SIN-CHN
SCIENTIFIC PRESS PTE.LTD.

Molecular & Cellular Biomechanics 2024, 21(4), 605.
https://doi.org/10.62617/mcb605

Article

Construction of tennis pose estimation and action recognition model based on improved ST-GCN

Yang Yu

Sports Institute, Baotou Teacher's College, Baotou 014030, China; yyojj070706@163.com

CITATION

Yu Y. Construction of tennis pose estimation and action recognition model based on improved ST-GCN. Molecular & Cellular Biomechanics. 2024; 21(4): 605. https://doi.org/10.62617/mcb605

ARTICLE INFO

Received: 24 October 2024
Accepted: 11 November 2024
Available online: 6 December 2024

Abstract: With the rapid growth of computer vision and deep learning technologies, the application of pose estimation and action recognition in sports training has become increasingly widespread. Due to factors such as complex movements, fast speed, and limb occlusion, pose estimation and action recognition in tennis face significant challenges. Therefore, this study first introduces selective dropout and pyramid region of interest pooling layer strategies in fast region convolutional neural networks. Secondly, a pose estimation algorithm based on multi-scale fusion pose residual network 50 is designed, and finally a spatiotemporal graph convolutional network model is constructed by fusing channel attention module and multi-scale dilated convolution module. The data showed that the average detection accuracy of the improved attitude residual network 50 was 70.4%, and the accuracy of object detection for small, medium, and large objects was 57.4%, 69.3%, and 79.2%, respectively. The continuous action recognition accuracy and inter action fluency detection time of the improved spatiotemporal graph convolutional network were 93.8% and 19.2 ms, respectively. When the sample size was 1000, its memory usage was 1378 MB and the running time was 32.7 ms. Experiments have shown that the improved model achieves high accuracy and robustness in tennis action recognition tasks, especially in complex scenes and limb occlusion conditions, where the model shows significant advantages. This study aims to provide an efficient and accurate motion recognition technology for tennis posture analysis and intelligent training.

Keywords: Spatial Temporal Graph Convolutional Network (ST-GCN); tennis; attitude estimation; action recognition; multi-scale dilated convolution module

1. Introduction

The progression of technology has resulted in a growing necessity for effective and intelligent training tools in the domain of sports [1]. In competitive sports, data-driven technology is gradually being introduced into athletes' daily training to improve their performance. Especially in tennis, the precision and speed variation of movements are crucial for athletes' technical performance [2,3]. To enhance the scientificity of training, Pose Estimation and Action Recognition (PEAR) technology has gradually become a focus of sports science research. These technologies can provide quantifiable motion analysis by capturing athletes' skeletal motion data, thereby providing strong support for athletes' motion optimization and training strategies [4]. In the development of Deep Learning (DL), Graph Convolutional Networks (GCNs) have gradually been introduced into the PEAR field, significantly improving the spatial feature extraction ability of models, especially achieving great success in image classification and Object Detection (OD) [5]. In response to the temporal dynamic characteristics of human movements, the Spatial Temporal Graph Convolutional Network (ST-GCN) has emerged.

ST-GCN combines the temporal and spatial dimensions to effectively capture the spatiotemporal dynamic characteristics of human bones, making it particularly suitable for PEAR in complex motion [6]. In view of the importance of human skeleton dynamics in action recognition, scholars such as S. Yan proposed a new dynamic skeleton model, ST-GCN, to overcome the problems of limited expression ability and difficulty in generalization of traditional skeleton modeling methods. ST-GCN enhances expression and generalization capabilities by automatically learning spatiotemporal patterns from data. Experimental results on two large datasets show that ST-GCN significantly outperforms existing mainstream methods [7]. Scholars such as M. Li proposed an action-structure graph convolutional network to address the problem of ignoring implicit joint correlations in action recognition based on skeletal data. The network combines active and structural links to capture higher-order dependencies and helps capture more detailed action patterns through self-supervision. Experimental results on NTU-RGB+D and Kinetics datasets show that the network shows significant improvements in both action recognition and future pose prediction [8]. Scholars such as Zhu introduced the channel attention module into ST-GCN++ to address the problems of information loss and redundancy in ST-GCN's processing of complex action data. Through experiments on the NTU60 data set, it is proven that this model has achieved the most advanced performance in the current field of action recognition, especially when processing complex data [9]. Keskes and other scholars proposed a visual system based on ST-GCN to solve the problem of insufficient robustness and versatility of methods based on manual features in fall detection. Through experiments on NTU RGB-D, TST fall detection v2 and Fallfree data sets, the efficiency and accuracy of the system were verified, reaching 100% accuracy, surpassing the existing state-of-the-art level [10].

In addition to its application in the field of general action recognition, ST-GCN and its improved versions have also been widely used in more specific scenarios, such as sports action recognition and medical health monitoring, and have demonstrated excellent performance. Tong et al. proposed a basketball pose recognition model based on enhanced GCN and ST-GCN. By combining the advantages of GCN and ST-GCN, graph structured data with time series relationships has been effectively processed. The improved ST-GCN achieved an accuracy of 95.58% in basketball pose recognition [11]. Lovanshi et al. developed a customized ST-GCN model for human activity recognition built on skeleton data. This model effectively utilized the spatial and temporal features in the skeleton data. The customized ST-GCN outperformed existing state-of-the-art methods in Top-1 and Top-5 accuracy across multiple databases [12]. Li et al. proposed a node attention-based ST-GCN model to address the limitation of ST-GCN's inability to learn non-adjacent node relationships in action recognition. The introduction of node attention module explicitly modeled the interdependence of global nodes, thereby effectively improving the recognition performance of actions that require global information [13]. Zhang et al. proposed a new spatial attention and temporal extension model to address the problem of fine-grained information loss in fixed time kernel size and action classification based on the existing ST-GCN model. By using the two GCN modules, data redundancy and noise were effectively reduced, and the robustness to various motion speeds and sequence lengths was improved [14]. Sabo et al. constructed a gait analysis model built on ST-GCN for detecting gait disorders in drug-induced Parkinson's disease. It encouraged the model to learn gait patterns of dementia patients through a self-supervised pre-training phase. The ST-GCN model performed better than traditional regression models and time convolutional networks on the 3D joint trajectories extracted by Kinect [15].

Currently, the field of action recognition has gradually become one of the research hotspots. Action recognition technologies for different scenarios and forms of motion are constantly emerging and showing broad application prospects. Wu et al. established a sports video standard action recognition method that integrates local and global features to address the current situation where existing action recognition algorithms cannot effectively work in sports competitions with high complexity, fine class granularity, and fast action speed. By using spatiotemporal compression and feature fusion algorithms, the underfitting problem of attention mechanism in extracting spatiotemporal features has been overcome [16]. Sun et al. proposed a motion video action recognition method based on Fish Swarm Algorithm (FSA). By improving the FSA, invariant features were constructed and feature dimensionality was reduced, effectively preserving key details of motion actions. This method had a recognition time of less than 326 s for actions such as walking and running, with a recognition rate of over 94% [17]. Ren et al. proposed an action recognition algorithm that combines Precise Time Protocol and CNN (PTP-CNN). In the testing of the human-computer interaction gymnastics action recognition system, the PTP-CNN achieved a recognition accuracy of 96.3%, a recall rate of 95.2%, and a running time of 3.4 s [18]. Barbon Junior et al. proposed a motion action mining framework for recognizing complex human movements, which combines position data and association rule mining to model actions based on human displacement trajectory sequences. The random forest classifier achieved a balanced accuracy of 93.3% in dribbling action recognition [19]. Shan et al. proposed an intelligent action recognition and correction system to address the issue of subjective judgment of action standards by coaches in traditional sports training. The system utilized RGB-D sensors to analyze the key points of athletes' bones in real time, and combined timing tracking algorithms to evaluate the differences between movements and standards [20].

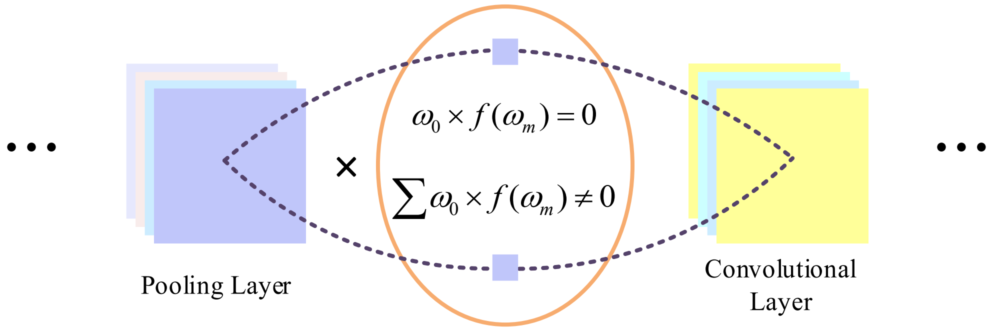
In summary, although many studies in recent years have improved the accuracy of action recognition by introducing ST-GCN and improved skeleton data analysis models, there are still certain limitations when dealing with scenarios such as multi-target, occlusion, and high-speed actions. Therefore, this study constructs a detection and recognition model for tennis events. The innovation lies in first improving the Faster Region-based CNN (Faster R-CNN) as an OD model. Secondly, based on Pose Residual Network 50 (PoseResNet50), a Multi-Scale Fused PoseResNet50 (MF-PoseResNet50) is proposed. Finally, an ST-GCN with Channel Attention and Dilated Convolution (ST-GCN-CAD) tennis PEAR model is constructed. This study aims to enhance the adaptability and recognition accuracy of the model for different actions in complex scenes by fusing the spatiotemporal features of GCN with the skeleton information of graph structures, providing more effective technical support for intelligent training of tennis.

