SIN-CHN
SCIENTIFIC PRESS PTE.LTD.

Molecular & Cellular Biomechanics 2025, 22(1), 776.
https://doi.org/10.62617/mcb776

Article

Recognition method of tennis swing based on time series convolution network

Bo Huang

Department of Sports, Tianjin Foreign Studies University, Tianjin 300204, China; huangbo2020tennis@163.com

CITATION

Huang B. Recognition method of tennis swing based on time series convolution network. Molecular & Cellular Biomechanics. 2025; 22(1): 776.
https://doi.org/10.62617/mcb776

ARTICLE INFO

Received: 12 November 2024
Accepted: 2 December 2024
Available online: 7 January 2025

Abstract: Tennis swings vary widely in type, and accurately identifying these motion patterns is crucial for swing analysis. With advancements in artificial intelligence, recent studies have achieved significant progress in human activity recognition through machine learning and sensor technologies. However, research specifically on tennis swing recognition remains relatively nascent, with limited exploration in this domain. This study focuses on recognizing tennis swing motions using a time-series convolution network, employing sensors to gather essential motion data. The MPU9250 sensor captures the intricate nuances of human movement, which often displays complexity and individual variation. Key challenges include effectively extracting features of tennis swings, designing suitable classifiers for recognition, and enhancing classifier generalization across different individuals. Addressing these challenges, this study introduces a temporal sensing network for swing recognition based on causal and dilated convolution techniques. The network effectively captures the temporal characteristics of swings, achieving a 94.73% recognition rate. Additionally, a comparative analysis between the sequential convolution network and traditional machine learning algorithms is conducted, providing insights into their performance and processing workflows.

Keywords: sequential convolution network; tennis; motion recognition; TCN

1. Introduction

Because of its own value and research significance, human motion recognition technology has attracted much attention in many fields and is regarded as the research target by many scholars, including machine learning, distributed computing and intelligent systems. It can be proved that this technology has a certain research prospect and is also very beneficial for future development and application. Nowadays, artificial intelligence technology is gradually advanced, which makes the development of sensor technology more and more rapid. This technology brings a new research direction for human motion recognition. In today's society, people attach great importance to sports and sports health, and movement recognition has a vital impact on sports events and training. For example, in some difficult movements, the mastery of movements is related to the final achievements of athletes, and the essentials and mastery of movements are also inseparable from the physical injury of athletes. Therefore, it can be proved that the research of motion recognition in sports is very promising. It can not only help athletes improve their performance and master the essentials more rationally, but also avoid injuries as much as possible.

Among many sports, tennis is of great research value. Firstly, there are many movements that athletes change in the training process, but if they want to do well, they need to combine their own characteristics. In the preparation of tennis events, coaches usually carefully analyze the opponent's movement characteristics through video data before the game, in order to find the opponent's attack and defense weaknesses, and find the strategy to defeat the enemy. In addition, the coach will arrange training personnel with similar styles for the athletes, so that the athletes can adapt to the opponent's style in advance and practice the fighting method of defeating the enemy. Therefore, the research on how to identify the tennis swing movement has at least the following two meanings: First of all, it can help the major tennis competitions to become more accurate in guidance and training, because it can be more in-depth analysis, including modeling and identification, while recognizing tennis swing movements. Secondly, more intelligent technologies can be used to speed up the development efficiency, including bracelets and tennis robots, which are of great significance to the daily training and competition of tennis. In this paper, the tennis swing recognition is the research object, and how to recognize the swing through the time convolution network is studied. The main research work is as follows:

1) Consult the relevant literature at home and abroad, deeply analyze the main factors affecting the accuracy of tennis swing motion recognition, and investigate the human motion recognition methods, especially the human motion recognition methods based on acceleration sensors.

2) Learn the basic principle of time-series convolution network (TCN) based on time series convolution network in deep learning, design a swing recognition network based on time series action input based on time series convolution idea, and verify its performance in tennis swing classification.

2. Literature review

As we all know, the technology of human motion recognition has been developing for more than 20 years, and the changes during this period are very rich. First, the sensor category has evolved from the original visual sensor to the wearable experience category. Second, there has been a certain breakthrough in learning methods. From traditional learning to today's deep learning, it has achieved great success in the construction of network models. So far, human body recognition methods can be classified into the following three categories, namely, vision methods, environmental sensor methods and wearable sensor methods. The tennis swing motion recognition studied in this paper also belongs to this field.

