Research Article

Recognition and Prediction Method of Music and Dance Movements Based on G-ResNet SPP and Attention Mechanism

Hongbo Shao

Shangqiu Institute of Technology, Shangqiu 476000, China

Correspondence should be addressed to Hongbo Shao; 1350024003@sqgxy.edu.cn

Received 2 June 2022; Revised 10 August 2022; Accepted 22 August 2022; Published 20 September 2022

Aiming at the problems of difficult action recognition and low prediction accuracy in the process of music and dance movement. This paper proposes a music and dance motion recognition and prediction method based on G-ResNet SPP (Spatial Pyramid Pooling) and attention mechanism, so as to improve the accuracy of motion feature recognition. Firstly, the action recognition model is described, and the related theoretical basis and the construction of sampling function and weight function in action recognition are described. Secondly, the recognition method of the encoder-decoder design framework is described; finally, a group of music and dance sequences are used to verify the system, and the results show that the recognition effect of the G-ResNet + APP + ATTEN model has better performance in different experimental data sets.

1. Introduction

The complex changes in dance movements, and the consistency of music and dance, are one of the challenges facing online video now. Therefore, based on the ST-GCN(Spatial Temporal Graph Convolutional network) model and the progressive prediction model, the dance movement recognition and prediction algorithm show the complex and changeable dance movements more real [1]. First, the original image is compressed to express the action information; then, by extracting the details of the action along the way, the sequence image as input, and finally the features of each attribute are extracted through the spatiotemporal convolution layer. The convolutional layer algorithm transfer function to dig out the relationship between attribute features and dance movements and realize the action correction function of dancers. In addition, the optimized objective function of dance movement matching and the optimized objective function combined with ant colony theory are obtained [2]. Finally, the pheromone evaporation factor was adaptively adjusted according to the pheromone concentration in all music-dance motion segments in the music choreography, and the pheromone update was dynamically performed according to the optimized scheme of dance motion matching [3]. The main challenges existing in the field of music and dance today are the postural recognition of the basic forms of dance and the consistency of music and dance. For such problems, this paper believes that the smoothness of action prediction and action transformation needs to be emphasized. The features experimentally extracted in [4] are 20 joint angles, using 30 joint positions to predict 50 positions from basic poses. The SPM (spatial pyramid matching) classifier of the PPM (Prediction by Partial Matching) model has the best attitude recognition effect [4]. By performing the analysis of loss rate and accuracy, we estimated the origin of these poses, and the test was done on our own Balletto dataset of 120 dancers with an accuracy of 97.14%. A 10-point scale scored the dancers’ expertise predictions of 50 poses, with an accuracy of 68.46% for grouping without postures and 89.80% with postures for [5]. Due to the rapid development of the computing video industry, the current video analysis means have been unable to meet the needs of the industry, so the derived action prediction model has become the main way of the video development industry. It needs to predict the possible state of future actions through the analysis of the current actions and predict human behavior based on incomplete actions. To establish a highly efficient and robust framework for action recognition and prediction, in reference
[6], we investigate the latest techniques for complete action recognition and prediction. Existing models, algorithms, techniques, action databases, evaluation protocols, and possible future directions are systematically discussed. In this work of identification and prediction of human behavior, the linear potential low-dimensional space representing high-dimensional nonlinear behavior is allowed. The algorithm of this paper uses the Balletto data set to extract some typical actions, design experiments to evaluate the performance index of the algorithm, and then in our algorithm, to achieve the high accuracy and low latency of action recognition. To identify and predict human behavior, this paper proposes a hybrid method for