JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 39, 19-37 (2023)
DOI: 10.6688/JISE.202301_39(1).0002

Adoption of Gesture Interactive Robot in Music Perception Education with Deep Learning Approach

JIA-XIN HU[1], YU SONG[2] AND YI-YAO ZHANG[3,+]
[1]*School of Music and Dance*
Qiqihar University
Qiqihar, 161000 P.R. China
E-mail: teacherhu0727@163.com
[2]*Development Research Centre of Music Industry*
Communication University of China
Beijing, 100024 P.R. China
E-mail: rain07122003@163.com
[3]*School of Art and Communication*
Beijing Normal University
Beijing, 100875 P.R. China
[+]*E-mail: 11112018044@bnu.edu.cn*

This work intends to help students perceive music, study music, create music, and realize the "human-computer interaction" music teaching mode. A distributed design pattern is adopted to design a gesture interactive robot suitable for music education. First, the client is designed. The client gesture acquisition module employs a dual-channel convolutional neural network (DCCNN) for gesture recognition. The convolutional layer of the constructed DCCNN contains convolution kernels with two sizes, which operate on the image. Second, the server is designed, which recognizes the collected gesture instruction data through two-stream convolutional neural network (CNN). This network cuts the gesture instruction data into *K* segments, and sparsely samples each segment into a short sequence. The optical flow algorithm is employed to extract the optical flow features of each short sequence. Finally, the performance of the robot is tested. The results show that the combination of convolution kernels with sizes of 5×5 and 7×7 has a recognition accuracy of 98%, suggesting that DCCNN can effectively collect gesture command data. After training, DCCNN's gesture recognition accuracy rate reaches 90%, which is higher than mainstream dynamic gesture recognition algorithms under the same conditions. In addition, the recognition accuracy of the gesture interactive robot is above 90%, suggesting that this robot can meet normal requirements and has good reliability and stability. It is also recommended to be utilized in music perception teaching to provide a basis for establishing a multi-sensory music teaching model.

Keywords: robot, gesture recognition, DCCNN, two-stream convolutional neural networks, deep learning

1. INTRODUCTION

Wireless sensor networks have wide application in fields such as home, industry, envir Music is an art form that takes musical sound as the carrier to express people's thoughts and feelings, and it is the expression of emotion. People's perception of music needs to be felt through hearing, not through touch or vision, but the emotion contained in music is not limited to acoustics. In ancient China, when people drank and played a string-

Received August 9, 2021; revised October 3, 2021; accepted January 13, 2022.
Communicated by Ching-Hsien Hsu.
[+] Corresponding author.

ed, they danced swords to help them enjoy themselves. They showed a visual and auditory musical feast through body movement, footwork and rhythm. Now, people combine sound, light and electricity to establish a wonderful audio-visual effect and achieve emotional interaction. Therefore, people have been exploring the interaction of multiple perception modes of music from vision, hearing and touch, to feel the pitch, loudness and timbre, experience the rhythm, melody and tonality of music, and realize the emotional interaction with music [1-3]. The traditional way of human-computer interaction has changed with the development of information technology and robot technology. Among them, gesture-based robot interaction can perform corresponding services according to people's understanding of gestures, and the behavior of the robot can be controlled through several simple gestures. Thereby, the gesture interaction system is of great significance to improve human lifestyle and production mode. Applying gesture interaction to music perception teaching can improve students' learning interest and learning accuracy [4].

This exploration aims to explore the application of gesture interactive robots in music perception education, improve the teaching efficiency of music classroom teaching and help teachers manage students in the classroom. A distributed robot gesture interactive application system with a client-server structure is designed. The client collects gesture instructions and obtains real-time video stream through a dual-channel convolutional neural network (DCCNN), analyzes and processes the video stream, extracts gesture instruction data and synchronizes it to the server. The server recognizes the collected gesture instruction data through a two-stream convolution neural network and returns the recognition results to the client to display to the user. Finally, the interactive robot is tested. The research innovation is to apply deep learning technology to music teaching, establish a human-computer interactive music teaching robot, and improve the efficiency of music teaching.

2. LITERATURE REVIEW

Gao *et al.* (2020) designed a single shot multibox detector based on deep learning network function map fusion to solve the problem of hand detection and positioning in space human-computer interaction. First, the background of this method was introduced, including astronaut assisted robot platform, hand detection and positioning difficulties, and deep learning network for object detection and positioning. Then, a single shot multibox detector was designed to detect and locate the position of the hand. In the experimental part, the single shot multibox detector was trained and tested through a self-made database and two public databases. The results show that compared with the existing technology, the proposed method takes into account the accuracy, speed and balanced performance, and has good advantages [5]. Simão *et al.* (2019) designed a human-computer interaction framework to classify the actions of interleaved static and dynamic gestures captured by wearable sensors. Dynamic gesture features were obtained by applying data dimensionality reduction (resampling through cubic interpolation and principal component analysis) to the original data from the sensor. Gesture datasets were used to conduct experimental tests on different samples. The results show that the accuracy of the classification model is 95%. For 24 static gesture libraries using random forest, it is 6%; for 10 dynamic gesture libraries using stochastic neural network, it is 99.3%. These results are different because different classifiers have different classification characteristics [6]. Neto *et al.* (2019) proposed a gesture-based human-computer interaction framework, which makes robots assist

human colleagues in delivering tools and parts to jointly complete assembly operations. Wearable sensors were used to capture gestures on the upper body of the human body. The captured data were divided into static data and dynamic data, which classified gestures through artificial neural network recognition to understand the practical significance of gestures. A parameterized robot task manager could be implemented through the human-computer interaction interface. According to the voice and visual feedback of the system, the functions of the robot could be selected by gesture to realize the human-computer interaction process. The experiment of assembly operation proves that the solution proposed can improve the work efficiency [7]. Gladence *et al.* (2014) studied the application of sequential pattern mining in heart disease prediction. Naive Bayes classifier combined with short-term heart rate variability measurement was used to find the severity of congestive heart failure, so as to realize the prediction of heart disease [8].

Thereby, the current research direction shows that the use of deep learning technology combined with relevant data can predict the future development of a model, and then respond effectively. At present, the research literature on the application of deep learning technology in music classroom teaching is less, and because of the particularity of the music curriculum, it needs a targeted design to meet the needs of music classroom teaching. Therefore, it is proposed to apply deep learning technology to music teaching to improve the teaching efficiency of music classrooms.

