*Journal of Multimedia Information System VOL. 10, NO. 1, March 2023 (pp. 69-78): ISSN 2383-7632 (Online)*
https://doi.org/10.33851/JMIS.2023.10.1.69

# 3D Dynamic Image Modeling Based on Machine Learning in Film and Television Animation

Yuwei Wang[1]

## Abstract

With the deep integration and rapid development of computer technology and film and television animation in recent years, computer animation technology has gradually created objective practical value. 3D animation technology also plays a key role in film and television special effects and advertising special effects. As a new type of multimedia data, human motion capture data can be used for 3D human model modeling and human motion simulation because of its high fidelity. A human motion pattern recognition method based on long short-term memory network (LSTM) is proposed. The model uses a deep learning neural network composed of a 2-layer LSTM network to automatically extract features from the collected human body feature information. Then, the multi-class motion patterns are modeled in time series to quickly identify different motion patterns in real time. To evaluate the performance of this model, the effectiveness of this method in identifying six different motion modes is validated using an open dataset. At the same time, this method is compared with four methods based on in-depth learning model. Experimental results verify the effectiveness of the method. It provides a feasible method for human motion recognition and modeling based on capture data in video animation.

**Key Words**: Dynamic Image Modeling, Film and Television Animation, Motion Recognition, LSTM, Video Animation.

## I. INTRODUCTION

With the increasing progress of computer technology, my country's 3D animation technology industry has also ushered in the spring of development. Its application range is very wide, involving architectural planning, product design, advertising animation, film and television special effects, virtual world and many other fields. Among them, 3D animation technology has made rapid progress in film and television special effects. Although my country's 3D animation technology is developing gradually, there is still a big gap compared with the same industry in other parts of the world. Therefore, we must correctly understand the restrictive factors in the development of 3D animation technology in the art of film and television special effects, and make full use of the existing production methods of film and television special effects. By combining 3D animation technology with film and television production, it will continue to promote the development of China's film and television industry.

3D human body animation modeling is an important branch of 3D animation modeling, which can promote more realistic film and television animation works [1-2]. Nowadays, machine learning and computer vision are widely used in human animation and other fields, including image/video-based human motion data acquisition technology, digital character and scene modeling, interactive character animation control and motion generation, etc. Computer vision techniques are widely used. Besides, machine learning theory is also widely used in the field of intelligent 3D human animation research. Three-dimensional human body animation technology can be generally regarded as two categories: one is model animation developed on the basis of traditional computer animation technology, especially traditional two-dimensional computer animation technology. The second is the production technology of human body animation based on captured data with the popularization of motion capture system. The simple academic definition of motion capture is: Motion capture is a comprehensive use of computer graphics, electronics, machinery, optics, computer animation and other technologies to capture the movements or expressions of the subject of the performance. Through the captured data of these actions or expressions, the direct drive to the animation image model is realized. Motion capture is divided into different categories such as mechanical motion capture, acoustic motion capture, electromagnetic motion capture, and optical motion capture.

**Manuscript received March 09, 2023; Revised March 17, 2023; Accepted March 19, 2023. (ID No. JMIS-23M-03-010)**
Corresponding Author (*): Yuwei Wang, +86-17630733074, wangyuwei1986@xxu.edu.cn
[1]Department of Fine Arts, Xinxiang University, Xinxiang, China, wangyuwei1986@xxu.edu.cn

Traditional animation utilizes mathematical models to produce animation results that meet user needs. Such methods can be classified as model-based animation methods, including key frame animation technology, joint animation technology based on kinematics knowledge, and physics/dynamics methods. Another type of animation production technology uses real collected 3D motion data to generate animation models. Generate 3D human animation by adopting a data-driven approach. In essence, it is a data-driven animation production method, including editing, compositing, and reusing technologies based on motion capture data. The popularization of commercial human motion capture systems makes obtaining realistic 3D human motion data no longer a limitation for making realistic 3D human animation. 3D human motion databases that can be reused have also appeared, which makes the data-driven approach an important means of making realistic 3D human animation.