anterior and posterior action-aware human action recognition and prediction based on the integration of convolutional neural networks and progressive prediction models [7]. The CNN (convolutional neural network) structure is embedded in the human and object information in the video image, and the identification and prediction of the front and rear video space are completed. The action sequence is analyzed through VMM (Virtual-Machine Monitor) to consider the current, past, and present best action states, so as to maximize the recognition of the dance movements and achieve the highest accuracy. Experimental evaluation of the effectiveness of the method, experimental surface action recognition, and prediction aspects have high accuracy [8]. Unlike conventional experiments, we first need to introduce a deep graph auto encoder for the learning task on symbolic scene graphs, rather than just relying on structured Euclidean data. Our encoder can divide operations into two branches, one for identifying input types and the other for predicting future action mapping. Network output is the detected and predicted by action image type label sets. We benchmark the new model proposed in this paper for different prior approaches on Balletto datasets, and experiments show that our model can have higher accuracy and lower loss rate [9]. A control point-random forest-based action recognition and prediction method is proposed for continuous action recognition of human bone sequences. Traditional methods often identify human behavior by constructing category classifiers and, unlike building an action recognition task, this paper identifies dance action based on the development of behavioral sequences [10]. In this paper, we introduce a new method for action prediction for 3D trajectory recognition, using the most 3D pose estimation of MOCCD (multocular contracting curve density algorithm) to track time with an encoder-decoder framework, relying on the LDT as a measure of similarity between trajectories. The true values obtained from the 10 angles of the same posture were studied while identifying the current action and predicting the next moment of action. The trajectory recognition rate is as high as 99%, and the prediction accuracy is also about 95%. Only a small number of training sequences are needed. In terms of experience, the method proposed in reference [11] has a very high reference utility. Specifically, the memory neural network is used to build an action predictor for all actions. These predicted actions can be synchronized with the music at the next moment. According to the prediction rules and removal rules of each step length, the prediction error can be eliminated, and the last prediction number can be used as the action sequence tag [12]. Pedestrian detection systems are an important part of the driver’s safe journey, and if these systems can identify and predict pedestrian behavior, and even estimate the time that each person crosses the road, the safety of road traffic will be significantly improved. The study in this paper not only focuses on pedestrian action recognition but also predicts whether current pedestrian behavior encounters danger in the future. It proposes a retinal recognition model, using a recurrent neural network to estimate pedestrian crossing time, and performing a recurrent network estimation to identify pedestrian intention, one predicting the time required to cross the road [13]. Deep learning is divided into unsupervised feature learning and supervised feature learning. In this paper, we propose a multimodal learning and show how deep learning is trained to complete the action recognition task. We demonstrate how to use cross-modal feature learning. The representation between learning modes is simulated, and the classifier is trained using audio data and then tested using video data. The experimental validation of the Balletto dataset demonstrated that AVLettes published the best visual speech consensus [14]. Skeleton-based action recognition has now become a popular 3D classification problem, integrating colony structure into deep network structures to learn simpler three-dimensional action identification of colony features. The input colony features are transformed into more desirable colony features by designing a rotating mapping layer. To reduce the dimensionality of high features, we use the log-mapping layer, regularly output the data, and perform the classification. Through the evaluation of standard 3D human action recognition datasets, the final experimental results surface this algorithm due to most traditional deep learning methods [15].

