Hindawi
Scientific Programming
Volume 2022, Article ID 5117546, 11 pages
https://doi.org/10.1155/2022/5117546

Research Article

Multiple Musical Instrument Signal Recognition Based on Convolutional Neural Network

Lei Lei

Minjiang University CAI Jikun Conservatory of Music, Fuzhou 350108, China

Correspondence should be addressed to Lei Lei; 1989@mju.edu.cn

Received 25 November 2021; Revised 10 January 2022; Accepted 8 February 2022; Published 25 March 2022

Academic Editor: Hangjun Che

To improve the accuracy of multi-instrument recognition, based on the basic principles and structure of CNN, a multipitch instrument recognition method based on the convolutional neural network (CNN) is proposed. First of all, the pitch feature detection technology and constant Q transform (CQT) are adopted to extract the signal characteristics of multiple instruments, which are used as the input of the CNN network. Moreover, in order to improve the accuracy of multi-instrument signal recognition, the benchmark recognition model and two-level recognition model are constructed. Finally, the above models are verified by experiments. The results show that the two-level classification model established in this article can accurately identify and classify various musical instruments, and the recognition accuracy is improved most obviously in xylophone. Compared with the benchmark model, the constructed two-level recognition has the highest accuracy and precision, which shows that this model has superior performance and can improve the accuracy of multi-instrument recognition.

With the rise of artificial intelligence technologies such as deep learning and the growth of massive music data, content-based music retrieval has become an urgent issue at present. In content-based music retrieval, how to identify music has become the focus of current music information retrieval research. Compared with the traditional speech signal, speech signal should have a richer spectrum, treble, and timbre. Therefore, based on the above characteristics, the recognition of music signal can be divided into recognition method, recognition accuracy, recognition time, and recognition scene. María and ValeroMas Jose applied the convolutional recursive neural network to music recognition, which greatly reduces the precision of musical note and number recognition [1]. Agarwal and Om applied the machine learning algorithm to music recognition and obtained the highest recognition accuracy by the improved method through the music emotion recognition of the ISMIR2012 dataset, NJU_V1 dataset, and self-built dataset [2]. Sarkar applied the deep learning algorithm to the recognition of music and audio by extracting MFCC features and finally using VGGNet for recognition. The results show that the method has obvious advantages in three datasets [3]. Yan uses the genetic algorithm to improve the T-S cognitive neural network and applies the model to music recognition for higher accuracy and robustness [4]. Liang used machine learning algorithms to build prediction models among audio features, individuals, and emotions, so as to propose suggestions on emotional influence in music [5]. Wang and others accurately identified different emotions including happiness, anger, sadness, and fear by establishing CLDNN's musical instrument emotion recognition model [6]; ATILA Orhan proposed a speech emotion recognition model based on 3D CNN-LSTM and evaluated speech from the perspectives of accuracy, sensitivity, specificity, and F1, which provided a reference for speech evaluation [7]. Solanki et al. also use the convolutional neural network to recognize musical instruments, but mainly focus on extracting the characteristic parameters of musical instruments [8–10]. As can be seen, the above research provides reference for the music retrieval and identification. However, the above research is mainly aimed at the musical identification of a single instrument. At present, there are relatively few references for the music identification of multiple instruments. In multi-instrument recognition of polyphony, not only the traditional single tone signal must be extracted, but also the tones of different instruments must be identified. Therefore, based on the reality of research, a convolutional neural network is used to identify the multi-instrument music signal. Therefore, this article attempts to identify the signals of different musical instruments by extracting and identifying the characteristics of musical instruments on the basis of traditional single instrument recognition.

1. Introduction to Convolutional Neural Network

CNN, a representative algorithm of deep learning, is a kind of feed-forward neural network, which includes convolution calculation and has depth structure [11]. CNN can learn the original data efficiently and quickly, so as to extract the features of the data, which means that it has the ability of representation learning. The specific structure is shown in Figure 1, which is mainly divided into five network layers and belongs to multilayer perceptron (MLP) [12]. The most important steps are convolution calculation and pooling operation.

1.1. Convolutional Layer. The convolution formula is as follows:

$$s(t) = x(t) * w(t) = \sum_{\tau=-\infty}^{\tau=+\infty} x(\tau) w(t-\tau), \tag{1}$$

where $s(t)$, $x(t)$, and $w(t)$ represent feature mapping, input features, and convolution cores, respectively. If it is two-dimensional matrices, it can be represented as

$$s(i,j) = \sum_{m=0}^{M} \sum_{n=0}^{N} \left(w_{m,n} x_{i+m} + w_b \right). \tag{2}$$

In the above formula, the size of convolution kernel is $M \times N$, which is shown in Figure 2 [13–15]. The advantages of convolution operation are mainly reflected in three aspects. Firstly, the realization of parameter sharing helps to reduce the size of the parameter set. Secondly, sparse connection reduces the number of parameters and improves the efficiency, which has certain advantages over full connection. Thirdly, because the same convolution kernel is used, when the value of input eigenmatrix changes, the corresponding result will change at the same position.

1.2. Pooling Layer. The pooling layer refers to the output of statistics for a specific region within the input eigenmatrix. Generally, two pooling methods can be adopted, namely, average pooling and maximum pooling. They take the mean and maximum values of the local region as the output, respectively. Except for these two methods, there is a random pooling, which selects neurons with greater probability values.

2. The Construction of Multi-Instrument Signal Recognition Model Based on Convolutional Neural Network

2.1. Multi-Instrument Signal Feature Extraction. In order to realize the recognition of multi-instrument signals, it is necessary to extract musical instrument signal features first. Conventional instrument signal extraction is usually only for a single instrument, which is relatively simple. It only needs to eliminate the instrument noise and then classify them. But for multi-instrument signals, it not only needs to deal with the noise, but also faces the knowledge of notes of different instruments. In other words, the conventional time-frequency feature extraction, such as MFCC, may not achieve the recognition effect. Therefore, on the basis of signal processing, the essential elements of music, such as pitch, harmony, and other signals, are combined to identify the signals of multiple instruments. The instrument signal characteristics are processed by pitch characteristic detection and constant Q transformation.

2.1.1. Pitch Feature Extraction. The multipitch detection based on a statistical model and spectral decomposition is the main method to extract pitch features. However, considering that an end-to-end neural network may have the problem of overfitting in the display feature extraction, filters are introduced to extract the time-frequency features of musical instrument signals in the primary feature extraction process of convolutional neural network. In other words, the first layer of the convolutional neural network is replaced by the filter, which can greatly reduce the overfitting problem. The specific extraction process is shown in Figure 3 [16–18].