This paper focuses on the data-driven 3D human animation modeling research program. Specifically, it uses real 3D human motion data and uses machine learning methods to realize 3D human body modeling to meet the needs of 3D human body modeling in film and television animation. Human motion has strong randomness and continuity. In order to improve the accuracy of motion recognition, time series information is needed to describe the characteristics of the motion process. Long short-term memory network (LSTM) is a variant of recurrent neural network (RNN), which has been widely used in many fields. LSTM can realize the modeling of variable-length time series information, and has certain feature extraction capabilities [2]. Therefore, this paper proposes a 3D human animation modeling method based on LSTM cyclic neural network to realize automatic recognition of human motion. The real sensing data used in this paper is obtained through inertial sensors. In this paper, based on the WISDM dataset [3], a two-layer LSTM neural network is used to extract time series features. By modeling the three-axis acceleration time series information of the front pocket of the right leg of the human body, real-time recognition of six human action modes: walking, jogging, going upstairs and downstairs, sitting, and standing. This paper verifies its effectiveness through comparative experiments. The experimental results show that the method in this paper can provide a feasible solution for the research of human motion recognition and modeling based on motion capture data, and provide a new solution for film and television animation production.

## II. RELATED WORK

Conde and Thalmann [4] used reinforcement learning theory to learn the virtual environment where the virtual character is located and analyze the hierarchical structure of the virtual scene. Noser et al. [5] and Kuffner and Latombe 0 established a multi-channel, high-level behavioral decision-making and driving model for virtual characters based on synthetic vision, memory and high-level reasoning and learning mechanisms, and realized autonomous roaming of virtual characters in obstacle scenes. Behavior animation generation. In the work of Ref [7] and Ref [8], machine learning techniques are used to provide a memory model for virtual characters, so that they can remember the information provided by the user and the instructions issued earlier. Ref [9] proposed a self-organizing structure for learning the virtual scene structure and the behavior of reaching a certain target point in the scene, etc., to realize autonomous animation generation. Ref [10] proposed the concept of virtual human imitation learning, that is, using machine learning theory to endow virtual characters with a certain autonomous learning ability, and enable them to simulate the physical behavior demonstrated by the user through training.

The essence of 3D human animation creation using motion capture technology is a data-driven animation creation method, which has the advantages of easy data acquisition, high precision, strong realism and high production efficiency. Motion synthesis is the focus and key technology of motion data reuse. It is also the most difficult part of the motion reuse process. Motion capture data has high dimensionality, large amount of information, complex structure, spatiotemporal continuity and Riemannian manifold structure, all of which bring challenges to motion synthesis. Motion hybrid [12-15] is a simple and efficient motion synthesis model. Such methods first preprocess motion segments of the same type, including using the DTW algorithm to align them in time sequence, and then make each motion frame have similar spatial coordinates through linear transformation, that is, coordinate alignment. The motions after time sequence alignment and coordinate alignment are unified in structure. By performing weighted interpolation on these unified motions, and then constrained reconstruction of the interpolated motion, a very realistic new motion can be obtained. However, the data organization method of such methods is too simple to mine the inherent laws in the data, and users cannot interact with the system in real time, making it difficult to control the results of motion synthesis to meet the needs of users. Another class of methods is parametric motion synthesis. Parametric motion models [16-18] can effectively solve the problems of motion graphs by exploiting some physical properties of motion. Kwon and Shin [18] introduced the type of motion, speed, acceleration and foothold into the synthetic model in the form of parameters, and controlled during the synthesis process, which can solve some problems such as foot sliding and orientation shaking. Heck and Gleicher [19] constructed the nodes of the motion graph as a continuous parameter space, which brought fine-grained control to the originally very limited

splicing and combination methods. For example, this method can synthesize richer and more delicate output by adjusting parameters Boxing sport. These methods greatly improve the controllability of the motion synthesis process, but the semantic level of these physical parameters is too low, and the content needs to be manually specified in advance, which cannot automatically adapt to changes in motion types.

To address the above challenges, researchers apply deep learning methods to motion synthesis. Nowadays, the emergence of various deep learning models has greatly promoted the development of related fields. One of the important advantages of the deep learning model is that it can automatically learn the characteristics of the data from the data set, which provides a new research direction for data editing and processing. Gatys et al. [20] used the deep network model to extract the style features and content features of the image in the hidden layer respectively, and then through the editing process in the hidden layer, a new image that maintained the content of the original image but had a different style could finally be obtained. In the field of motion data reuse, Taylor and Hinton [21] constructed a model of a restricted Boltzmann machine, and performed motion mixing by extracting motion-related parameters to generate new motions. Holden et al. [22] proposed a motion synthesis method based on a deep learning framework. This method has relatively broad requirements on the format of the training data and does not require the above-mentioned various operations. Any type and length of motion capture data can be exploited to train the model. The motion manifold learned by the deep model framework can be expressed by a hidden unit of a autoencoder, which can synthesize various types of complex motions based on the parameters given by the user. In addition, problems such as footstep sliding and orientation shaking can be solved by constraining the hidden unit space.