2. Methods and materials

In response to the problems of insufficient accuracy in pose estimation and insufficient extraction of dynamic features in existing tennis action recognition, this study first improves the basic framework of Faster R-CNN and PoseResNet50 from the spatial dimension of images. By introducing multi-scale fusion methods, the accuracy of feature extraction has been optimized. Secondly, by combining the Channel Attention Module (CAM) and the Multi-Scale Dilated Convolution Module (MSDCM), an improved ST-GCN-CAD tennis action recognition model is proposed to enhance the model's ability to extract spatiotemporal dynamic features of action sequences.

2.1. Design of HPE algorithm based on MF-PoseResNet50

Accurately locating the target person is a key step in algorithm design in Human Pose Estimation (HPE) tasks. To improve the accuracy of HPE, this study first uses OD technology to accurately locate the characters in the image before estimation. This study introduces the classic Faster R-CNN model, which can efficiently generate target candidate regions and perform accurate target recognition [21]. However, Faster R-CNN may face issues of multi-target occlusion and decreased performance in detecting fast-moving objects in complex motion scenes [22]. Therefore, this study attempts to introduce selective Dropout and Spatial Pyramid RoI Pooling Layer (SPP) to improve Faster R-CNN. The purpose is to improve its ability to detect fast-moving and dense targets in motion scenes, thereby providing more accurate target areas for subsequent HPE. The schematic diagram of selective Dropout is displayed in **Figure 1**.



$\omega_0 \times f(\omega_m) = 0$
$\sum \omega_0 \times f(\omega_m) \neq 0$
×
Pooling Layer
Convolutional Layer

Figure 1. Selective dropout diagram.

In **Figure 1**, selective Dropout first selects the feature map and weight of a hidden layer in the network as input, and uses the standard Dropout method for the first training. The weights that will be randomly set to zero during this process will be recorded. Subsequently, the feature maps that have been zeroed and those that have not been zeroed in the hidden layer are sent as negative and positive samples, respectively, to the Support Vector Machine (SVM) classifier for training. In the second training, SVM is utilized to determine whether the weight of each node is set to 0, and nodes classified as positive samples increase their probability of being dropped out. In the final training, based on the classification results of SVM, the network further performs selective dropout on each layer node to perfect the training period of the model. Subsequently, the structure of SPP is exhibited in **Figure 2**.

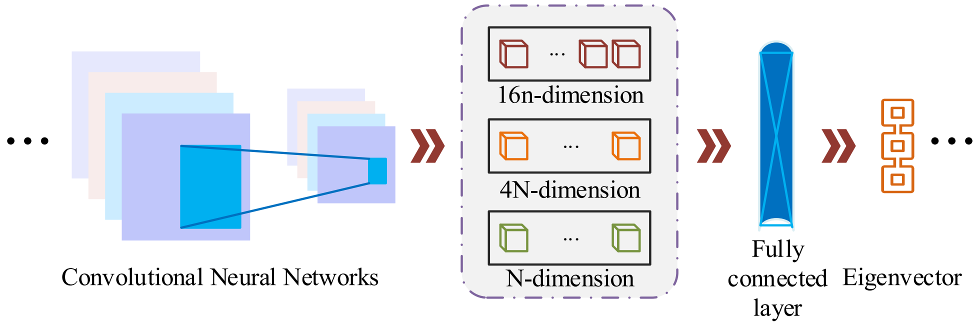


Figure 2. Schematic diagram of the SPP layer.

In **Figure 2**, SPP divides the input feature map into sub-regions of different sizes through multi-scale feature mapping, and performs pooling operations on these regions separately. The figure shows a three-layer pyramid structure, consisting of feature maps with resolutions of 16 times, 4 times, and n times, respectively. Through such multi-scale processing, the SPP layer can capture detailed information of different scales in the input image, thus adapting to changes in object shape and scale. Finally, these multi-scale pooled features are integrated through a fully connected layer to generate feature vectors for subsequent classification. Finally, to improve the generalization capacity and robustness, RReLU is introduced as the activation function. Compared to traditional ReLU, RReLU introduces a random slope on the negative half axis. This randomness can effectively prevent overfitting of the model, while alleviating the problem of neuron death and ensuring that more neurons participate in learning. In addition, RReLU performs better in handling noise and uncertainty, making it suitable for tennis motion detection in dynamic scenes. Its expression is shown in Equation (1).

$$f(x)=\begin{cases} x, & if \quad x>0 \\ a_i x, & if \quad x\le 0 \end{cases} \tag{1}$$

In Equation (1), x is the input feature value or the activation value of the neuron. a_i is a slope parameter randomly sampled from a uniform distribution $a_i \sim U(l,u)$ during training. l and u are preset upper and lower limits used to control the range of negative slope. After building an OD model grounded on improved Faster R-CNN, this study will attempt to construct the HPE algorithm for PoseResNet50. PoseResNet50 is based on the deep structure of residual networks, which can effectively extract human skeletal features and has strong feature expression ability [23,24]. Firstly, the constructed 2D pose point annotation is shown in **Figure 3**.

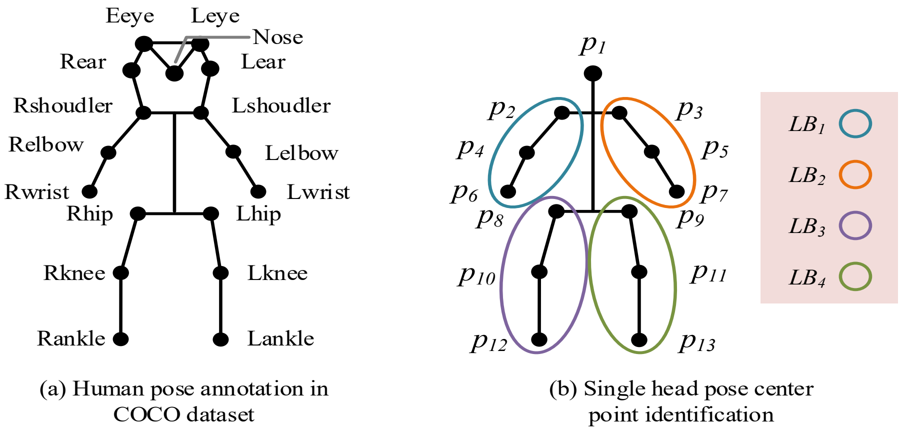


Figure 3. 2D human pose annotation diagram.

Figure 3a shows the multi-dimensional facial feature points in the traditional COCO annotation system, including nasal tip, eyes, and ears. **Figure 3b** shows a complex scenario for open sports. This study uses a single head posture center point identification to represent the head posture by calculating the geometric center coordinates of key areas of the head, to streamline the annotation process and improve operational efficiency. A whole-body skeletal dataset containing 13 core keypoints is constructed through precise recognition using HPE algorithm. The optimized data coordinate expression is shown in Equation (2).

$$P_i(t)=\left\{(x_j(t),y_j(t)),c_j\right\}_{j\in J}\qquad t\in\left\{1,\ldots,T\right\}\tag{2}$$

In Equation (2), $P_i(t)$ is the position of the j-th keypoint at time t. $x_j(t)$ and $y_j(t)$ are the horizontal and vertical coordinates of the j-th key point at time t. c_j is the confidence level of the j-th key point. In $j\in J$, J is the set of key points, indicating that there are a total of J key points that need to be annotated. $t\in\left\{1,\ldots,T\right\}$ is a different time frame in the time series, with a total of T time frames. The different sets of limb points contained in each of the four limb blocks in **Figure 3b** are shown in Equation (3).

$$\begin{cases} LB_1=\left\{p_2,p_4,p_6\right\} \\ LB_2=\left\{p_3,p_5,p_7\right\} \\ LB_3=\left\{p_8,p_{10},p_{12}\right\} \\ LB_4=\left\{p_9,p_{11},p_{13}\right\} \end{cases}\tag{3}$$

In Equation (3), LB_1, LB_2, LB_3, and LB_3 are the upper Left and Right (L&R), lower L&R limb masses. p_2 to p_{13} correspond to the L&R shoulders, elbows, wrists, pelvic bones, knees, and ankles in **Figure 3b**, respectively. Subsequently, in the pose estimation step, the deep residual network structure of PoseResNet50 can effectively handle complex image feature extraction tasks. Through multi-level convolution operations, shallow and deep features of the target character can be captured, ensuring that semantic information at different levels is fully expressed, thereby enhancing the ability to capture pose details [25,26]. In addition, the residual structure of PoseResNet50 helps to solve the common gradient vanishing problem in deep networks, making the network more stable during training and able to learn complex action patterns more efficiently [27,28]. Meanwhile, due to the complex actions and varying sizes of limb parts involved in pose estimation, this study introduces ResNet50 and improves the backbone network of PoseResNet50 using multi-scale feature fusion methods. By integrating feature information from both shallow and deep layers through multi-scale feature fusion, the model can not only capture the global skeletal structure but also focus on local detailed features. Therefore, the flowchart of the improved backbone network of MF PoseResNet50 is shown in **Figure 4**.