The specific process is as follows:

Firstly, the audio frame X is normalized, that is, $X \longrightarrow X/\|X\|^2$. The audible variables of each frame are standardized. Then, it is divided into Tp segments, and each segment is represented as x_t. The number of sampling points is s, which means $x_t = (x_{t_1}, x_{t_2}, \ldots, x_{t_i})$. To map X, it needs to use the filter banks in the log-frequency domain, including cosine and sine filters, and the total number is $n_p = 511$. Thus, the logarithmic frequency-time matrix $(n_p \times T_p)$ can be formed, and the log-frequency domain is $\log f_L$ to $\log f_H$. Then, the parameter of sine filter i is shown as follows [19]:

$$w_{i,\sin} = \left(\sin 2\pi f_i t_1, \ldots, \sin 2\pi f_i t_s \right). \tag{3}$$

The parameters of cosine filter i are shown as follows:

$$w_{i,\cos} = \left(\cos 2\pi f_i t_1, \ldots, \cos 2\pi f_i t_s \right). \tag{4}$$

In the above formula, $f_i = 10^{\log f_L + i(\log f_H - \log f_L/n)}$; according to the normalized amplitude $x_t = (x_{t_1}, x_{t_2}, \ldots, x_{t_i})$, the position $t_1, t_2, .., t_s$ at each time can be determined.

Then, let x_t do the inner product calculation with $w_{i,\sin}$ and $w_{i,\cos}$. Next, the square and the sum of the two can be calculated. So the output of filter i can be obtained as follows [20–22]:

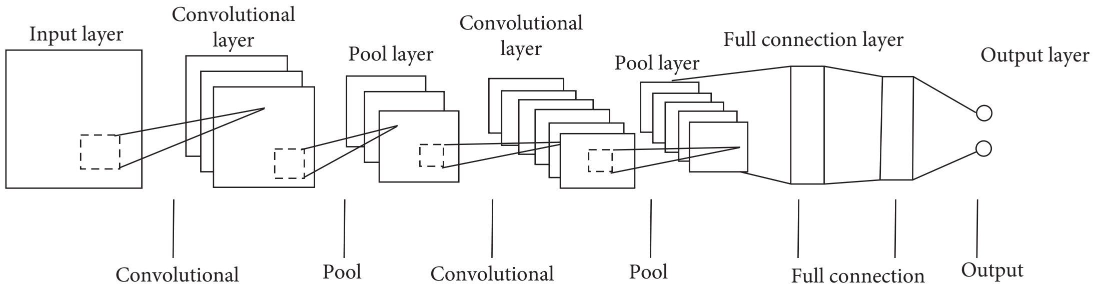


Figure 1: CNN neural network structure.

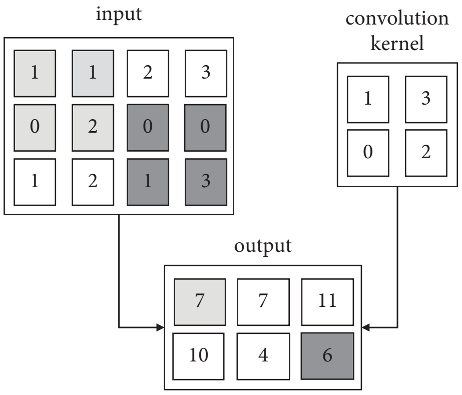


Figure 2: Example of convolution operation.

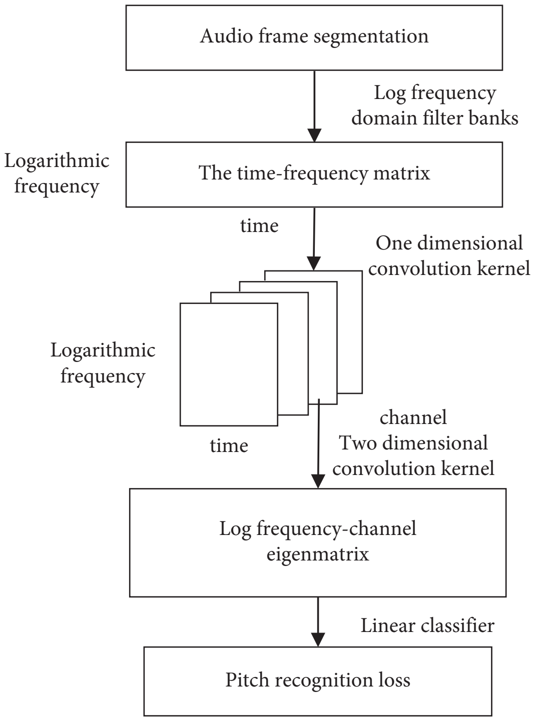

Audio frame segmentation
Log frequency domain filter banks
Logarithmic frequency
The time-frequency matrix
time
One dimensional convolution kernel
Logarithmic frequency
time
channel
Two dimensional convolution kernel
Log frequency-channel eigenmatrix
Linear classifier
Pitch recognition loss

Figure 3: Pitch feature extraction process.

$$\text{filter}_i = \left(w_{i,\sin}^T x_t\right)^2 + \left(w_{i,\cos}^T x_t\right)^2. \quad (5)$$

If the number of filters is n_p, the corresponding outputs are $\text{filter}_1, \ldots, \text{filter}_{n_p}$. Dividing the audio frames into segments, if the number of segments is T_p, the logarithmic frequency-time matrix is $n_p \times T_p$.

Secondly, processing the above matrix, the tensor of logarithmic frequency-time-channel can be obtained after convolution. The first layer of the convolutional network is the matrix obtained in the first step. However, setting up the mapping of the second layer is to do the convolution computation for the logarithmic frequency axis. The step size is set to 3, and the convolution kernel 128 × 1 is selected. The matrix after convolution is mapped to channels, and the tensor of $128 \times T_p \times C_i$ is obtained, where C_1 represents the number of channels.

Thirdly, continue the two-dimensional convolution for the tensor obtained in the previous step, so that the logarithmic frequency-channel matrix can be obtained. Mapping to the third layer with the same method, the height of convolution kernel $(T_p \times C_2)$ is 1; thus, the matrix of $128 \times C_2$ can be obtained, where C_2 represents the number of channels.

The full connection processing of the matrix obtained in the previous step is performed, and the corresponding pitch recognition vector can be obtained. It is necessary to connect the lines of the matrix in the previous step with the linear classifier. The number of pitch frequency is m_1. If the number of valid elements is the same, the vector is 1. If the number is different, the vector is 0.