## III. SYSTEM DESIGN

The 3D dynamic image modeling of the human body based on the data-driven method can be regarded as the synthesis and modeling of human motion using the motion data reuse technology. In recent years, various machine learning techniques such as subspace analysis, statistical learning, and manifold learning have been widely used to analyze and learn the existing 3D human motion data and guide the generation of new motion data. This paper proposes the use of deep learning models to achieve 3D human motion modeling. The data is acquired through the motion capture device and preprocessed by the information processing module. Then, a two-layer LSTM neural network is used to extract time series features and model the motion of human legs. The specific process of the method proposed in this paper is as follows.

### 3.1. Human Motion Data Acquisition and Preprocessing

The optical motion capture system is the most commonly used. Its working principle is to wear photosensitive nodes on each limb of the athlete, so that the movement of the athlete can be restored by the three-dimensional information of these nodes captured by the cameras installed around the capture field. This type of system can capture human movements very accurately, and the device itself does not impose too much constraints on the movement of the athlete.

In addition, commercial industry motion capture and analysis systems can also be used to track, capture and calibrate the face and body movements of the collector in real time. The EvaRT motion capture software can save the change data of the marker points on the actor and read it directly by the software, and then transfer data such as actions and models in the 3D modeling software. This paper uses public datasets for experimental evaluation.

Due to the influence of capture conditions and errors, or to meet specific application requirements, the captured 3D human motion data may require specific preprocessing before being applied to 3D human animation creation. Motion data preprocessing includes reconstruction of missing feature points in data, natural/realistic 3D human motion data evaluation, motion data compression, key frame extraction, and motion sequence segmentation and recognition. Common dataset preprocessing operations include data smoothing and data windowing. The motion capture technology captures the movement of the performer, and through preprocessing and post-processing, the original data is converted into model motion data in a standard format, which is used for driving various 3D models.

### 3.2. LSTM-Based 3D Human Behavior Pattern Modeling and Recognition Method

Recurrent Neural Networks (RNNs) can process sequence data and can model and describe human motion processes. However, ordinary RNNs have long-term dependence problems, and are prone to gradient disappearance and gradient explosion during network training [23]. Therefore, this paper proposes to select LSTM which is a variant of RNN, to build a human action recognition model.

#### *3.2.1. Preliminary on LSTM*

Like ordinary RNN, the input of the LSTM network at the current moment is still the output of the hidden state at the previous moment and the input feature at the current moment, and the network structure is also a chained neural network structure composed of a series of repeated neural

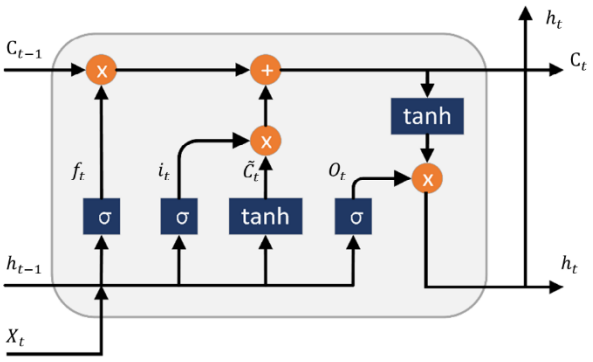


Fig. 1. Structure of LSTM neurons.

network units. Different from the traditional RNN, LSTM introduces a "gate" mechanism and a memory unit described by the cell state inside each loop body neuron, which can control the memory and forgetting degree of the previous information and the current moment information, thus solving the traditional RNN. The long-term dependency problem is widely used. The internal structure of LSTM neurons is shown in Fig. 1.

LSTM neurons are composed of cell states and "gate" mechanisms (forgetting gate, input gate, output gate). In the figure, $C_t$ represents the cell state, representing long-term memory. By adding or deleting state information on $C_t$ through the "gate" structure, the modified state information can be controlled to be transmitted to the next moment. σ represents the sigmoid activation function, which can output 0 to 1 the number between is mainly used to describe what information is passed after sigmoid, a value of 0 means that no information passes through sigmoid, and a value of 1 means that all information at this time passes through sigmoid. $h_{t-1}$ and $h_t$ represent the hidden state of the previous moment and the current moment, respectively, $\oplus$ represents vector addition, and $\otimes$ represents vector multiplication.