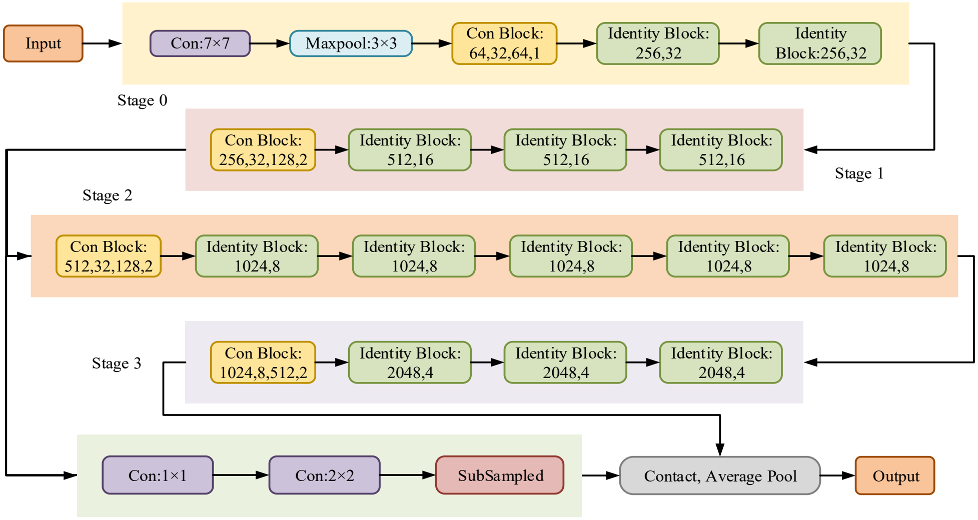


Figure 4. Introducing the PoseResNet50 backbone network structure of ResNet50.

In **Figure 4**, MF-PoseResNet50 consists of four stages, each consisting of a Conv Block and an Identity Block. In Stage 0, the input image undergoes convolution and max pooling operations, and the output feature map size is (64, 32, 32), which means 64 channels and a spatial size of 32 × 32. In Stage 1, the convblock extends the features to (256, 32, 32) through three convolution operations and maintains that dimension unchanged through identity blocks. Subsequently, Stage 2 further reduces the feature map to (512, 16, 16) through conv blocks, while maintaining the feature size through identity blocks. In Stage 3, the conv blocks continue to expand the features to (1024, 8, 8) and maintain the dimensionality of the feature map through identity blocks. Finally, through the global average pooling layer, the spatial size is reduced to 1 × 1 to obtain a fixed size output for subsequent pose estimation tasks. The entire network maintains effective gradient transfer through residual connections and integrates multi-level features at different scales, making it perform excellently in complex pose estimation tasks.

Subsequently, in response to the possible differences in human posture caused by occlusion and high-speed changes in tennis, a posture correction module is introduced after extracting features to correct the offset posture. This module first defines a function that calculates the average position of low confidence attitude points in the current frame by combining the attitude points of the previous frame, current frame, and next frame. Secondly, it adjusts the position of low confidence points by offsetting the position of high confidence pose points. The correction process generates the final correction point position by superimposing offset and weight. Subsequently, the algorithm traverses the pose sequence of each frame, determines whether there are low confidence pose points in the current frame, and obtains corresponding pose points in adjacent frames based on the contextual relationships of these points. Finally, the corrected pose point sequence is updated to the pose set.

Finally, since MF-PoseResNet50 is based on a convolutional neural network model, it has a large number of parameters. In order to further optimize the computational efficiency of the model and improve its practicality in environments with limited computing resources, the study introduced a model compression technology called weight pruning. The goal of weight pruning is to reduce the memory usage and computational complexity of the model by removing redundant weight parameters. It mainly includes weight importance evaluation, pruning strategy, and post-pruning training steps. First, the importance of each weight is evaluated by calculating its absolute value. Smaller weights are considered to have less impact on the model and can be removed. The weight importance evaluation Equation is shown in Equation (4).

$$\left|w_{ij}\right| = importance(w_{ij}) \tag{4}$$

In Equation (4), w_{ij} is the j-th weight in the i-th layer, and $importance(w_{ij})$ represents the absolute value of the weight. The importance of the weight is evaluated by calculating the absolute value of the weight. Smaller weights are considered to have less impact on the overall model and are removed first to achieve the goal of pruning.

Subsequently, in the pruning strategy, a pruning ratio of $p\%$ is set, that is, the least important $p\%$ weights in the model are removed. After all weights are sorted by their importance, these least important weights are removed. The removed weights are set to zero. The pruned model can be retrained through fine-tuning to restore some of the lost accuracy. The fine-tuning stage readjusts the remaining weights through back propagation to make the model adapt to the pruned structure again and minimize the performance degradation caused by pruning. The study adopts a global pruning strategy to remove the smaller important weights in the model, and the pruning ratio is set to 30%. Finally, the pruned model is trained through further fine-tuning to ensure that the performance is maintained at a high level. The loss function of fine-tuning is the same as that of the original training. The weight update Equation after pruning is shown in Equation (5).

$$w_{ij} \leftarrow w_{ij} - \eta \frac{\partial L}{\partial w_{ij}} \tag{5}$$

In Equation (5), η represents the learning rate and L represents the loss function. This pruning strategy aims to effectively reduce the computational complexity of the model while maintaining high accuracy. The pruned model is not only suitable for environments with abundant computing resources, but also performs well on devices with limited resources.

2.2. Tennis movement recognition algorithm based on ST-GCN-CAD

After completing the HPE algorithm design based on the improved PoseResNet50, this study further focuses on how to accurately identify and analyze complex tennis movements in dynamic motion scenes. Attitude estimation provides precise localization of skeletal points, but action recognition requires a deeper understanding of the dynamic changes of these skeletal points in the spatiotemporal dimension. Therefore, this study introduces ST-GCN. This model is capable of capturing both spatial and temporal features during motion, making it particularly suitable for fast and complex movements like tennis. The advantage of ST-GCN lies in its capacity to model the spatiotemporal relationships between human keypoints through graph convolution, effectively improving the accuracy and robustness of action recognition [29]. Therefore, choosing ST-GCN as the core algorithm for tennis action recognition can not only supplement the shortcomings of pose estimation, but also better handle complex motion patterns in dynamic scenes [30].

In the ST-GCN model, the input data is batch normalized to improve the training stability. Next, the data sequentially enter multiple ST-GC units. Each unit consists of three parts: Attention Module (ATT), GCN, and Temporal Convolutional Network (TCN) [31]. GCN is responsible for extracting spatial features, while TCN extracts time series features, and ATT is used to enhance the model’s attention to important information. Stacking multiple ST-GC units to capture complex spatiotemporal information at a deeper level. Subsequently, the features are dimensionality reduced through a Pooling layer (POOL) and finally passed into a Fully Connected layer (FC) for classification or prediction output. The entire structure achieves efficient analysis and recognition of input data by combining spatial and temporal features as well as attention mechanisms.Due to the fixed size of the time GCN kernel used in traditional ST-GCN models, this limits the model’s ability to capture temporal features, resulting in poor performance in processing complex dynamic data [32,33]. Therefore, this study attempts to introduce CAM and MSDCM to enhance the representation of temporal features. The design and function of these two key components are described in detail below.

First, CAM enhances feature representation by adjusting the importance of each channel [34]. In terms of specific design, it captures the global context information of the feature map through global average pooling and global maximum pooling, then generates weights through two fully connected layers and weights the input feature map. This can improve the network’s attention on important features, thereby improving the accuracy and robustness of action recognition. The expected effect is to improve the model’s ability to capture key information by optimizing the weight of channel features, especially in multi-objective or complex backgrounds, which can effectively enhance the model’s expressive ability. Second, MSDCM extracts multi-scale information through convolution with different expansion rates to expand the receptive field of the convolution kernel while maintaining a small amount of calculation. The design of this module can capture action details at different scales, and is particularly effective in identifying key frames or dynamic actions in long-term sequences. The expected effect is to enhance the generalization ability of the model on different time scales and improve the recognition accuracy of complex actions or rapidly changing postures. In short, CAM can better focus on key features by dynamically adjusting the weights of different channels. MSDCM enhances the model's ability to capture temporal features by introducing convolution kernels of different scales, enabling the model to more effectively handle multi-scale temporal information. Firstly, the introduction of FC and Batch Normalization (BN) CAM is shown in **Figure 5**.

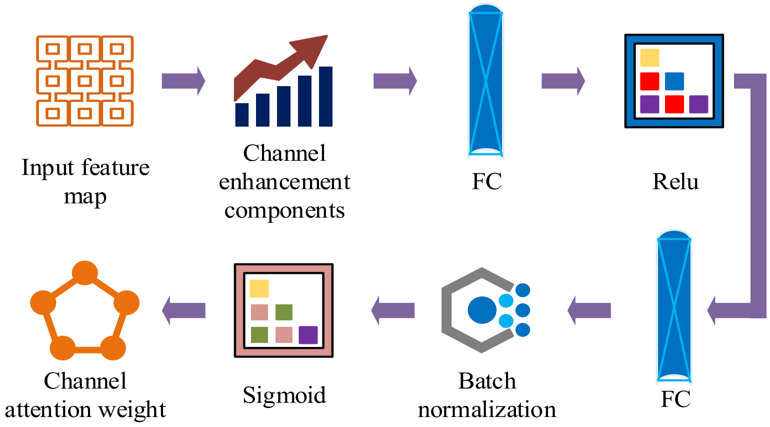


Figure 5. Schematic diagram of the improved CAM.

