Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2023.039004
Article

A Multi-Task Motion Generation Model that Fuses a Discriminator and a Generator

Xiuye Liu and Aihua Wu*

School of Information Engineering, Shanghai Maritime University, Shanghai, 201306, China
*Corresponding Author: Aihua Wu. Email: ahwu@shmtu.edu.cn
Received: 07 January 2023; Accepted: 10 April 2023; Published: 09 June 2023

Abstract: The human motion generation model can extract structural features from existing human motion capture data, and the generated data makes animated characters move. The 3D human motion capture sequences contain complex spatial-temporal structures, and the deep learning model can fully describe the potential semantic structure of human motion. To improve the authenticity of the generated human motion sequences, we propose a multi-task motion generation model that consists of a discriminator and a generator. The discriminator classifies motion sequences into different styles according to their similarity to the mean spatial-temporal templates from motion sequences of 17 crucial human joints in three-freedom degrees. And target motion sequences are created with these styles by the generator. Unlike traditional related works, our model can handle multiple tasks, such as identifying styles and generating data. In addition, by extracting 17 crucial joints from 29 human joints, our model avoids data redundancy and improves the accuracy of model recognition. The experimental results show that the discriminator of the model can effectively recognize diversified movements, and the generated data can correctly fit the actual data. The combination of discriminator and generator solves the problem of low reuse rate of motion data, and the generated motion sequences are more suitable for actual movement.

Keywords: Human motion; discriminator; generator; human motion generation model; multi-task processing performance; motion style

1 Introduction

The human motion generation model aims to generate more motion sequences through limited data. Since the 1970s, motion capture technology has been applied to video animation. With the rapid development of computer software and hardware technology, more and more motion capture devices appear. Motion capture technology has been widely used in military, entertainment, sports, and robot technology. However, motion capture devices have a lot of defects, and the data captured by different motion capture devices are not alike in authenticity and richness. To capture motion data that meets the demand, which will consume a lot of resources and materials. Although the types of motion in the motion capture database are more and more abundant, they cannot directly meet user needs. How to

effectively generate actual target motion sequences based on existing data has become a problem to be solved.

In recent years, the generalization ability of deep learning has become more and more powerful. More and more researchers have paid attention to the application of deep learning in motion generation. Many unlabeled and style-limited sequences reduce the possibility of reusing existing motion sequences. So motion style recognition and motion data generation have become two major research hotspots. Motion style recognition technology mainly analyzes frame segment and single-frame behavior. Motion generation technology aims at reconstructing the motion sequence by extracting the dynamic features from high-dimensional motion sequences that can fit actual motion. Therefore, according to the generative adversarial network (GAN) [1], we create a multi-task motion generation model that fuses a discriminator and a generator, as shown in Fig. 1.

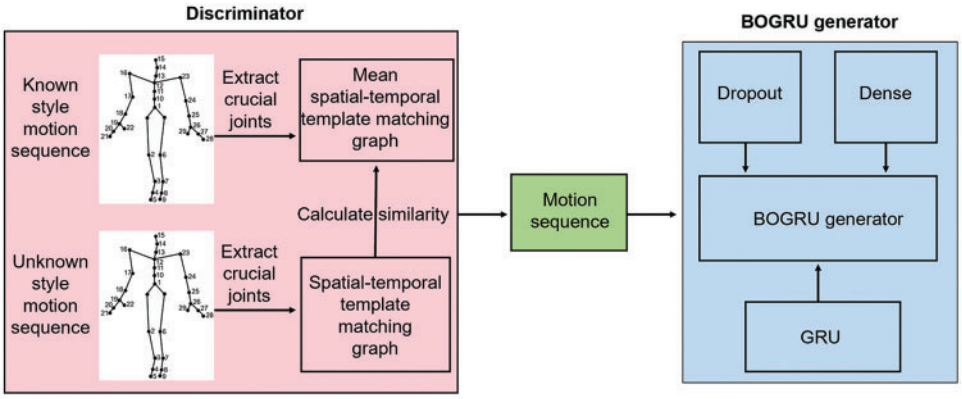


Figure 1: Multi-task motion generation model framework based on discriminator and generator

How to effectively identify and label complex movements, researchers have proposed different processing methods. Among them, the two representative methods are template distance measurement and machine learning based on a probability model. Inspired by the template measurement method and according to the periodic and continuous features of human motion joints, we construct a discriminator. It firstly establishes the semantic feature template structure based on the motion sequences, then calculates the similarity between the motion to be classified and all templates, and finally selects the motion label with the maximum similarity. Kadu et al. [2] used the tree-structure vector quantization method to establish the codeword template to approximate the motion posture. Kapsouras et al. [3] used the joint angle rotation data to construct the posture vector and the forward difference vector as the feature template of each motion. Raptis et al. [4] used the cascade classifier of multivariate time series data and the distance measurement method based on the dynamic time warping (DTW) [5] to recognize the dance moves. Compared with studies on segmented motion recognition, there are few studies on continuous motion recognition. By observing the motion capture data, we found that the joint angle of the human body changes periodically during the movement. Therefore, we have established a mean spatial-temporal template matching graph for each motion based on the joint angle data of the human. We use the DTW algorithm to calculate the similarity between the movement to be classified and all templates. Finally, we get the final recognition results based on the discrimination results of human joints in three degrees of freedom.

As known, there is a complex nonlinear structure between limb joints and a strict space-time dependence between adjacent motion frames. Because deep learning can learn essential features

from a few samples, the related works mainly focus on the spatial-temporal generative model. Taylor et al. [6,7], Chiu et al. [8], and Gan et al. [9] established a human motion generation model based on the restricted Boltzmann machine (RBM) [10]. RBM generative model mainly extracts low-dimensional features for coding, then reconstructs the original data according to the extracted features and decodes the data. The model obtains the probability distribution of the visible data through cyclic sampling. To solve the problem that the RBM model can not establish a long-term dependent model, Harvey et al. [11], Bayer et al. [12], Park et al. [13], and Mattos et al. [14,15] establish a human motion generation model based on recurrent neural network (RNN) [16]. These models realize the feedback of historical information through the autoregressive connection between hidden layers and generate the specified style motion sequences at any time using the limited motion data set. However, the RNN often encounters problems such as gradient explosion and long-term information forgetting. The gate recurrent unit (GRU) [17], based on the RNN structure, with a forgetting gate, has no such problems. So we created a generator based on the gated recurrent unit (BOGRU). We adjusted the number of layers of the GRU neural network according to the motion capture data and added the Dropout layer to avoid overfitting problems. Finally, the Dense layer generated the joint data we needed.