3. RESEARCH MODEL AND THE METHODOLOGY

3.1 Design of the Gesture Interactive Robot System

Sight-singing is the skill of reading musical scores using visual, auditory, and other perceptual methods. Solfege helps students to quickly master the melody of music, strengthens their perception and understanding of music, and is beneficial to heightening their performance level. A quick response from sight to singing is required during the process, which needs intensive training. Chen *et al.* (2020) used a convolutional neural network (CNN) combined with video tracking technology to identify the target [9]. However, how to show the dynamic rhythm of singing to make the teaching process more direct and effective remained undetermined. Curwen gestures (Fig. 1) were created by Hungarian mu-

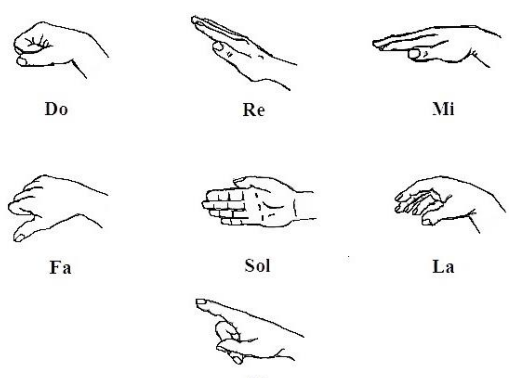

Fig. 1. Curwen gestures.

sic educator John Curwen and used in music teaching. Curwen gestures express different roll names through the high and low positions of seven different gestures, and the high-low relationship and the dynamic process are reflected through spatial relationships. Therefore, applying Curwen gestures to music perception teaching can increase the interest in learning, which also improves intonation and enriches learning content. How to use computers to recognize gestures in music teaching and establishing a human-computer interactive music perception teaching method are the major purposes of this exploration.

The robot gesture interaction system includes two parts: the client and the server. The client obtains the gesture instructions from users, performs preliminary static recognition, and transfers the data to the server for static recognition and optical flow recognition of gesture instructions [10]. The recognition results are sent to the client. If the gesture matches an instruction in the database, the recognition result will be converted into a control instruction. Otherwise, the data are transferred to the server to be processed. Separating the acquisition module can increase the stability of the system and reduce the requirements for local hardware performance [11].

Fig. 2 displays the workflow of the gesture interactive robot system. The client collects the gesture instruction data of user, and the video stream acquisition module is responsible for acquiring the video stream; the gesture instruction acquisition module performs preliminary analysis and processing on the video stream to extract the gesture instruction data; the data transceiver module synchronizes the gesture instruction data to the server, receives the recognition result from the server, and converts the corresponding gesture instruction; the client interface displays the recognition results corresponding to the gesture instruction from the user in real-time [12]. The server is mainly responsible for the recognition of user gesture instruction data. The data transceiver module is responsible for receiving the gesture instruction data transmitted by the client and returning the recognition result; the gesture instruction recognition module is responsible for recognizing the gesture instruction data; the server interface presentation module is responsible for the visualization of the working status [13, 14].

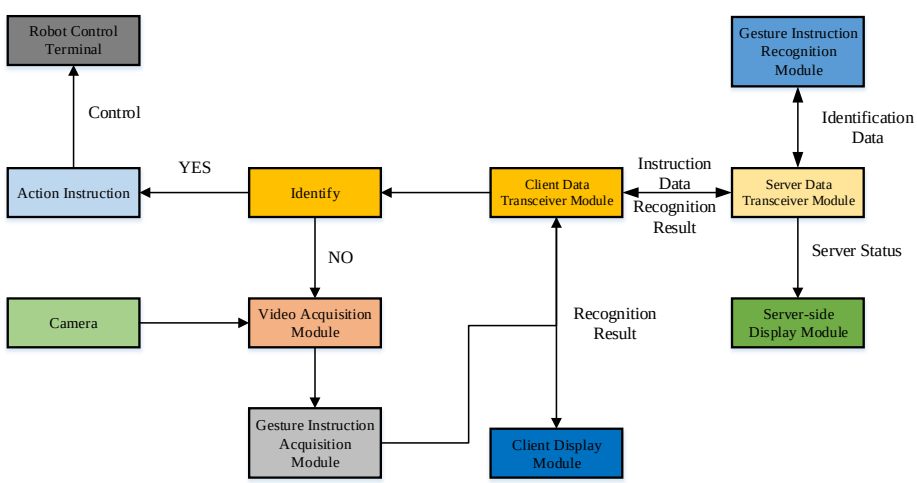

Fig. 2. Gesture interactive robot system.

3.2 Gesture Instruction Data Collection

CNN is a type of feedforward neural network that includes convolution calculations. It performs supervised learning and unsupervised learning by imitating the visual mechanism of the human eye. The convolution kernel parameters in the hidden layer can perform weight sharing and inter-layer connection, which enables learning features of Grid-like topology using a reduced amount of calculation. Therefore, CNN is often used to process images and speech [15]. Its structure includes the input layer, hidden layer, and output layer. The input layer processes multi-dimensional data, and the input features must be standardized. The output layer adopts a logical function or a normalization function to output classification labels. The hidden layer usually contains a convolutional layer, a pooling layer, and a fully connected layer, of which the convolutional layer and the pooling layer are unique structures to CNN. The convolutional layer contains weight coefficients, while the pooling layer doesn't [16].

The convolution operation refers to combining two functions and extracting features from the input image, including width, height, and multi-channel information. In image processing, the convolutional layer is superimposed on the image to multiply the convolution kernel by the image where it is located. By analogy, the features of the image are extracted through the convolution operation of the convolution kernel. In CNN, several convolutional units can be generated with a convolutional layer, and the parameters are learned through a direction propagation algorithm. After the network parameters are adjusted, the high-level features are obtained from the low-level features of the image through a multi-level convolution process [17].

The pooling operation reduces the number of learning parameters required by the network using a nonlinear down-sampling algorithm, thus preventing the over-fitting phenomenon of CNN [18]. The fully connected layer is generally set after the convolutional layer and the pooling layer. The previous layer is connected to the activation function to convert the extracted two-dimensional feature image into a one-dimensional feature vector, which has a feature classification function. The weighted sum of the output and the network training weight is acquired, and the maximum value is the recognition result [19]. In the CNN, the convolutional layer, the pooling layer, and the fully connected layer intersect each other (Fig. 3). The neurons in the fully connected layer are fully connected between layers, and features after the convolution and pooling are integrated [20].

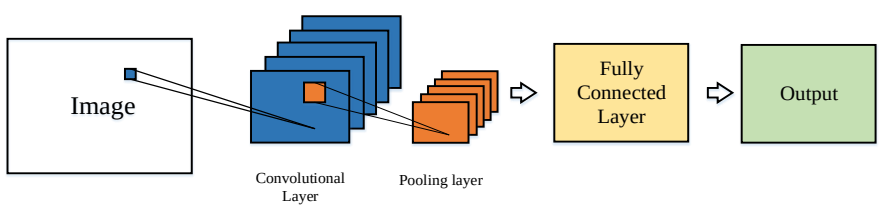

Fig. 3. CNN structure.

DCCNN is employed to recognize static gestures, extract high-level features, and realize advanced recognition of multiple gestures, so as to avoid pre-processing of the traditional gesture recognition. It can acquire advanced features of different granularities and enhances the recognition ability for gestures [21].