In **Figure 5**, in the improved CAM, after the input features are processed by the channel enhancement component, they are first reduced in dimensionality through FC, and then the non-linear expression ability of the features is improved through the ReLU function. Next, the features are restored to their dimensions through FC and processed through BN to ensure their balance across different channels. Finally, after Sigmoid, channel attention weights are generated to enhance the representation ability of important features. This process aims to enhance the model's attention to key features and further improve its performance by adaptively adjusting the weights of each channel. The expression of the output feature map of the channel enhancement component is shown in Equation (6).

$$Z = CAM(X) \tag{6}$$

In Equation (6), Z is the feature map after channel enhancement. Input data X is the input feature matrix of a batch. Subsequently, the attention weights of the channels are calculated through two layers of FC to control the importance of each channel, as shown in Equation (7).

$$ATT_c = Sigmoid(W_2 \times Relu(W_1 Z)) \tag{7}$$

In Equation (7), ATT_c is the final channel attention weight used to represent the importance of each channel. W_1 and W_2 are the weight matrices of two layers of FC, which respectively control the dimensionality reduction and enhancement of features, adjust the dimensionality of input features, and capture the interrelationships between channels. $Sigmoid$ and $Relu$ are both activation functions used to increase the non-linear expression ability and control the importance of each channel. Subsequently, in MSDCM, dilated convolution introduces intervals in the convolution operation to expand the receptive field, which can capture more global information without increasing computational complexity.

Its structure is shown in **Figure 6**.

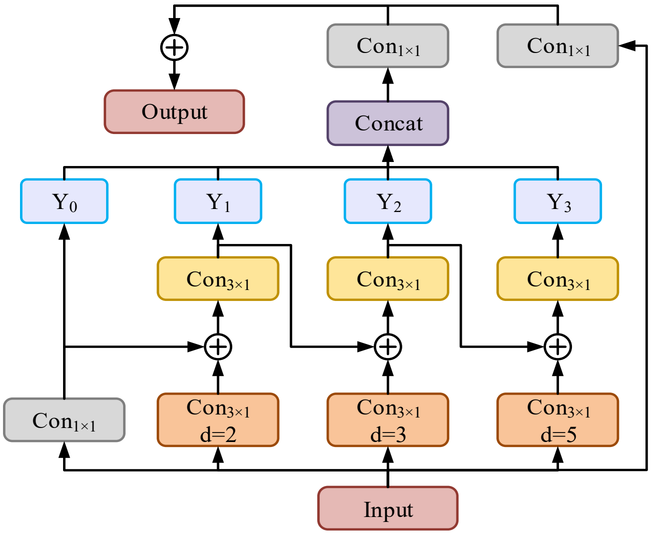


Figure 6. MSDCM structure.

In **Figure 6**, firstly, the input features are processed through dilated convolutions of various scales, i.e., dilated convolutions with kernel sizes of 3 × 1 and dilation rates of 2, 3, and 5, respectively, to capture features in different time ranges. Each dilated convolution operation generates a feature map Y_1, Y_2, Y_3. These features are further extracted through subsequent 3 × 1 convolutions, and finally all features are fused through feature concatenation. On the fused features, information is integrated through a 1 × 1 convolution operation to generate output features. Throughout the process, different scales of dilated convolutions help the model better capture feature changes at different time scales, enhance the model's perception of temporal information, and improve performance in time series analysis. Therefore, the Equation for extracting time equidistant feature information through dilated convolution operations with different dilation rates is shown in Equation (8).

$$x_i = C_d(X^{'}), \qquad d = 2,3,4 \quad i = 1,2,3 \tag{8}$$

In Equation (8), x_i is the feature information obtained through dilated convolutions with different dilation rates. C_d is a dilated convolution operation with an expansion rate of d. $X^{'}$ is the input feature information, with a shape of $[N, C, T, V]$. N is the batch size, C is the amount of channels, T is the time step, and V is the number of human skeletal nodes. To better capture the dependencies between local times in action sequences, this study uses a set of time convolutions to capture the correlations between different time steps, as shown in Equation (9).

$$Y_i = \begin{cases} Con_{1\times1}(X^{'}), & if \quad i = 0 \\ T_{G_i}(x_i + Y_{i-1}), & if \quad i > 0 \end{cases} \tag{9}$$

In Equation (9), Y_i and Y_{i-1} are the output features of the current and previous time steps. $Con_{1\times1}$ is a 1 × 1 convolution operation used to extract the basic features of the input at the initial layer. T_{G_i} is a time convolution operation with dilation rate used to extract temporal dependencies. Each branch aggregates contextual information from adjacent time frames through different convolution operations. Finally, the outputs of these branches are fused to form the overall temporal characteristics. The output result is shown in Equation (10).

$$\begin{cases} Z^{'} = Cat[Y_0, \dots, Y_s] \\ Output = Con_{1\times1}(Z^{'}) + Con_{1\times1}(X^{'}) \end{cases} \tag{10}$$

In Equation (10), $Z^{'}$ is the feature matrix formed by concatenating the outputs $Y_0, \dots, Y_s$ of each time convolution branch through a concatenation operation (Cat). $Con_{1\times1}(Z^{'})$ performs a 1 × 1 convolution operation on the concatenated feature $Z^{'}$ to integrate multi-scale feature information. $Con_{1\times1}(X^{'})$ performs a 1×1 convolution on input $X^{'}$ as another part of the residual connection to ensure the integrity of information transmission. $Output$ is the final output feature, which includes multi-scale temporal features and input residual information. Finally, based on the CAM and MSDCM modules mentioned above, the improved ST-GCN-CAD model structure is shown in **Figure 7**.

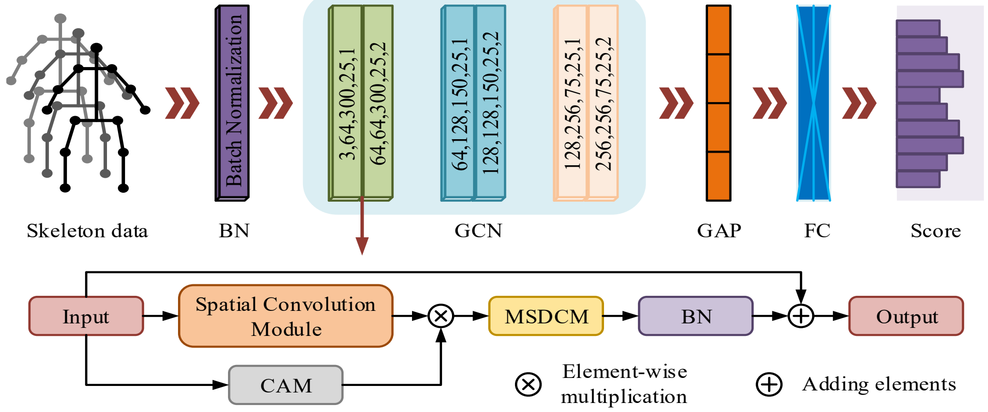


Figure 7. ST-GCN-CAD module structure diagram.

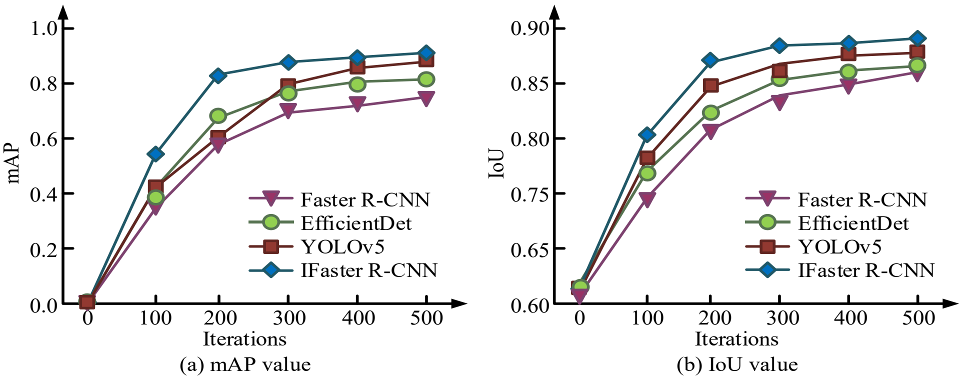
In **Figure 7**, firstly, the input skeleton data is processed and standardized by BN to improve the training stability of the network. Next, the data is sequentially processed through multiple layers of GCN. Each layer of GCN processes spatial features, extracts relationships and dynamic information between joint points. The network extracts features at different levels through three layers of GCN modules. Each layer corresponds to a different number of channels and time steps. After extracting preliminary features, a Global Average Pooling (GAP) layer was added and classified through FC, ultimately outputting action classification scores. In the feature extraction process, the model introduces spatial convolution module and CAM. This section captures spatial features through spatial convolution, combines MSDCM to enhance the perception ability of temporal features, and ultimately uses channel attention mechanism to further improve the weight of key features and enhance the discriminative ability. Through these improvements, the model can not only capture richer spatiotemporal information, but also greatly enhance the accuracy of complex action recognition.

3. Results

To verify the performance of the proposed tennis PEAR model based on improved ST-GCN-CAD, an experimental environment suitable for tennis sports scenes is first established, and the collected tennis sports data is preprocessed. Part of the data are used for model training. Secondly, this study comprehensively testes the recognition accuracy and robustness of the model through ablation experiments, comparison of similar models, and multi-index testing. Subsequently, simulation tests are conducted based on real tennis match scenes to verify the performance and application effects of the model in actual sports scenarios.

3.1. Performance testing of MF-PoseResNet50 attitude estimation algorithm

To verify the performance and effectiveness of the IFaster R-CNN OD algorithm and the MF PoseResNet50 attitude estimation algorithm, a suitable experimental environment is established in this study. The operating system is Windows 10, the CPU is Intel Core i7, and the base frequency is 4.2 Hz. The GPU is NVIDIA GeForce RTX 1660s, with 16 GB of video memory and 16 GB of memory. The dataset used is PASCAL VOC, which contains approximately 17,000 images with OD annotations. Firstly, in IFaster R-CNN, the learning rate is set to 0.001 and Cosine Annealing is used for dynamic adjustment, with a batch size of 16. The optimizer chooses AdamW to balance weight decay. In addition, the NMS threshold for non-maximum suppression is 0.5, and the scale range for Anchor generation is 32 to 512. The experiment uses a multi-scale training strategy to improve the detection performance of the model on targets of different sizes. To verify the effectiveness of the improvements made to IFaster R-CNN, its Mean Average Precision (mAP) and Intersection over Union (IoU) are compared with Faster R-CNN, Efficient Detection (EfficientDet), and YOLOv5 on the PASCAL VOC dataset. The test results are displayed in **Figure 8**.