2 Discriminator

Traditional motion recognition technology focuses on analyzing the motion behavior of a frame segment or single frame. The recognition rate of these models depends on the time frame. If the data of a time frame is wrong, the result will be an error. According to the continuous motion data, we found that the joint angle of the human changes periodically during the moving process. Therefore, we proposed a discriminator based on the template distance measurement method and solved the problem of time frame mismatching through the DTW. The discriminator does not need a complex feature extraction algorithm, and continuous motion data avoids accidental data recognition errors in the model.

2.1 Build Mean Spatial-Temporal Template Matching Graph

The spatial-temporal template matching graph is a broken line graph that reflects the changing trend of each joint angle during human movement. Our motion capture data comes from CMU and HDM05, two public databases, and the data in the capture file is the Euler angle of the human joint. It contains 29 joints, as shown in Fig. 2. Although the captured data is discrete, we show that the joint angle changes periodically during the continuous movement of the human body through the line graph. Different individuals have the same shape of motion curve. Still, their many characteristics are different when moving, such as joint angle, speed, and acceleration. To diminish individual differences, we averaged the spatial-temporal template matching graph. Algorithm 1 shows the process of establishing the mean spatial-temporal template matching graph.

Motion Capture Sequence: The human motion capture sequence record 29 joints' angle data at 59 freedom degrees. Because the human body has soft structure characteristics and conforms to biological motion characteristics, each joint has a different number of freedom degrees, which has at most three freedom degrees, respectively x, y, and z. The dimension of the T-frame motion capture sequence is 59, and each dimension records the joint angle of each freedom degree.

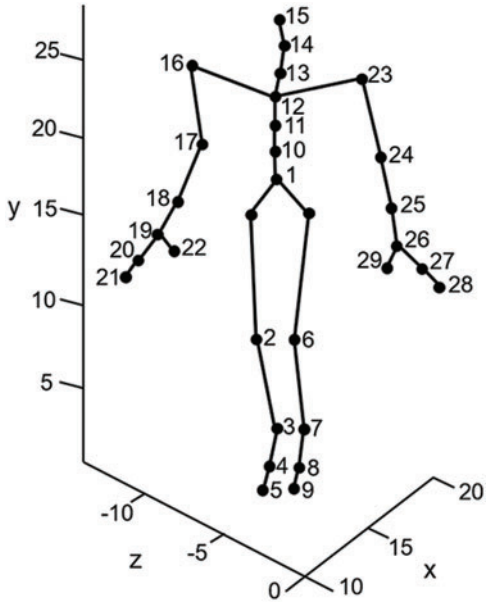


Figure 2: Human skeleton diagram

Spatial-Temporal Template Matching Graph: In the motion capture sequence with time length T, the motion sequence of joint i on the f freedom degree is $\boldsymbol{E}_{\boldsymbol{T}}^{\boldsymbol{i-f}} = \left\{e_1^{i-f}, e_2^{i-f}, e_3^{i-f}, \ldots, e_t^{i-f}, \ldots, e_T^{i-f}\right\}$, where $i = \{1, 2, 3, \ldots, 29\}$, representing human joints; $f = \{x, y, z\}$, representing joint freedom degrees. The spatial-temporal template matching graph is the line graph formed by the motion capture sequence $\boldsymbol{E}_{\boldsymbol{T}}^{\boldsymbol{i-f}} = \left\{e_1^{i-f}, e_2^{i-f}, e_3^{i-f}, \ldots, e_t^{i-f}, \ldots, e_T^{i-f}\right\}$. It reflects the joint angles are changing the trend of a joint in one freedom degree, so 59 spatial-temporal template matching graphs are constructed from the human motion sequence of 59 freedom degrees.

Mean Spatial-Temporal Template Matching Graph: Under the same motion, compare the spatial-temporal template matching graphs of different individuals in the same dimension, find the corresponding time frame of two individuals in the same posture through the DTW algorithm, average the Euclidean distance between the two, and add the smaller with the mean to get the joint angle data that eliminates the difference between the two individuals. The spatial-temporal template matching graph composed of all joint angles that eliminate individual differences is called the mean spatial-temporal template matching graph.

Under the same motion, compare the spatial-temporal template matching graphs constructed by the individual p and individual $p + 1$ motion sequences $\boldsymbol{E}_{\boldsymbol{pT}}^{\boldsymbol{i-f}}$ and $\boldsymbol{E}_{\boldsymbol{(p+1)Q}}^{\boldsymbol{i-f}}$, where Q represents the time length of the motion sequence. Due to the influence of human bone length, movement speed, and acceleration, the time frame corresponding to the same movement posture of different individuals is different. Through the DTW algorithm, we can find the corresponding time frames t and q of two individuals in the same pose and calculate their Euclidean distance as Eq. (1):

$$\mathrm{d}_{\mathrm{tq}} = \left|e_{pt}^{i-f} - e_{(p+1)q}^{i-f}\right| \tag{1}$$

where $|t-q| \leq 3$. Then, average $\mathrm{d_{tq}}$ and compare the joint angles e_{pt}^{i-f} and $e_{(p+1)q}^{i-f}$. The joint angle e_t^{i-f} that eliminates the difference between the two individuals is the sum of the smaller joint angle and the average Euclidean distance $\frac{1}{2}\mathrm{d_{tq}}$ as Eq. (2):

$$e_t^{i-f} = \frac{1}{2}\mathrm{d_{tq}} + \min\left\{e_{pt}^{i-f}, e_{(p+1)q}^{i-f}\right\} \tag{2}$$

After all the corresponding frames of individual p joint angle sequence $\boldsymbol{E}_{\boldsymbol{pT}}^{\boldsymbol{i-f}}$ and individual $p+1$ joint angle sequence $\boldsymbol{E}_{(\boldsymbol{p+1})\boldsymbol{Q}}^{\boldsymbol{i-f}}$ are averaged, the joint angle sequence $\boldsymbol{E}_{\boldsymbol{T}}^{\boldsymbol{i-f}}$ that eliminates the difference between the two individuals is obtained as Eq. (3):

$$\boldsymbol{E}_{\boldsymbol{T}}^{\boldsymbol{i-f}} = \left\{e_1^{i-f}, e_2^{i-f}, e_3^{i-f}, \ldots, e_t^{i-f}, \ldots, e_T^{i-f}\right\} \tag{3}$$

We extracted the motion capture sequence of m individuals and constructed mean spatial-temporal template matching graphs to eliminate the difference of m individuals according to Algorithm 1.

