ITC 3/54
Information Technology and Control
Vol. 54 / No. 3/ 2025
pp. 1077-1094
DOI 10.5755/j01.itc.54.3.40821

Volleyball Pose Recognition System Based on DLSTM-GCN Algorithm

Received 2025/03/13 | Accepted after revision 2025/07/22

HOW TO CITE: Wang, J., Wu, X., Feng, W., Liu, H. (2025). Volleyball Pose Recognition System Based on DLSTM-GCN Algorithm. *Information Technology and Control*, 54(3), 1077-1094. https://doi.org/10.5755/j01.itc.54.3.40821

Volleyball Pose Recognition System Based on DLSTM-GCN Algorithm

Jing Wang
Physical Education College, Shanxi Vocational University of Engineering Science and Technology, Jinzhong, 030606, China

Xue Wu*, Wei Feng, Hongyan Liu
Sports Work Department, Hebei Academy of Fine Arts, Shijiazhuang, 050700, China

Corresponding author: wuxue254065162@163.com

Aiming at the current problems of poor volleyball pose recognition and low recognition accuracy, this study proposes a volleyball pose recognition system based on dual long short term memory network graph convolutional network. A volleyball pose recognition system based on dual long short term memory network graph convolutional network is proposed. The new system firstly estimates and analyzes the volleyball sports data by multi-scale feature extraction and fusion network. Secondly, dual long short term memory network graph convolutional network is introduced to analyze the estimated video data. The outcomes revealed that among the different algorithmic model losses, multi-scale feature extraction and fusion network had the smallest loss value, which was reduced by 0.011 compared to the OpenPose algorithm. Meanwhile, the improved dual long short term memory network graph convolutional network was able to achieve the highest recognition accuracy of 93.68%, which was improved by 2.00% compared to the unimproved one. The running time of the improved dual long short term memory network graph convolutional network was shorter at only 2.1s, and the recall was also able to reach up to 92.68%. Meanwhile, in the actual running test of different algorithms, the improved dual long short term memory network graph convolutional network had more key point detection and more specific image action. In summary, the improved algorithm has better volleyball action recognition effect, which has a better application effect on volleyball sports pose recognition and volleyball action guidance research.

KEYWORDS: DLSTM-GCN, MFEFNet, Volleyball, Pose estimation, Pose recognition

1. Introduction

Accurate posture recognition in volleyball is crucial for athlete training and game analysis [22]. However, traditional methods rely on manual feature extraction, which is difficult to adapt to complex and changing motion scenes, and the model's feature representation ability is limited, resulting in insufficient recognition accuracy [28]. Although existing research has made some progress in action recognition, its effectiveness in specific volleyball sports still needs to be optimized [23]. In addition, Long Short Term Memory (LSTM) networks excel in temporal modeling, while Graph Convolutional Networks (GCN) can effectively handle non Euclidean data. However, how to combine the spatiotemporal features of both to improve action recognition performance remains a challenge [13].

As a result, designing a deep learning model that incorporates spatio-temporal features to achieve accurate recognition of volleyball sports poses has become the main problem of volleyball sports recognition at present. For example, in the study of Liu et al. a combined OpenPose and DeepSORT model for real-time pose estimation and tracking was proposed in an effort to improve the accuracy and real-time performance analysis of volleyball performance. The research results demonstrated a good pose recognition accuracy and real-time performance [12]. The method is able to estimate the motion poses but the effect on the recognition of motion poses still needs to be further explored. To improve the accuracy and versatility of the motion analysis technique, Bose et al. proposed a framework for analyzing athletes' pose and motion based on 3D pose estimation techniques and specific parameters. The results demonstrated that by using machine learning models, the framework could effectively analyze sports videos to provide performance information and real-time feedback. Its accuracy performance for training on various models was excellent [2]. Although the new method can effectively improve the recognition of sports, the recognition effect for specific sports still needs to be further explored. To improve the precise control ability of the robot in volleyball task, a sensor information integration method based on cross-modal self-attention mechanism is proposed in the study of Wang et al. and combined with GAN and migration learning to enrich the acquisition effect on image data. The outcomes demonstrated that the new method significantly improved the performance of the robot in the volleyball task [26]. Although the study has better results in robot action analysis, whether the method can recognize human pose action still needs to be further explored.

Based on this, the study innovatively proposes a volleyball action pose recognition system based on dual long short term memory network graph convolutional network (DLSTM-GCN) algorithm. The DLSTM-GCN algorithm can analyze the spike actions of volleyball players, with LSTM processing the temporal changes in consecutive frames and GCN modeling the joint spatial relationships. The new system firstly introduces multi-scale feature extraction and fusion network (MFEFNet) to realize the estimation and analysis of human pose movement, so as to improve the volleyball action recognition effect. When MFEFNet network detects volleyball serve movements, small-scale convolution captures finger joint details, and large-scale convolution recognizes overall body posture. The study introduces a multi-scale feature extraction and fusion network. The network significantly improves the accuracy and stability of pose estimation by capturing finger joint details and overall body pose through convolutional layers at different scales. The study also proposes an improved dual long short-term memory network graph convolutional network (DLSTM-GCN) for analyzing spatio-temporal features in video data. The temporal variations of consecutive frames are handled by the LSTM and the GCN models the joint spatial relationships to improve the accuracy of action recognition. And by removing redundant data and introducing recognition termination strategy, the inference cost of the model is reduced and real-time performance is improved. Second, to increase the accuracy of volleyball action detection, the system implements the DLSTM-GCN method to identify the human posture action. The new system aims to promote the progress of volleyball and the whole sports science field, and provide more scientific and accurate analysis tools for athletes' training and competition.

2. Methods and Materials

2.1. Analysis of Human Pose Action Estimation Based on DLSTM-GCN Algorithm

2.1.1. Estimation and Correction of Human Posture Points

During rapid motion, the human background and obstacles can cause deviations with the human motion process. This makes the estimator for human pose capture not able to capture the human pose accurately. At the same time the initial pose point position will also change with the change of pose movement resulting in inaccurate recognized movements [4, 24]. Therefore, for volleyball posture capture the human pose points need to be detected and tracked. The study proposes a feature extraction network for multi-scale fusion to correct and estimate the human pose point locations. The recognition process of the network is mainly to determine the posture points of the acquired images through the network first, and then finally realize the determination and correction of the human posture points through the human posture correction algorithm. Since most of the human poses in the open volleyball sports scene do not involve the determination of specific pose points of the human face, it is necessary to label different pose points during human pose recognition. Equation (1) shows the labeled pose point formula [1, 30].

$$I_j(t) = \{(x_i(t), y_i(t)), z_i\}_{i \in I}. \quad (1)$$

In Equation (1), $I_j(t)$ denotes the j th bit pose point as a collection of information data at time t. $(x_i(t),y_i(t))$ denotes the horizontal and vertical coordinates of the bit pose point j at time t. z_i denotes the j confidence level of the bit position point. The value range of t is the total duration of the video. Define the coordinates and confidence level of the i-th pose point at time t. Since the network belongs to deep convolutional network contains multiple residual blocks, the determination and accuracy of human body's bit pose needs to be further improved. Therefore, a pose feature extraction method using convolutional residuals is investigated by filtering and analyzing multiple different convolutional layers from a simple deep network. The network layer will first normalize the convolutional layers for analysis and activation function activation to generate a new convolutional network and scalar map [11, 6, 25, 7]. In the study MFEFNet uses the traditional network structure for feature extraction network structure for multi-scale fusion, as shown in Figure 1. In MFEFNet, residual blocks help the network learn multi-scale features more stably.

In Figure 1, the feature extraction network structure for multi-scale fusion consists of two main stages of network operation. In the first stage, the network first convolves the pose point data through different convolutional layers, which also contain normaliza-

Figure 1
Feature extraction network structure for multi-scale fusion.