Equations (1) to (6) describe how LSTM updates the state of the neural network unit according to the "gate" mechanism at any time step. The input is fed into each "gate" unit of the LSTM unit. The first step is to control which previously recorded information should be retained by the forget gate. It can be seen from Fig. 1 that the input of the forget gate at the current moment is the hidden state $h_{t-1}$ at the previous moment and the input information $x_t$ at the current moment, as shown in equation (1). The second step is to update $C_t$ by the input gate. First, pass $x_t$ into the sigmoid function and tanh function respectively to obtain the $i_t$ vector and the new candidate value vector $\tilde{C}_t$. Afterwards, it is multiplied by the two-part vector of $\tilde{C}_t$ and $i_t$ to determine whether the input information of the network at the current moment is saved in $C_t$ to update the cell value. The cell update formula is shown in equation (2) −(4). Finally, the final output value of the LSTM neuron at a time step is determined by the output gate. First, $x_t$ is calculated by the sigmoid function to obtain the vector $O_t$. Then multiply the $C_t$ and $C_t$ vectors processed by the tanh function to determine the final output information of the LSTM neuron at the current moment. The output gate formula is shown in equation (5) and equation (6).

$$f_t = \sigma\left(W_f[h_{t-1}, x_t] + b_f\right). \tag{1}$$

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i). \tag{2}$$

$$\tilde{C}_t = tanh(W_c[h_{t-1}, x_t] + b_c). \tag{3}$$

$$C_t = i_t \circ \tilde{C}_t + f_t \circ C_{t-1}. \tag{4}$$

$$O_t = \sigma(W_o[h_{t-1}, x_t] + b_o). \tag{5}$$

$$h_t = O_t \circ tanh(c_t). \tag{6}$$

In the formula: $W_f$, $W_i$, $W_c$, and $W_o$ represent the weight matrix. $b_f$, $b_i$, $b_c$, and $b_o$ represent the bias vector; [,] represent the splicing of two vectors.

*3.2.2. Network Structure Design*

In order to identify which kind of motion the person acts in real time, the structure of the LSTM network designed in this paper consists of an input layer and a hidden layer: including two LSTM layers, a fully connected layer and an output layer.

(1) Input layer: The input is the preprocessed data. The input dimension of RNN for the data is [number of samples, number of time steps, number of input features], namely: [54906, 90, 3].

(2) The first layer of LSTM layer: the time step is n=90, and the number of neurons in each time step of LSTM is 32. Since the input is the data of the accelerometer *x*, *y*, and *z* axes, the number of input features is 3. In addition, the hidden state output of each time step is used as the input of the next LSTM layer. The selection of the time step n and the number of neurons needs to be set experimentally, which will be explained later.

(3) Second LSTM layer: The number of neurons inside the LSTM unit at each time step is 32. Since the sample set and its corresponding category need to be used as input in the process of action pattern recognition, LSTM only needs to output at the last time step as the input of the fully connected layer.

(4) Fully connected layer: There are 32 neurons. The

Relu function is adopted in our model as the activation function.

(5) Output layer: Since the network recognizes six human action patterns of standing, jogging, going upstairs, walking, and sitting, the softmax classifier is used as the output of the six action patterns, that is, the output layer will output the probability values of six categories. The calculation formula is shown in equation (7):

$$softmax(y_t) = \frac{e^{y_t}}{\sum_i e^{y_i}}. \quad (7)$$

In the formula: $i$ represents the action mode category, $y_t$ and $y_i$ represent the probability distribution of the human action category. Finally, according to the maximum likelihood estimation method, the attribute of the action mode is judged as the category with the highest probability.

Besides, a Dropout layer is added after the first LSTM layer, the second LSTM layer, and the fully connected layer. The Dropout layer will discard neurons with a certain probability at random when the model is trained each time. Since the neurons ignored each time are different, the trained networks are also different. Finally, the trained model is integrated to predict the average probability.