Algorithm 1: Build mean spatial-temporal template matching graph

Input:
$\boldsymbol{I}$: human joints i
$\boldsymbol{F}$: degree of freedom f of human joint angle
$\boldsymbol{E}$: joint angle sequences of m individuals $\{\boldsymbol{E}_1, \boldsymbol{E}_2, \boldsymbol{E}_3, \ldots, \boldsymbol{E}_m\}$
T, Q: duration of human joint angle sequence.
Output: $\boldsymbol{E}_{\boldsymbol{T}}^{\boldsymbol{i-f}}$.

For $p = 1 to m$
  For $t = 1 to T$
    For $q = 1 to Q$
      Find the corresponding time frames t and q of $\boldsymbol{E}_{\boldsymbol{p}}^{\boldsymbol{i-f}}$ and $\boldsymbol{E}_{\boldsymbol{p+1}}^{\boldsymbol{i-f}}$ based on DTW;
      Calculate Euclidean distance $\mathrm{d_{tq}}$ by Eq. (1);
      Mean Euclidean distance by $\frac{1}{2}\mathrm{d_{tq}}$;
      The joint angle $\boldsymbol{e}_{\boldsymbol{t}}^{\boldsymbol{i-f}}$ to eliminate the difference between two individuals is obtained by Eq. (2);
      $t = t + 1$;
    End For
  End For
  Return the joint angle sequence $\boldsymbol{E}_{\boldsymbol{T}}^{\boldsymbol{i-f}}$ that eliminates the differences between individual p and individual $p+1$ by Eq. (3);
  $p = p + 1$;
  $\boldsymbol{E}_{\boldsymbol{p}}^{\boldsymbol{i-f}} = \boldsymbol{E}_{\boldsymbol{T}}^{\boldsymbol{i-f}}$;
End For
Return the joint angle sequence $\boldsymbol{E}_{\boldsymbol{T}}^{\boldsymbol{i-f}}$ that eliminates the varies of m individuals.

2.2 Extract Crucial Joints

Crucial Joint: When the joint plays a decisive role in human motion, and the recognition accuracies of all its freedom degrees are not less than 99.5%, the joint is a crucial joint.

Voter: The voter is a counter that records the number of times the discriminator correctly recognizes the motion capture sequence.

The discriminator recognizes the motion style based on the human joint angle sequence. By constructing the mean spatial-temporal template matching graphs for all joints, we found that some joints have small periodic changes, and these joints have the same motion template matching map in different movements. Therefore, combining all joints to identify motion will cause data redundancy and reduce the accuracy of model recognition. We extract crucial joints from 29 human joints according to Algorithm 2. We extracted k motion sequences and analyzed the accuracy of joint i identifying k samples on f freedom degree. According to the DTW, the similarity between the motion sequence of joint i on f freedom degree and the mean spatial-temporal template matching graphs corresponding to all motions are calculated. The recognition result is the motion with the highest similarity. After each sample data is correctly identified, the voter adds one as Eq. (4):

$$count_{i-f} = count_{i-f} + 1 \tag{4}$$

After recognizing k motion sequences, calculate the recognition accuracy of the discriminator for joint i on f freedom degree as Eq. (5):

$$R_{i-f} = \frac{count_{i-f}}{k} \tag{5}$$

If the recognition accuracies of joint i in all freedom degrees are not less than 99.5%, joint i is a crucial joint.

Algorithm 2: Extract crucial joints

Input:
$\boldsymbol{I}$: human joints i
$\boldsymbol{F}$: f-freedom degree of human joint angle
$\boldsymbol{B}$: mean spatial-temporal template matching graph of various movements
$\boldsymbol{E}$: k human motion joint angle sequences $\{\boldsymbol{E_1}, \boldsymbol{E_2}, \boldsymbol{E_3}, \ldots, \boldsymbol{E_k}\}$
$count_{i-f}$: the number of times the sequence of joint i on f freedom degree is correctly recognized.
Output: crucial joints.

$count_{i-f} = 0$;
For $i = 1$ to 29;
  For $f \in \{x, y, z\}$
    For $j = 1$ tok
      Calculate the similarity between the sample of $\boldsymbol{E}_j^{i-f}$ and B based on DTW;
      Identify style = the style with the greatest similarity;
      If identify style == correct movement
        $count_{i-f} = count_{i-f} + 1$;
      End If
    End For
    $R_{i-f} = \frac{count_{i-f}}{k}$;
  End For
  If the correct rate $R_f = \frac{count_f}{k}$ on the joint i with all freedom degrees $\geq$ 99.5%
    Joint i is a crucial joint;
  End If
End For
Return crucial joints.

2.3 Motion Style Recognition

The goal of the discriminator is to recognize the movement style of the human joint angle sequences. The mean spatial-temporal template matching graphs are composed of the crucial joints' angle sequences as the comparison object of the discriminator. Calculate the similarity between the data of each joint in each freedom degree in the motion sequence to be recognized and the corresponding mean spatial-temporal template matching graph according to DTW, in which the result of each recognition is the motion with the highest similarity. The voter records the recognition results of each sequence. Combined with all crucial freedom degrees, add one to the number of votes for the motion identified on each freedom degree. If the maximum number of votes is not less than $\frac{9}{5}h$, the recognition result is the motion with the highest number of votes, where h represents the number of crucial joints. Otherwise, the individual's movement style is not in the database, such as in Algorithm 3. When the number of votes is not less than $\frac{9}{5}h$, the discriminator not only avoids the over-fitting phenomenon of the model but also achieves 99.7% accuracy.