After the frame segmentation is completed, each audio frame is processed based on the above process. The corresponding pitch feature matrix can be obtained. If the pitch frequency in the pitch set is M_p, and the number of frame is N_p, so the corresponding matrix size is expressed as $M_p \times N_p$.

2.1.2. Constant Q Transform. In order to better display the pitch frequency on the spectrum space of DFT or STFT, the constant Q transform (CQT) is adopted in this study to transform the time-frequency of music signal analysis. The specific steps are as follows [23–25]:

(1) Find the spectral kernel matrix corresponding to the octave with the highest frequency.

(2) The corresponding CQT frequency band of the input signal $x(n)$ is calculated by the DFT transform vector, and the input signal $x(n)$ is marked as $x_0(n)$.

(3) Sample the signal.

(4) Calculate the CQT frequency band by the corresponding DFT transform vector of the next octave.

(5) Repeat steps (3) and (4) until the calculation is complete, as shown in Figure 4.

In Figure 4, $G(f)$ represents the low-pass filter, and $\downarrow 2$ represents downsampling with a downsampling factor of 2. Here, the downsampling of $x_d(n)$ is $f_s/2d$ $(d \geq 1)$, and the CQT transform X_d^{CQ} of each octave is

$$X_d^{CQ} = A^* X_d, \tag{6}$$

where A^* represents the conjugate transpose of the complex numerical spectrum kernel matrix, which is usually used to calculate the CQT of octaves.

2.2. Construction of Multi-Instrument Signal Recognition Model

2.2.1. Construction of Benchmark Model. First of all, the benchmark model needs to be established, and then the modification and improvement can be achieved on this basis. In this article, combining the convolutional network model designed by liu and Yang, the model can realize automatic music labeling. The data used in this training have a frame-level accuracy label, which is used as a supervisory signal.

This model is divided into multiple layers, including the batch standardization layer, convolution layer, pooling layer, etc., and the specific structure is shown in Figure 5 [26].

Due to the problem of internal covariable offset in the training process of the convolutional network, the batch standardization layer can be used to deal with it. It is necessary to ensure the consistency in the distribution of training and test data, and it is helpful to improve generalization ability. However, when there are many parameters and the number of network layers increases, the data distribution will change after the parameter update. At this time, the difficulty of training will increase. To solve the above problems, the batch standardization can be adopted to adjust the data distribution. It makes the intermediate characteristic data become normal distribution, which is realized by processing the input or output data of the intermediate hidden layer. This is a standardized processing procedures, and its formula is shown as follows [27]:

$$a_i^n = \gamma_i \times \frac{a_i - \mu}{\sigma} + \beta_i. \tag{7}$$

Here, β_i and γ_i represent translation and zoom factor, respectively; a_1 represents initial activation value; μ and σ are as follows, respectively:

$$\begin{aligned} \mu &= \frac{1}{m} \sum_{k=1}^{m} a_k, \quad k \in S, \|S\| = m, \\ \sigma &= \sqrt{\frac{1}{m} \sum_{k=1}^{m} (a_k - \mu)^2 + \varepsilon}, \quad k \in S, \|S\| = m. \end{aligned} \tag{8}$$

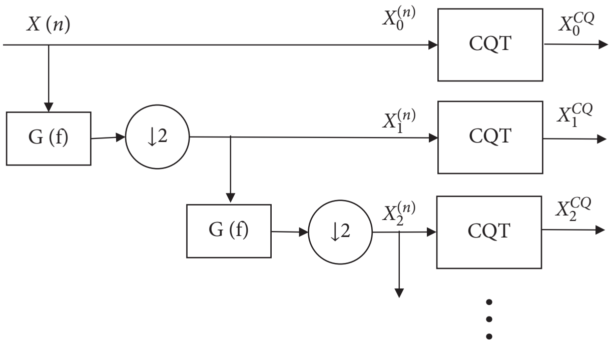


Figure 4: The method of CQT calculation for each octave.

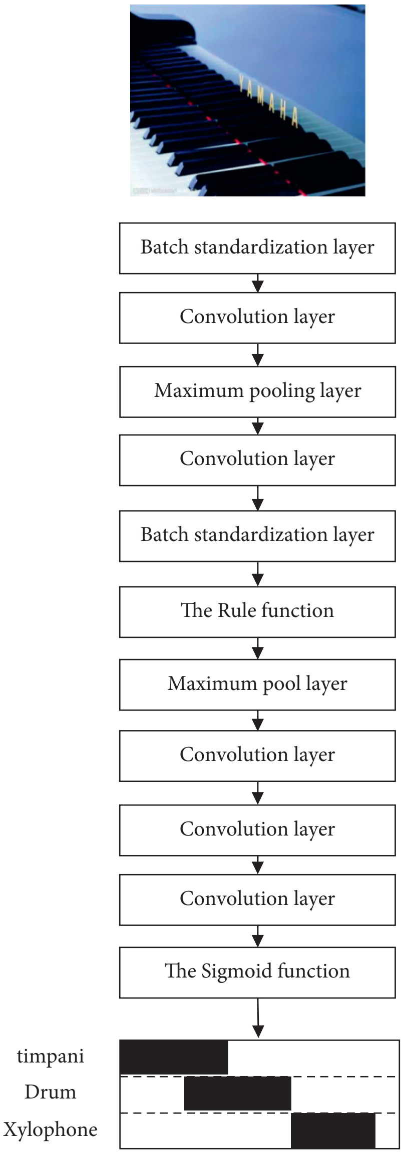


Figure 5: Network structure of the benchmark model.

In the formulas, S represents the neuron set with a size of M, but it has different meanings for different networks. If it is a convolutional network, m actually represents the total number of all activation values formed based on the convolutional kernel channel. For a fully connected network, it represents the number of activation values formed by all instances in a particular batch. ε represents the constants related to training stability.

Batch standardization is a key process. The problem of gradient explosion can be solved by adjusting data distribution, because the noise caused by scaling and other operations can help to get the parameters with higher generalization performance. In addition, to improve the efficiency of convergence, a relatively large learning step can be set.