CNN can extract the required features after training. Generally, a CNN has multiple convolutional layers and pooling layers. There are many convolution kernels in a convolutional layer to extract local features. Each convolution kernel can map a feature image [22]. A small convolution kernel can increase the amount of feature information and reduces the number of parameters under the same conditions, thereby improving recognition effects. Therefore, the recognition effects (the recognition accuracy of the neural network for gestures) of multiple small convolution kernels are higher than that of a large convolution kernel, whereas if the convolution kernel is too small, image feature extraction may be impossible. CNN outputs the feature map of an input image. The design of the first convolutional layer is particularly important. If the convolution kernel of the first convolutional layer is too large, some detailed information of the image will be lost; if the convolution kernel is too small, it is unable to present the characteristics of the image [23]. Traditional CNN uses a fixed-size convolution kernel, and the image granularity is also fixed. As a result, some features are lost in the learning process, which reduces the accuracy of network recognition [24]. Consequently, the DCCNN with dual-size convolution kernels has two convolutional layers and two pooling layers, and a fully connected layer. The convolution kernel of the first convolutional layer of the two convolutional networks is different in size, which is 5×5 and 7×7, respectively, as shown in Fig. 4. The fully connected layers of the two CNNs are combined through a fully connected map, which is input into the classifier for feature classification [25].

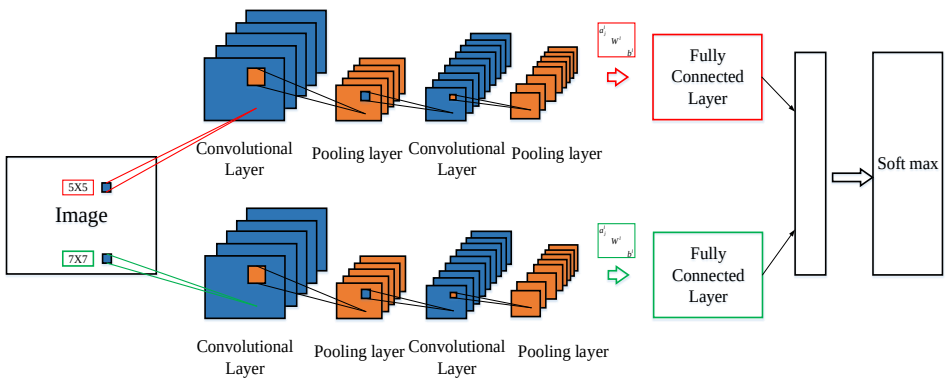

Fig. 4. DCCNN structure.

The robot gesture interaction system can recognize the gestures in the video stream. The user's gestures are recorded by the camera, and the acquired data are undertaken as the input image for static gesture recognition The DCCNN is then applied to check whether there are gestures in the video, and COUNT and HAND are adopted to mark the video frame by frame [26]. HAND = 1 means that the current frame contains static gestures, and HAND = 0 means no static gestures; COUNT = 1 means that the previous frame contains static gestures, and COUNT = 0 means that the previous frame does not contain static gestures. *X* indicates the *X*th frame. The acquired images containing gestures are stored [27, 28]. Fig. 5 presents the specific process.

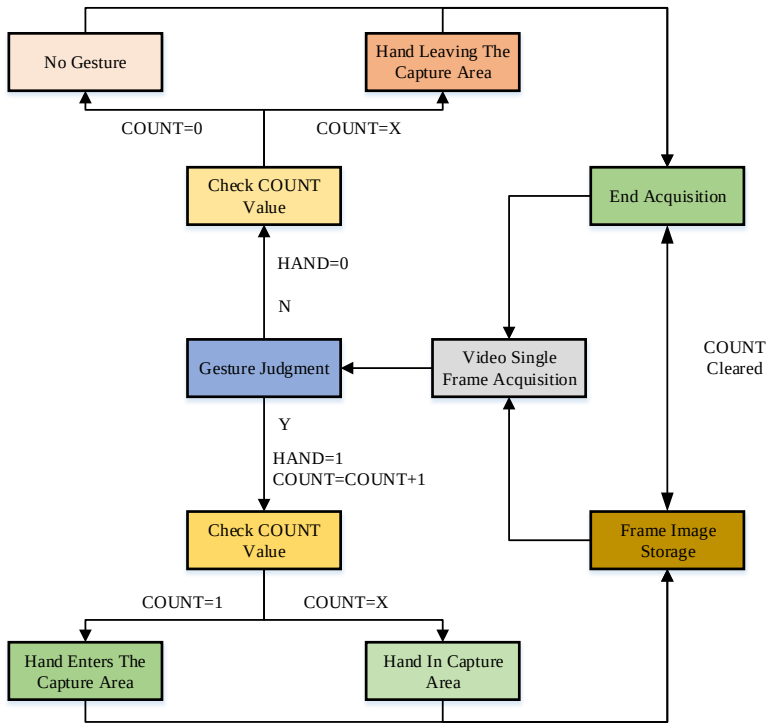

Fig. 5. Acquisition of video stream gesture instructions.

In this section, the three-layer CNN structure is introduced to recognize static gestures. The gestures in the video stream are analyzed by DCCNN, and the video frames containing the gestures are stored, providing a data basis for gesture recognition [29].

3.3 Gesture Instruction Recognition Network

The dynamic gestures contain more motion information than static gestures. Hence, this section focuses on how to recognize motion gestures. The optical flow characteristics of video are analyzed, with the motion information recognized using DCCNN. Video information is classified into spatial information and temporal information [30]. Spatial information refers to the surface information of video frames; temporal information refers to the optical flow among frames. Therefore, DCCNN is applied to recognize the motion information by extracting spatial and temporal information from the RGB image and the optical flow image. These two images are input into the network respectively, and the prediction results of each network are merged to form the final recognition results [31].

The DCCNN is composed of two dimensions of space and time, which process the temporal and spatial information respectively. One network is a single-frame CNN. The video is input through a single-frame RGB image to describe the spatial characteristics of the image. The other CNNs input the video through the multi-frame dense optical flow to express the temporal characteristics of the video. The two networks are combined to recognize the video [32]. DCCNN can effectively obtain the motion information of video data.

The spatial feature image and the temporal image are convolved, pooled, and fully connected through a CNN, and the video is recognized by merging spatial and temporal features [33].

I. The early DCCNN convolves the single-frame image and multi-frame optical flow image through the convolution kernel in the convolutional layer. Fig. 6 presents its structure. The final results are acquired by the weighted sum of the output results [34].

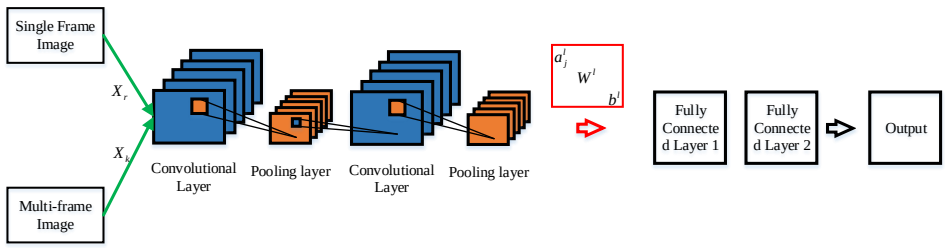

Fig. 6. DCCNN with early merging strategy.