Algorithm 3: Motion style recognition

Input:
$\boldsymbol{E}$: human motion joint angle sequences $\{\boldsymbol{e}_{1x}, \boldsymbol{e}_{1y}, \boldsymbol{e}_{1z}, \boldsymbol{e}_{2x}, \boldsymbol{e}_{2y}, \boldsymbol{e}_{2z}, \ldots, \boldsymbol{e}_{hx}, \boldsymbol{e}_{hy}, \boldsymbol{e}_{hz}\}$
$\boldsymbol{B}$: mean spatial-temporal template matching graph of various movements
$count_{style\_q}$: number of votes for sports style q.
Output: motion style.

$count_{style\_q} = 0$;
For $i = 1$ to h
  For j$\in$ {x, y, z}
    Calculate the similarity between $\boldsymbol{e}_{ij}$ and B at the same latitude based on DTW;
    Identify style = the style with the highest similarity;
    $count_{style\_q} = count_{style\_q} + 1$;
  End For
End For
If $\max\{count_{\text{style}_1}, count_{\text{style}_2}, count_{\text{style}_3}, \ldots, count_{\text{style}_z}\} \geq \frac{9}{5}h$
  Return identify style = $\max\{count_{\text{style}_1}, count_{\text{style}_2}, count_{\text{style}_3}, \ldots, count_{\text{style}_z}\}$.
Else
  Return the motion style to be recognized does not exist in the database.

3 BOGRU Generator

To solve the problem that the accuracy of the motion generation model predicts long-time frame joint angle data and smooths the transition frame of different motion styles, we propose a BOGRU generator. GRU realizes the feedback of historical information through the autoregressive connection between hidden layers and generates any length joint angle sequences by learning the characteristics of the data front and back frames. We train the BOGRU based on motion sequences, and the generator can ensure prediction accuracy and prevent the model from overfitting.

Build a BOGRU generator by combining the GRU, Dropout, and Dense, as shown in Fig.3. The BOGRU input is the human joint angle sequence. It uses the GRU to learn human joints' structural and temporal characteristics. Its output is to convert the GRU layers' predicted data into the joint angle

data we need through the Dense. BOGRU trained different styles of motion capture sequences and learned a set of model parameters corresponding to the motion style. BOGRU generates the human joint angle sequence of the corresponding motion style by calling parameters. The model solves the problem that different motion transition frames are not smooth by learning the relationship between transition frames of motion style.

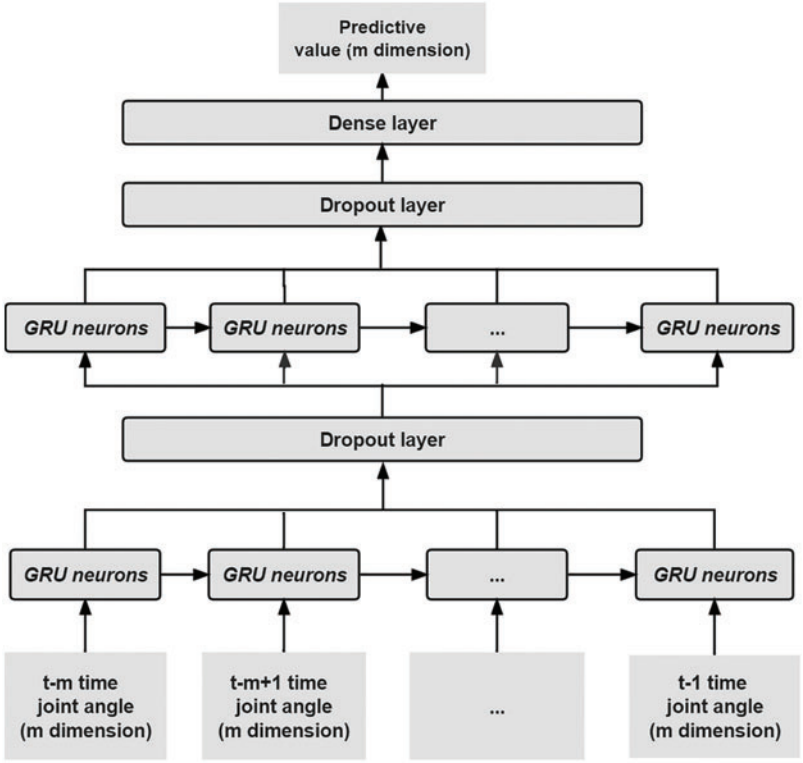


Figure 3: BOGRU generator

3.1 GRU Layer

The GRU layer is the recursive layer of the BOGRU, and it captures the nonlinear characteristics and time characteristics between joints through GRU neurons. The input of the GRU neural network is the joint angle sequence X_t of time frame t and the hidden state S_{t-1} of time frame $t-1$. After training, GRU obtains the hidden unit S_t at time t. S_t is the hidden unit of the GRU neural network to predict the time t, and it is also the hidden unit S_t input to the next neuron. Each GRU neuron stores a time frame of human joint angle data. The accuracy of BOGRU varies with the length of the human joint angle sequence (the number of GRU neurons). The deeper the structure level of the neural network, the richer the feature information obtained, and the model has a more powerful performance. One layer of the GRU neural network can't fully capture the features between human joints. According to Table 1, we selected different length motion sequences and trained single-layer GRU, double-layer GRU, and three-layer GRU.

According to Table 1, the generator model with GRU neural network layers of 2 has the lowest prediction error and the highest accuracy. The prediction accuracy of the generator will continue to improve as the number of neurons increases. From the evaluation of the model, when the number of

neurons in each layer is 240, the goodness of fit of training data and test data is significant. When the number of neurons is 480, the training data has high goodness of fit, but the test data is not good, and the model appears to be overfitted. Therefore, we built a BOGRU generator with two layers of GRU neural network and 240 neurons in each layer.

Table 1: The joint angle error predicted by BOGRU with different structures

Number of network layers and number of neurons in each layer	Judgment coefficient R^2_*training*	Judgment coefficient R^2_*test*	Mean square error (MSE)
Single-layer (120 neurons)	0.89975	0.90568	0.00831
Single-layer (240 neurons)	0.95624	0.96245	0.00327
Single-layer (480 neurons)	0.99006	0.55326	0.00156
Double-layer (120 neurons)	0.93654	0.95248	0.00529
Double-layer (240 neurons)	0.99015	0.99635	0.00131
Double-layer (480 neurons)	0.99138	0.54378	0.00097
Three-layer (120 neurons)	0.95306	0.96006	0.00346
Three-layer (240 neurons)	0.98264	0.99264	0.00264
Three-layer (480 neurons)	0.99325	0.52136	0.00083

3.2 Dropout Layer

Because it is difficult to capture the joint angle data of human motion, there are too few training samples. The trained model has a high prediction accuracy on the training set, while the prediction accuracy on the test set is extremely low, so the model appears overfitting. The BOGRU solves the overfitting problem of the model by adding a dropout layer. In the training process, the output of the GRU layer is the hidden node of t time frames, and the dropout will disable some nodes of the hidden layer with a 50% probability. It simplifies the network topology and generator parameters, as shown in Fig. 4.