Except for the above layers, the function of the convolution layer is to extract the required intermediate features. The maximum pooling layer is an important part, which can realize the function of compression features. And it can reduce the difficulty of calculation. Specifically, the maximum value dividing the pooling area is taken as the output value. ReLu function is adopted in the middle, and the specific form is as follows [28].

$$f(x) = \begin{cases} x, & x > 0, \\ 0, & x \leq 0, \end{cases} \tag{9}$$

where some activation values of output are equal to zero, which makes the network sparse. Compared to the Sigmoid function, a higher convergence speed can be achieved. The output layer is mainly applied, and its form is shown as follows [29]:

$$f(x) = \frac{1}{1 + \exp(-x)}. \tag{10}$$

Based on the function, the normalization has been achieved. The output value of the instrument recognition model is placed in the range of 0–1, namely, the existence probability of various musical instruments. Then, the binarization method can be adopted to determine instruments' existing situation. This process depends on the proper threshold. If the threshold is set to 0.5, there is no guarantee for good performance. Therefore, a kind of threshold selection algorithm is designed. It means that the method of maximizing the F1 score of the training set is used to set the threshold value. There are 99 candidates' threshold values, which are 0.1, 0.2, . . ., 0.99, respectively.

The loss function adopted in the training is binary cross-entropy, and the specific form is shown as follows [30]:

$$l = -\sum_{k=1}^{11} \widehat{y}_k \log y_k + (1 - \widehat{y}_k)\log(1 - y_k). \tag{11}$$

Here, k represents the specific musical instrument category, and y_k and $\widehat{y}_k$ represent the identification of each time frame and real label. Considering the imbalance of categories, a certain weight is set for each category, which is expressed as ω_k. The specific form is shown as follows [30]:

$$\omega_k = \left(\frac{\overline{p}}{p_k} \times \frac{1 - p_k}{1 - \overline{p}}\right)^{\eta}. \tag{12}$$

In the formula, η represents the hyperparameter, which is generally valued at 0.3; $\overline{p}$ represents the mean value of all of p_k; and p_k represents the proportion occupied by category k. l_{ban} is expressed as follows:

$$l_{\text{ban}} = -\sum_{k=1}^{11} \omega_k \left[\widehat{y}_k \log y_k + (1 - \widehat{y}_k)\log(1 - y_k)\right]. \tag{13}$$

According to the above analysis, the weights need to be set in conjunction with the proportion size occupied by a specific categories of instruments. For example, when the proportion occupied by a specific categories of instruments is low, a higher weight should be set to improve the accuracy of instrument recognition results. So when the frequency of occurrence is not high, the effective recognition even can be ensured. The momentum algorithm is adopted in the calculation, in which weight attenuation factor, learning rate, and batch are 2×10^{-4}, 0.01, and 80, respectively.

2.2.2. Multi-Instrument Signal Recognition Based on Two-Level Classification. When multiple instruments are played at the same time, the traditional classification model based on the attention network has a poor recognition effect on harmonic instruments. The main reason is the category imbalance, which means that the difference in the proportion of different categories interferes with the learning of model parameters. Therefore, combined with the basic principle of undersampling or oversampling, the two-level classification model is proposed. This model is mainly divided into the first-level and the second-level convolutional neural network classification models.

The first-level classification model takes the constant Q transform matrix as the input feature. Firstly, the instrument families in audio signals are rough classified. The constant Q transform matrix reflects the time-frequency energy distribution of audio signals, and it can be used as an effective feature of rough classification.

The second-level classification model is composed of three residual network models with the same architecture. Each residual network model is specially trained to identify various instruments under a certain musical instrument family. There is a special network model for each of the three musical instrument families. The specific process of the two-level classification model is as follows [30, 31].

Figure 6 shows the network architecture of first-level classification model. From the top to the bottom, there are batch standardization layer, convolution layer, batch standardization layer, convolution layer, convolution layer, batch standardization layer, ReLu layer, maximum pooling layer, convolution layer, and Sigmoid layer.

Figure 7 shows the residual network model architecture of three same structures in the second-level classification model. From the top to the bottom, there are batch standardization layer, convolution layer, residual block, maximum pooling layer, residual block, convolution layer, maximum pooling layer, batch standardization layer, ReLu layer, convolution layer, and Sigmoid layer.

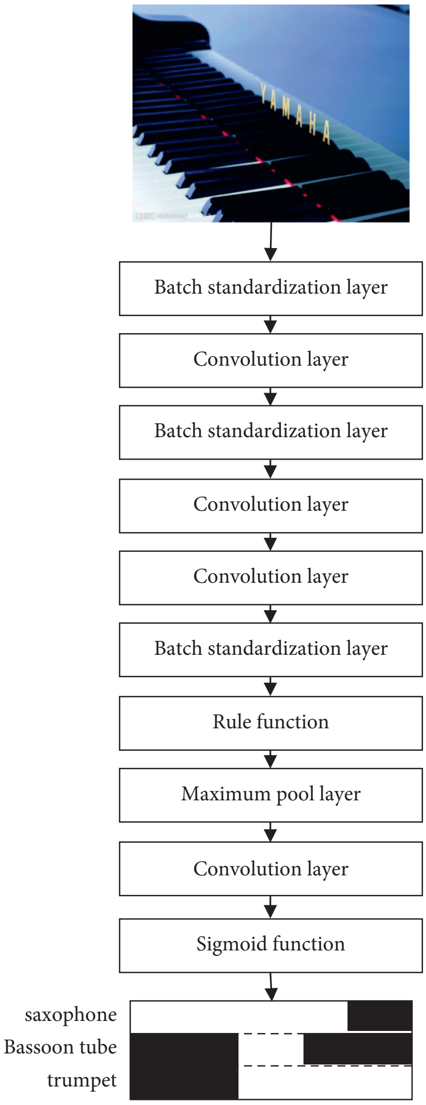


Figure 6: The network structure of the first-level classification model in the two-level classification model.

The residual block is divided into two parts. The first part includes batch standardization layer, convolution layer, batch standardization layer, ReLu layer, convolution layer, and batch standardization layer. And the second part is a convolution layer. The input of the residual block enters these two parts to obtain the output, and the output of two parts is summed as the final output of the residual block.

The residual structure of the residual network can effectively solve the problem of gradient disappearance in deep network. In this article, it is applied to detect and recognize the time-frequency characteristic spectrum, and the intermediate feature extraction of musical instrument recognition is realized successively.

