C. N. Silla Jr, V. D. Feltrim, D. Bertolini, and Y. M. Costa, "A multimodal approach for multi-label movie genre classification," *Multimedia Tools and Applications*, vol. 81, no. 14, pp. 19071–19096, 2022.
[20] T.-Y. Wu, H. Li, S. Kumari, and C.-M. Chen, "A Spectral Convolutional Neural Network Model Based on Adaptive Fick's Law for Hyperspectral Image Classification," *Computers, Materials & Continua*, vol. 79, no. 1, pp. 19–46, 2024.
[21] F. Zhang, T.-Y. Wu, J.-S. Pan, G. Ding, and Z. Li, "Human motion recognition based on SVM in VR art media interaction environment," *Human-centric Computing and Information Sciences*, vol. 9, no. 1, 2019.
[22] Y. Ma, Y. Peng, and T.-Y. Wu, "Transfer learning model for false positive reduction in lymph node detection via sparse coding and deep learning," *Journal of Intelligent & Fuzzy Systems*, vol. 43, no. 2, pp. 2121–2133, 2022.
[23] P. Kumar, and A. Shankar Hati, "Convolutional neural network with batch normalisation for fault detection in squirrel cage induction motor," *IET Electric Power Applications*, vol. 15, no. 1, pp. 39–50, 2021.
[24] M. Qin, F. Xie, W. Li, Z. Shi, and H. Zhang, "Dehazing for multispectral remote sensing images based on a convolutional neural network with the residual architecture," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 5, pp. 1645–1655, 2018.

[25] W. Liu, X. Liu, and X. Chen, "An inexact augmented lagrangian algorithm for training leaky ReLU neural network with group sparsity," *Journal of Machine Learning Research*, vol. 24, no. 212, pp. 1–43, 2023.

[26] C. Van Hinsbergen, A. Hegyi, J. Van Lint, and H. Van Zuylen, "Bayesian neural networks for the prediction of stochastic travel times in urban networks," *IET Intelligent Transport Systems*, vol. 5, no. 4, pp. 259–265, 2011.

[27] Y. Chen, Y. Mai, J. Xiao, and L. Zhang, "Improving the antinoise ability of DNNs via a bio-inspired noise adaptive activation function rand softplus," *Neural Computation*, vol. 31, no. 6, pp. 1215–1233, 2019.

[28] X. Cai, and H. Zhang, "Music genre classification based on auditory image, spectral and acoustic features," *Multimedia Systems*, vol. 28, no. 3, pp. 779–791, 2022.

[29] S. H. Park, J. M. Goo, and C.-H. Jo, "Receiver operating characteristic (ROC) curve: practical review for radiologists," *Korean Journal of Radiology*, vol. 5, no. 1, pp. 11–18, 2004.