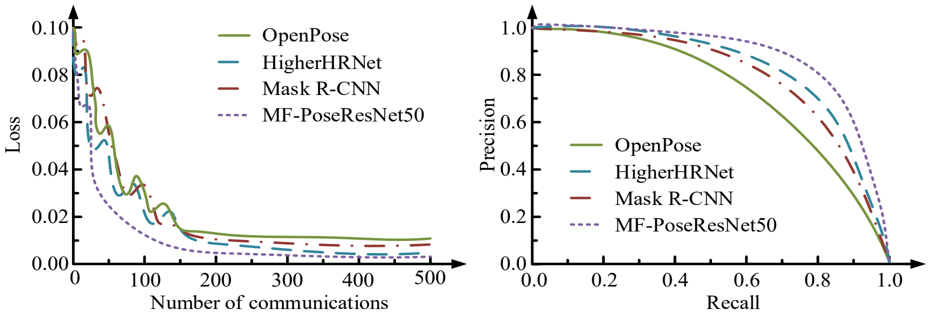
(a) mAP value

(b) IoU value

Figure 8. The mAP and IoU test results for different models.

Figure 8 shows the mAP and IoU test results of Faster R-CNN, EfficientDet, YOLOv5, and IFaster R-CNN as a function of iteration times. In **Figure 8a**, when the iteration reaches 500 times, the mAP values of the four models are 0.76, 0.81, 0.87, and 0.91. In the early stages of iteration, IFaster R-CNN quickly outperforms the other three models, with its mAP value increasing from 0.5 to nearly 0.75, enabling faster capture of target features. In **Figure 8b**, when the iteration reaches 500 times, the IoU values of the four models are 0.86, 0.87, 0.88, and 0.89. IFaster

R-CNN has the optimal mAP and IoU values, demonstrating the effectiveness of Faster R-CNN in introducing selective Dropout and SPP strategies. This fusion makes it more stable and accurate when dealing with complex backgrounds and small targets. Subsequently, this study utilizes the COCO dataset, which contains over 250,000 annotated keypoints and over 80,000 images. The experiment introduces OpenPose, Higher Resolution Network (HigherHRNet), Mask Region-based CNN (Mask R-CNN) models, and compares them with MF-PoseResNet50. The loss curves and Precision-Recall (PR) curves of the four models as they vary with the number of iterations are shown in **Figure 9**.

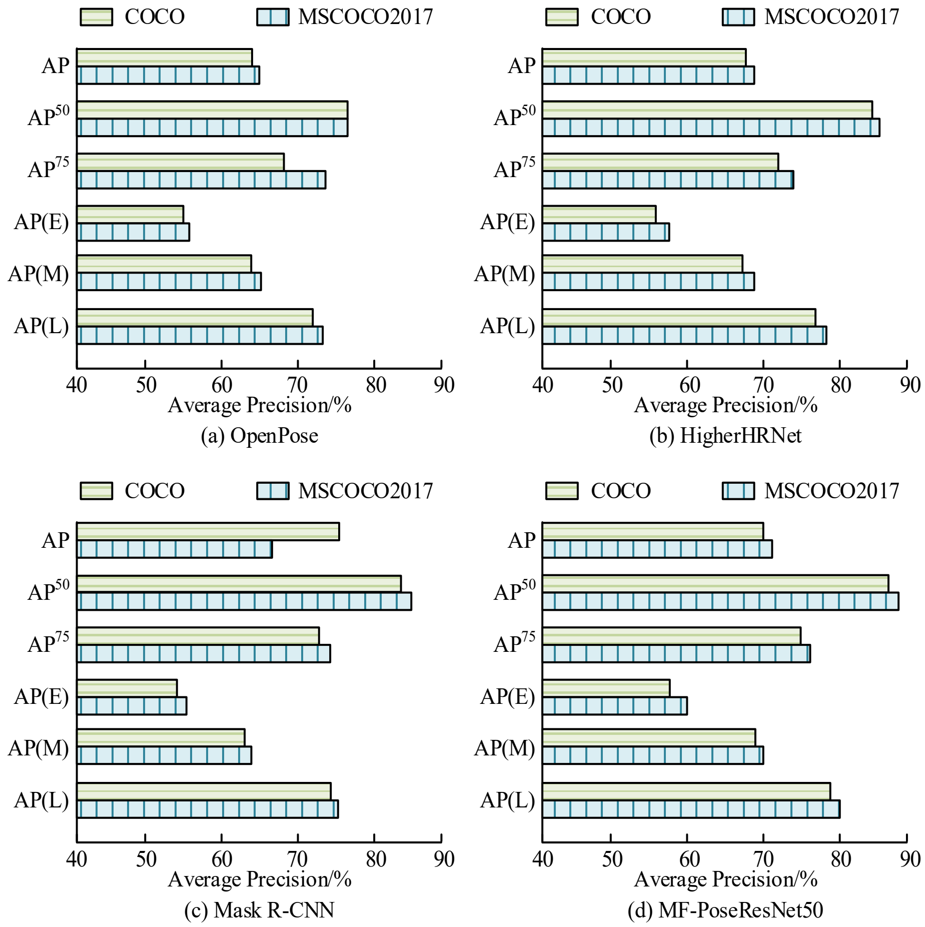


(a) Loss curve test

(b) PR curve test

Figure 9. Test results of loss curve and PR curve.

In **Figure 9a**, when iterating 500 times, the loss values of OpenPose, HigherHRNet, Mask R-CNN, and MF-PoseResNet50 are 0.011, 0.006, 0.009, and 0.003. In contrast, although HigherHRNet and Mask R-CNN perform better in the early iteration stage, their relatively complex model structures result in slower convergence speed and higher loss values at higher iteration times. OpenPose, due to its bottom-up nature, struggles to capture high-resolution details, resulting in relatively high loss values throughout the entire process. In **Figure 9b**, the area enclosed by the PR curve and the coordinate axis is the Area Under the Precision Recall Curve (AUC). As the AUC value increases, the accuracy and recall of the model at varying thresholds also rises, indicating that the model has good comprehensive performance. Therefore, the larger the area of the PR curve, the better the detection performance of the attitude estimation algorithm, which can more accurately and completely capture and identify key attitude points. Therefore, the curve of MF-PoseResNet50 completely surrounds other curves and has the highest AUC value. This study introduces the MSCOCO2017 dataset, which contains approximately 67,000 images with human keypoint annotation results, to test the Average detection Precision (AP) of each model on two datasets, as shown in **Figure 10**.



(a) OpenPose

(b) HigherHRNet

(c) Mask R-CNN

(d) MF-PoseResNet50

Figure 10. AP test results of each model.

Figure 10a–d show the AP values of four models on the COCO and MSCOCO2017 datasets. Among them, AP50 and AP75 are APs detected with IoU thresholds of 0.50 and 0.75. AP (E), AP (M), and AP (L) are APs used for detecting simple samples, medium-sized targets, and large targets. The above different AP indicators can comprehensively evaluate the detection performance of the model under different IoU thresholds and target sizes. In **Figure 10d**, the MF-PoseResNet50 model is applied on COCO, AP = 70.4%, AP50 = 88.5%, AP75 = 75.6%. In addition, the model has an AP (E) of 57.4% for small objects, 69.3% for medium objects, and 79.2% for large objects on objects of different sizes. At MSCOCO2017, the performance of the MF-PoseResNet50 model slightly improves, AP = 71.3%, AP50 = 89.1%, and AP75 = 76.2%, AP (E) = 58.2%, AP (M) = 70.1%, and AP (L) = 80.1%. This indicates that the MF-PoseResNet50 model exhibits high accuracy on different datasets, especially in the detection of larger object AP (L), where the performance advantage is more pronounced. Compared to other models, MF-PoseResNet50 has shown better accuracy and stability in all indicators. Finally, to further validate the comprehensive performance of the models, **Table 1** shows the comparative results of each model at different resolutions.

Table 1. Experimental results at different resolutions.