The loss function is expressed as

$$\varsigma'\mathrm{ban} = -\sum_{k=1}^{K} \omega_k \left[\hat{y}_k \log y_k + (1 - \hat{y}_k)\log(1 - y_k)\right]. \qquad (14)$$

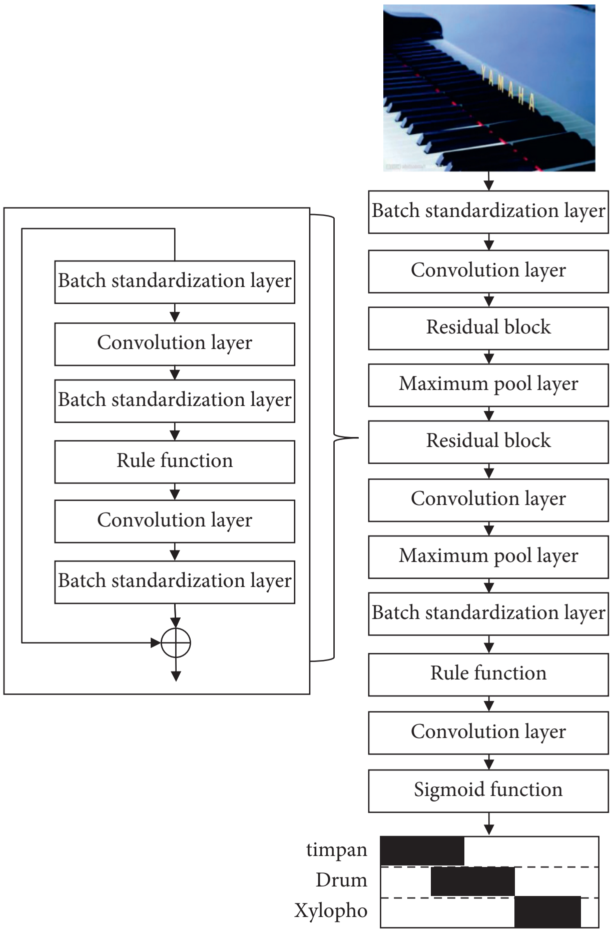


Figure 7: Structure of three residual network models with the same structure in the second-level classification model.

In the first-level classification model, the K of equation (14) is valued at 3, representing the three musical instrument family categories of string, wind, and percussion. In the second-level classification model, the K in the string music classification network is valued at 5, representing piano, violin, viola, guitar, and bass. In the wind music classification network, the K is valued at 3, representing saxophone, bassoon, and trumpet. The value of K in the percussion classification network is valued at 3, representing timpani, small drum, and xylophone.

3. Simulation Verification

3.1. Experimental Environment. In order to achieve better experimental results, Intel i7-7800X CPU is selected as the hardware system in this experiment. The main frequency is 3.5 GHz, and the farce frequency is 4.0 GHz. It uses 6 cores and 12 threads. The memory is 16 GB, and the graphics card is an NVIDIA GTX 2080 dual-channel GPU.

Software system: Ubuntu 16.04, 64 bit operating system, Anaconda3-4.4.0, deep learning framework PyTorch0.4.1, and acceleration module CUDA 10.0.

3.2. Dataset Sources. At present, the commonly used dataset includes Bach10 dataset, MedleyDB dataset, and MIXING SECRETS dataset. Among them, Bach10 dataset includes ten large choral works by J.S.Bach, each of which contains four monophonic parts. The audio recordings of each monophonic part are performed by violin, clarinet, saxophone, and bassoon; the MedleyDB dataset consists of 122 songs, in which 108 songs are vocal and instrumental melodies; the MIXING SECRETS dataset contains 258 multitrack audio songs, and there are a variety of music genres involved. However, the scale of the above three public datasets is still not large, and there is no note label in the MIXING SECRETS dataset, only instrument label. Moreover, there are 14 songs in the Medley DB dataset and no note label. In order to solve the above problems, the label annotation information of the MIDI score in an open-source music platform is aligned to the original audio by means of self-built dataset, and then, it is manually calibrated by people with professional music background. Finally, there are 307 useable extended datasets obtained, including various musical instruments and music types, and every frame has the annotation label.

3.3. Processing of Pitch Feature Matrix. Musical instruments of the same family have certain similarities in pitch range. In order to better identify, the energy ratio of harmonics needs to be considered. Therefore, based on the extracted pitch feature matrix, lines 7–94 of the extracted matrix are expressed as Y_1. According to available information, the matrix is sparse. Then, the fundamental frequency position value within Y_1 is moved to 12 grids $2^{(12/12)} = 2$ distance from it. Thus, the matrix Y_2 is formed. Using the same way to move to 19 grid $2^{(19/12)} = 3$, 24 grid $2^{(24/12)} = 4$, 28 grid $2^{(28/12)} = 5$, and 31 grid $2^{(31/12)} = 6$ in turn, the matrix is respectively represented as Y_3, Y_4, Y_5, and Y_6.

The basic form of harmonic sequence matrix is as follows:

$$S_n = Y_1 + Y_2 + \ldots + Y_n. \quad (15)$$

The matrix actually represents a sequence combination of fundamental frequencies and corresponding harmonics. Therefore, when determining the meaning of Y_6, S_1–S_6 matrices can be obtained, which are the input feature of the CNN.

3.4. Experimental Results and Analysis

3.4.1. Multi-Instrument Recognition Results under Benchmark Model. There are ten kinds of musical instruments, which are divided into three categories: percussion instruments, string instruments, and wind instruments. The percussion instruments are xylophone, timpani, trumpet, and the string instruments are guitar, piano, bass, viola, and violin. In addition, the wind instruments are bassoon and saxophone. In the experiment, an appropriate experimental environment should be configured first, which is basically consistent with the previous pitch feature extraction experiment. The dataset is divided into two parts, namely, training set and test set. The ratio of the two parts is 9 : 1. Moreover, the possibility of the inexistence and unlabeled instruments in the training set should be considered.

The extracted constant Q transform matrix X^{CQ} (88 × 165) is processed, which is spliced with S_1–S_6. The corresponding input matrices can be obtained, which can be represented as I_1–I_6, namely, the harmonic mapping matrix. Then, they are input into the model, and the corresponding class-time series matrix can be obtained. The feature changes in the benchmark model are shown in Table 1.

The experimental results are obtained according to Table 2. The instrument-type recognition results are evaluated through F1. In the total number of instrument recognition, there are three cases of unrecognized, misrecognized, and correct recognition. It can be seen that the overall accuracy can identify the proportion of accuracy times.

The real pitch label matrix is adopted to design the harmonic mapping matrix I_n^g (n is 1–6). At this time, the values in Y_1 are all accurate. According to the information in Tables 2 and 3, it can be clearly seen that compared with the estimated pitch labels, the harmonic mapping matrix obtained by using real labels can achieve higher overall accuracy and F1 value. In addition, compared with xylophone and timpani, the recognition results of estimated and real pitch of a small drum are basically the same, which is mainly related to the unfixed pitch. In this study, the pitch features are mainly used to recognize the musical instruments. So the recognition scores of different types of musical instruments are different. Compared with percussion instruments, the recognition scores of orchestral instruments are higher, which verifies the effectiveness of pitch feature extraction.