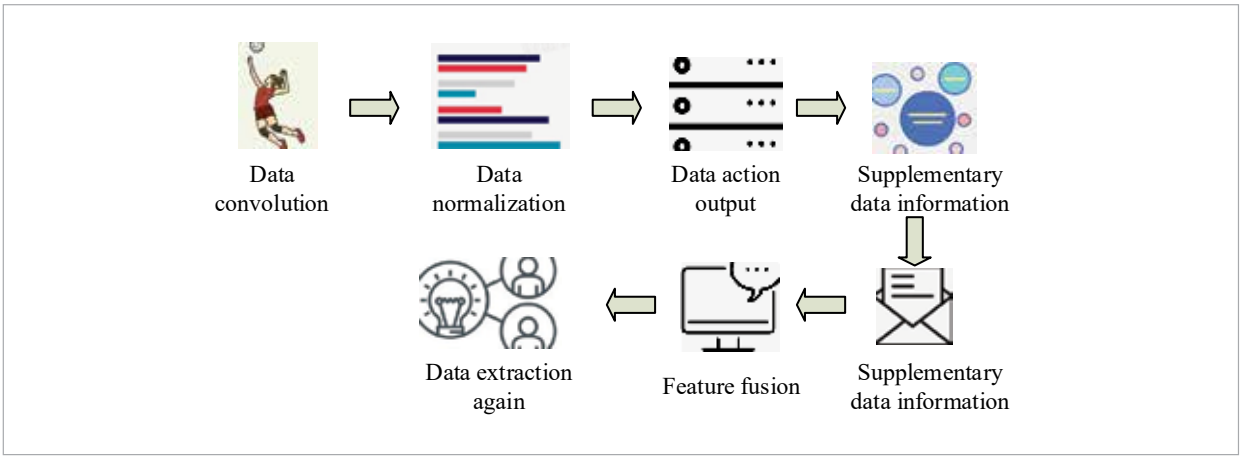
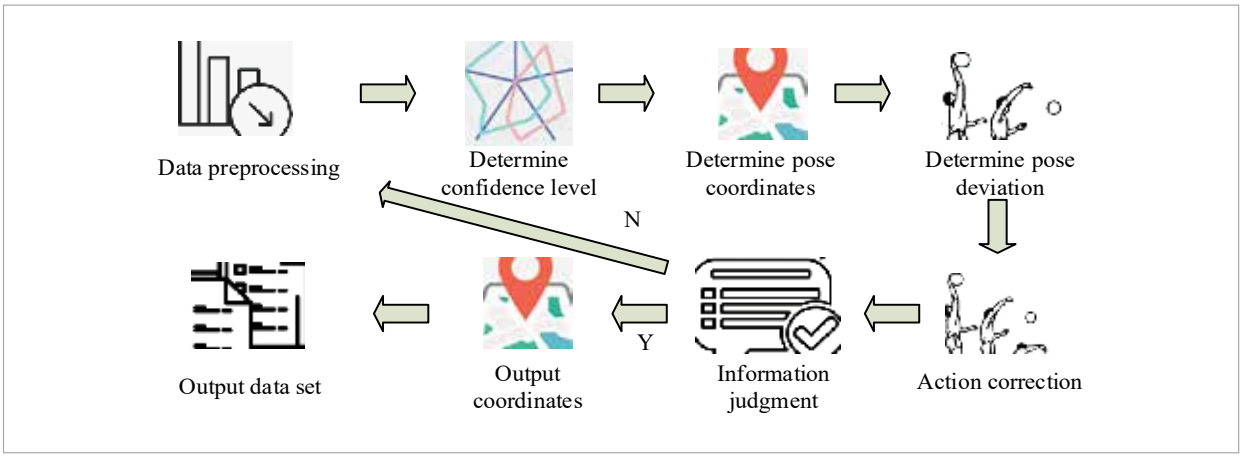


Figure 2
Algorithm running process.



tion, activation function, and data output layers. In the second stage of the network, the pose point data information processed in the first stage is used as an input layer for feature supplementation, and these data are fused and new feature fusion information is obtained. Finally, the network will extract the feature fusion information to realize a better fusion effect. The final output data feature set will be convolved with 1*1 again to reduce the feature fusion channel of the data and reduce the amount of data computation. To improve the model training effect, the study uses the pose correction algorithm. Figure 2 shows the algorithm operation flow.

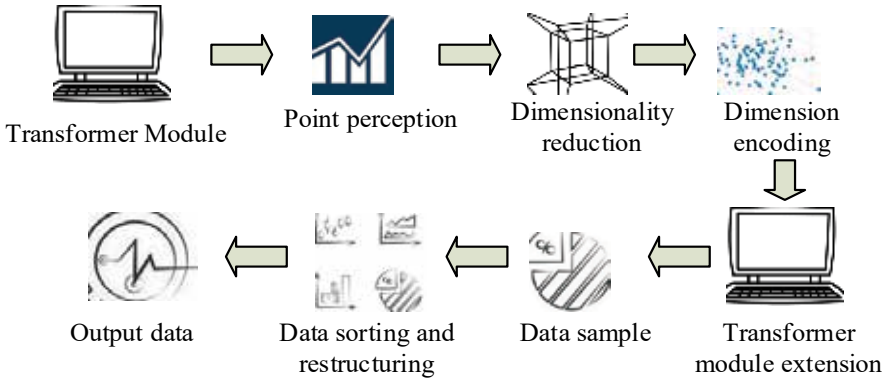
In Figure 2, the algorithm runs by first sorting and preprocessing the input human body position information data, followed by determining the confidence size of the current position information. Then the average coordinate of the position point is determined according to the confidence level of the position point. Then the average position change is determined according to the average position change, and the average position deviation size of the position point is corrected by the deviation size of the human position point sequence data. It then judges whether the current position point information is correct or not. If it is correct, the coordinates of the current position point information will be output directly, and if it is wrong, the data will be re-input for preprocessing. Until the current position information data is correct, the final set of all position point data information is output.

2.1.2. Three Dimensional Human Pose Data Capture

The evaluation of human pose is analyzed using stereo dimensional pose estimation method can reduce the data loss due to photography and angular bias and thus enhance the details of motion capture for volleyball movement [7, 15]. Therefore, the study uses 3D human pose data point capture method to capture the actual human pose variation. The 3D human pose data capture network structure is shown in Figure 3.

In Figure 3, in the human pose capture network the network consists of multiple Transformer modules. Different modules have different effects on the capture of pose movements. Among them, the spatial Transformer module is able to share the data of the pose data points in the space and realize the information exchange between different data points. The network structure in the first stage first senses each bit-posture point location. After that, it is downscaled to one-dimensional coding. Finally, the one-dimensional code is input as the initial data for the next stage. The second stage of the network structure requires the extension of the Transformer module of the target, followed by dimensionality enhancement and data sampling. Finally, the dimensionally upgraded bit position points are sorted and reorganized, and then encoded by the Transformer module to finally output a single linear bit position point data.

Figure 3
Network structure for capturing 3D human pose data



2.2. Design of DLSTM-GCN Based Estimation Algorithm for Bit Position Recognition

2.2.1. Joint Modeling of Spatiotemporal Features

The volleyball motion capture and estimation in open scenarios can improve the correlation of the motion points. Therefore, the classification and identification of bit-posture point location data information is more important. Thus, the study uses the DLSTM-GCN bit-posture point analysis algorithm on the basis of the volleyball movement bit-posture point estimation method in the previous section to estimate and identify the actual volleyball movement points. The DLSTM-GCN algorithm is an improved algorithm based on LSTM and GCN. In LSTM and GCN networks will first analyze the human body's movements by classifying them through modules [27, 21, 29]. The network will first assume that the current network exists a time sequence $[R_t,R_{t+1},\ldots,R_{t+T}]$ consisting of human body position point locations. Among them, R_t denotes the position point states that change in the network at time t, and the matrix Y_t is obtained according to the different position point changes. Therefore, the input of the whole network is shown in Equation (2).

$$\ddot{Y}_t = \arg\max L(Y_r \mid Y_{t-T},\cdots,Y_{t-1}). \quad (2)$$

In Equation (2), $Y_{t-T},\ldots,Y_{t-1}$ denotes the adjacency matrix between the first T matrices. $\ddot{Y}_t$ denotes the prediction matrix at moment t. Y_t denotes the true matrix at moment t. The network model first judges the previous network state through the forgetting gate (FG). The current situation of lost network units is judged as shown in Equation (3) [16, 14].

$$D_t = o(Y_t W_D + GCN_D^k(\ddot{Y}_{t-1}, h_{t-1}) + b_D). \quad (3)$$

In Equation (3), D_t denotes the output of the FG at moment t. o is the activation function. W_D denotes the weight matrix of the FG. GCN_D^k is the k layer network in GCN. $\ddot{Y}_{t-1}$ is the prediction matrix at moment t–1. h_{t-1} is the hidden state at moment t–1. b_D is the bias term of the FG [5, 19]. At this point, the bias matrix of the OG is judged as shown in Equation (4) [3, 17].

$$E_t = \theta_t \odot \tanh(\dddot{C}_t). \quad (4)$$

In Equation (4), E_t denotes the deviation matrix of the OG. The network passes through the decoder to output the network probability changes and obtains the final matrix case as shown in Equation (5) [18, 20].