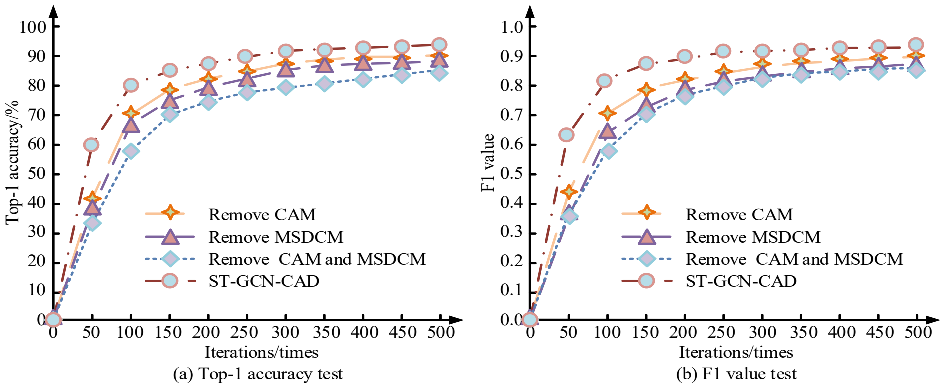
Model Name	Resolution/px	FPS/Frames per second	Average inference time/ms	Memory usage/MB
	256 × 256	44.7	22.4	948
MF-PoseResNet50	512 × 512	31.5	31.7	1248
	1024 × 1024	18.3	55.9	2113
	256 × 256	29.8	33.2	1214
OpenPose	512 × 512	20.5	50.2	1723
	1024 × 1024	12.2	83.5	2802
	256 × 256	37.9	26.1	1145
HigherHRNet	512 × 512	28.3	35.9	1523
	1024 × 1024	16.5	62.7	2417
	256 × 256	21.7	45.7	1325
Mask R-CNN	512 × 512	14.1	71.6	1934
	1024 × 1024	8.3	125.3	3223

Table 1 shows the Frames Per Second (FPS), inference time, and memory usage of each model at different resolutions. At a low resolution of 256 × 256, MF-PoseResNet50 performs the best with 44.7 FPS, inference time of 22.4 ms, and memory usage of only 948 MB, indicating that the lightweight design of this model significantly improves computational efficiency. At a high resolution of 1024 × 1024, although the FPS and inference time of all models have significantly decreased, MF-PoseResNet50 still maintains a high FPS of 18.3 FPS. Compared to other models, its inference time and memory consumption are still at a relatively low level. In contrast, Mask R-CNN achieves inference time of 125.3 ms at high resolution and significantly increases memory usage. This indicates that MF-PoseResNet50 can effectively control computational overhead while ensuring high accuracy, and its optimized multi-scale fusion strategy is the key to improving efficiency.

3.2. Experimental analysis of ST-GCN-CAD action recognition model

After verifying the application effectiveness of IFaster R-CNN and MF-PoseResNet50, this study conducts performance testing on the ST-GCN-CAD model. The NTU-RGB+D dataset is used, which contains over 56,000 video segments covering 60 different actions. Due to the fact that ST-GCN-CAD is composed of CAM and MSDCM, to verify the effectiveness, the ablation experiment results are shown in **Figure 11**.

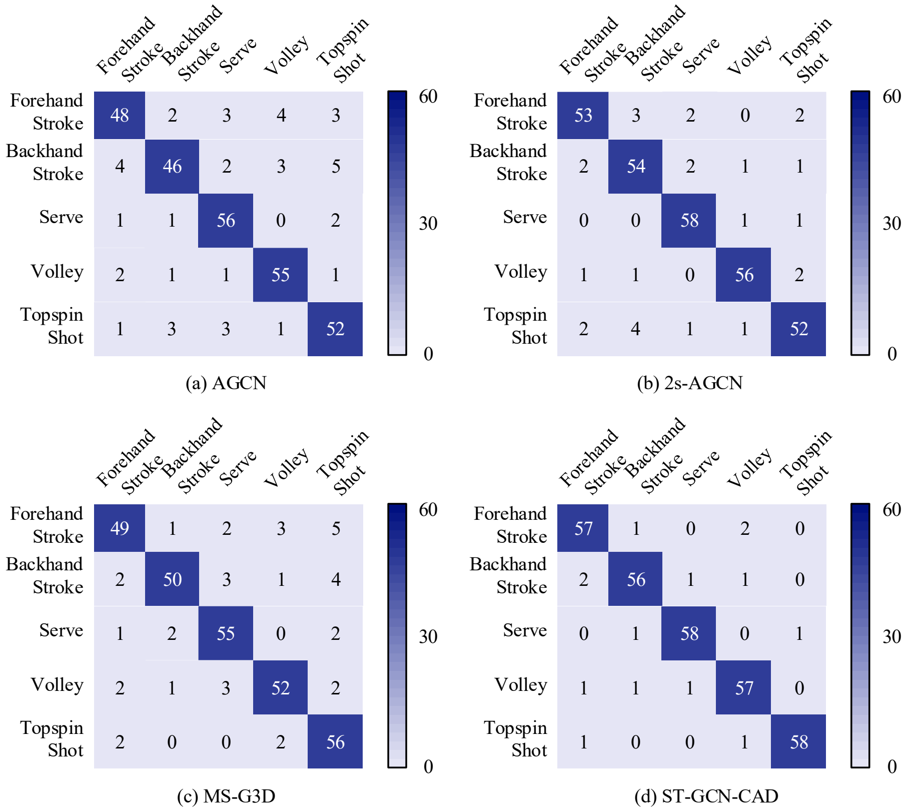
Figure 11a,b show the ablation test results for Top-1 accuracy and F1 Score with CAM and MSDCM removed, both modules removed simultaneously, and the complete model removed. In the Top-1 accuracy test of **Figure 11a**, the complete ST-GCN-CAD exhibits optimal accuracy as the number of iterations increases. When iterates 500 times, the detection accuracies for the four situations are 90.1%, 88.5%, 75.3%, and 94.3%, respectively. The F1 test results in **Figure 11b** also show a similar trend, with F1 values of 0.90, 0.87, 0.86, and 0.93 for the four models after 500 iterations. The main reason for this difference is that CAM effectively enhances the channel selection of spatiotemporal features, improving the accuracy of feature expression, while MSDCM enhances the ability to capture spatiotemporal information at different scales, improving the performance of action recognition. It also confirms the effectiveness of this study in improving ST-GCN-CAD. Each module has played a certain positive role in the final model. Subsequently, Adaptive GGCN (AGCN), Two-Stream AGCN (2s-AGCN), and Multi-Scale Graph Temporal Convolutional Networks (MS-G3D) are introduced as comparative models. The experiment selects the TSR tennis action dataset, which includes various technical actions such as forehand, backhand, and serve, and provides relevant video data. This study selects five representative tennis movements: forehand stroke, backhand stroke, serve, intercept, and topspin. The experimental results of the confusion matrix for the four models are shown in **Figure 12**.



(a) Top-1 accuracy test

(b) F1 value test

Figure 11. Ablation test results.



(a) AGCN

	Forehand Stroke	Backhand Stroke	Serve	Volley	Topspin Shot
Forehand Stroke	48	2	3	4	3
Backhand Stroke	4	46	2	3	5
Serve	1	1	56	0	2
Volley	2	1	1	55	1
Topspin Shot	1	3	3	1	52

(b) 2s-AGCN

	Forehand Stroke	Backhand Stroke	Serve	Volley	Topspin Shot
Forehand Stroke	53	3	2	0	2
Backhand Stroke	2	54	2	1	1
Serve	0	0	58	1	1
Volley	1	1	0	56	2
Topspin Shot	2	4	1	1	52

(c) MS-G3D

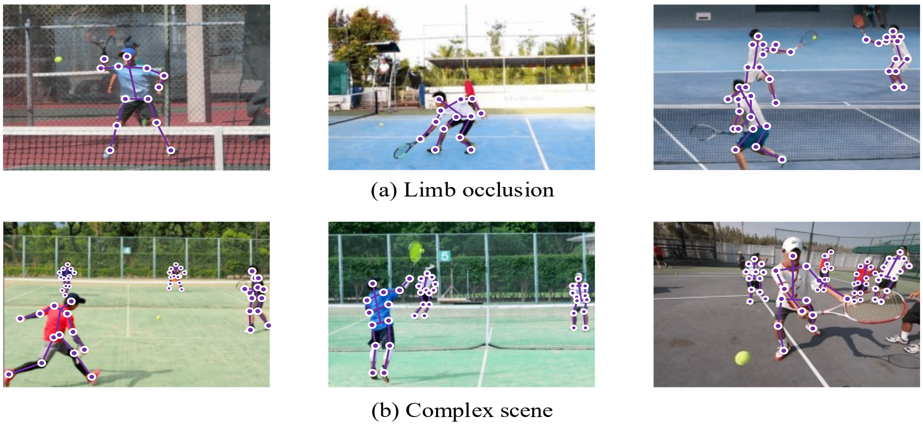
	Forehand Stroke	Backhand Stroke	Serve	Volley	Topspin Shot
Forehand Stroke	49	1	2	3	5
Backhand Stroke	2	50	3	1	4
Serve	1	2	55	0	2
Volley	2	1	3	52	2
Topspin Shot	2	0	0	2	56

(d) ST-GCN-CAD

	Forehand Stroke	Backhand Stroke	Serve	Volley	Topspin Shot
Forehand Stroke	57	1	0	2	0
Backhand Stroke	2	56	1	1	0
Serve	0	1	58	0	1
Volley	1	1	1	57	0
Topspin Shot	1	0	0	1	58

Figure 12. Confusion matrix test results of each model.

Figure 12a–d show the confusion matrices of AGCN, 2s-AGCN, MS-G3D, and ST-GCN-CAD for recognizing different tennis movements. The recognition accuracy of ST-GCN-CAD is significantly higher in five representative tennis movements. It can accurately distinguish subtle differences in movements, such as significantly reducing the confusion rate between forehand and backhand shots, and further improving the accuracy of interception and topspin movements. In **Figure 12a**, AGCN is prone to confusion when recognizing similar actions such as forehand and backhand shots due to the lack of precise capture of action details. In **Figure 12b**, 2s-AGCN improves its ability to extract spatiotemporal information from action sequences by introducing a dual stream architecture, but there is still some error in complex actions such as interception. In **Figure 12c**, MS-G3D improves the capture of global motion features by introducing multi-scale graph convolution, but is relatively inadequate in handling local motion details. In **Figure 12d**, ST-GCN-CAD combined with CAM and MSDCM enhances the ability to capture spatiotemporal details, resulting in more stable and accurate performance in various tennis movements. To further verify the model's performance in dealing with complex scenes and body occlusion, the study selected tennis action images with complex scenes such as body occlusion and multi-person interaction in the dataset and performed posture labeling analysis. The model's robustness and accuracy in dealing with partial body occlusion and complex background interference are demonstrated through visualization. The experimental results are shown in **Figure 13** below.