3.4.2. Multi-Instrument Classification Results under Two-Level Recognition. Firstly, the configuration of the experimental environment is consistent with the previous section. The momentum algorithm (0.9) is adopted, where the weight attenuation factor, learning rate, and batch are $2 * 10^{-4}$, 0.05, and 60, respectively. In this experiment, a tensor is input, including I_1–I_6. Furthermore, the output result is class-time series matrix. The specific characteristic changes are shown in Table 4.

I_3 and I_5 as the benchmark model of input are represented as BI_3 and BI_5, respectively. The classification model is represented as MA. According to the information in Table 5, it can be seen that compared with the previously adopted benchmark model, a higher overall accuracy is achieved by adopting the attention network model, and the recognition scores of all instruments except xylophone are improved.

Based on the analysis of the above phenomenon, it is found that the attention network actually is to set the appropriate weights for the intermediate feature graph. Compared with the other types of musical instruments, the melodic instrument features are more conducive to recognition, which means the weights are higher. The common melodic instruments are piano and guitar, and the xylophone is rarely used as a melodic instrument. Therefore, if it exists at the same time with other instruments, the

Table 1: Change process of the feature size in the benchmark model.

Input size	Operation	Output size
176 × 165 × 1	2 × 1 Convolution kernel, 352 channels	352 × 164 × 1
352 × 164 × 1	3 × 1 Maximum pooling	352 × 54 × 1
352 × 54 × 1	3 × 1 Convolution kernel, 704 channels	704 × 52 × 1
704 × 52 × 1	3 × 1 Channels	704 × 17 × 1
704 × 17 × 1	2 × 1 Channels, 704 channels	704 × 8 × 1
704 × 8 × 1	1 × 1 Channels, 11 channels	11 × 8

Table 2: F1 and overall accuracy of ten instruments under the benchmark model (using estimated pitch).

Harmonic mapping matrix order	Piano	Violin	Viola	Guitar	Saxophone	Bassoon tube	Timpani	Xylophone	Bass	Trumpet	Overall accuracy
I_1	0.88	0.88	0.84	0.88	0.82	0.82	0.76	0.73	0.77	0.80	0.77
I_2	0.89	0.89	0.84	0.89	0.84	0.83	0.78	0.75	0.73	0.81	0.77
I_3	0.90	0.90	838.00	0 892	0.83	0 836	0.78	0.76	0.80	0.74	0.77
I_4	0.90	0.40	832.00	0.89	0.84	0.84	0.79	0.75	0.75	0.79	0.77
I_5	0.89	0.89	0.84	0.89	0.85	0.85	0.80	0.77	0.80	0.76	0.77
I_6	0.89	0.89	0.83	0.89	0.85	0.85	0.04	0.77	0.77	0.79	0.78

Table 3: F1 scores and overall accuracy of ten instruments under the benchmark model (using real pitch).

Harmonic mapping matrix order	Piano	Violin	Viola	Guitar	Saxophone	Bassoon tube	Timpani	Xylophone	Bass	Trumpet	Overall accuracy
I_1^g	0.90	0.90	0.85	0.89	0.84	0.83	0.77	0.74	0.77	0.83	0.80
I_2^g	0.91	0.91	0.86	0.89	0.85	0.83	0.80	0.76	0.75	0.85	0.80
I_3^g	0.91	0.91	0.85	0.91	0.85	0.84	0.79	0.77	0.76	0.84	0.80
I_4^g	0.91	0.91	0.85	0.90	0.85	0.86	0.81	0.78	0.77	0.85	0.81
I_5^g	0.91	0.91	0.84	0.90	0.86	0.87	0.82	0.79	0.80	0.81	0.80
I_6^g	0.91	0.91	0.85	0.90	0.86	0.87	0.77	0.78	0.79	0.82	0.80

Table 4: Changes of feature size in the classification model based on the attention network.

Input size	Operation	Output size
176 × 165 × 6	2 × 1 Convolution kernel, 352 channels	352 × 164 × 6
352 × 164 × 6	3 × 1 Maximum pooling	352 × 54 × 6
352 × 52 × 6	3 × 1 Convolution kernel, 704 channels	704 × 52 × 6
704 × 52 × 6	3 × 1 Channels	704 × 17 × 6
704 × 17 × 6	2 × 1 Channels, 11 channels	11 × 8 × 6
704 × 17 × 6	Attention subnet	Six attention weights
11 × 8 × 6	The sum using the weighting of attention weight	11 × 8

Table 5: F1 and overall accuracy and comparison of ten musical instruments based on the attention network classification model.

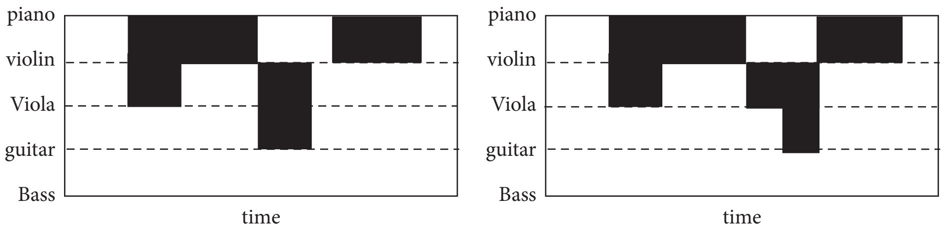
	Piano	Violin	Viola	Guitar	Saxophone	Bassoon tube	Timpani	Xylophone	Bass	Trumpet	Overall accuracy
BI_3	0.90	0.90	0.84	0.89	0.83	0.84	0.78	0.76	0.81	0.73	0.77
BI_5	0.89	0.89	0.84	0.89	0.85	0.85	0.80	0.77	0.76	0.88	0.77
MA	0.91	0.90	0.85	0.91	0.86	0.86	0.81	0.74	0.80	0.86	0.83

characteristics that need to be recognized can be easily masked. In this perspective, after adding the attention network, it is beneficial for melody instrument recognition, which is helpful to improve the overall accuracy. However, it is not possible to improve all instrument recognition scores, which needs to be further studied.