In the direction of multi-modal fusion, Khanam et al. summarized the progress of action recognition in a monitoring environment based on the combination of manual design features and deep learning, pointing out that multi-modal feature fusion can effectively improve the sensitivity and accuracy of complex actions [1]. Wu proposed a fusion model that combines fast and slow networks to significantly optimize recognition performance by enhancing fine-grained feature capture capabilities of actions [2]. Yawen and Yi studied a dance video movement recognition method based on computer vision and image processing technology [3]. Through the accurate segmentation and feature extraction of video frames, this study realized the accurate recognition of dance movements under the complex dynamic background, providing an important reference for vision-based movement recognition.

In the field of skeletal action recognition, Zhonghua et al. proposed a graph convolutional network combining channel attention and temporal attention to enhance the recognition performance of skeletal action [4]. By focusing on key information in both spatial and temporal dimensions, this method effectively improves the expression ability of skeleton data, especially in scenes with high action complexity. In addition, Shehzad et al. proposed a two-stream deep learning architecture for human motion recognition [5]. This architecture combines spatial flow and temporal flow, and has significant advantages in capturing short- and long-term characteristics of human motion, demonstrating the potential of multi-flow fusion architecture in motion recognition.

Researchers in the field of human motion recognition have fully studied and demonstrated the feasibility of various sensors and their combinations in human motion recognition. There are a method based on a single sensor, a method having a plurality of single type sensors, and a method combining a plurality of multi type sensors. For example, Kau proposed a fall detection algorithm based on the electronic compass and triaxial accelerometer, which analyzes whether the elderly fall based on the angle information obtained from the electronic compass and the acceleration data measured by the accelerometer [6]. In order to urge patients to take medicine, Kalantarian and Sarrafzadeh used the inertial sensor integrated in the smart watch to analyze the motion data of human wrists, and thus designed a motion classifier that can recognize the bottle cap screwing and medication gesture [7]. This work is expected to be applied in the field of postoperative rehabilitation and patient monitoring. In addition to these familiar sensors, Jin et al. investigated in detail wearable sensors designed using new material technologies (such as temperature sensing and strain materials) [8]. These monitoring sensors based on new materials can complete the basic physiological monitoring of the human body, and can be used to monitor brain activity and fatigue state.

In recent years, with the popularity of deep learning, many researchers have introduced deep learning into the field of human motion recognition based on wearable sensors. Although a lot of achievements have been obtained, few work and teams focusing on tennis swing motion recognition have been found. At present, there are no products and applications specially used for tennis motion recognition in the market, so there are still many challenges and problems to be solved by researchers.

3. TCN tennis swing recognition method based on time convolution network

In this chapter, a learning strong classifier is mainly designed. Its main purpose is to have more accurate research on tennis swing motion recognition. The specific design idea is evolved from the effective data in the acceleration sensor. In the process of design, we summarize its characteristics, mainly including: 1) the whole process is not concise and clear enough. Only after the design of each sub classifier is completed can we obtain the learning classifier 2) In the process of practical application, this method is mainly verified on the basis of experiments, but if you want to select its characteristics, it needs to rely on manual experience 3) It is not very accurate in performance because there is no close relationship between the individuals in this classifier.

In view of this, this chapter explores a motion recognition scheme based on deep learning. Compared with ensemble learning, the deep learning method is simple in operation, without feature selection, and can learn the feature expression according to the characteristics of the data set and give the action classification results. With the proposal of “alexnet” in 2012, deep learning has made rapid development and breakthrough. In many recognition problems, deep learning methods have been crushed by traditional machine learning algorithms. In deep learning, recurrent neural network (RNN) is a kind of neural network specially used for processing sequence data. It was first applied in the field of natural language processing, such as speech recognition, speech modeling and translation. Compared with the traditional deep neural networks RNN network, the biggest feature is the memory function of historical information [9]. This memory ability can help the deep learning network to discover the potential information of data in the time series, so as to complete more complex classification tasks. In addition, it is worth noting that this memory ability of RNN is also used in the field of human motion recognition. Since human actions have time-domain characteristics, the feature expression of human actions in time-domain can be learned through RNN. In addition to RNN, TCN is also a time series modeling method, in 2018, Although TCN is based on the improvement of CNNC (evolutionary neural network) method, TCN has almost the same model size as RNN. In addition, the structure of RNN network determines that it can only process one time step at a time, and the next step must wait for the previous step to complete the operation, which means that RNN. It can not perform large-scale parallel processing like CNN, but TCN improved based on CNN has parallel processing capability.

3.1. TCN rationale

3.1.1. TCN network model

TCN network is similar to CNN network architecture, and its foundation is convolution operation, such as one-dimensional full convolution, causal convolution technology applicable to sequences, void convolution (or expansion convolution) based on function, residual connection and other structures.