(a) Limb occlusion

(b) Complex scene

Figure 13. Qualitative analysis of experimental results.

Figure 13a,b are the experimental results of posture estimation in the presence of limb occlusion and complex background. In **Figure 13a**, the model can still accurately locate key points when dealing with limb occlusion, especially when the limbs are partially occluded by the net or other objects, the model can still restore relatively complete posture information, indicating that it has a strong recognition ability for incompletely visible actions. **Figure 13b** shows that in the scene of multi-person interaction and complex background, the model can effectively distinguish the actions of different athletes, and maintain a high posture recognition accuracy under the condition of lighting changes and background interference. The experimental results verify the robustness of the model in actual complex scenes, and further demonstrate its application potential in multi-target and multi-athlete scenes.

Subsequently, the study used a self-built data set for further simulation tests. First, the data set contains no less than 10 different tennis actions, including common actions such as serving, receiving, forehand, backhand, volley, and so on. These actions are accurately annotated to ensure that the starting and ending points of each action are consistent, and the number of samples of each action is evenly distributed. Specifically, serving and receiving each contain about 1500 samples, forehand and backhand shots each contain 1200 samples, volley and volley actions each contain 1000 samples, and other special actions (such as difficult volleys) have 800 samples each. Data annotation uses a combination of manual and automatic annotation, of which 70% of the data is annotated by an automated tool based on motion capture technology, and 30% is reviewed by an expert team to ensure the consistency and accuracy of the annotation. In addition, the data set includes athletes from different industries and experience levels. Participants include 50 athletes, covering different groups ranging in age from 18 to 40 years old, of which 30% are professional players and 70% are amateurs. In terms of gender, the dataset balances the ratio of male and female athletes, and each gender group provides sufficient sample size. In terms of environmental conditions, data collection was carried out under a variety of lighting and venue conditions, including outdoor venues (such as sunny days, cloudy days) and indoor lighting environments. In order to ensure the robustness of the model, 70% of the data was collected in outdoor venues and 30% in indoor environments. In addition, lighting conditions are subdivided into three categories: strong light, weak light, and shadow environments, accounting for 40%, 30%, and 30% of the total data, respectively. Through these settings, the complex environmental conditions that may be encountered in reality are reflected. First, using Action Sequence Accuracy (ASA) and Action Smoothness Detection Time (ASDT) as indicators, each model is run 5 times, and the results are shown in **Figure 14**.

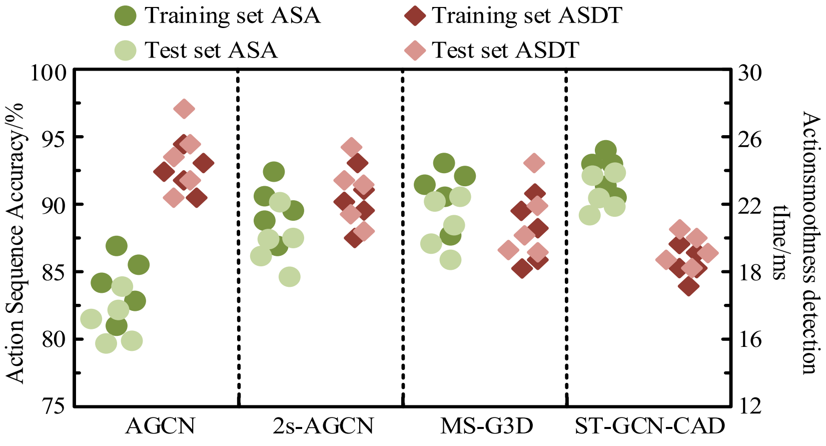


Figure 14. ASA and ASDT test result.

Figure 14 shows the ASA and ASDT obtained by running each model 5 times on the training and testing sets, respectively. ST-GCN-CAD performs the best in the continuous recognition test of tennis serving and receiving movements, with an average ASA of 93.8% and an average ASDT of 19.2 ms. AGCN shows the weakest performance with 84.7% ASA and 25.4 ms ASDT, and has a higher delay in detecting motor fluency. The performance of 2s-AGCN and MS-G3D on ASA and

ASDT is relatively similar, with ASA of 89.4% and 91.9% in the test set, and ASDT of 22.8 ms and 21.2 ms. ST-GCN-CAD reduces computational redundancy and improves model processing efficiency through optimized 13 skeleton points and network structure. It has higher accuracy and shorter detection time in continuous recognition of tennis serve and receive actions. Its advantage lies in its ability to capture fine-grained actions more accurately and process them in real-time. Finally, this study further expands the sample size and evaluates the model's performance in recognizing complex action sequences on different dataset sizes to test its scalability and robustness to large-scale datasets. **Table 2** shows the test results.

Table 2. Complex action sequence recognition test results.

Model	Data size/samples	CASA/%	SCM/%	Processing time/ms	Memory usage/MB
AGCN	100	85.6	80.1	30.2	1049
	500	84.5	78.2	32.6	1121
	1000	82.3	74.6	35.9	1224
2s-AGCN	100	90.4	85.9	28.3	1098
	500	89.1	82.7	31.2	1142
	1000	87.5	80.3	34.5	1227
MS-G3D	100	93.8	88.2	26.1	1199
	500	92.5	86.4	29.8	1289
	1000	90.7	83.9	32.2	1378
ST-GCN-CAD	100	95.4	91.3	24.7	1023
	500	94.7	90.1	29.3	1108
	1000	93.5	88.7	32.7	1221

In **Table 2**, there are significant differences in the performance of each model at different sample sizes. When the sample size is 1000, ST-GCN-CAD achieves a Complex Action Sequence Accuracy (CASA) of 93.5 for action combination recognition. The CASA of AGCN at the same scale is only 82.3%. Meanwhile, ST-GCN-CAD also performed well in the Sequence Continuity Maintenance (SCM) metric, reaching a maximum of 91.3%. Its processing time is slightly higher, at 32.7ms. In addition, as the sample size increases, the memory usage of each model also increases. MS-G3D achieves a memory usage of 1378 MB at 1000 samples, but overall it remains reasonable. Overall, ST-GCN-CAD performs well in balancing accuracy and efficiency, making it suitable for complex action recognition tasks on large-scale datasets.

4. Discussion

In order to improve the recognition accuracy and the ability to handle complex scenes in tennis action recognition tasks, the IFaster R-CNN target detection algorithm, the pose estimation algorithm based on MF-PoseResNet50 and the ST-GCN-CAD action recognition model were researched and designed. First of all, compared with the lightweight single-branch Pose distillation network proposed by Zhang et al. [3], the MF-PoseResNet50 model significantly improves the action recognition accuracy by fusing multi-scale features. On the COCO data set, the AP values of Zhang et al.'s method at different scales fluctuate around 70%, while the AP, AP50, AP75, AP (E), AP (M), and AP (L) values of MF-PoseResNet50 They are 70.4%, 88.5%, 75.6%, 57.4%, 69.3% and 79.2% respectively. It shows that its advantages in target detection at different scales are more significant. This improvement is due to the fusion of multi-scale features, allowing the model to not only capture global bone structure but also better focus on local details. When the number of iterations reaches 500, the mAP value and IoU value of IFaster R-CNN are 0.91 and 0.89 respectively.

For the ST-GCN-CAD model, compared with the multi-task model combining ST-GCN and YOLO proposed by Liu et al. [5], ST-GCN-CAD has better performance in complex scenes. Liu et al.'s model only achieved 82.3% on CASA, while ST-GCN-CAD achieved 93.5% and SCM achieved 91.3%. At the same time, its maximum running time is 35.9ms and the memory usage is 1378MB. This result is mainly attributed to the introduction of the channel attention module, which effectively enhances the model's attention to important features, thereby showing stronger generalization ability and robustness in complex action sequences. It is worth mentioning that the structural design of the ST-GCN-CAD model gives it strong generalization ability and is also suitable for action recognition tasks in other sports. For example, when processing basketball, football and other highly dynamic scenes, the channel attention mechanism can effectively capture important motion features, while the multi-scale dilated convolution module helps the model adapt to actions at different scales. In addition, this structure makes the ST-GCN-CAD model potentially adaptable when facing different groups of people, such as different ages, genders, and exercise levels, which lays a good foundation for expanding the application scope of the model in the future.

In summary, MF-PoseResNet50 successfully integrates the multi-scale feature fusion capabilities of the residual network, demonstrating excellent accuracy and computational efficiency. ST-GCN-CAD significantly improves the accuracy and robustness of action recognition through the introduction of channel attention and dilated convolution. Overall, it provides new research directions and technical means for the future action recognition field, which helps to improve the accuracy, robustness and wide applicability of the model in practical applications.

5. Conclusion

In response to the complex PEAR problem in tennis, this study proposed the IFaster R-CNN algorithm, the pose estimation algorithm based on MF-PoseResNet50, and the ST-GCN-CAD model. The experimental results show that the model has good loss function value and PR curve performance. The AP test results at different scales also show excellent detection capabilities. At the same time, the test results at different resolutions show that the model can effectively control the computational overhead while ensuring high accuracy. In the simulation test, the model not only achieved high-precision performance in complex scenes and body occlusion tests, but also showed strong adaptability in multi-target recognition and dynamic environments. The channel attention mechanism effectively enhances the model’s attention to important features, thereby achieving more accurate posture estimation and action prediction. In summary, the ST-GCN-CAD model performs well in processing dynamic and complex scenes, demonstrating its great application potential in tennis action recognition tasks.