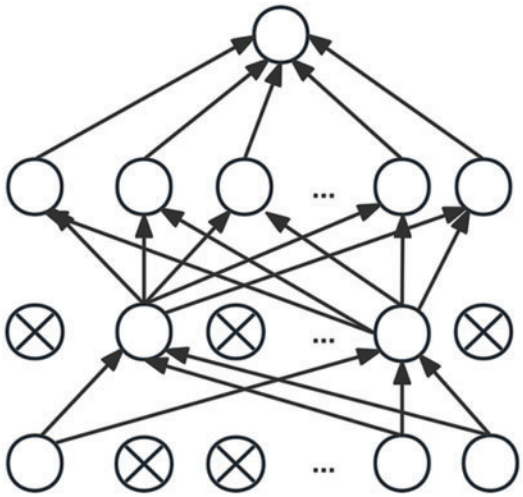


Figure 4: Dropout layer

3.3 Dense Layer

The BOGRU obtains the human joint angle sequences through the dense layer. The input of the dense layer is the hidden sequence of the output of the GRU neural network of the second layer. The network has the learning ability of nonlinear mapping through the Relu nonlinear activation function. So we can convert the hidden state units entered by the dense layer into visible state units that meet our requirements. The output of the dense layer is the sequence of human motion joint angles predicted by BOGRU in the specified time.

To reduce the training error, BOGRU uses the Adam optimization algorithm [18]. It combines the characteristics of an adaptive gradient algorithm and root mean square propagation, which controls the occurrence of the overfitting phenomenon and improves the accuracy of the model prediction. The loss function is the combination of the MSE function and weight subtraction. MSE function enhances the prediction accuracy and adds weight attenuation term to prevent the model from overfitting.

4 Experiment

Experimental Environment: The running environment of the experiment is a 2.5 GHz CPU, 16 GB memory PC, using the Windows 10 operating system, the programming language is Python 3.7, and the development platform is PyCharm. The generator uses the TensorFlow framework to model it.

Data Set: The experimental data set comes from two public databases, CMU and HDM05, which are dataset 1 and dataset 2. Dataset 1 has eight motions styles, namely run, jog, walk, slow walk, stride, forward jump, jump, and high jump; Dataset 2 has 13 motions styles, namely throw, stand, squat, sneak, sit, rotate, lie, jump, jog, hop, grab, elbow, and depositr. Both databases capture the motion data of 29 joints of performers and record the Euler angle of human joints, so there are 59 freedom degrees in total.

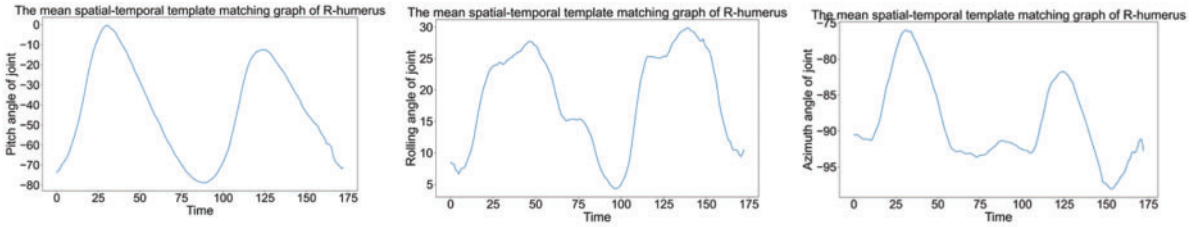
4.1 Analysis of Discriminator Experiment Results

To solve the problem that individual differences affect the recognition accuracy of the discriminator, we extracted the human motion joint angle sequence of 35 actors for each movement. Then the discriminator constructed the mean spatial-temporal template matching graphs based on Algorithm 1. Each sample contains 29 joint angles of the human, and the data of each frame is 59 dimensions. Therefore, the discriminator can construct 59 mean spatial-temporal template-matching graphs for each movement. We analyzed the change of human joint angle during running and walking, and the discriminator constructed mean spatial-temporal template matching graphs. Fig. 5 only shows the r-humerus joint.

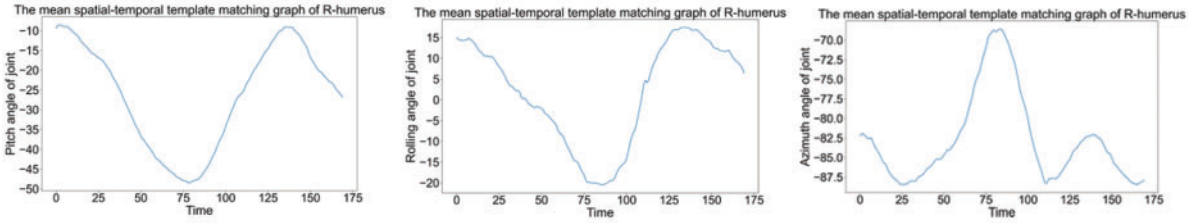
Among the 29 joints, we extract 17 crucial joints based on Algorithm 2. This way prevents the data redundancy of the discriminator and improves the recognition accuracy of the discriminator. For each sport, we took 64 samples, 45 samples as the training set and the other 19 samples as the test set. Table 2 shows the crucial joints extracted by the discriminator.

After the discriminator extracts the crucial joints, it draws the mean spatial-temporal template matching graphs of 17 crucial joints (32 dimensions) from the mean spatial-temporal template matching graphs of 29 joints (59 dimensions) and uses them as recognition objects. Finally, the discriminator identifies the movement style of the captured data according to Algorithm 3. We use the confusion matrix of two data sets for quantitative analysis. The row direction represents the original motion style, and the column direction represents the movement style recognized by the discriminator. Each block in the figure is the correct rate of the discriminator to identify the capture sequences.