1) One dimensional full convolution structure

Figure 1 below shows the basic structure of one-dimensional full convolution. The upper layer result can be obtained by convolution operation of the lower two inputs [10]. Based on the prediction of each time-series element, TCN can realize element level perception and prediction. According to the basic concept of convolutional neural network, the high-level neuron nodes have higher receptive fields and are more sensitive to feature transformation, which is the basis for TCN to realize time-domain modeling.

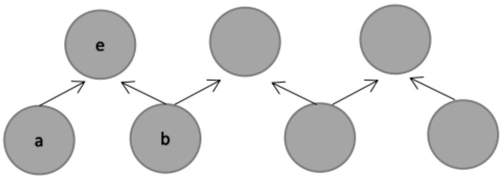

Figure 1. Schematic diagram of one-dimensional full convolution.

2) Causal perception technology

Causal perception was first proposed in WaveNet [6], and was initially used in the field of speech processing. The emergence of causal awareness technology is the basis for TCN network to have sequence awareness. **Figure 2** shows the basic principle of causal convolution.

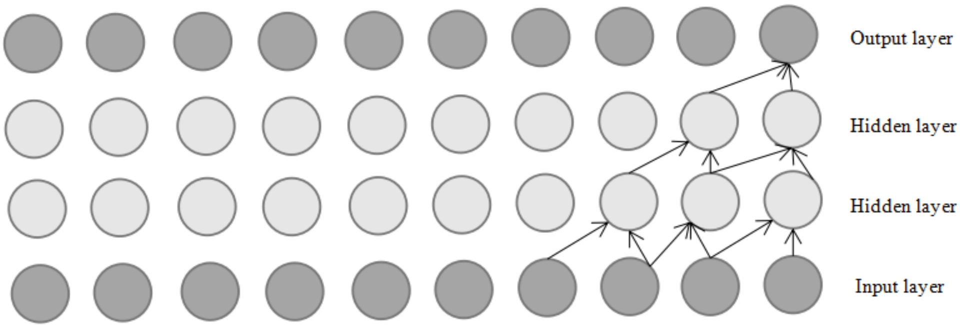

Figure 2. Schematic diagram of causal perception.

Assuming that the convolution filter is $F = (f_1, f_2, \cdots, f_k)$ and the sequence to be perceived is $X = (x_1, x_2, \cdots, x_T)$, the causal perception result at x_t can be obtained by Equation (1). Taking **Figure 2** as an example, the figure shows the causal perception result with convolution kernel of 2. It is assumed that the last two nodes of the input layer are $(x_{t-1}, x_t)$and the filter is (f_1, f_2). According to the perception result y_t at formula x_t. Can be expressed as $f_1 x_{t-1} + f_2 x_t$.

$$F \times X(x_t) = \sum_{k=1}^{K} f_k x_{t-K+k} \tag{1}$$

It can be seen from the perception process in **Figure 2** that causal perception does not consider future information but only historical information, and the longer the causal perception traces the information, the more hidden layers. Suppose we take the hidden layer of the second layer as the output, and its last node is associated with three nodes of the input layer. If we take the output layer as the output, its last node is associated with four nodes of the input layer. Secondly, as a new structure proposed for the time series problem, the causal convolution input is a sequence, and the prediction result can also be a sequence.

3) Void convolution

Cavity convolution is also called dilated convolution. Cavity convolution expands the receptive field by ignoring some receptive nodes. In convolutional neural networks, the expansion of receptive fields is realized by adding a pool layer. Because some receptive nodes are ignored, information loss is inevitable. In TCN, the parameter dilation rate is used to express the degree of cavity. The expansion rate expresses the number of nodes neglected between each two cores. **Figure 3** is a schematic diagram of hole convolution operation. In the figure, D represents the expansion rate (expansion rate of 1 indicates no expansion of receptive field) [11]. The left side of the dotted line is the filling node, which is used to ensure that the number of input and output nodes is the same. The filling number is equal to the convolution kernel size minus and multiplied by the expansion coefficient.

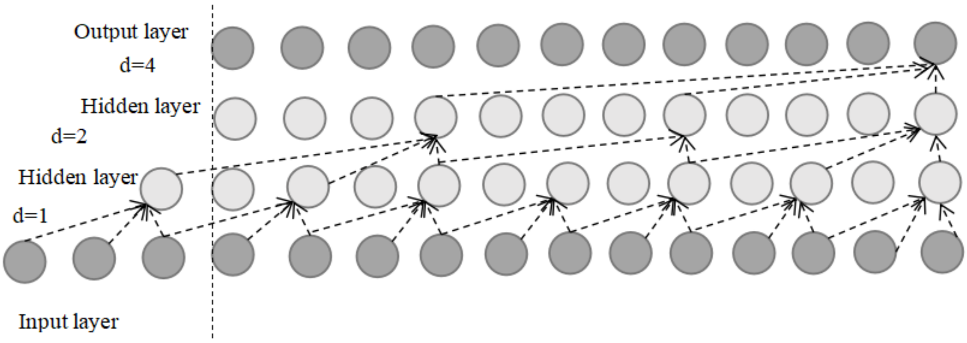

Figure 3. Schematic diagram of void perception.