In this article, the constant Q transform matrix of the first-level classification model is used as input and output instrument family-time series matrix. In the second-level classification model, the third-order harmonic mapping matrix I_3, the fifth-order harmonic mapping matrix I_5, and the sixth-order harmonic mapping matrix I6 are used as the input features of the string music classification network, wind music classification network, and percussion music classification network, respectively. Then, the output of the three networks is summarized to obtain the final instrument class-time series matrix. The recognition scores and overall accuracy of various musical instruments in the two-level classification model (MT) are obtained, and the comparison is shown in Table 6.

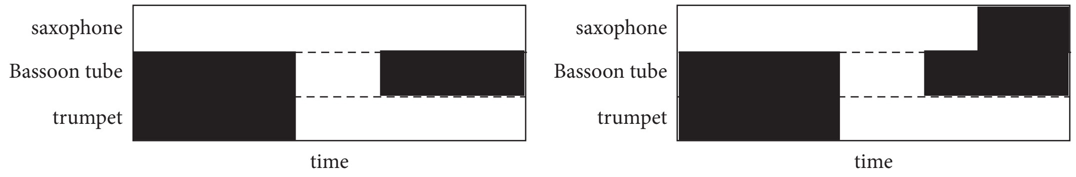
Table 6: Comparison of F1 and overall accuracy of ten musical instruments in the two-level classification model.

	Piano	Violin	Viola	Guitar	Saxophone	Bassoon tube	Timpani	Xylophone	Bass	Trumpet	Overall accuracy
BI_3	0.90	0.90	0.84	0.89	0.83	0.84	0.78	0.76	0.79	0.75	0.77
BI_5	0.89	0.89	0.84	0.89	0.85	0.85	0.80	0.77	0.71	0.83	0.77
MA	0.91	0.90	0.85	0.91	0.86	0.86	0.81	0.74	0.75	0.91	0.83
MT	**0.91**	**0.90**	**0.85**	**0.91**	**0.86**	**0.87**	**0.82**	**0.81**	**0.83**	**0.89**	**0.86**



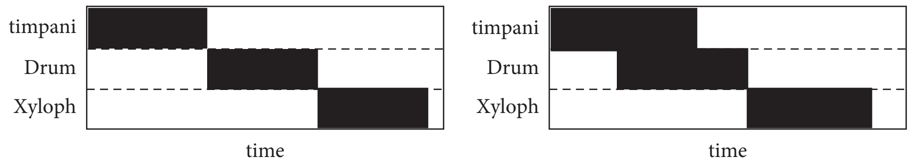
(a) (b)

Figure 8: Recognition effect of string music classification network in the second-level classification model.



(a) (b)

Figure 9: Identification effect of the wind music classification network in the second-level classification model.



(a) (b)

Figure 10: Recognition effect of the percussion classification network in the second-level classification model.

After comprehensively analyzing the charts, what can be found is that the recognition scores of most musical instruments are improved, especially xylophone. And it can be seen that the two-level classification model proposed in this article can balance the classification of musical instruments well, and the overall accuracy is further improved.

Figure 8 is the recognition effect diagram of the string music classification network. The upper part (a) represents the real label, the lower part (b) represents the recognition result, and the black part represents the existence of musical instruments.

As can be seen intuitively from the figure above, piano and violin can be accurately identified, while there is confusion in the viola, and the recognition accuracy needs to be improved.

Figure 9 is the identification effect diagram of the wind music classification network. As can be seen from the picture, the trumpet can be identified accurately, while there are confusions in the the other two instruments, and the recognition accuracy needs to be improved.

Figure 10 is the identification effect diagram of the percussion music classification network. It can be clearly seen from the figure that all three musical instruments have been accurately identified, which means that there are obvious differences between these three musical instruments, so that they can be well identified and classified.

The comprehensive analysis shows that the two-level classification model constructed in this article has the best comprehensive performance, and it has the more accurate recognition effect.

To further verify the effectiveness of the proposed method, the experiment compares the accuracy of the proposed model with that of the existing duets, trios, and quartet, and the obtained comparison results are shown in Table 7.

Table 7: Comparison between the proposed model and existing methods.

	Piano	Violin	Viola	Guitar	Saxophone	Bassoon tube	Timpani	Xylophone	Bass	Trumpet	Overall accuracy
MA	0.912	0.903	0.849	0.908	0.855	0.859	0.809	0.743	0.762	0.889	0.825
Bipolar classification model	**0.913**	**0.901**	**0.853**	**0.907**	**0.863**	**0.866**	**0.822**	**0.811**	**0.834**	**0.880**	**0.857**
Duet	0.827	0.881	—	0.829	—	—	—	—	—	—	0.841
Trio	0.683	0.825	—	0.828	—	—	—	—	—	—	0.778
Quartet	0.573	0.799	—	0.791	—	—	—	—	—	—	0.731

As can be seen from the above table, compared with the other three methods, the recognition accuracy of the two-level classification model proposed in this article is as high as 85.7%. The recognition accuracy of duet is 84.1%. There are 77.8% for trio, and there are 73.1% for quartet. The method proposed in this article is much higher than the other three methods, which shows that the method proposed in this article has a higher recognition accuracy and better performance.

4. Conclusion

In conclusion, the two-level classification model based on the convolutional neural network proposed in this article has a good classification effect and recognition accuracy. It has certain validity. Through comparative experiments, it is found that the recognition accuracy of the proposed method is 1.6%, 8.1%, and 13.4% higher than that of the method of duet, trio, and quartet. So the recognition accuracy and classification effect of the proposed method are better. The validity of the proposed classification model is further verified by comparing the benchmark classification model with the classification model based on the attention network. However, due to the lack of experience and adequate experimental conditions, the research needs to be further improved and perfected. Specifically, the original audio and scores of various musical instruments can be added to obtain more datasets, so as to further improve the experiment scientificity.

Data Availability

The experimental data are available from the corresponding author upon request.

Conflicts of Interest

The author declares that there are no conflicts of interest regarding this work.

Acknowledgments

This project was funded by the First-class Undergraduate Curriculum Development Program of Minjiang University (MJU2021KC314).

References

[1] A. C. María and J. ValeroMas Jose, “Exploiting the two-dimensional nature of agnostic music notation for neural optical music recognition,” *Applied Sciences*, vol. 11, no. 8, p. 3621, 2021.

[2] G. Agarwal and H. Om, “An efficient supervised framework for music mood recognition using autoencoder-based optimised support vector regression model,” *IET Signal Processing*, vol. 15, no. 2, pp. 98–121, 2021.

[3] R. Sarkar, S. Choudhury, S. Dutta, A. Roy, and S. Saha, “Recognition of emotion in music based on deep convolutional neural network,” *Multimedia Tools and Applications*, vol. 79, no. 1, pp. 765–783, 2020.