2. Dance Movement Recognition Model

2.1. ST-GCN Network Model. ST-GCN differs from the end-to-end motion recognition model in that it applies GCN to the human behavior recognition system of the skeleton, adds the factor of a spatial relationship between joints, connects the natural connections between joints and the cross-continuous time connections of the same joints, and then constructs multiple space-time map convolutional layers to integrate information [16]. The ST-GCN-based human motion recognition process is shown in Figure 1.

2.2. Construction of Space-Time Map Convolution. The traditional two-dimensional convolution algorithm is [17] for convolution operation by using filter and image pixel matrix. For a filter of size and an input feature matrix with several channels of c, a 2D convolutional operation output at position x can be defined:

$$f_{out}(x) = \sum_{h=1}^{K} \sum_{\omega=1}^{K} f_{in}(p(h, \omega)) \cdot W(h, \omega),$$

(1)

where P is the sampling function, acting on the neighborhood of position x (h, \omega); W as a weight function, a weight vector is provided in c-dimensional space;
Later, by redefining the sampling \( P \) function and the weight function \( W \), you can extend the above convolution formula to graph convolution formula.

### 2.2.1. Sampling and Weight Functions.

For 2D convolution operations, the sampling function \( p(h, w) \) is defined on a pixel matrix centered at position \( x \), with the filter as the region. Define the sampling function on \( v_{ij} \), the neighbor set of the node \( B(v_{ij}) = \{v_{ij} | d(v_{ij}, v_{ij}) \leq D\} \). This \( d(v_{ij}, v_{ij}) \) represents \( v_{ij} \), the \( v_{ij} \) minimum length of any path from to \( [18] \). The sampling function \( P \) in this article selects the adjacent set of \( D = 2B(v_{ij}) \). Therefore, the sampling function can be \( p(v_{ij}, v_{ij}) \) defined as follows:

\[
p(v_{ij}, v_{ij}) = v_{ij}. \quad (2)
\]

The weight function of the graph convolution takes the weight value from each position of the filter and then maps the graph nodes \( B(v_{ij}) \rightarrow \{0, \ldots, K - 1\} \). You can also simplify this operation by breaking down adjacent areas of graph nodes. The simplified weight function is as follows:

\[
w(v_{ij}, v_{ij}) = w(I_{ti \leftrightarrow tij}). \quad (3)
\]

These \( w_c \) are the parameters of the multi-category classifier, \( \oplus \) which represent the channel stitching operation.

### 2.2.2. Construction of Spatial Graph Convolution.

Through the above-given sampling function and weight function, the two-dimensional convolution operation can be calculated to obtain the graph convolution formula on the space:

\[
f_{\text{out}}(v_{ij}) = \sum_{v_{ik} \in B(v_{ij})} Z_{l_{ti}} \cdot f_{\text{in}}(v_{ij}, v_{ik}) \cdot w(v_{ij}, v_{ij}). \quad (4)
\]

where normalized terms \( Z_{l_{ti}}(v_{ij}) = \{|v_{ik} | l_{ti}(v_{ik}) = l_{ti}(v_{ij})\}| \) represent the cardinality of the corresponding subset in order to balance the contribution of different subsets to the output. According to the above-given calculation method, the final graph convolution formula on the space is as follows:

\[
f_{\text{out}}(v_{ij}) = \sum_{v_{ik} \in B(v_{ij})} Z_{l_{ti}}(v_{ij}) \cdot f_{\text{in}}(v_{ij}, v_{ik}) \cdot w(I_{ti}(v_{ij})). \quad (5)
\]

### 2.2.3. Construction of a Space-Time Map Convolution.

When constructing a skeleton spatiotemporal map, you can select a skeleton frame sequence [19] with a time range of \( F \). By applying the spatial map convolution in this sequence range, you can add a time dimension to define the spatiotemporal map convolution formula:

\[
B(v_{ij}) = \{v_{ij} | d(v_{ij}, v_{ij}) \leq K, |q - t| \leq \left\lfloor \frac{F}{2} \right\rfloor \}, \quad (6)
\]

where the \( \Gamma \) time range in the adjacent graph, that is, the time kernel size.

To simplify the space-time graph convolution operation, the adjacent regions of the bone points defined by the sampling function and the weight function can be mapped \( l_{ST} \) with the result of:

\[
l_{ST}(v_{ij}) = l_{ti}(v_{ij}) + (q - t + \left\lfloor \frac{F}{2} \right\rfloor) \times K, \quad (7)
\]

where \( l_{ti}(v_{ij}) \) is \( v_{ti} \) the mapping result of the bone points.

### 3. Dance Movement Prediction Model

The action prediction task aims to predict the action category \( y \) corresponding to future video frames by means of the observed local video \( V \). Given a video \( V^t_i = [I_{t_1}, I_{t_2}, \ldots, I_{t_l}] \), where \( L \) is the total number of frames of the video. Giving the first \( t \)-frame of the observable part of the video \( V^t_i = [I_{t_1}, I_{t_2}, \ldots, I_{t_l}] \), this task requires predicting the action category that will occur from the \( t + 1 \) to the \( L \) frame, that is, giving the action category label price of the unobserved part \( V^F_i = [y_{t+1_1}, y_{t+1_2}, \ldots, y_{t+1_{F}}] \).