$$Q_d^{(1)} = \mathrm{Re}\,LU(W_d^{(1)} E_t + b_d^{(1)}). \quad (5)$$

In Equation (5), $Q_d^{(1)}$ denotes the output of the decoder. ReLU denotes the activation function $W_d^{(1)}$ num-

ber, which represents the first weight moment of the decoding go. $b_d^{(1)}$ denotes the first layer bias vector of the decoder. At this point, the multilayer output of the decoder is obtained as shown in Equation (6).

$$Q_d^{(k)} = \mathrm{Re}\,LU(W_d^{(k)} E_d^{(k)} + b_d^{(k)}). \quad (6)$$

In Equation (6), $Q_d^{(k)}$ is the layer k output of the decoder. $W_d^{(k)}$ is the kth weighting moment of the decoder going. $b_d^{(k)}$ is the kth layer bias vector of the decoder. The probabilistic output matrix of the decoder is shown in Equation (7) [9, 8].

$$U_t = \theta_t \odot \tanh(\ddot{C}_t). \quad (7)$$

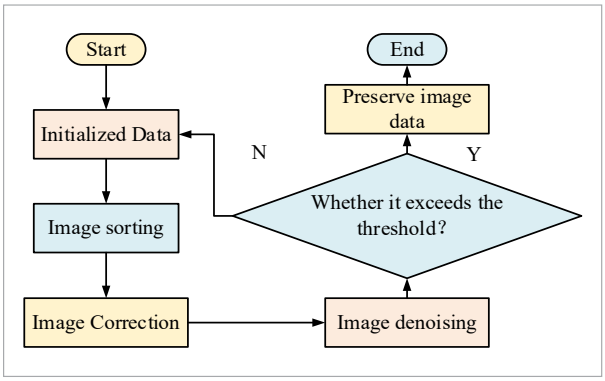
In Equation (7), U_t denotes the probabilistic output matrix of the decoder.

2.2.2. Redundant Data Processing and Identification Termination Strategy

According to the output of the network the different key points of the human body position are analyzed and the position data time points are ranked. Figure 4 shows the process of classifying the network structure bit pose point data.

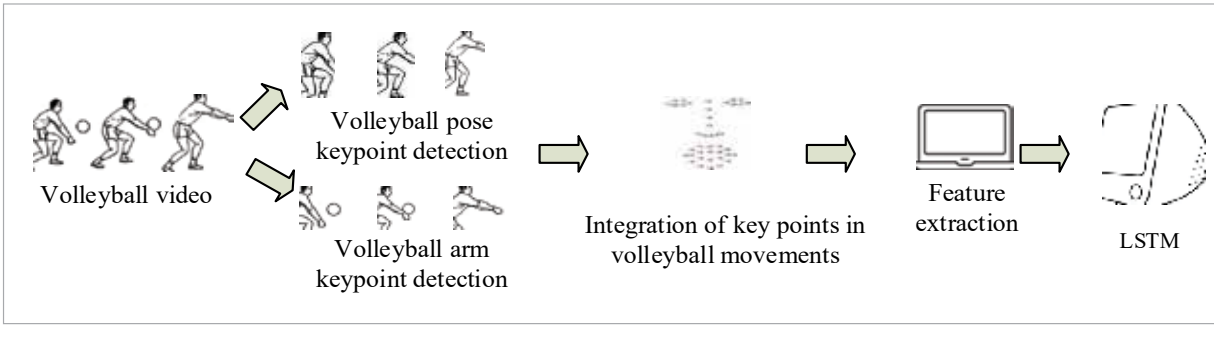
In Figure 4, in the classification of the point information in the network structure backline through the input video situation to get the key point information of the human body position and the human body's arm and other key parts of the information. Secondly, the key points are spliced according to the two key point information. After the splicing is completed the graph data information of the points is feature extracted by GCN and finally the classification of the point data is completed using LSTM. DLSTM-GCN algorithm is based on the combination of traditional LSTM and GCN model and then add the lightweight module to improve, and add the recognition termination strategy to achieve the termination of the recognition of the positional gesture points. DLSTM-GCN algorithm model in the volleyball image action recognition process will first de-clutter the image, the video image of the action change situation and data points redundant gesture to remove, reduce the inference cost of the model. Figure 5 shows the model de-cluttering process.

Figure 5
Model de-clutter process.



In Figure 5, in the process of de-duplication of video images, the algorithmic model will first sequence sort the initialized video images and correct the sorted images to store the de-duplicated human body position sequence images. Then the redundant detection information is compared. If the current redundant image matrix is P then P is compared with the image of the next video frame. When P is

Figure 4
Network structure pose point data classification process.



the filtered image matrix then the matrix is updated and incorporated into the sequence collection. Then again it is compared with the matrix of the next video frame until all the non-redundant data are able to complete the update. The extracted redundant data is eliminated by redundancy elimination formula as shown in Equation (13) [10].

$$V(P_i, P_j) = 1[oks(P_i, P_j)|\Delta| \geq \lambda]. \quad (8)$$

In Equation (3), $V(P_i, P_j)$ denotes the coordinates of data points after removing redundant data. *oks* denotes the judgment criterion to determine whether the two coordinates are the same. Δ denotes the set. λ denotes the threshold for removing redundant data information. When the size of the removal threshold is greater than or equal to the threshold value, the function coordinate at this time is 1, indicating that the two bit position points are duplicated or have not shown large deviation. Then it means that the current point position coordinates are redundant coordinates and the current coordinate information is deleted. If the size of the removal threshold is smaller than the threshold, the current function coordinate is 0, indicating that there is a large deviation between the two point position coordinates. This states that the current point coordinates are not redundant coordinates and are retained.

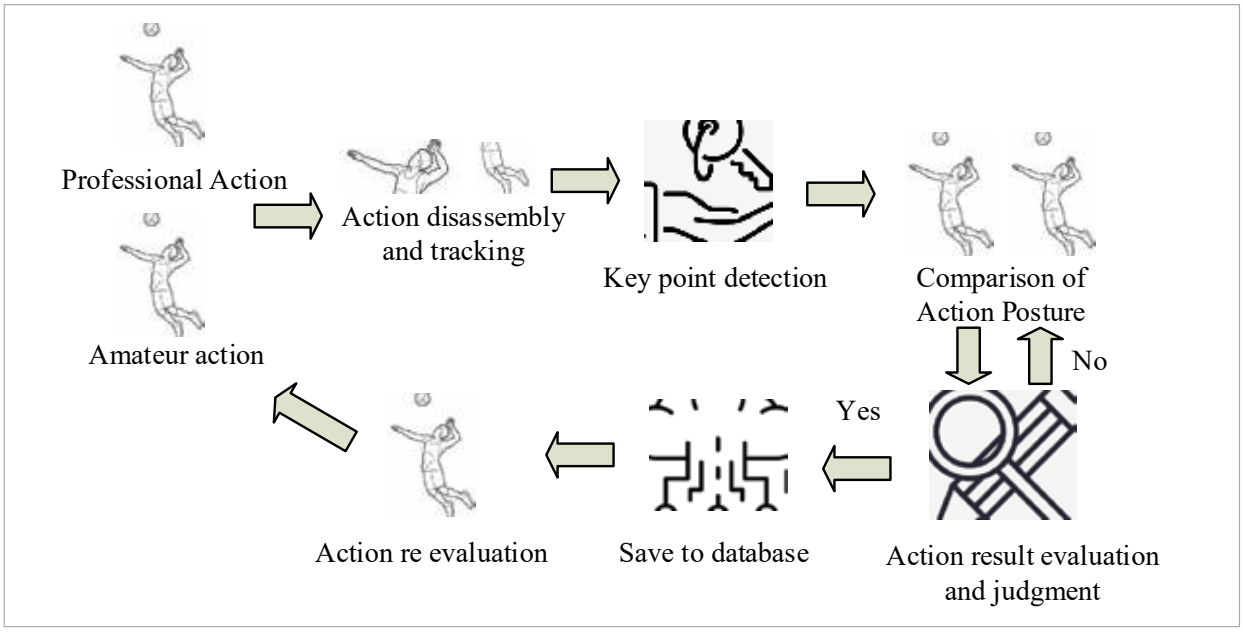
The recognition termination strategy incorporated in the algorithm can reduce the data information dependency of the model due to the bias of the processing effect on the data. The recognition termination strategy will average pool the data in the processed data sequence, and then classify the hidden layer of the current feature data through the LSTM module to obtain a new bit position probability matrix. Then the threshold value and probability distribution changes are judged by the set threshold value. If the recognition probability is greater than the threshold, the current recognition action is terminated. If the recognition probability is less than the threshold value the current recognition action continues.