Usually, the merging weights of the two are both 0.5. The image of a single frame of RGB image is X_r, the optical flow image is X_k, and then the jth Neuron in the merged layer is expressed as follows.

$$a_j^l = f_F\left(W^l\left(0.5\bullet X_r \bullet k_{ij}^l + 0.5\bullet\sum_1^n X_k \bullet k_{ij}^l\right) + b^l\right) \tag{1}$$

a_j^l is the output value of the jth neuron of the merged layer; W^l and b^l are the weight and bias of CNN, respectively. $f_F(\ )$ is the SoftMax activation function. The fusion weights of spatial and temporal feature fusion are both 0.5. However, the difference between the optical flow image and the single-frame image is ignored, so this method is suitable for the merging of the initial images only [35].

II. The later DCCNN inputs the single-frame image and the optical flow image into the same CNN, followed by feature extraction, respectively, which is divided into the spatial neural CNN and the temporal CNN, as shown in Fig. 7.

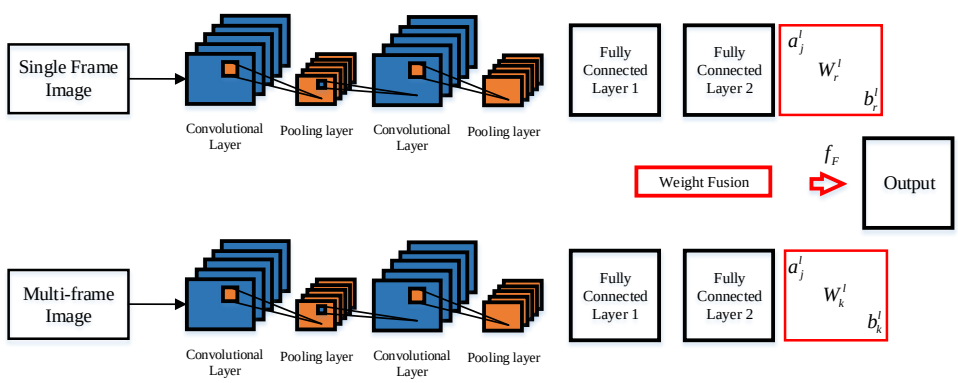

Fig. 7. DCCNN with late merging strategy.

The neuron in spatial CNN is a_{rj}^{l-1}, and the neuron in temporal CNN is a_{kj}^{l-1}. The equation to fully connect the two fully connected layers is as follows.

$$a_j^l = f_F\left(0.5\left(W_r^l \bullet \sum_{i \in M_{rj}^l} a_{rj}^{l-1} + b_r^l\right) + 0.5\left(W_k^l \bullet \sum_{i \in M_{kj}^l} a_{kj}^{l-1} + b_k^l\right)\right) \quad (2)$$

a_j^l is the output value of the neural network; W_r^l and b_r^l are the weight and biases of the spatial CNN; W_k^l and b_k^l are the weights and biases of the temporal CNN; M_{rj}^l is the feature map set of the spatial CNN, M_{kj}^l is the feature map set of temporal CNN, and f_F is the activation function. The correlation between RGB image and optical flow image is not considered in the following merging, which is suitable for merging at the decision-making level.

III. The merging of the fully connected layer refers to realizing the weight sharing of the spatial CNN and the temporal CNN in the first fully connected layer, so as to reduce the independence in the later merging. The spatial CNN and temporal CNN have the same structure, as shown in Fig. 8.

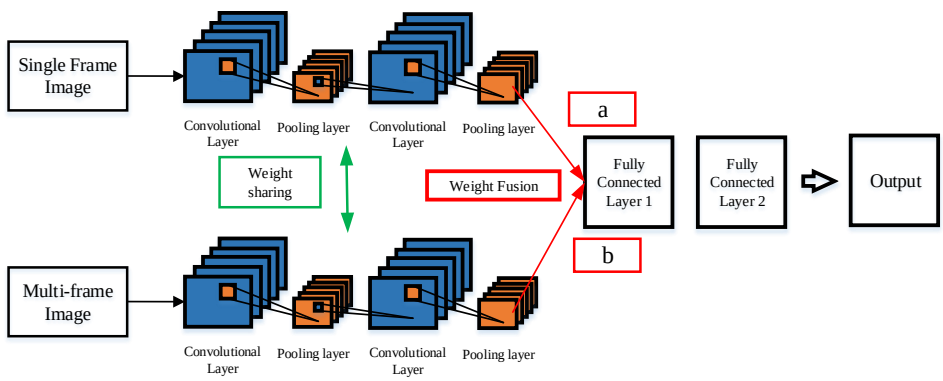

Fig. 8. DCCNN of fully connected layer merging strategy.

The neuron in the pooling layer of spatial CNN is a_{rj}^{l-1}, and the neuron in the pooling layer of temporal CNN is a_{kj}^{l-1}. The output results of the two pooling layers are fully connected, and then the equation of the *j*th neuron of the merging layer is as follows [36].

$$a_j^l = f_F(W^l(\sum_{i \in M_{rj}^l} a_{rj}^{l-1} \bullet \alpha + \sum_{i \in M_{kj}^l} a_{kj}^{l-1} \bullet \beta) + b^l) \quad (3)$$

α and β are the merging coefficients of spatial CNN and temporal CNN, respectively. The weight value equalling to the bias value can improve the network recognition effects, which is suitable for the merging in the fully connected layer [37].

IV. The convolutional layer merging refers to inputting a single-frame image and a multi-frame optical flow image into two CNNs, respectively. Fig. 9 displays the structure. Multiple convolutions and pooling are then performed, and the output results of the two networks are convolved with the same convolution kernel. The weighted values are merged after the convolution, and convolution, pooling, and full connection are repeated [38].

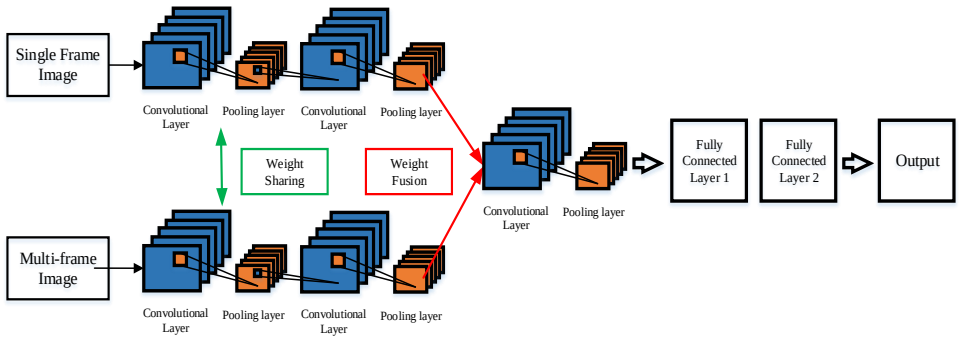

Fig. 9. DCCNN of convolutional layer merging strategy.