This article solves these two problems through a concise and efficient framework, and the framework is named TTPP. First, the long video of the loser is split into multiple video clips without overlap \( [I_{t_1'}, I_{t_2'}, \ldots, I_{t_l'}] \), each containing the same number of consecutive frames; secondly, the video \( g_{\text{enc}} \) clips of the losers are encoded into corresponding fragment features through the feature encoding network \( f_{\text{enc}} \); subsequently, the time attention module aggregates \( t \) consecutive video clip features into historical features \( S_t = g_{\text{enc}}(f_{t')}, \ldots, f_{t_l}) \); finally, the Progressive Prediction module progressively predicts future video features and action category scores [20]. The progressive prediction module shares the progressive prediction module by the initial \( g_{\text{pred}}^0 \) prediction module and parameters \( g_{\text{pred}} \), in which \( g_{\text{pred}}^0 \) the aggregated historical feature \( S_t \) is lost as the module, and the fragment characteristics and corresponding action category scores are predicted at subsequent moments; \( g_{\text{pred}} \) By accumulating previous prediction results and historical features \( S_t \), subsequent features and action prediction results are generated. The model uses an encoder-decoder framework for text processing, so let us briefly introduce the codec framework.

#### 3.1. Encoder-Decoder Framework.

As shown in Figure 2, the encoder-decoder framework is commonly used in the field of text processing. In the case of machine translation, for example, the encoder transforms the loser sentence into an intermediate semantic vector \( C \) [21], which encodes the information of the entire loser statement:

\[
C = f(x_1, x_2, \ldots, x_m), \quad (8)
\]
Figure 2: Encoder-decoder frame diagram.

where \( m \) is the length of the loser sentence and \( f(\cdot) \) is the encoding function. The decoder generates sentence information for the current moment by taking the semantic encoding vector \( C \) and the sentence information of the historical moment that has been generated as the loser. Taking the generation of \( t \)-moment as an example, the decoding process is expressed as follows \( y_t \):

\[
\begin{align*}
    y_1 &= g(C), \\
    y_2 &= g(C, y_1), \\
    & \vdots \\
    y_t &= g(C, y_1, y_2, \ldots, y_{t-1}),
\end{align*}
\]

(9)

where \( g(\cdot) \) the decoder function is represented. The decoder takes the result of each decode as input to the next decade, iteratively outputting a decoding sequence of length \( n \). The above-given joint distribution can be decomposed into an ordered conditional probability distribution:

\[
p(y) = \prod_{t=1}^{n} p(y_t|y_1, y_2, \ldots, y_{t-1}, C),
\]

(10)

where \( y = \{y_1, y_2, \ldots, y_n\} \) is the decoding sequence of the output.

Similarly, the progressive motion prediction model based on time attention designed in this chapter uses the time attention module TTM as the encoder and the progressive prediction module PPM as the decoder. Next, the design ideas for both modules will be highlighted.

### 3.2. Progressive Prediction Module

Inspired by the WaveNet model, this chapter elaborately designs the progressive prediction module PPM to mine aggregated historical information to produce more accurate action prediction results. The PPM module consists of an initial prediction submodule and a parameter-sharing progressive prediction submodule, each consisting of two fully connected layers, a ReLU activation function, and a layer normalized LN [22].

Suppose that the future \( t + 1 \) to \( t + l \) total of \( l \) time points will be predicted, and the first time point in the future \( t + 1 \) will be at the moment; the initial prediction module uses the aggregated historical information as the \( S_i \) \( \in \mathbb{R}^{d_m} \) action characteristics and action probability scores of the loser to predict the future \( t \tilde{f}_{t+1} \) \( \in \mathbb{R}^{d_m+1} \) moment \( \tilde{p}_{t+1} \) \( \in \mathbb{R}^C \). This process is expressed as follows:

\[
\begin{align*}
    p_i &= \text{Softmax}(W_c f_i), \\
    \tilde{f}_{t+1} &= g_{\text{pred}}(S_i \oplus \tilde{f}_{t+1} \oplus \tilde{p}_{t+1}), \\
    \tilde{p}_{t+1} &= \text{Softmax}(W_c \tilde{f}_{t+1}).
\end{align*}
\]

(11)