2.3. Volleyball Action Position Recognition System

2.3.1. System Architecture and Hardware Configuration

To realize the practical application of the system, the system needs to meet the following operating conditions. Firstly, the system needs to be able to

Figure 6
Framework structure of the system.



realize the recognition and detection of volleyball movements, and to be able to track and recognize several different human targets and data points in high motion scenes. Second, after receiving fresh data, the system must be able to identify and categorize human motions. The system must simultaneously be able to identify and evaluate volleyball moves. Lastly, the system must function and make use of various network technologies. For this study the software tools used in building the system include Visual Studio for development, TensorFlow for system development framework, MariaDB as database for system development, Windows 10 for the computer used for the system, and Radeon RX 6800 for the graphics card. Figure 6 illustrates how the actual runtime is used to determine the system's overall framework structure.

In Figure 6, in the system framework mainly two different actions will be recognized, including the recognition of the volleyball action video of professional athletes, and secondly, the recognition and analysis of the volleyball action of amateurs. The two different recognition processes are detected and analyzed by the system to determine and track the positional points of the two different actions. Secondly, the system uses a point tracker to extract the volleyball action points and get the video action data information after extracting the key points. Then the video action data is identified and analyzed by DLSTM-GCN algorithm, and the actions are evaluated and analyzed according to the corresponding volleyball action rules. Based on the assessment results, the similarity of the movement position is judged again. If the current movement does not meet the volleyball movement criteria then it is re-identified and compared with the professional movement. If it meets the standard, the current movement is stored in the database.

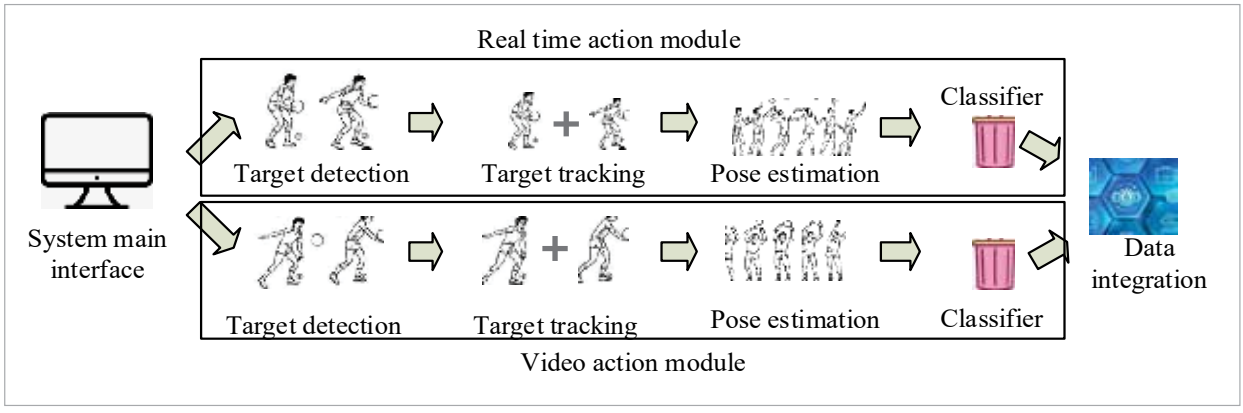
2.3.2. Real Time and Offline Action Recognition Process

Through the system recognition, it can recognize and classify the movements of athletes in different situations, which is more helpful to improve the movement standard of amateur athletes. At the same time, there are two main recognition modules in the process of action recognition, one is the real-time action recognition module, and the other is the action detection module of the video image. The action recognition process of the two different modules is shown in Figure 7.

In Figure 7, during the volleyball action recognition, the system operates on the volleyball action through the main interface. Both modules perform the same point recognition process. Firstly, the target in the video is detected. Secondly, the action process of the target is tracked. Then, the pose of the target is estimated and classified by DLSTM-GCN algorithm. Finally, according to the structure of classification, the target bit-posture point data is inputted into the classifier and finally the recognition of the video action is realized. After the action recognition is completed by the two modules, the model will integrate the recognition output data of different video information and finally output the integrated video data.

Figure 7

Action recognition process of two different modules.



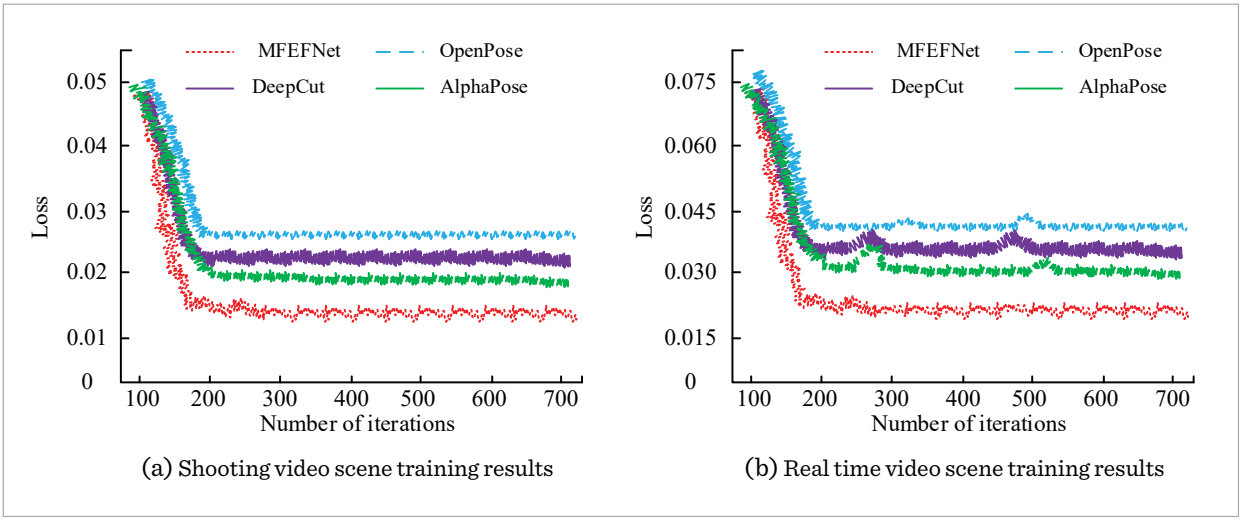
3. Results

3.1. Analysis of the Effect of Pose Point Estimation

The experiments in this paper use the Leeds Sports Pose database, which contains about 2000 volleyball action pose samples, covering both professional athletes' game scenarios and amateur training scenarios. Real-time match video clips with resolution ≥1080p, capturing athletes' actions such as serving, dunking, blocking, etc., including complex backgrounds. Amateur scene contains indoor training video resolution 720p, actions including basic passing, padding, etc., with simple background. Each frame contains 17 keypoints, which are manually labeled and pre-labeled by OpenPose with a confidence threshold of 0.8. The preprocessing process removes the repetitive action data in consecutive frames by inter-frame differencing. Then the coordinates of key points were normalized to the interval [-1, 1] to eliminate the effect of resolution difference. Finally, random rotation (±15°), horizontal flip, and brightness adjustment of 20% were performed. The system is equipped with AMD Radeon RX 6800 XT graphics card and Intel Core i7-12700K CPU, supporting 4K video real-time input (30 FPS). After the camera captures the image, MFEFNet is used for multi-scale feature extraction, and the DLSTM-GCN algorithm completes the action classification with a throughput of 50.1 FPS, and the results are fed back to the training interface in real time. The central processor used for the study is Intel Core i7-12700K and the image processor is AMD Radeon RX 6800 XT. The size of the memory used for the study is DDR4-3200MHz and the hard disk is Crucial P5 Plus NVMe M.2 SSD. The operating system used for the study is Windows 10. The software used for the study is Python 3.9, Anaconda 2023.10 64bit, and CUDA 12.0. The study's model's learning rate is set at 0.001. 2000 videos are used for training, and they are split into training and test sets in a 7:3 training to test ratio. During the training process, the model's learning rate decays to 1/10, and 1000 iterations are used. The judgment criterion for model bit position estimation is judged using the average bit position point deviation value O. The number of residual block layers in MFEFNet is 4 layers of residual blocks. When increasing to 6 layers, the GFLOPs increase to 25.3, but the accuracy only improves by 0.7%, so a shallow layer structure is chosen. 2-layer LSTM for the spatio-temporal module of DLSTM-GCN can effectively model the timing dependence, and more than 3 layers will easily lead to overfitting. 3-layer GCN can capture the joint space relationship, and too many layers will introduce noise. Initial learning rate is 0.001 to avoid gradient explosion. Decay to 1/10 every 200 iterations, and the final learning rate is 1e-5. This strategy allows the model to fine-tune

Figure 8

Comparison of loss results of video models in different scenarios.