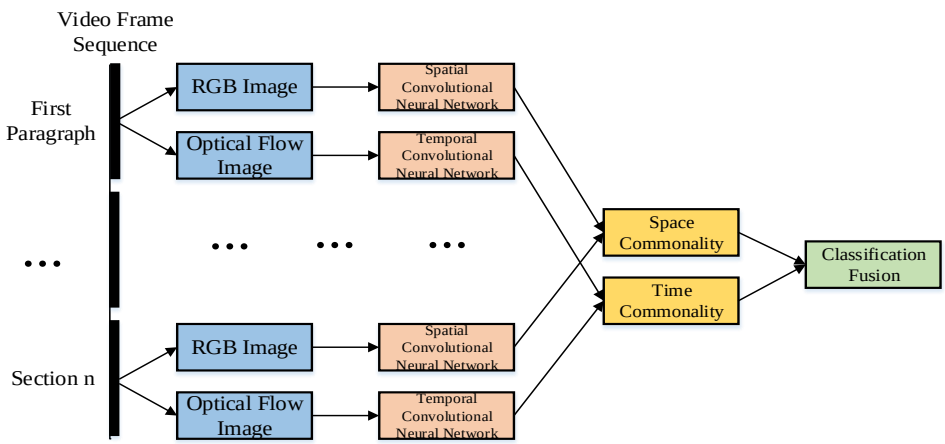

Fig. 10. Gesture instruction recognition network.

The *j*th neuron in the merging layer is expressed as follows,

$$a_j^l = f_c\left\{W^l \bullet \left[\sum_{i\in M_{ij}^l}\left(\alpha \bullet a_i^{l-1} \bullet k_{ij}^l\right) + \sum_{i\in M_{kj}^l}\left(\beta \bullet a_i^{l-1} \bullet k_{ij}^l\right)\right] + b^l\right\}. \tag{4}$$

The convolutional layer merging means that the two convolutional networks undergo multiple convolutions and poolings to extract the inherent features of the image. Then, features are merged using the same convolution kernel to ensure the relevance between the single-frame image features and optical flow image features. The advantage of this network is that it ensures the independence of RGB features and optical flow features, thereby reducing the number of training parameters and improving training and recognition efficiency [39].

The temporal information of the video is called the optical flow feature, representing the velocity vector of the object in the video, which illustrates the change of a single-frame image in the two-dimensional vector field. The velocity field of the three-dimensional movement of the object point is represented by the two-dimensional image. Optical flow reflects the changes in the image as a result of motion, and the direction and velocity of the image can be determined by optical flow [40]. Fig. 10 displays the structure of the

DCCNN designed. The complete information of the video is utilized to perform a two-stream time network with time segmentation for the task of predicting video results. This network contains *K* spatial CNN and *K* temporal CNN compared to the traditional two-stream CNN. The input value of the temporal CNN is the optical flow characteristic image of consecutive frames processed by a series of cut fragments of the entire video [41]. Each cut segment will produce its own cut video prediction result, and then the prediction result of each segment is exported as the prediction input of the entire video. In the learning process, the loss value of the entire video prediction is optimized by iteratively updating the model parameters.

4. EXPERIMENTAL DESIGN AND PERFORMANCE EVALUATION

4.1 Experimental Dataset, Hyperparameters, Test Environment Settings

Different datasets are utilized to test and train static gesture recognition and dynamic gesture recognition modules, so as to test the performance of human-computer interaction. The static gesture dataset uses the CGD2013 dataset, which contains 11,000 color images of 20 gestures (550 for each gesture). 400 images of each gesture are randomly selected from the dataset, and a total of 8000 gesture images are taken as the training dataset of the DCCNN. In another 150 images, a total of 3000 gesture images are taken as the dataset for the network test. Fig. 11 presents the datasets. The constructed neural network adopts a small batch stochastic gradient descent method to learn the network parameters. The batch size is set to 256, and the momentum is set to 0.9. The initial weight of the spatial convolutional network is set to 0.95, and that of the temporal convolutional network is set to 1.5. In the experiment, the learning rate is initialized to 0.01 for the spatial convolutional network, which is reduced to 1/10 of it every 2000 iterations, and the training process is iterated 4000 times in total. For the time convolutional network, the learning rate is initialized to 0.005 and is reduced to 1/10 of it after the 12,000 and 15,000 iterations. The training process is iterated 25,000 times in total.

Fig. 11. CGD2013 data set.

The interactive system is completed by a robot with a three-layer structure. The bottom layer is the robot motion base, which walks by controlling the movement and speed. The middle layer is the hardware layer, which is mainly used to connect hardware devices

such as rangefinders, steering gears, and gyroscopes to realize management and control of various hardware. The top layer is the development layer based on the Raspberry Pi. Raspberry Pi is a microcomputer motherboard with multiple USB ports and network ports, which has the basic functions of a microcomputer. Table 1 shows the robot configuration information and network hyperparameter settings.

Table 1. Robot configuration information.

Parameters	Client	Servers
Display program	Qt 5.7 VS2015	Linux-dash
System	Win7	CentOS 7.3.1611
CPU	i7-7700	Intel Xeon E5-2680
GPU	GeForce GTX 960	GeForce GTX Titan V
Camera	Kinect v1	
Hard disk capacity	32GB+512GB	512GB+4TB

4.2 Recognition Results of Static Gesture Instruction

To verify the impact of the size of the DCCNN kernel on the recognition effects, two parallel experiments are performed with different sizes of convolution kernels. Fig. 12 displays the results.

Fig. 12 illustrates that the recognition accuracy of DCCNN reaches as high as 96%, and the recognition accuracy of different convolution kernels is different. The comparison reveals that the recognition effects of DCCNN are affected by the size of the convolution kernel. The combination of convolution kernels with the size of 5×5 and 7×7 improves the recognition accuracy rate to 98%. The experiment also suggests that the DCCNN can recognize static gesture images of different scales, and merging the information of these images can obtain richer feature information and better identification results.

Fig. 12 suggests that the recognition rate of gestures by different DCCNNs is 98%. The experimental results reveal that the size of the convolution kernel has a certain impact on the recognition rate of the network. The network that combines two smaller convolution kernels has a lower recognition rate for gestures than the network generated by combining two larger convolution kernels. The network generated by the combination of convolution kernels with a larger gap has a lower recognition rate of gestures than the network generated by the combination of convolution kernels with a smaller gap. Combining 5×5 and 7×7 convolution kernels can increase the recognition accuracy to 98%. Experiments also show that DCCNN can recognize static gesture images of different scales, and fusion of the information of these images can help obtain richer feature information and better recognition results.