At other points in the future \( t + i (i > 1) \), the feature representation predicted at the previous moment \( t + i - 1 \) is predicted Stitch together \( \tilde{f}_{t+1-i} \) the action probability score \( \tilde{p}_{t+1-i} \) and aggregation features \( S_i \) on the channel and send people to the progressive prediction module. This process is expressed as follows:

\[
\begin{align*}
    \tilde{f}_{t+i} &= g_{\text{pred}}(S_i \oplus \tilde{f}_{t+i} \oplus \tilde{p}_{t+i-1}), \\
    \tilde{p}_{t+i} &= \text{Softmax}(W_c \tilde{f}_{t+i}).
\end{align*}
\]

(12)

Due to the channel stitching operation, the loser dimension of the parameter-sharing progressive prediction module is \( 2d_m + C \). Both submodules have the same structure and consist of two fully connected layers. Among them, the first fully connected layer first reduces the feature dimension to \( d_m/2 \), and the second fully connected layer raises the feature dimension to \( d_m \) [23], using feature transformation to learn effective feature representation. It is worth noting that the progressive prediction submodule is parameter-sharing, so the entire PPM module satisfies lightweight network design.

### 3.3. Model Training with Loss Function

The entire TTP framework can achieve end-to-end training under the action of the supervision signal of the PPM prediction module. Specifically, two types of loss functions are used: feature reconstruction loss \( L_r \) and action classification loss \( L_c \) [24]. Here, \( L_r \) by measuring the loss of mean squared error between the predicted feature and the corresponding truth feature, is defined as follows:

\[
L_r = \frac{1}{i} \sum_{i=1}^{l} \| \tilde{f}_{t+i} - f_{t+i} \|^2,
\]

(13)

where \( \tilde{f}_{t+i} \) it is the prediction feature of the \( t + f_{t+i} \) moment and the truth value feature of the \( t + i \) moment. \( L_c \) is the sum of cross-entropy losses at all predicted moments, defined as follows:

\[
L_c = -\sum_{i=1}^{l} \sum_{j=1}^{C} y(t+i, j) \log \tilde{p}(t+i, j),
\]

(14)

where \( y(t+i,: \) is the truth vector one-hot encoding of the \( t + i \) moment. The optimization objective function in this chapter consists of a feature reconstruction loss and a categorical loss function, expressed as follows:

\[
\text{Loss} = L_c + \lambda L_r,
\]

(15)

where \( \lambda \) is the hyper parameter used to balance the two loss functions.
4. Music and Dance Movement Loss Function Optimization Design

The following are the various loss functions used in the three network components of NTS-Net and their optimization process.

4.1. Filter for Losses. First, in the filtering network, we mark M information regions as distinct, and \( R = \{ R_1, R_2, \ldots, R_M \} \) the amount of information they contain is expressed as [25]. \( I = \{ I_1, I_2, \ldots, I_M \} \). At the same time, in optimizing the network, we will do this. The confidence level of \( M \) regions is expressed as \( C = \{ C_1, C_2, \ldots, C_M \} \). Then, the loss function of the filter network can be defined as follows:

\[
L_I(I, C) = \sum_{(i,j): \, C_i < C_j} f(I_j - I_i),
\]

where the function \( f \) is a nonincreasing function that represents: if \( C_i > C_j \), then \( I_j > I_i \). In the experiment, for the function \( f \), we use the hinge loss function, which has the formula:

\[
f(x) = \max \{1 - x, 0\}. \tag{17}
\]

The expectation of the loss function for \( (I, C) \) is that \( I \) and \( C \) are in the same order. The loss function of the filtered network is differentiable and using the chain rule in back-propagation, it is possible to calculate its \( W_I \) derivative about \( \frac{\partial L_I(I, C)}{\partial W_I} = \sum_{(i,j): \, C_i < C_j} f'(I_j - I_i) \cdot \left( \frac{\partial f(x)}{\partial W_I} \bigg|_{x=R_i} - \frac{\partial f(x)}{\partial W_I} \bigg|_{x=R_j} \right). \tag{18}\]

This equation can be derived directly from \( I_i = I(R_i) \) the definition.