*3.2.3. Model Training*

The collected data is input into the LSTM neural network as the motion pattern feature, and the six human motion pattern categories of standing, jogging, going upstairs, going downstairs, walking, and sitting are used as outputs. The training is realized by minimizing the loss function, and the loss function adopts the cross entropy. Loss function, the calculation formula is shown in equation (8).

$$loss = -\frac{1}{m}\sum_{i=1}^{m} \tilde{y}_i log y_i. \quad (8)$$

In the formula: $\tilde{y}_i$ represents the true value of the i-th category, and $y_i$ represents the predicted value of the ith category of the model. The Adam optimization algorithm [24] is used to adaptively optimize the learning rate, which has the advantages of efficient computation and less memory. The parameters of the model are initialized with random values of truncated normal distribution. In the process of backpropagation, different from the BP algorithm of other neural networks, the backpropagation along time (BPTT) algorithm is used to update the parameters. In order to prevent the model from overfitting, the method of early stopping is used in the iterative process. If the accuracy of the model on the test set does not improve by 0.001 within 10 iterations, the model stops training. After the model training is over, save the optimal parameters, and then use the saved optimal parameters to identify the human actions in the test set.

# IV. EXPERIMENTAL EVALUATIONS

## 4.1. Experimental Settings

The experiment in this paper is based on the Windows 10 system, the CPU model is Intel Core i5-9300H, and the memory is 8 GB. The GPU is a notebook computer with NVIDIA GTX1650 graphics processor and 4 GB video memory. The algorithm is implemented using python language based on Google's open-source deep learning framework Tensorflow2.0, and the experimental integrated development environment is Pycharm.

In this paper, we use public datasets to evaluate the experimental results. This dataset is the public dataset WISDM dataset of the Wireless Data Mining Laboratory of Fordham University. The WISDM dataset is a public dataset released by the Wireless Sensor Data Mining Laboratory (2012). This data set uses an Android smartphone as the data collection platform, and the smartphone is placed in the right front trouser pocket of the subject. The subjects completed 6 exercise modes including walking, jogging, going upstairs, going downstairs, sitting, and standing within a specific time. During this period, the built-in accelerometer of the mobile phone collects the data of the *x*, *y*, and *z* axes of the three-axis accelerometer at a sampling frequency of 20 Hz. The data set contains a total of 1,098,207 sample point data from 36 healthy subjects (the number of movements of each subject is not equal), and the distribution of the number of motion pattern samples is shown in Table 1. The continuous activity signal is segmented using a sliding window with a time length of 2.56s and an overlap rate of 50%. In this paper, 70% of the data is used as the training set and 30% of the data is used as the test set. For the convenience of processing, the data set is normalized. The processing flow is as follows:

$$X_{normalize} = \frac{X-\mu_x}{\sigma_x}. \quad (9)$$

$$Y_{normalize} = \frac{Y-\mu_y}{\sigma_y}. \quad (10)$$

$$Z_{normalize} = \frac{Z-\mu_z}{\sigma_z}. \quad (11)$$

In the formula: $X_{normalize}$, $Y_{normalize}$, and $Z_{normalize}$ represent the normalized acceleration value. *X*, *Y*, and *Z* represent the raw data of the acceleration sensor; $\mu_x$, $\mu_y$, and $\mu_z$ represent the average values of the accelerometer's *x*, *y*, and *z* axes, respectively; $\sigma_x$, $\sigma_y$, and $\sigma_z$ represent the var-

Table 1. Sports mode data distribution.

| Sports mode | Ratio (%) |
|---|---|
| Walking | 38.6 |
| Jogging | 31.2 |
| Up Stairs | 11.2 |
| Down Stairs | 9.1 |
| Sitting | 5.5 |
| Standing | 4.4 |

iance of the accelerometer's $x$, $y$, and $z$ axes, respectively.

In the evaluation, the model conducts 10 experiments on the test set, and takes the average result of 10 runs as the final value. The parameters of the LSTM network have a great influence on the recognition effect of the action pattern, so it is necessary to conduct experimental analysis on different parameters. In the experiment, the training model is aimed at 6 kinds of human action patterns, the weight and bias parameters are continuously updated, and the accuracy and loss values of the test data and training data after each iteration are recorded for comprehensive comparison and analysis. In addition, in the selection of model hyperparameters (such as the time step of LSTM neural network and the number of neurons), this paper determines the parameters through comparative experiments. This accuracy rate is selected as the indicator:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}. \quad (12)$$

In the formula, True Positive (TP)and True Negative (TN) represent the number of samples that predict all positive samples as positive and negative samples, respectively. False Positive (FP) and False Negative (FN) represent the number of samples that predict all negative samples as positive and negative samples, respectively.