(a) Shooting video scene training results

(b) Real time video scene training results

the parameters in the later stages of training and stabilize the convergence. The redundancy threshold is set to 0.05, if the difference of keypoint coordinates between two frames is <5%, it is recognized as redundancy. The size of the deviation value is the size of the Euclidean distance between two point positions. Comparison of OpenPose algorithm, AlphaPose algorithm, DeepCut algorithm, and MFEFNet network for the comparison of the loss situation of the bit position estimation algorithm is obtained as shown in Figure 8.

In Figure 8(a), in the video scene analysis, the loss value of the model deduces relatively after the quantity of iterations increases, and the model tends to be relatively stable after reaching a certain loss value. Among them, the OpenPose model has the highest loss value after the loss value is stabilized, with the highest value of 0.027, while the MFEFNet algorithm among the other algorithms has the lowest loss value of only 0.016. The difference in the loss value of the two algorithmic models is 0.011. It can be observed that the MFEFNet network among the different algorithmic models is more effective and more stable during the training process of the video scene model. This may be due to the reason that the algorithm evaluates the bit pose better. In Figure 8(b), the loss values of different algorithmic models are increased in real-time video scene analysis, which may be due to the reduced operational effectiveness of more complex models for real-time scene processing. Meanwhile, from the loss value after stabilization, the MFEFNet network has a lower loss value of only 0.021 in real-time scene estimation, while the OpenPose model can reach a maximum of 0.041. There is a 0.020 discrepancy between the two models' loss values. Moreover, the loss values of the algorithms other than the MFEFNet network have slight fluctuations, which may be due to fluctuations of the algorithms caused by the transformation of the scene. The results display that the MFEFNet network has better algorithmic stability and better results for the estimation of real-time scenes and video scenes. Paired t-test showed that the average loss value of MFEFNet in video scenes was significantly lower than OpenPose, with a significant statistical difference (p<0.01) and a confidence interval size of 0.008-0.014. To compare the estimation effect of different algorithmic models, the accuracy, recall, and F1 score of different algorithms are compared and analyzed. Table 1 shows the detailed information.

In Table 1, in the comparison of the effect of different model runs, the video scene is compared better than the real-time scene, and the different models are applied better. This may be due to the simpler background of the video scene. In the video scene, the MFEFNet network has the best running effect (RE), the accuracy can reach 93.16%, compared with the OpenPose algorithm its accuracy is improved by 5.51%. It can be noted that in different algorithmic models, its recall is improved by 6.87% and F1 score is improved by 9.64%. The bit pose estimation algorithm MFEFNet used in the study runs better in different video scene runs, which may be due to the ability of the network to go through multi-scale feature extraction. Since the video varies from frame to frame, in an effort to compare the actual model video runs, the study compares the model runs at different frame numbers. ANOVA analysis showed that MFEFNet had significantly higher accuracy in video scenes than other models (F=12.4, p<0.001), and there was also a significant difference in accuracy between simple and real-time scenes (p=0.003). The

Table 1

Comparison of training results for models in different scenarios.

Scene	Network model	OpenPose	AlphaPose	DeepCut	MFEFNet
Video scene	Accuracy (%)	87.65	89.54	90.35	93.16
	Recall (%)	84.56	85.62	89.15	91.52
	F1 (%)	80.12	82.35	86.47	89.76
Real time scene	Accuracy (%)	84.63	87.52	88.26	90.15
	Recall (%)	82.10	83.64	87.26	89.11
	F1 (%)	79.84	80.01	85.16	87.95

Table 2
Comparison of running effects of different video step models.

Video stride	Scene	Network model	OpenPose	AlphaPose	DeepCut	MFEFNet
The video step size is 4	Video scene	Key-frames (FPs)	35.4	36.7	35.7	40.5
		All frames (FPs)	36.8	37.2	37.1	40.6
		O	27.5	26.3	24.1	20.2
		GFLOPs (V)	48.5	35.6	31.8	18.9
	Real time scene	Key-frames (FPs)	34.3	35.7	33.8	39.4
		All frames (FPs)	35.9	36.9	36.7	40.3
		O	28.6	27.1	25.3	22.4
		GFLOPs (V)	51.2	38.6	33.7	22.7
The video step size is 10	Video scene	Key-frames (FPs)	35.4	36.7	35.7	40.5
		All frames (FPs)	36.8	37.2	37.1	40.6
		O	27.3	26.5	24.9	20.1
		GFLOPs (V)	48.3	35.5	31.2	18.7
	Real time scene	Key-frames (FPs)	34.3	35.7	33.9	39.4
		All frames (FPs)	35.8	36.9	36.7	40.4
		O	30.1	28.4	26.8	24.0
		GFLOPs (V)	51.2	38.1	33.7	22.2

model's real-world implementation is contrasted for the general frame number, video step variation, and critical point frame number. Table 2 displays the findings. GFLOPs measure the computational complexity of a model, with smaller values indicating higher computational efficiency.

In Table 2, GFLOPs denotes the number of gigafloating point operations per second of the model, the smaller its value the better the model effect. Different algorithms have better results in key frame comparison in MFEFNet network with higher number of comparison frames. Among them, when the step size is set to 4, the key frame number of MFEFNet model is improved by 5.1FPS than OpenPose algorithm, and the total frame comparison of the model is also improved by 3.8FPS. It can be concluded that MFEFNet is more effective in the comparison of key frames of the model. The comparison results of the deviation value of the model's average positional point show that the deviation value of MFEFNet is the smallest only 20.2. Meanwhile, the GFLOPs value of the model is also the smallest only 18.9V. It can be concluded that the MFEFNet network has a better practical RE. Meanwhile, when the step size setting is increased, the deviation effect of the model changes less, which indicates that the change of the step size does not have much influence on the deviation of the model's operation effect. However, the deviation of the model in the real-time scene has a large change, which indicates that the change of the scene has a large effect on the operation effect of the model. Linear regression shows that increasing the video step size has no significant effect on the bias value (O) of MFEFNet (p=0.12), while the changes in GFLOPs in real-time scenes are strongly correlated with scene complexity (r=0.89, p<0.01)

3.2. Volleyball Sports Pose Recognition Results Analysis

To analyze the actual RE of the proposed improved DLSTM-GCN algorithm, the study compares the actual RE of the model under different algorithm runs based on the experimental scenarios in the previous section. The comparison is made between the RE of the model for complex scenes and a single simple scene under the actual RE of the model. Figure 9

shows the comparison of confusion matrices for different action scenarios. The diagonal line represents the accurately anticipated outcomes, the vertical coordinate represents the predicted results, and the horizontal coordinate represents the actual results. Simple action scenarios are categorized as basketball, table tennis, football, and tennis. Complex action scenarios are respectively receiving, serving, setting, and faking for volleyball actions. 40 of the simple action scenarios are extracted for prediction analysis, and 15 of the complex action scenarios are extracted for prediction analysis.

In Figure 9(a), the number of correct predictions in the simple action scene prediction results are above 30. This indicates that the improved DLSTM-GCN algorithm has a better model prediction ability when predicting simple scenes, with the highest value being able to reach 38. In Figure 9(b), in the prediction of complex action scenes, the actual prediction result of the model has a significant reduction, and its highest result of correct prediction can reach 15. It can be demonstrated that the improved DLSTM-GCN algorithm had good prediction effect when predicting different scenes of video data, which may be due to the algorithm added action recognition algorithm. The chi square test (p<0.001) showed that the improved DLSTM-GCN had significantly higher prediction accuracy in complex scenarios than the traditional algorithm MoveNet. To compare the action recognition effect of the improved DLSTM-GCN before and after the improvement, the improved module of the algorithm is analyzed for volleyball action recognition. Table 3 displays the findings.