The learning curve displays the performance of the network on the new data by calculating the accuracy of the training set and cross-validation when the size of the training set is different. In this way, it can determine whether the variance of the network or the deviation is too high and whether increasing the training set can reduce the over-fitting phenomenon of CNN. Fig. 13 indicates that there is no over-fitting phenomenon in the algorithm.

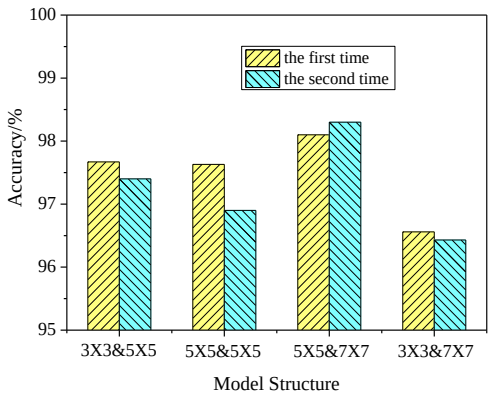

Fig. 12. The recognition accuracy of CNNs of different sizes of kernels.

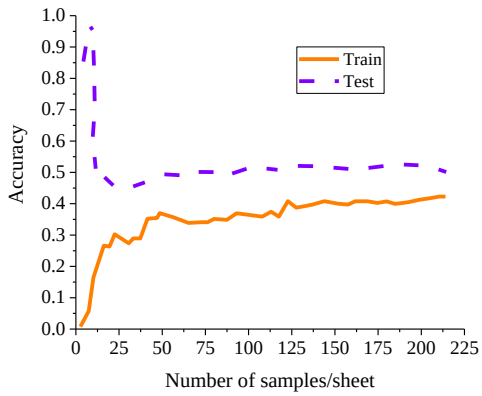

Fig. 13. The learning curve.

4.3 Recognition Results of Dynamic Gesture Instructions

The three types of CNNs involved are tested using training from scratch, pre-training spatial networks, and cross-input pre-training strategies. Fig. 14 presents the results.

Fig. 14 reveals that both spatial CNN and temporal CNN have general effects in dynamic gesture recognition, with an accuracy of less than 90%; while the DCCNN exhibits accuracy of about 90%, which is higher than the other two networks. Besides, the recognition results are changeable if the same CNN is trained using different algorithms. Among them, the recognition effects of cross-input pre-training are the best among the three, with a recognition accuracy of 91%. The reason may be that cross-input pre-training can effectively reduce the over-fitting phenomenon of CNN.

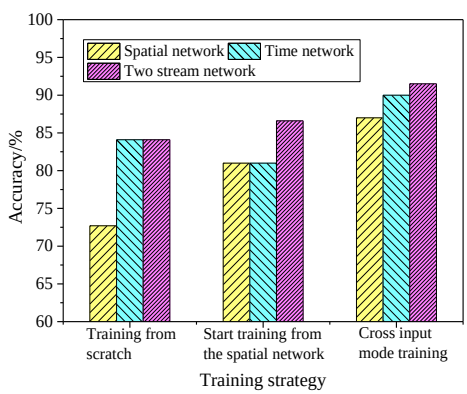

Fig. 14. CNN test results under three different training methods.

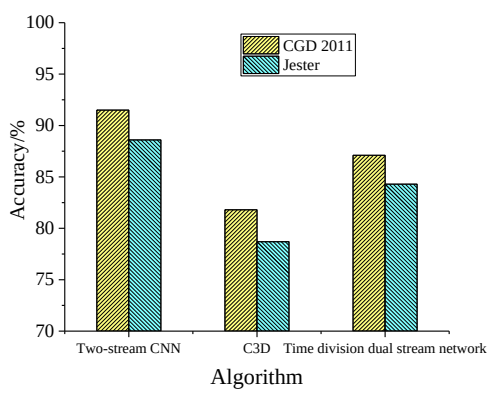

Fig. 15. The dynamic gesture recognition results.

The two-stream CNN, Convolutional 3D (C3D), and the time-segmented double-channel algorithm are compared for recognition accuracy using two datasets of CGD2011 and Jester for training, as shown in Fig. 15.

Fig. 15 reveals that the recognition accuracy of the double-channel is 91% and 88% in CGD2011 and Jester datasets respectively. They are higher than the recognition results of C3D and time division double-channel networks after training on the two datasets. Therefore, the algorithm designed demonstrates better video recognition effects and can avoid the loss of video information during the video recognition process.

4.4 Two-Stream CNN Test With Different Layer Merging Strategies

The different two-stream CNNs mentioned are compared, including early fusion strategy, late fusion strategy, fully connected layer fusion strategy, and convolutional layer fusion strategy, using the CGD2013, CGD2011, and Jester datasets for training and testing. Fig. 16 presents the results.

Fig. 16 suggests that the recognition accuracy of the unified dataset of the two-stream CNN using different layer merging strategies has obvious differences. Among them, the two-stream CNN using the convolutional layer fusion strategy has the best recognition accuracy, which is above 85%, followed by the fully connected layer fusion strategy, the late fusion strategy, and the early fusion strategy. Therefore, the accuracy of the algorithm for image recognition is continuously improved with the improvement of the two-stream CNN algorithm.

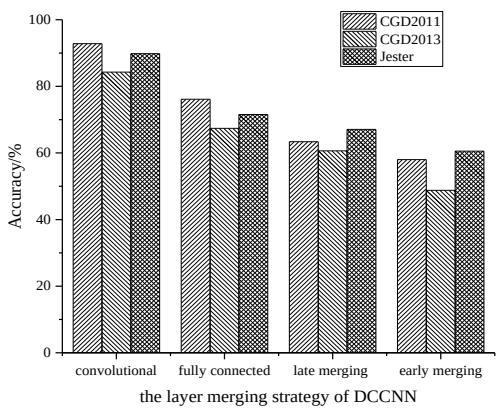

Fig. 16. Test results of two-stream CNN with different layer merging strategies on different data sets.

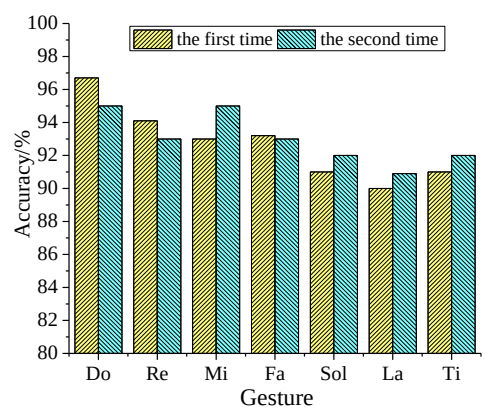

Fig. 17. Gesture instruction recognition results of gesture interactive robot.