### 4.2. Experimental Results on Different Sport Modes

In this section, this paper evaluates the experimental results of our method on the WISDM dataset. Set the time step to 90 and the Dropout parameter to 0.2. The results of 6 different types of action patterns are shown in Table 2. The action mode Sitting has the highest accuracy rate, reaching 96.52%. The accuracy rate of Standing also exceeds 96%, slightly lower than Sitting, which is 96.36%. This is mainly because the movements of sitting and standing are the simplest, there is no change in movement, and the prediction model is easier to fit. Secondly, the accuracy rates of Jogging and Up Stairs reached 95.48% and 95.37%, respectively. Compared with the previous action modes, the accuracy rate of Down Stairs has dropped significantly, and its value is 93.11%, which is 3.41% lower than that of the Sitting category. The lowest experimental result is the

Table 2. Confusion matrix of accuracies of activity recognition.

| | Walking (%) | Jogging (%) | Up stairs (%) | Down stairs (%) | Sitting (%) | Standing (%) |
|---|---|---|---|---|---|---|
| Walking | 92.95 | 0.55 | 3.14 | 3.36 | 0 | 0 |
| Jogging | 0 | 95.48 | 1.02 | 0 | 0.71 | 2.79 |
| Up stairs | 0.11 | 0.92 | 95.37 | 2.46 | 0 | 1.14 |
| Down stairs | 4.20 | 0 | 2.69 | 93.11 | 0 | 0 |
| Sitting | 0.41 | 0 | 1.64 | 0.64 | 96.52 | 0.79 |
| Standing | 0 | 2.13 | 1.51 | 0 | 0 | 96.36 |

Walking category, with an accuracy rate of 92.95%. For Walking, 3.14% and 3.36% of the data were identified as Up Stairs and Down Stairs. This is mainly because Walking has similarities in body swing and leg movements between walking and going up and down stairs, so there are relatively large misidentifications.

### 4.3. Experimental Results Compared with Different Methods

To further evaluate the effectiveness, the proposed method is compared with existing research. The four methods involved in the comparison (Methods in Ref [25], Ref [26], Ref [27], and Ref [28]) are all methods designed based on the deep learning architecture. The compared results can be seen in Fig. 2. The horizontal axis represents 6 different action pattern categories, and the last item is the average of all category results. The vertical axis is accuracy. The experimental performance of the method in Ref [25] is the worst among all 5 methods, and its average accuracy is 83.27%. The method in Ref [25] is even less than 80% accurate on the upstairs category dataset. Among all the five methods, the algorithm in this paper shows better performance, with an average accuracy rate of 94.97%. The average accuracy of Methods in Ref [27] and Ref [28] is very close, with a difference of only 0.1%. However, the experimental results of the two on different categories are quite

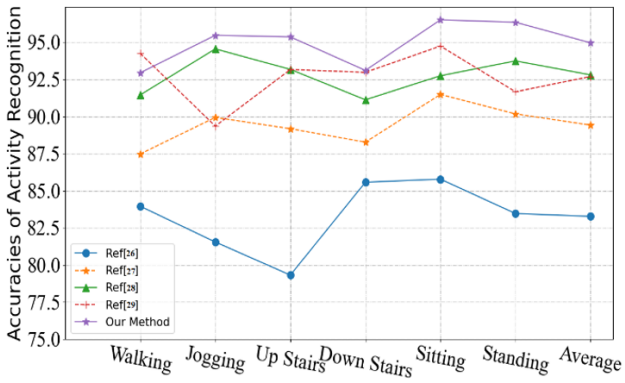


Fig. 2. Comparative experimental results with existing studies.

different. The experimental results of Ref [27] are more stable in different categories with less fluctuation. The experimental results of Ref [28] in different categories fluctuate much more. The experimental results of this method in the two categories of Walking and Down Stairs are not lower than those of the method in this paper. Another very interesting observation is that several methods involved in the comparison all use more complex network structures. However, our method performs better in all datasets. The reason for this situation may be that the method in this paper is more applicable to the 6 simple action modes in the WISDM dataset. Models with more complex network structures are more prone to overfitting and performance degradation when dealing with these datasets.