Table 3

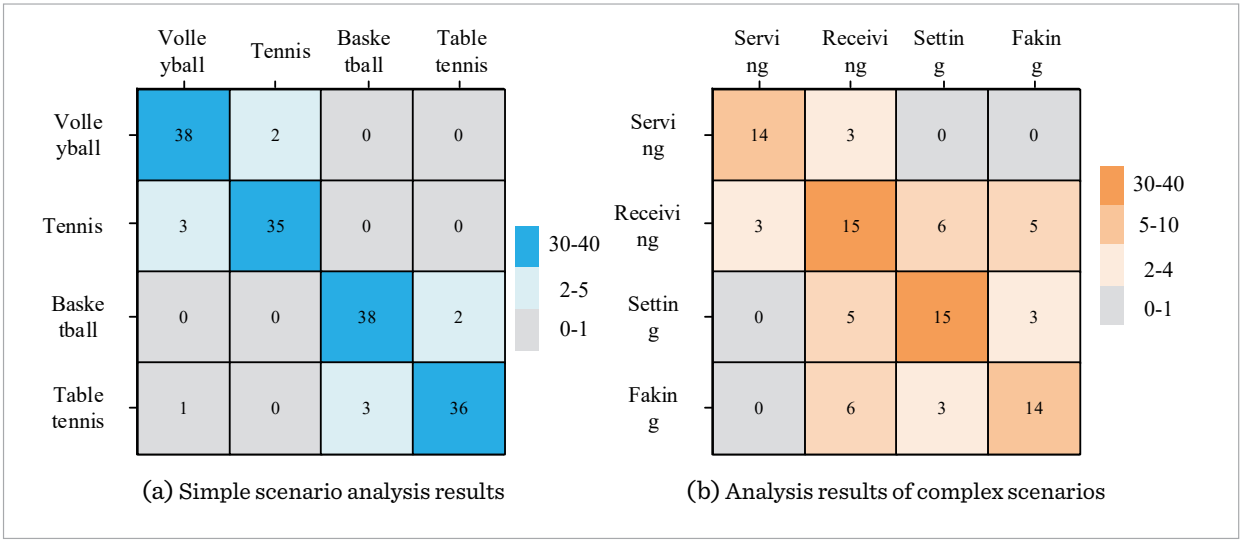
Comparison of recognition accuracy of different improved modules.

Volleyball moves	Improve DLSTM-GCN	DL-STM-GCN	MFEFNet
Serving	92.51%	90.32%	87.36%
Receiving	93.14%	91.03%	82.68%
Passing	92.34%	90.25%	84.67%
Setting	93.68%	91.68%	86.95%
Spiking	91.68%	90.35%	87.16%
Blocking	92.03%	91.02%	87.15%
Back row attack	93.28%	91.58%	80.36%
Dinking	92.68%	90.35%	81.26%

In Table 3, the action recognition accuracy of the model is significantly improved after adding different improved models. Among them, the recognition accuracy of the DLSTM-GCN algorithm before improvement can reach 91.68% in the second pass action. After the improvement, its action recognition effect can reach 93.68%, which is a 2.00% increase in

Figure 9

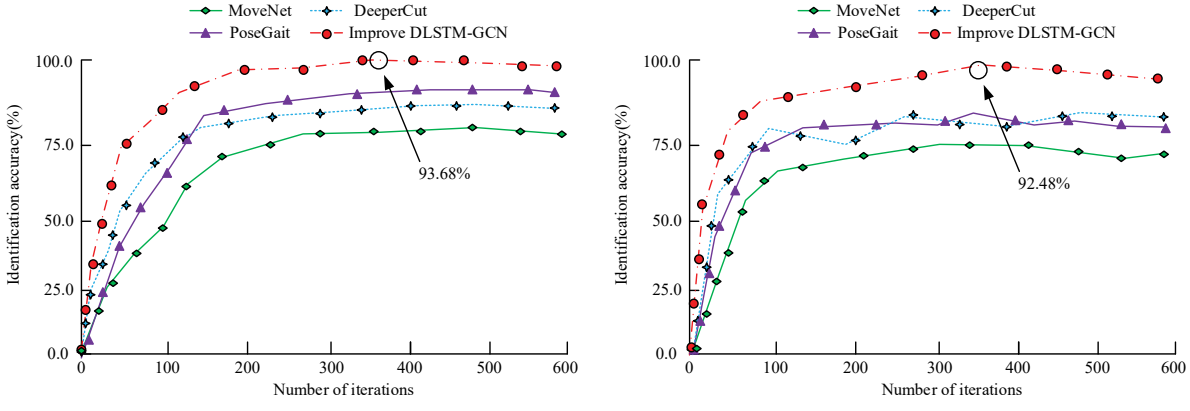
Comparison results of predicted values and true values in different scenarios.



(a) Simple scenario analysis results

(b) Analysis results of complex scenarios

Figure 10
Comparison results of recognition accuracy of different algorithm models.



(a) Simple action scene (b) Complex action scenes

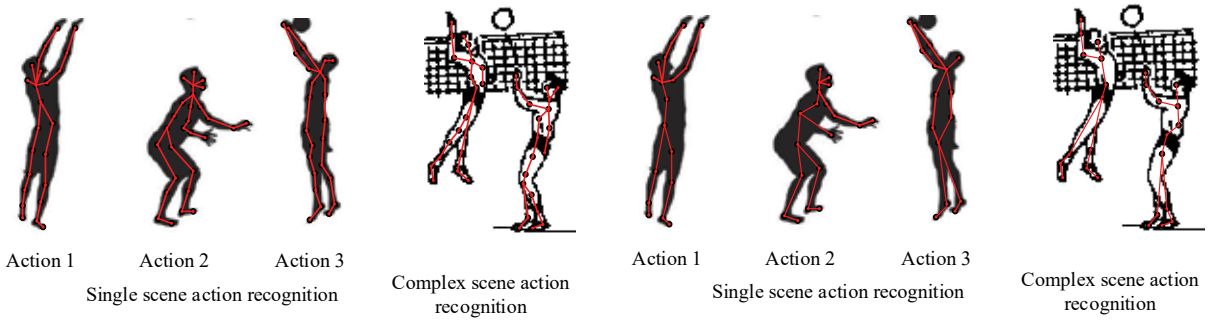
accuracy. This is due to the addition of action recognition model to the improved algorithm. Meanwhile for the comparison of action recognition network its recognition accuracy has a large decrease. Among them, in the two-pass action, the recognition accuracy of MFEFNet is only 86.95%, which is 6.73% lower compared to the accuracy of the improved DLSTM-GCN algorithm. It shows that the MFEFNet network is less effective in recognizing actions, which may be due to the fact that the network is more suitable for action estimation. To compare the actual operational effectiveness of different algorithms, the study compares the recognition accuracy of MoveNet network, DeeperCut algorithm, and PoseGait algorithm. Figure 10 presents the findings.

In Figure 10(a), in the comparison of recognition accuracy of different algorithms in simple action scenarios, the recognition accuracy of the improved DLSTM-GCN algorithm increases with the number of model iterations. The maximum recognition accuracy of the model is 93.68%, while the maximum recognition accuracy of MoveNet network is only 77.35%. Compared to the improved DLSTM-GCN algorithm, the recognition accuracy of MoveNet network is 16.33% lower. In Figure 10(b), there is a significant decrease in the recognition accuracy of the model in complex scenes. Among them, the highest recognition accuracy of the improved DLSTM-GCN algorithm is only 92.48%, which is 1.20% lower compared to the simple scene. This may be due to the fact that the complexity of the scene places higher demands on the model. Meanwhile, the recognition accuracy of the improved DLSTM-GCN algorithm is improved by 19.22% compared to the MoveNet network. The outcomes display that the improved DLSTM-GCN algorithm has better action pose recognition effect in different scenarios. Bootstrap confidence interval analysis of 1000 samples showed that the improved DLSTM-GCN had significantly higher accuracy in complex scenarios than MoveNet (73.26%, 95% CI 18.1). To compare the actual operation of different models, the study compares the data parameter information such as running speed and number of running parameters of different models to get as shown in Table 4.

Table 4
Comparison results of running conditions of different models.

Model	Run time (s)	Parameter quantity	Throughput (FPS)	Recall (%)
DLSTM-GCN	3.5	35421	45.8	90.25
MoveNet	6.8	53261	41.5	87.65
DeeperCut	5.6	42454	40.2	88.67
PoseGait	4.2	42357	43.3	89.12
Improve DLSTM-GCN	2.1	23457	50.1	92.68

Figure 11
Volleyball action recognition in different scenarios.



(a) Improve the DLSTM-GCN model (b) MoveNet model

In Table 4, in the comparison of the RE of different models, the improved DLSTM-GCN model has better RE. The shortest running time of the model is only 2.1s, and the quantity of parameters required for the model to run is only 23457. The throughput of the model can reach 50.1FPS, and the recall of the model can reach 92.68%. It displays that the actual operation of the improved DLSTM-GCN algorithm has better results compared with other models. Compared to the MoveNet network its runtime is improved by 4.7s, throughput by 8.6FPS and recall by 5.03%. This may be due to the inclusion of the action position estimation algorithm in the model. To compare the actual RE of the system, the study analyzes the actual running videos of different volleyball actions. Figure 11 shows the volleyball action recognition situation.