4.2. Optimization Loss. We define the loss of optimizing the network \( L_C \) as follows:

\[
L_C = - \sum_{i=1}^{M} \log C(R_i) - \log C(X), \tag{19}
\]

where \( C \) is the confidence function that represents the degree of truth for the specified area. The first term of equation (19) represents the sum of all losses, and the second term represents the cross-entropy loss of the complete image.

4.3. Check for Losses. When the screening network obtains the \( K \{ R_1, R_2, \ldots, R_K \} \) regions with the most information, the inspection network will get fine-grained recognition results \( P = S(X, R_1, R_2, \ldots, R_K) \). We use cross-entropy loss as a classification loss, which is expressed as follows:

\[
L_S = -\log S(X, R_1, R_2, \ldots, R_K). \tag{20}
\]

4.4. Joint Losses. In the end, we combine the various losses for joint training. The final complete loss function is defined as follows:

\[
L_{\text{total}} = L_I + \lambda \cdot L_S + \mu \cdot L_C, \tag{21}
\]

where \( \lambda \) and \( \mu \) are hyperparameters, in this experiment we all set them to 1. We end up using the stochastic gradient descent method to optimize \( L_{\text{total}} \).

5. Case Studies

5.1. Evaluation Indicators. In the experiment, the prediction accuracy of the calculation model for action prediction is in the form of average Euler error, specifically by comparing each frame sequence to obtain the corresponding frame sequence result, and then drawing the observable result by means of a line chart, of which the Euler angle is calculated as follows:

\[
\text{Error}_i = \sqrt{\sum_j \left( \text{Real}_{ij} - \text{Predict}_{ij} \right)^2}, \tag{22}
\]

\( \text{Error}_i \), where the error of the \( i \)-frame \( \text{Real}_{ij} \) is represented, the value of the Euler angle of the real data at the \( j \)-th of the \( i \)-frame, and \( \text{Predict}_{ij} \) the predicted data at the \( j \)-th of the \( i \)-frame. The numeric value of Euler angles. After finding the sum of the squares of the errors of the two, the Euler error of this frame can be obtained by finding the root number.

5.2. Datasets. In this paper, 250 dance movements in the Ballet database will be used for video decomposition, from which some key frame movement pictures of ballet dance will be obtained, and some of the selected data will be experimented with using the ST-GCN model and the progressive prediction model to verify the effectiveness of the algorithm as shown in Figure 3.

5.3. Experimental Results and Analysis. In this paper, SPP (Spatial Pyramid Pooling) is added to ResNet34, and then the attention mechanism (Atten) is fused with the GRU network, and the pyramid pooling and attention mechanism are introduced. The FSAG-ResNet network model and the G-ResNet network model were first experimented on the UCF101 dataset, UCF101. The comparison of training loss and accuracy of the dataset is shown in Figure 4. As can be seen from Figure 4(a), the improved network after increasing the SPP layer is smaller than the G-ResNet network model. On the basis of increasing the SPP layer and then integrating the attention mechanism, the network training loss is smaller than increasing the SPP layer and compared with the G-ResNet network, the final network training loss is smaller, indicating that the network is easier to optimize. According to Figure 4(b), after adding the SPP layer, the accuracy of the network in the UCF101 data set is improved compared with the G-ResNet network, and after integrating the attention mechanism, the accuracy is improved compared with the original network, and the accuracy is also improved compared with increasing the SPP layer.

Similarly, the improved network is experimented on the HMDB51 dataset, and the training loss and accuracy of the
HMDB51 dataset before and after the network improvement are compared as shown in Figure 5.

As can be seen from Figures 5(a) and 5(b), the training loss of the network in the HMDB51 dataset becomes smaller and the value is smaller after increasing the SPP layer and attention, and the network is in HMDB51. The accuracy of the dataset is also improved compared to the original network. Experimental results from the above-given two datasets show that the improved G-ResNet network model has less training loss and higher accuracy, which verifies the superiority of the proposed model in this paper.

The comparison results of the improved G-ResNet network model with the G-ResNet database in the UCF101 database and the HMDB51 database are counted, and the identification accuracy rate of the two databases is shown in Figure 6 as shown.