### 4.4. Experimental Results over Number of Epochs During Training

This section evaluates how the experimental results change as the number of iteration training increases. The accuracy curves of the training set and the test set during the training process are shown in Fig. 3. It can be seen from the figure that in the initial stage, the accuracy of the model on the training and testing data sets can reach 64.75% and 74.46%, respectively. This shows that our model has an advantage in handling action recognition. With the increase of the number of iterations, the recognition rate of the model gradually increases whether it is the training data set or the test data set. The experimental results on the training data set rise rapidly as the model iterates and cross with the experimental results on the test set. When the iteration reaches about 55 times, the experimental results on the test data set gradually converge and become stable. In this paper, the time step is set to 90, and the dropout parameter is 0.2. It can be seen that the model is better in terms of recognition rate stability and overfitting. This is because when the dropout parameter is 0.2, the dropout layer will randomly generate the network structure, which can effectively prevent overfitting. Therefore, the hyperparameter time step used by the final LSTM neural network is 90, and the dropout layer parameter is set to 0.2.

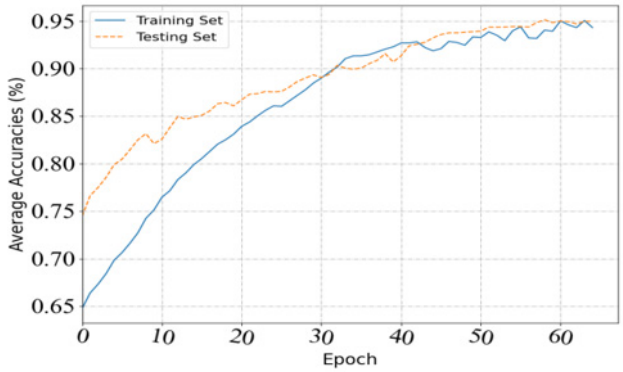


Fig. 3. Curves of accuracy on training set and testing set during training.

## V. CONCLUSION

In the field of film and television animation, the use of motion capture technology to model three-dimensional dynamic images of the human body can meet the requirements of a certain degree of professionalism faster and more conveniently than traditional hand K animation. At the same time, it can shorten the production cycle and improve the efficiency of 3D animation modeling. Considering the strong randomness and continuity of human motion, time series information is needed to describe the characteristics of the motion process to increase the accuracy of action recognition. In this paper, a human motion recognition method based on LSTM neural network is designed using open dataset WISDM as raw data. A two-layer LSTM network is constructed to model and describe human temporal motions. The experimental results show that the average recognition accuracy is 94.97%. To measure the performance of this method, this method is compared with four methods based on in-depth learning model. The experimental results verify the validity of this method. In the future, the research group will study the human multi-node motion information, and further explore the human motion capture method based on inertial information.

## REFERENCES

[1] P. Ratner, *3-D Human Modeling and Animation*, John Wiley and Sons, 2012.

[2] Y. Li, "Film and TV animation production based on artificial intelligence AlphaGd," *Mobile Information Systems*, 2021.

[3] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Computation*, vol. 31, no. 7, pp. 1235-1270, 2019.

[4] G. M. Weiss, "Wisdm smartphone and smartwatch activity and biometrics dataset," UCI Machine Learning Repository: WISDM Smartphone and Smartwatch Activity and Biometrics Dataset Data Set, vol. 7, pp. 133190-133202, 2019.

[5] T. Conde and D. Thalmann, "Learnable behavioural model for autonomous virtual agents: Low-level learning," in *Proceedings of the Fifth international Joint Conference on Autonomous Agents and Multiagent Systems*, May 2006, pp. 89-96.

[6] H. Noser, O. Renault, D. Thalmann, and N. M. Thalmann, "Navigation for digital actors based on synthetic vision, memory, and learning," *Computers and Gra-*

*phics*, vol. 19, no. 1, pp. 7-19, 1995.

[7] J. J. Kuffner and J. C. Latombe, "Fast synthetic vision, memory, and learning models for virtual humans," in *Proceedings Computer Animation 1999*, *IEEE*, May 1999, pp. 118-127.

[8] I. Wang and J. Ruiz, "Examining the use of nonverbal communication in virtual agents," *International Journal of Human–Computer Interaction*, vol. 37, no. 17, pp. 1648-1673, 2021.