[4] F. Yan, “Music recognition algorithm based on T-S cognitive neural network,” *Translational Neuroscience*, vol. 10, no. 1, pp. 135–140, 2019.

[5] X. Liang, L. Xu, X. Wen, J. Shi, S. Li, and X. Qian, “Effects of individual factors on perceived emotion and felt emotion of music: based on machine learning methods,” *Psychology of Music*, vol. 49, no. 5, pp. 1069–1087, 2021.

[6] J. Wang, Q. Wang, and H. Liu, “Emotion recognition of musical instruments based on convolution long short time memory depth neural network,” *Journal of Physics: Conference Series*, vol. 1976, no. 1, 2021.

[7] A. Orhan and Ş. Abdulkadir, “Attention guided 3D CNN-LSTM model for accurate speech based emotion recognition,” *Applied Acoustics*, vol. 182, 2021.

[8] A. Solanki and S. Pandey, “Music instrument recognition using deep convolutional neural networks,” *International Journal of Information Technology*, pp. 1–10, 2019.

[9] I Maliki, “Musical instrument recognition using mel-frequency cepstral coefficients and learning vector quantization,” *IOP Conference Series: Materials Science and Engineering*, vol. 407, no. 1, 2018.

[10] K. Patil and M. Elhilali, “Biomimetic spectro-temporal features for music instrument recognition in isolated notes and solo phrases,” *EURASIP Journal on Audio Speech and Music Processing*, vol. 2015, no. 1, 27 pages, 2015.

[11] V. Gupta, S. Juyal, G. P. Singh, C. Killa, and N. Gupta, “Emotion recognition of audio/speech data using deep learning approaches,” *Journal of Information and Optimization Sciences*, vol. 41, no. 6, pp. 1309–1317, 2020.

[12] T. Y. Chin, B. N. I Eskelson, K. Martin, and V. LeMay, “Automatic bird sound detection: logistic regression based acoustic occupancy model,” *Bioacoustics*, vol. 30, no. 3, pp. 324–340, 2021.

[13] E. N. N. Ocquaye, Q. Mao, Y. Xue, and H. Song, “Cross lingual speech emotion recognition via triple attentive asymmetric convolutional neural network,” *International Journal of Intelligent Systems*, vol. 36, no. 1, pp. 53–71, 2020.

[14] Y. Yin, D. Tu, W. Shen, and J. Bao, “Recognition of sick pig cough sounds based on convolutional neural network in field situations,” *Information Processing in Agriculture*, vol. 8, no. 3, pp. 369–379, 2021.

[15] Mustaqeem and S. Kwon, “Optimal feature selection based speech emotion recognition using two-stream deep convolutional neural network,” *International Journal of Intelligent Systems*, vol. 36, no. 9, pp. 5116–5135, 2021.

[16] T. Anvarjon, J. Y. Choeh, and S. Kwon, "Age and gender recognition using a convolutional neural network with a specially designed multi-attention module through speech spectrograms," *Sensors*, vol. 21, no. 17, p. 5892, 2021.

[17] D.-H. Jung, N. Y. Kim, S. H. Moon et al., "Deep learning-based cattle vocal classification model and real-time livestock monitoring system with noise filtering," *Animals*, vol. 11, no. 2, p. 357, 2021.

[18] F. Misbah, M. Farooq, F. Hussain, N. Baloch, F. Raja, and Y. Zikria, "Impact of feature selection algorithm on speech emotion recognition using deep convolutional neural network," *Sensors*, vol. 20, no. 21, p. 6008, 2020.

[19] W. C. Yin, W. Ei Hlaing, and M. Myo Khaing, "Myanmar continuous speech recognition system using convolutional neural network," *International Journal of Image, Graphics and Signal Processing*, vol. 13, no. 2, pp. 44–52, 2021.

[20] K. Mukul, M. Kumar, N. Katyal, N. Ruban, E. Lyakso, and G. Richard, "Transfer learning based convolution neural net for authentication and classification of emotions from natural and stimulated speech signals," *Journal of Intelligent and Fuzzy Systems*, vol. 41, no. 1, pp. 2013–2024, 2021.

[21] X. Chen, "Simulation of English speech emotion recognition based on transfer learning and CNN neural network," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 2, pp. 2349–2360, 2021.

[22] M. Seo and M. Kim, "Fusing visual attention CNN and bag of visual words for cross-corpus speech emotion recognition," *Sensors*, vol. 20, no. 19, p. 5559, 2020.

[23] V. Passricha and R. K. Aggarwal, "PSO-based optimized CNN for Hindi ASR," *International Journal of Speech Technology*, vol. 22, no. 4, pp. 1123–1133, 2019.

[24] G. Wang, W. Li, L. Zhang, L. Sun, and X. Ning, "Encoder-X: solving unknown coefficients automatically in polynomial fitting by using an autoencoder," *IEEE Transactions on Neural Networks and Learning Systems*, 2021.

[25] V. Passricha and R. K. Aggarwal, "Convolutional support vector machines for speech recognition," *International Journal of Speech Technology*, vol. 22, no. 3, pp. 601–609, 2019.

[26] S. Wang, T. H. Wu, T. Shao, and Z. X. Peng, "Integrated model of BP neural network and CNN algorithm for automatic wear debris classification," *Wear*, vol. 426-427, pp. 1761–1770, 2019.

[27] Q. Zheng, P. Zhao, Y. Li, H. Wang, and Y. Yang, "Spectrum interference-based two-level data augmentation method in deep learning for automatic modulation classification," *Neural Computing & Applications*, vol. 33, no. 13, pp. 7723–7745, 2020.

[28] C. Lin, Y. Shi, J. Zhang, C. Xie, W. Chen, and Y. Chen, "An anchor-free detector and R-CNN integrated neural network architecture for environmental perception of urban roads," *Proceedings of the Institution of Mechanical Engineers - Part D: Journal of Automobile Engineering*, vol. 235, no. 12, pp. 2964–2973, 2021.

[29] R. Vidhya and G. Vadivu, "Towards developing an ensemble based two-level student classification model (ESCM) using advanced learning patterns and analytics," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 7, pp. 7095–7105, 2020.

[30] J. Ma and T. W. S. Chow, "Label-specific feature selection and two-level label recovery for multi-label classification with missing labels," *Neural Networks*, vol. 118, pp. 110–126, 2019.

[31] S. Qi, X. Ning, G. Yang et al., "Review of multi-view 3D object recognition methods based on deep learning," *Displays*, vol. 69, no. 1, 2021.