Compared with the results of the G-ResNet experiment, the SPP layer was added after UCF101 the recognition rate on the HMDB51 reached 95.6%, an increase of 3.2%, and on the HMDB51 the recognition rate reached 62.6%, an increase of 4.2%. After adding attention mechanisms on the basis of SPP, the recognition rate on UCF101 was 96.3%, an increase of 0.8%, and on HMDB51 the recognition rate on 64.6% increased by 2.1%. Compared to the G-ResNet network model, the FSAG-ResNet network model presented in this chapter ends up in UCF101. The recognition rate on
HMDB increased by 3.6% at HMDB51. The recognition rate increased by 6.0%, and from the results, pyramid pooling was introduced for ResNet34 on the basis of the G-ResNet network. After the grus network fusion attention mechanism, the recognition rate of the model on the UCF101 and HMDB51 datasets has been improved, which proves the effectiveness of the improvement method.

This paper conducts a user survey on the authenticity of dance and the nature of dance and music. The dance-generated videos of the five different models were scored by 100 observers, and the average value of each score was calculated, obtaining the true user evaluation of each model.

As can be seen from Figure 7, the user of model 5 has the highest evaluation, which reflects the higher authenticity of the video processed by the model.

As seen from Figure 8, our model is better than the other models in terms of musical consistency. Statistically, the Balletto dataset had higher musical consistency than other categories of dance. It shows that the Balletto choreography is more consistent with the music. Specifically, the Model-scored 4.56 on the Kpop dataset, 2.89 on the Poppin dataset, 3.21 on the Hiphop dataset, Model 2 scored 5.61 on the Kpop dataset.
dataset, 4.33 on the Poppin dataset, 4.21 on the Hip hop dataset; Model 3 scored 6.56 on the Kpop dataset, 5.33 on the Poppin dataset and 5.41 on the Hip hop dataset.

From the score of Model Four, it can be seen that our model has received the best user reviews relative to other models in terms of dance authenticity and musical consistency.

6. Conclusion

In this paper, the current situation of the computer video network industry in the field of music and dance movement recognition and prediction, as well as the application prospect of deep learning research, and then mainly for the use of deep learning, the identification and prediction of music and dance movements are proposed ST-GCN model and progressive prediction model based on attention mechanism. The main work of this article is as follows:

(1) Establish a dance movement recognition model. First, the spatial graph convolution is constructed by sampling functions and weight functions; then the constructing weight function is simplified; finally, the action recognition model under the new partitioning strategy is constructed. Experiments show that the accuracy of this model is greatly improved compared with the original model.

(2) Progressive action prediction based on temporal attention. This paper uses an attention mechanism to capture historical information, well supports parallel transportation, and combines the idea of iterative decoding of the neural network to make progressive action feature prediction. On the dataset UCF101, the model outperforms the ordinary codec model in performance and efficiency, which further verifies the effectiveness of the model.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declared that they have no conflicts of interest regarding this work.

References

[1] V. Bloom, V. Argyriou, and D. Makris, “Linear latent low dimensional space for online early action recognition and prediction,” Pattern Recognition, vol. 72, pp. 532–547, 2017.

[2] J. Zhang, P. Wang, and R. X. Gao, “Hybrid machine learning for human action recognition and prediction in assembly,” Robotics and Computer-Integrated Manufacturing, vol. 72, Article ID 102184, 2021.

[3] M. M. Briegel, “Multi-agent action recognition and prediction using support vector machines [J],” University of Birmingham, vol. 35, no. 30, pp. 703–707, 2012.

[4] Y. Yin, N. Sun, and M. Liu, “Action recognition and prediction of human skeleton based on BSCPs-RF [J],” Jaipur/Robot, vol. 39, no. 6, pp. 795–802, 2017.

[5] S. Cho and H. Foroosh, “A Temporal Sequence Learning for Action Recognition and Prediction,” in Proceedings of the IEEE Winter Conference on Applications of Computer Vision, pp. 352–361, Lake Tahoe, NV, USA, March 2018.