4.5 Recognition Results of Robot Gesture Interaction System

Finally, the designed robot interactive system is tested to verify the usability and reliability of the gesture interactive robot in practice. The seven music gestures (Curwen gestures) are recognized and tested through two parallel tests, as shown in Fig. 17.

Fig. 17 illustrates that the gesture interactive robot can recognize various gesture instructions, and the overall recognition accuracy is over 90%. It may be because that there is little content related to Curwen gestures in the training set used. As a result, the corresponding gestures cannot be accurately recognized. However, the test shows that the gesture interactive robot system can meet the normal interaction requirements, which have good reliability and stability. It can be used for music perception teaching.

To sum up, the combination of 5×5 and 7×7 kernels can improve the recognition accuracy of DCCNN up to 98%, and can effectively collect gesture instruction data. The recognition accuracy of DCCNN can reach 90% after training, higher than the mainstream dynamic gesture recognition algorithm. The recognition accuracy rate of the gesture interactive robot is above 90%. Therefore, the gesture interactive robot system can meet normal requirements and is of good reliability and stability.

4.6 Experimental Comparison

Seven music gestures are input into the network designed and the original two-stream network for testing. Fig. 18 displays the test results.

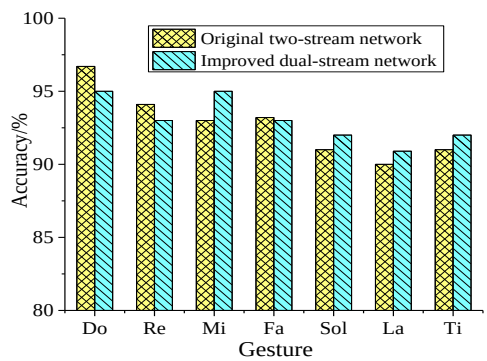

Fig. 18. Comparison of test results of different algorithms.

In Fig. 18, the gesture recognition interactive system containing the original two-stream network can recognize various gesture commands, and the overall recognition accuracy rate exceeds 93%. The overall recognition accuracy rate of gesture recognition interactive system with improved dual-stream network exceeds 95%. The two algorithms have the highest recognition rate for Do and the lowest recognition rate for La. The reason may be that Do has a scale that is easier to identify, but La doesn't.

5. CONCLUSIONS

This exploration is to explore the application of gesture interactive robots in music perception education. First, the DCCNN is designed to collect the user's gesture instructions on the client, and upload the collected results to the server. Then, the two-stream CNN is designed to recognize the collected data and send the recognition results back to the client. Next, the client-server distributed human-computer interaction system is introduced. Finally, the designed interactive robot is tested. The results show that the recognition accuracy of the designed gesture interactive robot is more than 90%, which can meet the requirements of normal human-computer interaction, and the interactive system has good reliability and stability. Therefore, the gesture interactive robot based on deep learning artificial intelligence technology can be applied to music perception teaching. However, the current research still has some defects. When dual-stream CNN performs video recog-

nition, it is realized by the average sampling of the video. Therefore, the time information contained in the video will be lost in the process of feature learning, which will affect the recognition results of dual-stream CNN. Moreover, the average sampling of video will lead to the allocation of wrong tags, making the recognition result inconsistent with the actual situation. Therefore, in the experiment process, the model will be optimized combined with the updated algorithm to improve the recognition effect of the model, and applied to the actual music classroom teaching.

ACKNOWLEDGEMENT

Art science planning project: This article is the staged research results of Heilongjiang Province art science planning project "Research on the Artistic Value of Promoting Xi Jinping's New Era Music works by Daur nationality in Our Province". B106 number: 2021.

REFERENCES

1. I. Rodriguez, J. M. Martínez-Otzeta, I. Irigoien, and E. Lazkano, "Spontaneous talking gestures using generative adversarial networks," *Robotics and Autonomous Systems*, Vol. 114, 2019, pp. 57-65.
2. M. Jeon, "Robotic arts: Current practices, potentials, and implications," *Multimodal Technologies and Interaction*, Vol. 1, 2017, p. 5.
3. H. Liu and L. Wang, "Gesture recognition for human-robot collaboration: A review," *International Journal of Industrial Ergonomics*, Vol. 68, 2018, pp. 355-367.
4. Q. Wu, S. Wang, J. Cao, B. He, C. Yu, and J. Zheng, "Object recognition-based second language learning educational robot system for Chinese preschool children," *IEEE Access*, Vol. 7, 2019, pp. 7301-7312.
5. Q. Gao, J. Liu, and Z. Ju, "Robust real-time hand detection and localization for space human-robot interaction based on deep learning," *Neurocomputing*, Vol. 390, 2020, pp. 198-206.
6. M. A. Simão, O. Gibaru, and P. Neto, "Online recognition of incomplete gesture data to interface collaborative robots," *IEEE Transactions on Industrial Electronics*, Vol. 66, 2019, pp. 9372-9382.
7. P. Neto, M. Simão, N. Mendes, and M. Safeea, "Gesture-based human-robot interaction for human assistance in manufacturing," *International Journal of Advanced Manufacturing Technology*, Vol. 101, 2019, pp. 119-135.
8. L. M. Gladence, T. Ravi, and M. Karthi, "Heart disease prediction using naive bayes classifier-sequential pattern mining," *International Journal of Applied Engineering Research*, Vol. 9, 2014, pp. 8593-8602.
9. Y. Chen, S. Hu, H. Mao, W. Deng, and X. Gao, "Application of the best evacuation model of deep learning in the design of public structures," *Image and Vision Computing*, Vol. 102, 2020, pp. 103975.
10. R. Qin, C. Zhou, H. Zhu, M. Shi, F. Chao, and N. Li, "A music-driven dance system of humanoid robots," *International Journal of Humanoid Robotics*, Vol. 15, 2018, pp. 18-23.