Fig. 6 shows the recognition effect of dataset 1 and dataset 2 on each movement style. The value of the first row and the first column in Fig. 6a is 1.00, which means that the correct rate of the discriminator in recognizing the run motion sequence is 100%. The discriminator has identified 8% of the motion sequence incorrectly by observing the jog motion in line 2. Through experiments, we find that the difference between mean spatial-temporal template matching graphs of similar motion is minuteness, and the interference discriminator recognizes. However, for the motion style with a significant difference, as shown in Fig. 6b, the discriminator's recognition accuracy reaches 100%.



(a) The mean spatial-temporal template matching graphs of R-humerus under jogging



(b) The mean spatial-temporal template matching graphs of R-humerus under walking

Figure 5: The mean spatial-temporal template matching graphs of r-humerus under jogging and walking

Table 2: Crucial joints of human movement

Number	Number of joints	Crucial joint	Number	Number of joints	Crucial joint
1	1	Root	10	12	Thorax
2	2	L-femur	11	13	Lower-neck
3	3	L-tibia	12	14	Upper-neck
4	4	L-foot	13	15	Head
5	6	R-femur	14	17	L-humerus
6	7	R-tibia	15	18	L-radius
7	8	R-foot	16	24	R-humerus
8	10	Lower-back	17	25	R-radius
9	11	Upper-back			

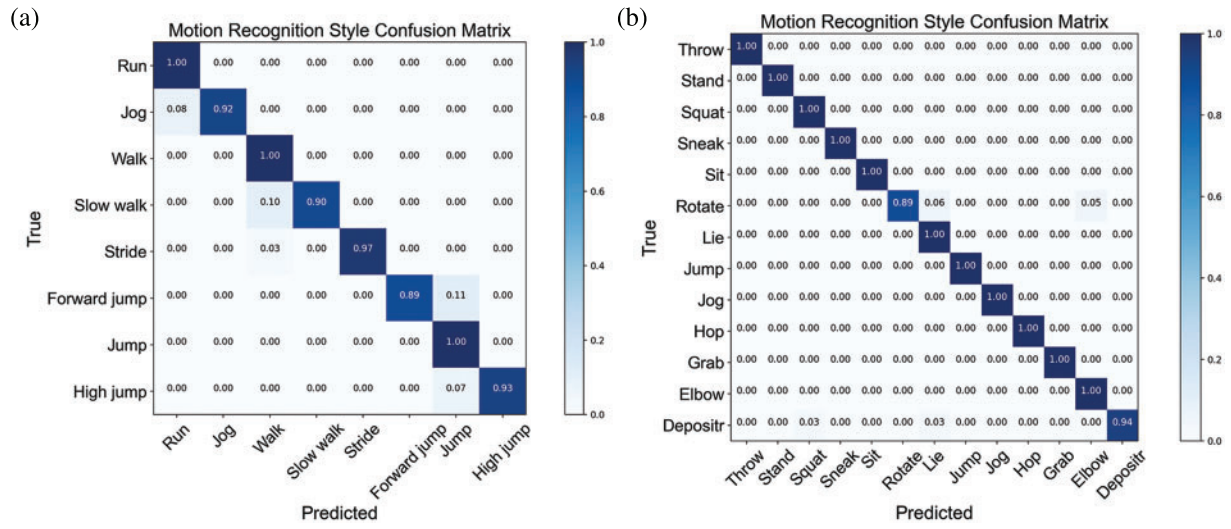


Figure 6: Confusion matrix of dataset 1 (a) and dataset 2 (b)

To verify the recognition effect of the recognizer on the motion capture sequence, we compared the recognition effect of the support vector machine (SVM) based on radial function and our discriminator. Fig. 7 shows the comparison results of the two recognition algorithms. The recognition rate of both simple motion models reached 100%. But the recognition rate of similar motion SVM models was lower than 80%, mainly because SVM models ignored the timing information of human motion, resulting in a poor recognition effect. Our discriminator captures the periodic and continuous characteristics of human movements in temporal data and then combines the crucial joints of the human body for voting recognition, significantly improving the recognition accuracy of the model.

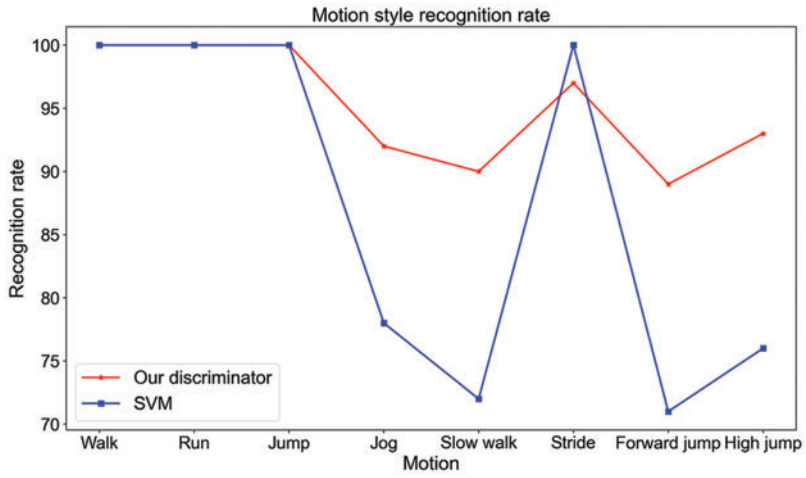


Figure 7: Comparison of style motion recognition rate

4.2 Analysis of Generator Experiment Results

We use the joint angle sequences of multiple actors walking, running, and jumping in the CMU database and HDM05 database to train BOGRU. So the generator can learn the temporal relationship

of all joints between different movements. We extracted 150 walking sequences, 280 jogging sequences, and 150 jumping sequences. After training, the discriminator obtains a series of optimal model parameters representing each motion. At the same time, when the time step was 240 frames, the model had the best prediction effect. According to Fig. 8, we can get the ability of the BOGRU to generate pitch angle sequences of the r-radius joint under walk and jog motions.