In Figure 11(a), the key point recognition of the improved DLSTM-GCN model in simple scene recognition is better, and more point information can be recognized, including the recognition of points such as the knee joints of the characters. In Figure 11(b), most of the key points are not recognized by the MoveNet network when recognizing the characters, which leads to the deviation of the overall recognition effect. This is the reason for the poor recognition effect of MoveNet network. Therefore, it can be observed that by improving the DLSTM-GCN model can significantly improve the recognition of volleyball movements, and the detection and recognition of point locations are also the best. To test the actual operational performance of the current method and more advanced methods, the DLSTM-GCN method was compared with other advanced methods and the results are shown in Table 5.

From Table 5, it can be seen that the improved DLSTM-GCN algorithm performs better than other advanced algorithms in terms of performance com-

Table 5
Comparison of Running Performance of Different Algorithms.

Model Name	Accuracy (%)	Recall (%)	F1 Score (%)	Run Time (s)	Parameter Count	GFLOPs
Improved DLSTM-GCN	93.68	92.68	93.16	2.1	23457	18.9
DLSTM	87.65	84.56	80.12	3.2	56268	48.5
GCN	89.54	85.62	82.35	5.6	63524	35.6
MFEFNet	77.35	87.65	86.52	6.8	53261	31.6
Swin Transformer	92.65	91	91.52	3.4	42685	32.8
MobilePose	90.55	89.5	90.26	3.5	36584	36.7
Multi-Modal Fusion	92.51	91.5	92.34	3.9	36846	38.4

parison among different algorithms. The recognition accuracy of the improved DLSTM-GCN algorithm has increased by 1.17% compared to the Multi Modal Fusion algorithm, and the recall rate has increased by 1.18% compared to the Multi Modal Fusion algorithm. In comparison with other indicators, the performance of the improved algorithm is significantly better than that of other algorithm models. To analyze the model validation effectiveness of different indicators, a comparative analysis was conducted on end-to-end delay, decreased anti occlusion accuracy, energy consumption ratio, and F1 values of other datasets, as shown in Table 6.

Table 6
Comparison results of quality indicators.

Metrics	MFEF-Net-DL-STM-GCN	Move Net	Alpha-Pose
Accuracy (%)	93.68	77.35	89.54
End-to-End Latency (ms)	42	68	55
Accuracy Drop under Occlusion (%)	6.2	14.5	9.8
Energy Efficiency (FPS/Watt)	12.3	7.1	9.4
F1 Score on External Dataset (%)	85.3	72.6	80.1

From Table 6, it can be seen that the MFEFNet DLSTM-GCN algorithm used in the study has improved accuracy by 16.33% compared to the MoveNet model. At the same time, the end-to-end latency of the MFEFNet DLSTM-GCN algorithm decreased by 38.2% compared to the MoveNet model. The occlusion accuracy decreased by 57.3% compared to the MoveNet model. Compared to the MoveNet model, the energy consumption has increased by 73.2%. And in external dataset testing, the F1 value increased by 17.5% compared to the MoveNet model. It can be seen that the MFEFNet DLSTM-GCN algorithm also has a good comparative effect in quality index comparison. The sensitivity of the model was analyzed and the results are shown in Table 7.

As can be seen from Table 7, the number of residual block layers and redundancy threshold θ have a significant impact on performance, and it is necessary to balance depth and efficiency. If the number of STM layers exceeds 2, it is easy to cause overfitting. In the selection process, it is necessary to choose a better number of layers for fitting, and the dynamic learning rate attenuation strategy significantly improves the convergence stability of the model.

The study addresses complex situations such as fast continuous movements, occlusion and low-light scenes, and 480 samples of complex scenes are specially classified from 2000 samples in the LSP database, of which 320 are professional competition scenes and 160 are amateur training scenes. Statistical examination of complex scenarios through stratified sampling shows that the confidence interval of MFEFNet's accuracy in complex scenarios is [88.7%, 91.3%] (95% CI), which is significantly different from that in simple scenarios (92.1%, 94.5%) (p=0.003), indicating that environmental complexity has a significant impact on model performance.

The data enhancement technique employs a multi-stage synergistic strategy to improve model robustness. First, random rotations are applied to the input video frames to simulate the athlete's sideways or tilted movements; second, horizontal flipping enhances the model's ability to generalize the left-right symmetric movements; and finally, the lighting fluctuations and sensor noises of the game scene are restored through dynamic brightness adjustment and Gaussian noise injection.

Table 7
Sensitivity analysis results.

Hyperparameter	Parameter Range	Accuracy (%)	GFLOPs	Sensitivity Rating	Optimal Value
Residual Block Layers	2 → 6	91.2 → 93.8	15.3 → 25.3	High	4 layers
LSTM Layers	1 → 3	92.5 → 92.5 (1)	18.9 → 22.1	Medium	2 layers
Dropout Rate θ	0.02 → 0.1	89.3 → 90.1	18.9 → 19.5	High	0.05
Loss Weight α	0.5 → 0.9	91.3 → 90.8	18.9 → 19.2	High	0.7

During the training phase of the DLSTM-GCN-MFEF-Net model, the peak memory for batch 32 hours is 10.2GB, and the inference single frame memory is 1.8GB. The embedded device supports real-time processing at 25 FPS. Maintain a throughput of 18 FPS. Scalability is efficiently deployed through dynamic frame sampling, INT8 quantization, and distributed training. The long-term running test shows that the memory leakage rate is less than 0.3%, and the throughput fluctuation under high load is less than or equal to 4%. The model is both efficient and flexible in edge computing and large-scale video analysis. Therefore, it has good scalability.

4. Discussion and Conclusion

The study achieved significant performance improvement in volleyball action recognition through the improved DLSTM-GCN algorithm and MFEF-Net network. In complex action scenes, the improved DLSTM-GCN algorithm has a recognition accuracy of up to 93.68%, which is 2.00% higher than the unimproved version. In simple action scenarios, the accuracy of the improved algorithm is further improved, significantly better than traditional algorithms. The MFEFNet network achieved an accuracy of 93.16% in video scenes by fusing multi-scale features, which is 5.51% higher than the OpenPose algorithm, and performs well in recall and F1 score. The improved DLSTM-GCN algorithm only takes 2.1 seconds to run, which is 69% shorter than MoveNet and significantly improves real-time performance. The GFLOPs of MFEFNet network are 18.9, much lower than OpenPose's 48.5, indicating its higher computational efficiency. In dynamic and complex real-time competition scenarios, the improved algorithm maintains a recognition accuracy of 90.15% and a recall rate of 89.11%, making it more robust than traditional algorithms. Improved algorithms can detect more key points in complex scenes, while algorithms such as MoveNet suffer from overall performance bias due to missed detection of key points. The system can recognize 8 types of volleyball movements, including serving, blocking, and attacking from the back row, among which the recognition accuracy of the "passing" movement has been significantly improved. Finally, in the comparative analysis of movements between amateur and professional athletes, the system can effectively identify differences in movements and provide scientific feedback for training. It can be seen that different methods can also be used to analyze and predict multiple poses. It can be concluded that the improved DLSTM-GCN algorithm has better volleyball action recognition performance and can significantly improve the recognition effect on volleyball actions. While the study has yielded some findings, there are still some limitations to be addressed. For instance, the study only analyzes the recognition of volleyball, and thus, the recognition effect of other sports requires further investigation in subsequent research. Furthermore, the study only examines the accuracy of the action video and other parameters. Consequently, additional models will be evaluated in subsequent studies to ascertain the recognition performance. The DLSTM-GCN-MFEFNet model performs well in volleyball action recognition, however, the model has limitations in complex scenes - double occlusion leads to lower accuracy and a low detection rate of 18% in low light environments. Although the new algorithm needs to be adjusted for specific scenarios, its spatiotemporal fusion architecture provides a high-precision and lightweight universal solution for multi domain action analysis. In the future, its robustness can be further improved through multimodal sensor fusion and adaptive threshold mechanisms.