11. A. Phinyomark and E. Scheme, "EMG pattern recognition in the era of big data and deep learning," *Big Data and Cognitive Computing*, Vol. 2, 2018, p. 21.
12. C. Li, C. Xie, B. Zhang, C. Chen, and J. Han, "Deep fisher discriminant learning for mobile hand gesture recognition," *Pattern Recognition,* Vol. 77, 2018, pp. 276-288.
13. M. Jeon, R. Fiebrink, E. A. Edmonds, and D. Herath, "From rituals to magic: Interactive art and HCI of the past, present, and future," *International Journal of Human-Computer Studies*, Vol. 131, 2019, pp. 108-119.
14. D. Dalmazzo and R. Ramírez, "Bowing gestures classification in violin performance: a machine learning approach," *Frontiers in Psychology*, Vol. 10, 2019, p. 344.
15. M. Schedel, S. Yuditskaya, and S. E. Green, "New interfaces for musical expression (NIME) conference," *Computer Music Journal*, Vol. 43, 2019, pp. 159-166.
16. D. Avola, M. Bernardi, L. Cinque, G. L. Foresti, and C. Massaroni, "Exploiting recurrent neural networks and leap motion controller for the recognition of sign language and semaphoric hand gestures," *IEEE Transactions on Multimedia*, Vol. 21, 2018, pp. 234-245.
17. B. Hu and J. Wang, "Deep learning based hand gesture recognition and UAV flight controls," *International Journal of Automation and Computing*, Vol. 17, 2020, pp. 17-29.
18. Z. Duan, S. Essid, C. C. Liem, G. Richard, and G. Sharma, "Audiovisual analysis of music performances: Overview of an emerging field," *IEEE Signal Processing Magazine*, Vol. 36, 2018, pp. 63-73.
19. I. V. Brito, E. O. Freire, E. A. Carvalho, and L. Molina, "Analysis of cross-cultural effect on gesture-based human-robot interaction," *International Journal of Mechanical Engineering and Robotics Research*, Vol. 8, 2019, pp. 18-29.
20. Y. Lavinia, H. Vo, and A. Verma, "New colour fusion deep learning model for large-scale action recognition," *International Journal of Computational Vision and Robotics*, Vol. 10, 2020, pp. 41-60.
21. M. Almagro, V. Fresno, and F. de la Paz, "Speech gestural interpretation by applying word representations in robotics," *Integrated Computer-Aided Engineering*, Vol. 26, 2019, pp. 97-109.
22. Y. Kim, K. Lee, and U. Oh, "Understanding interactive and explainable feedback for supporting non-experts with data preparation for building a deep learning model," *International Journal of Advanced Smart Convergence*, Vol. 9, 2020, pp. 90-104.
23. J. A. Ruiz-Vanoye, R. A. B. Cámara, O. Díaz-Parra, A. Fuentes-Penna, M. A. Ruiz-Jaimes, and J. Pérez-Ortega, "Editorial: Can machines play musical instruments?" *International Journal of Combinatorial Optimization Problems and Informatics*, Vol. 10, 2019, pp. 1-6.
24. O. Bălan, G. Moise, A. Moldoveanu, M. Leordeanu, and F. Moldoveanu, "An investigation of various machine and deep learning techniques applied in automatic fear level detection and acrophobia virtual therapy," *Sensors*, Vol. 20, 2020, p. 496.
25. F. Alonso-Martín, J. J. Gamboa-Montero, J. C. Castillo, Á. Castro-González, and M. Á. Salichs, "Detecting and classifying human touches in a social robot through acoustic sensing and machine learning," *Sensors*, Vol. 17, 2017, p. 1138.
26. P. G. Esteban, P. Baxter, T. Belpaeme, *et al.*, "How to build a supervised autonomous system for robot-enhanced therapy for children with autism spectrum disorder," *Paladyn, Journal of Behavioral Robotics*, Vol. 8, 2017, pp. 18-38.

27. T. Gifford, S. Knotts, J. McCormack, S. Kalonaris, M. Yee-King, and M. d'Inverno, "Computational systems for music improvisation," *Digital Creativity*, Vol. 29, 2018, pp. 19-36.
28. J. Zheng, Q. Zhang, S. Xu, H. Peng, and Q. Wu, "Cognition-based context-aware cloud computing for intelligent robotic systems in mobile education," *IEEE Access*, Vol. 6, 2018, pp. 49103-49111.
29. J. Joo, F. F. Steen, and M. Turner, "Red Hen Lab: Dataset and tools for multimodal human communication research," *KI-Künstliche Intelligenz*, Vol. 31, 2017, pp. 357-361.
30. D. St-Onge, U. Côté-Allard, K. Glette, B. Gosselin, and G. Beltrame, "Engaging with robotic swarms: Commands from expressive motion," *ACM Transactions on Human-Robot Interaction*, Vol. 8, 2019, pp. 1-26.
31. M. E. Cabrera and J. P. Wachs, "A human-centered approach to one-shot gesture learning," *Frontiers in Robotics and AI*, Vol. 4, 2017, pp. 8-17.
32. S. Dargan, M. Kumar, M. R. Ayyagari, and G. Kumar, "A survey of deep learning and its applications:A new paradigm to machine learning," *Archives of Computational Methods in Engineering*, Vol. 6, 2019, pp. 1-22.
33. N. Jaouedi, N. Boujnah, and M. S. Bouhlel, "A new hybrid deep learning model for human action recognition," *Journal of King Saud University-Computer and Information Sciences*, Vol. 32, 2020, pp. 447-453.
34. H. Liu, T. Fang, T. Zhou, and L. Wang, "Towards robust human-robot collaborative manufacturing:multimodal fusion," *IEEE Access*, Vol. 6, 2018, pp. 74762-74771.
35. K. Tatar and P. Pasquier, "Musical agents: A typology and state of the art towards musical metacreation," *Journal of New Music Research*, Vol. 48, 2019, pp. 56-105.
36. J. J. Gamboa-Montero, F. Alonso-Martin, J. C. Castillo, M. Malfaz, and M. A. Salichs, "Detecting, locating and recognizing human touches in social robots with contact microphones," *Engineering Applications of Artificial Intelligence*, Vol. 92, 2020, p. 103670.
37. G. Lampropoulos, E. Keramopoulos, and K. Diamantaras, "Enhancing the functionality of augmented reality using deep learning, semantic web and knowledge graphs: A review," *Visual Informatics*, Vol. 4, 2020, pp. 32-42.
38. D. Alu, E. Zoltan, and I. C. Stoica, "Voice based emotion recognition with convolutional neural networks for companion robots," *Science and Technology*, Vol. 20, 2017, pp. 222-240.
39. A. Cangelosi and S. Invitto, "Human-Robot Interaction and Neuroprosthetics: A review of new technologies," *IEEE Consumer Electronics Magazine*, Vol. 6, 2017, pp. 24-33.
40. D. Herremans, C. H. Chuan, and E. Chew, "A functional taxonomy of music generation systems," *ACM Computing Surveys*, Vol. 50, 2017, pp. 1-30.
41. O. Alemi, J. Françoise, and P. Pasquier, "GrooveNet: Real-time music-driven dance movement generation using artificial neural networks," *Networks*, Vol. 8, 2017, pp. 26.

Jia-Xin Hu was born in Qiqihar, Heilongjiang Province in 1986. She received her master's degree from Qiqihar University. He is now an Associate Professor in the School of Music and Dance of Qiqihar University. His research interests include vocal singing and vocal music teaching.

Yu Song was born in Yichang, Hubei, P.R. China, in 1989. he received the Master degree from Central Conservatory of Music, P.R. China. Now, he works in Development Research Centre of Music Industry, Communication University of China, His research interests include musical education, object based audio and musical timbre perception.

Yi-Yao Zhang was born in Xuzhou, Jiangsu, P.R. China, in 1991. He received the Master degree from Milan Giuseppe Verdi Conservatory of Music, Italy. Now, he works in School of Art and Communication, Beijing Normal University. His research interests include sound therapy, music perception and opera.