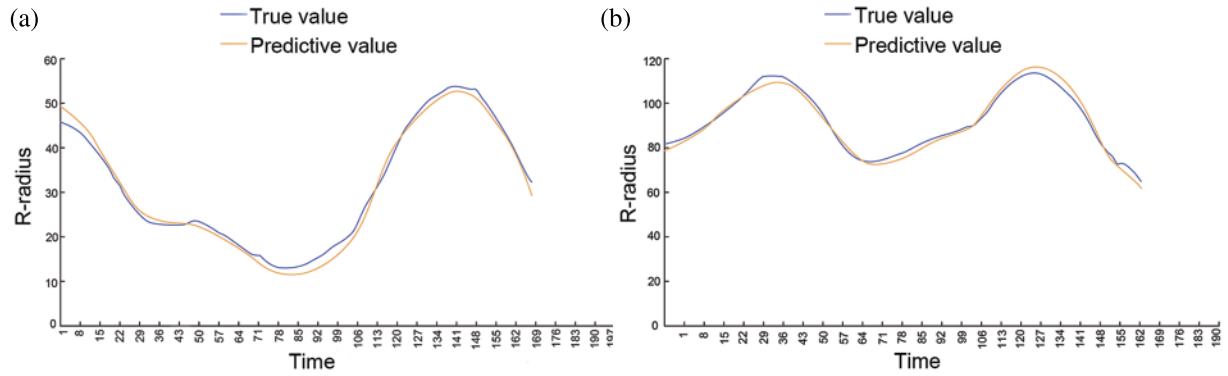


Figure 8: The pitch angle of r-radius predicted by BOGRU for a walk (a) and a jog (b)

The BOGRU generator can generate single-style movements of any length and can splice movements of different styles to form smoother multi-style movements. To avoid shaking during the transition of different motion styles, we trained 130 motion sequences in the HDM05 database, including running and leaping, and some transition frames between the two motions. The trained model can learn the transition between the two motions and generate smooth joint angle sequences. Taking the pitch angle of the r-humerus joint as an example, Fig. 9 shows the ability of the BOGRU to reconstruct the run and leap sequence.

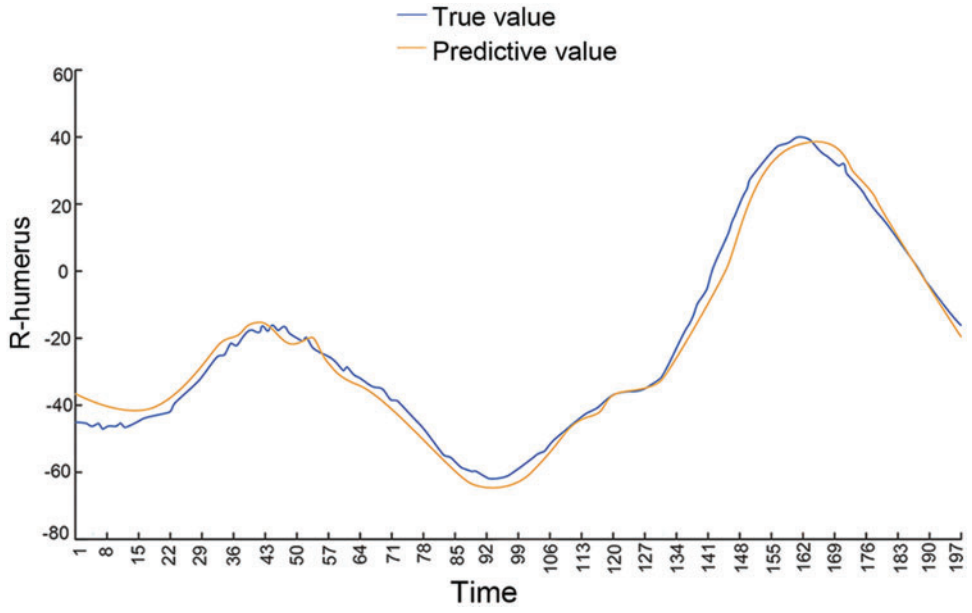


Figure 9: Pitch angle of r-humerus predicted by BOGRU for run and leap

4.3 Comparison with Other Generation Models

The application of deep learning in the human motion generation model mainly includes the following four studies: the temporal structure generation model based on RBM, the spatial structure generation model based on RNN, the spatial-temporal domain generation model based on convolutional neural network (CNN) [19], and the motion generation model based on hybrid deep learning model. We use the motion generation model based on a hybrid deep learning model. Compared with other models, our model can recognize the motion style through a discriminator and generate the target motion joint angle sequences of any length through a generator. The combination of discriminator and generator enables the model to realize the generation of multi-task and multi-style target motion sequences. Table 3 shows the multi-task scenarios suitable for different motion generation models.

Table 3: Multi-task scenarios suitable for different motion generation models

Model	Joint repair	Motion transition		Sequence generation		Style synthesis	Training method	
		Random	Controlled	Short-term	Long-term		Unsupervised	Supervised
Conditional RBM (CRBM) [6]	√	√		√		√	√	
Factored CRBM (FCRBM) [20]	√		√	√		√		√
Implicit mixture CRBM (IMCRBM) [7]			√	√		√	√	√
Encoder-RNN-decoder (ERD) [21]					√		√	
Structural RNN (SRNN) [22]					√	√	√	
Factored conditional temporal sigmoid belief network (FCTSBN) [23]		√			√		√	
Recurrent Gaussian process (RGP) [14]			√		√		√	
GRU [18]					√		√	
Ours	√		√	√	√	√	√	

In addition, our human motion generation model is superior to other human motion generation models in terms of prediction data and reconstruction sequences. Table 4 shows the prediction errors of different motion generation models in predicting walking motion sequences of different lengths. Table 5 shows the reconstruction errors of various motion generation models in reconstructing walking and running motion data.