However, there are still some shortcomings in the research, that is, the performance of the model still has room for improvement when dealing with multi-target interference or complex backgrounds. Although the model performs well in tennis action recognition, its adaptability in other types of sports action recognition and different populations has not been fully verified. The generalization ability of the model still needs further discussion and testing. Future research will focus on the following aspects: First, further improve the performance of the model by introducing multimodal data fusion technology and combining athlete biosignal data to improve the understanding and prediction of sports behavior. Second, explore the application of the model in multi-athlete interaction scenarios, such as collective tactical analysis in team sports, which will greatly expand the scope of application of the model. In addition, the adaptability of the model in other sports and different populations will be further explored, and its generalization ability in different sports scenarios such as basketball and football will be verified, and the model will be ensured to have robust performance for different populations. Finally, the focus of future research will also be on how to transform the research results into practical sports training tools, develop real-time action recognition and feedback systems, help athletes optimize training strategies, and provide coaches with instant feedback, thereby improving training efficiency and sports performance.

Ethical approval: Not applicable.

Conflict of interest: The author declares no conflict of interest.

Abbreviations

Pose Estimation and Action Recognition	PEAR
Deep Learning	DL
Graph Convolutional Networks	GCNs
Object Detection	OD
Spatial Temporal Graph Convolutional Network	ST-GCN
Fish Swarm Algorithm	FSA
Precise Time Protocol and CNN	PTP-CNN
Faster Region-based CNN	Faster R-CNN
ST-GCN with Channel Attention and Dilated Convolution	ST-GCN-CAD
Channel Attention Module	CAM)
Multi-Scale Dilated Convolution Module	MSDCM
Human Pose Estimation	HPE
Spatial Pyramid RoI Pooling Layer	SPP
Support Vector Machine	SVM
Left and Right	L&R
Attention Module	ATT

Temporal Convolutional Network	TCN
Pooling layer	POOL
Fully Connected layer	FC
Batch Normalization	BN
Global Average Pooling	GAP
Mean Average Precision	mAP
Intersection over Union	IoU
Efficient Detection	EfficientDet
Higher Resolution Network	HigherHRNet
Mask Region-based CNN	Mask R-CNN
Area Under the Precision Recall Curve	AUC
Average detection Precision	AP
Frames Per Second	FPS
Adaptive GGCN	AGCN
Two-Stream AGCN	2s-AGCN
Multi-Scale Graph Temporal Convolutional Networks	MS-G3D
Action Sequence Accuracy	ASA
Action Smoothness Detection Time	ASDT
Complex Action Sequence Accuracy	CASA
Sequence Continuity Maintenance	SCM

References

1. Mokayed H, Quan TZ, Alkhaled L, Sivakumar V. Real-time human detection and counting system using deep learning computer vision techniques. Artif. Intell. Appl. 2023;1(4):221-229.
2. Zhang J, Gong K, Wang X, Feng J. Learning to augment poses for 3D human pose estimation in images and videos. IEEE Trans. Pattern Anal. Mach. Intell. 2023;45(8):10012-10026.
3. Zhang S, Qiang B, Yang X, Zhou M, Chen R, Chen L. Efficient pose estimation via a lightweight single-branch pose distillation network. IEEE Sens. J. 2023;23(22):27709-27719.
4. Zhang M, Zhou Z, Deng M. Cascaded hierarchical CNN for 2D hand pose estimation from a single color image. Multimed. Tools Appl. 2022;81(18):25745-25763.
5. Kong L, Pei D, He R, Huang D, Wang Y. Spatio-temporal player relation modeling for tactic recognition in sports videos. IEEE Trans. Circuits Syst. Video Technol. 2022;32(9):6086-6099.
6. Liu C, Li X, Li Q, Xue Y, Liu H, Gao Y. Robot recognizing humans intention and interacting with humans based on a multi-task model combining ST-GCN-LSTM model and YOLO model. Neurocomputing. 2021;430:174-184.
7. S. Yan, Y. Xiong, and D. Lin, "Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition," Proc. AAAI Conf. Artif. Intell., vol. 32, no. 1, pp. 12328, January, 2018, DOI:10.1609/aaai.v32i1.12328.
8. M. Li, S. Chen, X. Chen, Y. Zhang, Y. Wang, and Q. Tian, "Actional-Structural Graph Convolutional Networks for Skeleton-Based Action Recognition," IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), pp. 3595-3603, June, 2019, DOI:10.1109/CVPR.2019.00371.
9. J. Zhu and C. Zhu, "Enhancing Action Recognition with Channel Attention Modules in GCN," 2023 3rd Int. Conf. Electron. Inf. Eng. Comput. Sci. (EIECS), pp. 625-628, September, 2023, DOI:10.1109/EIECS59936.2023.10435515.
10. Keskes and R. Noumeir, "Vision-Based Fall Detection Using ST-GCN," IEEE Access, vol. 9, pp. 28224-28236, February, 2021, DOI:10.1109/ACCESS.2021.3058219.
11. Tong J, Wang F. Basketball sports posture recognition technology based on improved graph convolutional neural network. J. Adv. Comput. Intell. Intell. Inform. 2024;28(3):552-561.

12. Lovanshi M, Tiwari V. Human skeleton pose and spatio-temporal feature-based activity recognition using ST-GCN. Multimed. Tools Appl. 2024;83(5):12705-12730.
13. Li Q, Wan J, Zhang W, Kweh QL. Spatial-temporal graph neural network based on node attention. Appl. Math. Nonlinear Sci. 2022;7(2):703-712.
14. Zhang J, Ye G, Tu Z, Qin Y, Qin Q, Zhang J, Liu J. A spatial attentive and temporal dilated (SATD) GCN for skeleton-based action recognition. CAAI Trans. Intell. Technol. 2022;7(1):46-55.
15. Sabo A, Mehdizadeh S, Iaboni A, Taati B. Estimating parkinsonism severity in natural gait videos of older adults with dementia. IEEE J. Biomed. Health Inform. 2022;26(5):2288-2298.
16. Wu S. Image recognition of standard actions in sports videos based on feature fusion. Trait. Signal. 2021;38(6):1801-1807.
17. Sun J, Lu L. Action recognition method in sports video shear based on fish swarm algorithm. J. Inf. Process. Syst. 2023;19(4):554-562.
18. Ren Y, Sun K. Application effect of human-computer interactive gymnastic sports action recognition system based on PTP-CNN algorithm. Int. J. Adv. Comput. Sci. Appl. 2024;15(1):136-145.
19. Barbon Junior S, Pinto A, Barroso JV, Caetano FG, Moura FA, Cunha SA, et al. Sport action mining: Dribbling recognition in soccer. Multimed. Tools Appl. 2022;81(3):4341-4364.
20. Shan S, Sun S, Dong P. Data driven intelligent action recognition and correction in sports training and teaching. Evol. Intell. 2023;16(5):1679-1687.
21. Jin G. Player target tracking and detection in football game video using edge computing and deep learning. J. Supercomput. 2022;78(7):9475-9491.
22. Niknejad N, Caro JL, Bidese-Puhl R, Bao Y, Staiger EA. Equine kinematic gait analysis using stereo videography and deep learning: stride length and stance duration estimation. J. ASABE. 2023;66(4):865-877.
23. Kolekar S, Gite S, Pradhan B, Alamril A. Predicting vehicle pose in six degrees of freedom from single image in real-world traffic environments using deep pretrained convolutional networks and modified Centernet. Int. J. Smart Sens. Intell. Syst. 2024;17(1):384-406.
24. Tej B, Bouaafia S, Hajjaji MA, Mtibaa A. AI-based smart agriculture 4.0 system for plant diseases detection in Tunisia. Signal Image Video Process. 2024;18(1):97-111.
25. Yang P, Liu Q, Wang B, Li W, Li Z, Sun M. An empirical study of fault diagnosis methods of a dissolved oxygen sensor based on ResNet-50. Int. J. Sens. Netw. 2022;39(3):205-214.
26. Xu X, Guo Y, Wang X. Human pose estimation model based on DiracNets and integral pose regression. Multimed. Tools Appl. 2023;82(23):36019-36039.
27. Ma X, Li Z, Zhang L. An improved ResNet-50 for garbage image classification. Tech. Gaz. 2022;29(5):1552-1559.
28. Dewi C, Chen RC. Combination of ResNet and spatial pyramid pooling for musical instrument identification. Cybern. Inf. Technol. 2022;22(1):104-116.
29. Mu T, Zhang C, Huang M, Ning B, and Wang J. Partitioning leakage detection in water distribution systems: a specialized deep learning framework enhanced by spatial-temporal graph convolutional networks. ACS ES&T Water. 2024;4(8):3453-3463.
30. Yang S, Li Z, Wang J, He D, Li Q, Li D. ST-GCN human action recognition based on new partition strategy. Comput. Integr. Manuf. Syst. 2023;29(12):4040-4059.
31. Wang H, Zhang R, Cheng X, Yang L. Hierarchical traffic flow prediction based on spatial-temporal graph convolutional network. IEEE Trans. Intell. Transp. Syst. 2022;23(9):16137-16147.
32. Alsawadi MS, Rio M. Skeleton split strategies for spatial temporal graph convolution networks. arXiv. 2021;23:4643-4658.
33. Tsai MF, Huang SH. Enhancing accuracy of human action recognition system using skeleton point correction method. Multimed. Tools Appl. 2022;81(5):7439-7459.
34. Wang Y, Wang W, Li Y, Jia Y, Xu Y, Ling Y, et al. An attention mechanism module with spatial perception and channel information interaction. Complex Intell. Syst. 2024;10(4):5427-5444.