References

1. Agostini, F., de Sire, A., Furcas, L., Finamore, N., Farì, G., Giuliani, S., Sveva, V., Bernetti, A., Paoloni, M., Mangone, M. Postural Analysis Using Rasterstereography and Inertial Measurement Units in Volleyball Players: Different Roles as Indicators of Injury Predisposition. Medicina, 2023, 59(12), 2102-2103. https://doi.org/10.3390/medicina59122102
2. Bose, D., Arora, B., Srivastava, A. K., Garg, P. A Computer Vision-Based Framework for Posture Analysis and Performance Prediction in Athletes. In Proceed-

ings of the 2024 International Conference on Communication and Computational Sciences and Engineering (IC3SE), 2024, 23(6), 942-947. https://doi.org/10.1109/IC3SE62002.2024.10593041

3. Cieśluk, K., Sadowska, D., Krzepota, J. The Use of Modern Measuring Devices in the Evaluation of Movement in the Block in Volleyball Depending on the Difficulty of the Task Determined by Light Signals. Applied Sciences, 2023, 13(20), 11462-11463. https://doi.org/10.3390/app132011462
4. Demirel, B., Ozkan, H. Decompl: Decompositional Learning with Attention Pooling for Group Activity Recognition from a Single Volleyball Image. In Proceedings of the 2024 IEEE International Conference on Image Processing (ICIP), 2024, 27(10), 977-983. https://doi.org/10.1109/ICIP51287.2024.10647499
5. Deng, H., Zhang, Z., Li, C., Xu, W., Wang, C., Wang, C. Spatiotemporal Information Complementary Modeling and Group Relationship Reasoning for Group Activity Recognition. The Journal of Supercomputing, 2024, 16(6), 1-21. https://doi.org/10.1007/s11227-024-06288-2
6. Ding, W., Li, W. High Speed and Accuracy of Animation 3D Pose Recognition Based on an Improved Deep Convolution Neural Network. Applied Sciences, 2023, 13(13), 7566-7567. https://doi.org/10.3390/app13137566
7. Geisen, M., Seifriz, F., Fasold, F., Slupczynski, M., Klatt, S. A Novel Approach to Sensor-Based Motion Analysis for Sports: Piloting the Kabsch Algorithm in Volleyball and Handball. IEEE Sensors Journal, 2024, 11(9), 35654-35663. https://doi.org/10.1109/JSEN.2024.3455173
8. Huang, S. A Sparse Representation-Based Local Occlusion Recognition Method for Athlete Expressions. International Journal of Biometrics, 2024, 16(3-4), 287-299. https://doi.org/10.1504/IJBM.2024.138224
9. Jiang, X., Qing, L., Huang, J., Guo, L., Peng, Y. Unveiling Group Activity Recognition: Leveraging Local-Global Context-Aware Graph Reasoning for Enhanced Actor-Scene Interactions. Engineering Applications of Artificial Intelligence, 2024, 133(6), 108412-108413. https://doi.org/10.1016/j.engappai.2024.108412
10. Khan, M. A., Javed, K., Khan, S. A., Saba, T., Habib, U., Khan, J. A., Abbasi, A. A. Human Action Recognition Using Fusion of Multiview and Deep Features: An Application to Video Surveillance. Multimedia Tools and Applications, 2024, 83(5), 14885-14911. https://doi.org/10.1007/s11042-020-08806-9
11. Li, Z., Chang, X., Li, Y., Su, J. Skeleton-Based Group Activity Recognition via Spatial-Temporal Panoramic Graph. arXiv Preprint arXiv:2407.19497, 2024, 2407(6), 2407-2408. https://doi.org/10.48550/arXiv.2407.19497
12. Liu, L., Dai, Y., Liu, Z. Real-Time Pose Estimation and Motion Tracking for Motion Performance Using Deep Learning Models. Journal of Intelligent Systems, 2024, 33(1), 288-289. https://doi.org/10.1515/jisys-2023-0288
13. Liu, Y., Cheng, X., Ikenaga, T. Motion-Aware and Data-Independent Model Based Multi-View 3D Pose Refinement for Volleyball Spike Analysis. Multimedia Tools and Applications, 2024, 83(8), 22995-23018. https://doi.org/10.1007/s11042-023-16369-8
14. Miao, Y., Ge, Y., Wang, D., Mao, D., Song, Q., Wu, R. Effects of Visual Disruption on Static and Dynamic Postural Control in People with and without Chronic Ankle Instability. Frontiers in Bioengineering and Biotechnology, 2024, 12, 1499684-1499685. https://doi.org/10.3389/fbioe.2024.1499684
15. Pang, J. An Early Warning Method for Volleyball Players with Human Bioelectric Energy. Renewable Energy and Power Quality Journal, 2024, 6(6), 111-120. https://doi.org/10.52152/3983
16. Pei, D., Huang, D., Kong, L., Wang, Y. Key Role Guided Transformer for Group Activity Recognition. IEEE Transactions on Circuits and Systems for Video Technology, 2023, 33(12), 7803-7818. https://doi.org/10.1109/TCSVT.2023.3283282
17. Petersen, J. M., Drummond, M., Elliott, S., Drummond, C., Smith, J. A., Wadham, B., Prichard, I. Examining the Promotion of Mental Health and Wellbeing in Australian Sports Clubs. Sport, Education and Society, 2025, 30(6), 742-753. https://doi.org/10.1080/13573322.2024.2351990
18. Phan, L. A., Ngo, H. Q. Application of the Artificial Intelligence Technique to Recognize and Analyze from the Image Data. In Deep Learning and Other Soft Computing Techniques: Biomedical and Related Applications. Cham: Springer Nature Switzerland, 2023, 27(6), 77-89. https://doi.org/10.1007/978-3-031-29447-1_8
19. Rallis, E., Tertipi, N., Sfyri, E., Kefala, V. Prevalence of Skin Injuries in Beach Volleyball Athletes in Greece. Journal of Clinical Medicine, 2024, 13(7), 2115-2116. https://doi.org/10.3390/jcm13072115
20. Ren, L., Wang, Y., Li, K. Real-Time Sports Injury Monitoring System Based on the Deep Learning Algorithm. BMC Medical Imaging, 2024, 24(1), 122-123. https://doi.org/10.1186/s12880-024-01304-6

21. Salian, P., Kulkarni, S. Group Activity Recognition in Visual Data Using Deep Learning Framework. In Proceedings of the 2023 2nd International Conference on Futuristic Technologies (INCOFT), 2023, 22(11), 1-6. https://doi.org/10.1109/INCOFT60753.2023.10425408

22. Salim, F. A., Postma, D. B., Haider, F., Luz, S., Beijnum, B. J., Reidsma, D. Enhancing Volleyball Training: Empowering Athletes and Coaches Through Advanced Sensing and Analysis. Frontiers in Sports and Active Living, 2024, 6(1), 1326807-1326808. https://doi.org/10.3389/fspor.2024.1326807

23. Shang, X., Liu, Y., Cheng, X., Ikenaga, T. Visible Joint Classification and Temporal Segment Matching Based 3D Pose Refinement for Volleyball Receive Analysis. Journal of Physics: Conference Series, 2023, 2522(1), 012017-012018. https://doi.org/10.1088/1742-6596/2522/1/012017

24. Thilakarathne, H., Nibali, A., He, Z., Morgan, S. Group Activity Recognition Using Unreliable Tracked Pose. Neural Computing and Applications, 2024, 5(10), 1-8. https://doi.org/10.1007/s00521-024-10470-1

25. Wang, L., Feng, W., Tian, C., Chen, L., Pei, J. 3D-Unified Spatial-Temporal Graph for Group Activity Recognition. Neurocomputing, 2023, 556(11), 126646-126647. https://doi.org/10.1016/j.neucom.2023.126646

26. Wang, M., Liang, Z. Cross-Modal Self-Attention Mechanism for Controlling Robot Volleyball Motion. Frontiers in Neurorobotics, 2023, 17(10), 1288463-1288464. https://doi.org/10.3389/fnbot.2023.1288463

27. Xu, S. Research on Deep Learning-Based Group Recognition. Highlights in Science, Engineering and Technology, 2023, 72(10), 742-755. https://doi.org/10.54097/55emqr27

28. Yuan, G. Application of Posture Estimation Optimization Algorithm in the Analysis of College Air Volleyball Teaching Movements. Systems and Soft Computing, 2024, 6(10), 200135-200136. https://doi.org/10.1016/j.sasc.2024.200135

29. Zhang, Y., Hou, X. Application of Video Image Processing in Sports Action Recognition Based on Particle Swarm Optimization Algorithm. Preventive Medicine, 2023, 173, 107592-107593. https://doi.org/10.1016/j.ypmed.2023.107592

30. Zhu, S., Luo, J., Du, L. Research on Rapid Reconstruction and Tracking Method of Volleyball Spike Trajectory Based on YOLOv5 Algorithm. In Proceedings of the 2024 International Conference on Machine Intelligence and Digital Applications, 2024, 3(1), 462-468. https://doi.org/10.1145/3662739.3670858