Table 4: Prediction error of different motion models

Model	100 frames	200 frames	400 frames
ERD	0.0093	0.0125	0.0178
SRNN	0.0081	0.0108	0.0130
3 Long short term memory layers (LSTM-3LR) [24]	0.0077	0.0091	0.0147
Literature [25]	0.0063	0.0085	0.0098
Ours	0.0033	0.0058	0.0091

Table 5: Reconstruction error of different motion models

Model	Walk	Run
CRBM	0.0967 ± 0.0268	0.0845 ± 0.0203
FCTSBN	0.0813 ± 0.0026	0.0588 ± 0.0025
RGP	0.0512 ± 0.0050	0.0485 ± 0.0126
Ours	0.0225 ± 0.0037	0.0208 ± 0.0026

5 Conclusion and Prospect

The existing motion generation models generate new human motion data by extracting structural features from existing lossless data, which cannot recognize unknown style motion and generate target motion sequences based on current data. We propose a multi-task motion generation model combining a discriminator and a generator. The discriminator can recognize the motion capture sequences of unknown attributes. According to the identified data, the generator can continue to generate the motion sequence that does not exist in the target motion. The discriminator and generator improve the reuse rate of motion capture data. When the human motion generation model adds a recognition function, the model generates motion sequences that are more similar to the actual motion. Compared with the existing motion generation models, our model adds recognition tasks. From the perspective of generation ability, our model can generate motion sequences of any length, and the predicted motion is closer to the actual movement. At the same time, our model can also well repair the missing motion capture data.

Although our model has achieved good results in human motion generation, we only study periodic motion, which is also a worthwhile scheme for non-periodic movement. Confronted learning has powerful feature extraction and reconstruction capabilities and can be used to identify continuous behaviors in natural scenes. Therefore, generation models can add a recognition function to discriminate behavior in a complex environment. The model can generate more life-fitting motion sequences.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, X. Bing, D. Warde-Farley *et al.,* "Generative adversarial networks," *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[2] H. Kadu, M. Kuo and C. C. J. Kuo, "Human motion classification and management based on mocap data analysis," in *Human Gesture and Behavior Understanding, Joint ACM Workshop*. New York, NY, USA, 73–74, 2011.

[3] I. Kapsouras and N. Nikolaidis, "Action recognition on motion capture data using a dynemes and forward differences representation," *Journal of Visual Communication and Image Representation*, vol. 25, no. 6, pp. 1432–1445, 2014.

[4] M. Raptis, D. Kirovski and H. Hoppe, "Real-time classification of dance gestures from skeleton animation," in *Computer Animation, ACM SIGGRAPH/Eurographics Symp.*, New York, NY, USA, pp. 147–156, 2011.

[5] S. Salvador and P. Chan, "Toward accurate dynamic time warping in linear time and space," *Intelligent Data Analysis*, vol. 11, no. 5, pp. 561–580, 2007.

[6] G. W. Taylor, G. E. Hinton and S. Roweis, "Modeling human motion using binary latent variables," *Neural Information Processing Systems*, vol. 19, pp. 1345–1352, 2007.

[7] G. W. Taylor, L. Sigal, D. J. Fleet and G. E. Hinton, "Dynamical binary latent variable models for 3D human pose tracking," in *Computer Vision and Pattern Recognition, IEEE Computer Society Conf.*, San Francisco, USA, pp. pp 631–638, 2010.

[8] C. C. Chiu and S. Marsella, "A style controller for generating virtual human behaviors," in *Autonomous Agents and Multiagent Systems-Volume 3, the 10th Int. Conf.*, Taiwan, China, pp. 1023–1030, 2011.

[9] Z. Gan, C. Y. Li, R. Henao, D. E. Carlson and L. Carin, "Deep temporal sigmoid belief networks for sequence modeling," in *Neural Information Processing Systems, the 28th Int. Conf.*, Montreal, Canada, pp. 2467–2475, 2015.

[10] R. Salakhutdinov, A. Mnih and G. Hinton, "Restricted Boltzmann machines for collaborative filtering," in *Machine Learning, the 24th Int. Conf.*, New York, NY, USA, pp. 791–798, 2007.

[11] F. G. Harvey, J. Roy, D. Kanaa and C. Pal, "Recurrent semi-supervised classification and constrained adversarial generation with motion capture data," *Image and Vision Computing*, vol. 78, no. 1, pp. 42–52, 2018.

[12] J. Bayer and C. Osendorfer, "Learning stochastic recurrent networks," arXiv: 1411.7610, 2014.

[13] Y. Park, S. Moon and I. H. Suh, "Tracking human-like natural motion using deep recurrent neural networks," arXiv: 1604.04528, 2016.

[14] C. L. C. Mattos and G. A. Barreto, "A stochastic variational framework for recurrent gaussian processes models," *Neural Networks*, vol. 112, no. 1, pp. 54–72, 2019.

[15] J. Martinez, M. J. Black and J. Romero, "On human motion prediction using recurrent neural networks," in *Computer Vision and Pattern Recognition, IEEE Conf.*, Honolulu, HI, USA, pp. 2891–2900, 2017.

[16] R. J. Williams and D. Zipser, "A learning algorithm for continually running fully recurrent neural networks," *Neural Computation*, vol. 1, no. 2, pp. 270–280, 1989.

[17] J. Chung, C. Gulcehre, K. H. Cho and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," arXiv: 1412.3555, 2014.

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv: 1412.6980, 2014.

[19] Y. LeCun, Y. Bengio and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[20] G. W. Taylor and G. E. Hinton, "Factored conditional restricted Boltzmann machines for modeling motion style," in *Machine Learning, the 26th Annual Int. Conf.*, Montreal, Canada, pp. 1025–1032, 2009.

[21] K. Fragkiadaki, S. Levine, P. Felsen and J. Malik, "Recurrent network models for human dynamics," in *Computer Vision, IEEE Int. Conf.*, Santiago, Chile, pp. 4346–4354, 2015.

[22] A. Jain, A. R. Zamir, S. Savarese and A. Saxena, "Structural-RNN: Deep learning on spatio-temporal graphs," in *Computer Vision and Pattern Recognition, IEEE Conf.*, Las Vegas, NV, USA, pp. 5308–5317, 2016.

[23] J. M. Song, Z. Gan and L. Carin, "Factored temporal sigmoid belief networks for sequence learning," in *Machine Learning, Int. Conf.*, New York, NY, USA, PMLR, pp. 1272–1281, 2016.
[24] T. Yongyi, M. Lin, L. Wei and Z. Weishi, "Long-term human motion prediction by modeling motion context and enhancing motion dynamic," arXiv: 1805.02513, 2018.
[25] L. Chen, Z. Zhen, W. S. Lee and G. H. Lee, "Convolutional sequence to sequence model for human dynamics," in *Computer Vision and Pattern Recognition, IEEE Conf.*, State of, Utah, USA, pp. 5226–5234, 2018.