[6] D. O. Pop, A. Rogozan, and C. Chatelain, “Multi-Task Deep Learning for Pedestrian Detection, Action Recognition and Time to Cross Prediction,” IEEE Access, vol. 7, no. 99, p. 1, 2019.

[7] S. Abdussami, S. Singh and A. Thyagarajamurthy, A review on action recognition and action prediction of human(s) using deep learning approaches, “International Journal of Computer Application,” vol. 177, pp. 1–5, 2019.

[8] I. Fatima, M. Fahim, Y. K. Lee, and S. Lee, “A unified framework for activity recognition-based behavior analysis and action prediction in smart homes,” Sensors, vol. 13, no. 2, pp. 2682–2699, 2013.

[9] G. A. Carpenter, “Distributed learning, recognition, and prediction by ART and ARTMAP neural networks,” Neural Networks, vol. 10, no. 8, pp. 1473–1494, 1997.

[10] Z. Wang, C. Zou, and W. Cai, “Small sample classification of hyperspectral remote sensing images based on sequential joint deep learning model,” IEEE Access, vol. 8, pp. 71353–71363, 2020.

[11] Marsojo, “Ancient Chinese named entity recognition based on deep learning,” Computer Science and Application, vol. 10, no. 07, pp. 1359–1366, 2020.

[12] U. N. DP. Summary, “Human development report 2002: deepening democracy in a fragmented world,” [J]. Epfl, vol. 36, no. 2, pp. 205–206, 2002.

[13] J. Zhang, L. Dai, and F. Cheng, “Corn seed variety classification based on hyperspectral reflectance imaging and deep convolutional neural network,” Journal of Food Measurement and Characterization, vol. 15, no. 1, pp. 484–494, 2020.

[14] S. O. Long, Y. S. Akande, and R. W. Purdy, “Deepening learning and inspiring rigor: bridging academic and experiential learning using a host country approach to a study tour [J],” Journal of Studies in International Education, vol. 14, no. 1, pp. 89–111, 2010.

[15] X. Sun, A. Zimmer, S. Mukherjee, N. K. Kottayil, P. Ghuman, and I. Cheng, “DeepInSAR—a deep learning framework for SAR interferometric phase restoration and coherence estimation,” Remote Sensing, vol. 12, no. 14, p. 2340, 2020.

[16] M. Safaei, P. Balouchian, and H. Foroosh, “TICNN: a hierarchical deep learning framework for still image action recognition using temporal image prediction,” in Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), pp. 3463–3467, Athens, Greece, October 2018.

[17] J. Gall, A. Yao, N. Razavi, L. Van Gool, and V. Lempitsky, “Hough forests for object detection, tracking, and action recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, pp. 2188–2202, 2011.

[18] Y. Demiris, “Prediction of intent in robotics and multi-agent systems,” Cognitive Processing, vol. 8, no. 3, pp. 151–158, 2007.

[19] L. Meng, Y. Liang, and Q. Wang, “Group sparse regression-based learning model for real-time depth-based human action prediction,” Mathematical Problems in Engineering, vol. 2018, no. 1, pp. 1–7, 2018.

[20] T. Liu, K. Chen, and X. Dai, “Action recognition fused with RGB-D multi-modal features and radius-margin bounded extreme learning,” Moshi Shibie ju Rengong Zhineng/Pattern
[21] S. Gaglio and G. Martorella, “An AMI system for user daily routine recognition and prediction,” Advances in Intelligent Systems and Computing, vol. 260, pp. 33–45, 2014.

[22] C. Rother, V. Kolmogorov, and A. Blake, “‘Grabcut’: interactive foreground extraction using iterated graph cuts [J],” ACM Transactions on Graphics, vol. 23, no. 3, pp. 309–314, 2004.

[23] S. Q. Ren, K. M. He, R. Girshick, and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137–1149, 2017.

[24] K. M. He, G. Gkioxari, P. Dollar, and R. Girshick, “Mask R-CNN,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 2, pp. 386–397, 2020.

[25] S. Han, W. Tao, X. Wu, Xc Tai, and T. Wang, “Fast image segmentation based on multilevel banded closed-form method,” Pattern Recognition Letters, vol. 31, no. 3, pp. 216–225, 2010.