[9] P. Budzianowski, T. H. Wen, B. H. Tseng, I. Casanueva, S. Ultes, and O. Ramadan, et al., "MultiWOZ--a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling," arXiv preprint arXiv:1810.00278, 2018.

[10] C. Gershenson, V. Trianni, J. Werfel, and H. Sayama, "Self-organization and artificial life," *Artificial Life*, vol. 26, no. 3, 391-408, 2020.

[11] A. J. Ijspeert, J. Nakanishi, and S. Schaal, "Movement imitation with nonlinear dynamical systems in humanoid robots," in *Proceedings 2002 IEEE International Conference on Robotics and Automation*, *IEEE*, May 2002, vol. 2, pp. 1398-1403.

[12] M. Oshita, "Interactive motion synthesis with optimal blending," *Computer Animation and Virtual Worlds*, vol. 25, no. 3-4, pp. 311-319, 2014.

[13] M. Geilinger, R. Poranne, R. Desai, B. Thomaszewski, and S. Coros, "Skaterbots: Optimization-based design and motion synthesis for robotic creatures with legs and wheels," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 4, pp. 1-12, 2018.

[14] J. Wang, S. Yan, B. Dai, and D. Lin, "Scene-aware generative network for human motion synthesis," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 12206-12215.

[15] G. Carbone, E. C. Gerding, B. Corves, D. Cafolla, M. Russo, and M. Ceccarelli, "Design of a two-DOFs driving mechanism for a motion-assisted finger exoskeleton," *Applied Sciences*, vol. 10, no. 7, p. 2619, 2020.

[16] L. Kovar and M. Gleicher, "Automated extraction and parameterization of motions in large data sets," *ACM Transactions on Graphics (ToG)*, vol. 23, no. 3, pp. 559-568, 2004.

[17] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli, "Gait and trajectory optimization for legged systems through phase-based end-effector parameterization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1560-1567, 2018.

[18] L. Y. Chen, H. Huang, E. Novoseller, D. Seita, J. Ichnowski, and M. Laskey, et al., "Efficiently learning single-arm fling motions to smooth garments," arXiv preprint arXiv:2206.08921, 2022.

[19] T. Kwon and S. Y. Shin, "Motion modeling for on-line locomotion synthesis," in *Proceedings of the 2005 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2005, Jul, pp. 29-38.

[20] R. Heck and M. Gleicher, "Parametric motion graphs," in *Proceedings of the 2007 Symposium on Interactive 3D Graphics and Games*, 2007, Apr. pp. 129-136.

[21] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," arXiv preprint arXiv: 1508.06576, 2015.

[22] G. W. Taylor and G. E. Hinton, "Factored conditional restricted Boltzmann machines for modeling motion style," in *Proceedings of the 26th Annual International Conference on Machine Learning*, Jun. 2009, pp. 1025-1032.

[23] D. Holden, J. Saito, and T. Komura, "A deep learning framework for character motion synthesis and editing," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 4, pp. 1-11, 2016.

[24] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: LSTM cells and network architectures," *Neural Computation*, vol. 31, no. 7, pp. 1235-1270, 2019.

[25] I. K. M. Jais, A. R. Ismail, and S. Q. Nisa, "Adam optimization algorithm for wide and deep neural network," *Knowledge Engineering and Data Science*, vol. 2, no. 1, pp. 41-46, 2019.

[26] F. J. Ordóñez and D. Roggen, "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition," *Sensors*, vol. 16, no. 1, p. 115, 2016.

[27] C. Xu, D. Chai, J. He, X. Zhang, and S. Duan, "InnoHAR: A deep neural network for complex human activity recognition," *Ieee Access*, vol. 7, pp. 9893-9902, 2019.

[28] S. Mekruksavanich and A. Jitpattanakul, "Lstm networks using smartphone data for sensor-based human activity recognition in smart homes," *Sensors*, vol. 21, no. 5, p. 1636, 2021.

[29] K. Xia, J. Huang, and H. Wang, "LSTM-CNN architecture for human activity recognition," *IEEE Access*, vol. 8, pp. 56855-56866, 2020.

## AUTHOR

**Yuwei Wang** received his Bachelor Degree at Xi'an Academy of Fine Arts in 2011 and received his Master Degree at Henan Normal University in 2018. He has been a lecture at Xinxiang University since 2011. Currently, his research interests includes Dynamic Image Modeling, Film and Television Animation, and Computer Science.