The convolution result under cavity convolution can be calculated by Equation (2).

$$F^d \times X(x_t) = \sum_{k=1}^{K} f_k^x t - K - k \times d \tag{2}$$

The symbol definition in the formula is the same as that in Equation (1), where F^d represents the hole convolution operation.

4) Residual connection

In convolutional neural networks, with the increase of network layers, the learned features will be richer, more abstract and closer to semantic information. However, simply increasing the number of network layers may make the gradient "disappear" or "explode". A conventional solution to this problem is to initialize the training weights effectively and use the regularization layer. After solving the problem that the number of layers cannot be increased due to the gradient problem, the convolutional neural network may also fall into the performance degradation problem due to the increase of layers. At first, the neural network designer did not know the optimal number of network layers [12]. In order to solve the network degradation problem, TCN adopted the residual connection method to solve the network degradation problem. The purpose of residual connection is to make redundant networks generate identity mapping, thus eliminating redundancy. Network impact. In general, it is difficult for a certain layer of the network to learn the identity mapping function (f(x)=x). However, if the learning identity mapping function is converted to learning a residual function $(r(x) = f(x) - x)$, it only needs $r(x) = 0$.

Figure 4 shows the connection diagram of a residual module. Residual connection provides two transition modes for the network, one is identity mapping connection (also known as s-shortcut connection) and the other is residual mapping connection. When the network is degraded due to redundant connection, this structure can make the result of residual mapping tend to zero, activate identity connection, and eliminate the role of redundant network. In addition, identity mapping will not increase network parameters and computational complexity. Relu in the figure represents a linear modified activation function [13].

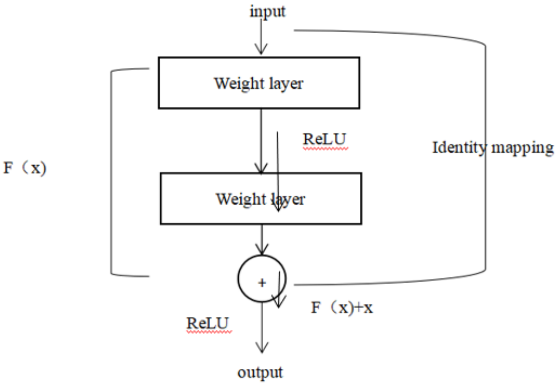

Figure 4. Residual connection.

The TCN network uses residual connections to build a network model. Its basic structure is a plurality of residual blocks composed of residual connections, as shown in **Figure 5**. Conv1d in the figure represents a 1 × 1 convolution or an identity map [14]. Generally, a residual module includes two basic networks, as shown by the dotted line in the figure.

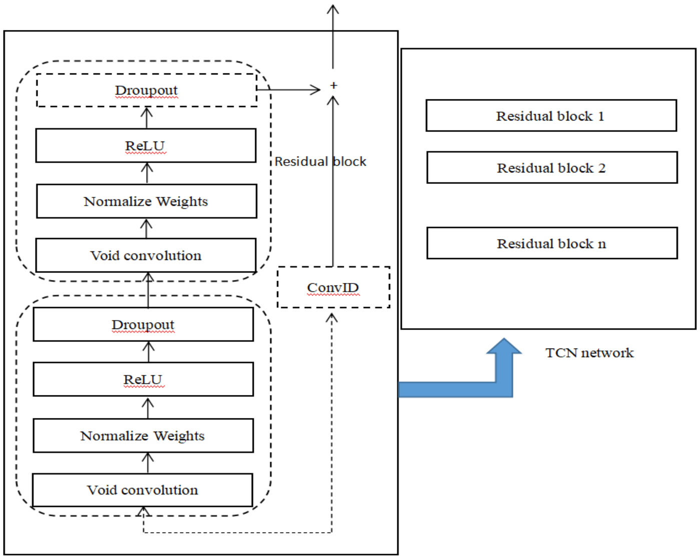

Figure 5. Basic structure of TCN network.

3.1.2. TCN sequential convolution idea

In the research of tennis swing motion recognition, the length of input data is very strict, and it can only be in the same sequence. In this chapter, the main research goal is to form the recognition network of tennis swing, and the sample data passed is the three-axis acceleration data with the same length in the training process. Secondly, the input point is confirmed to be the three-five acceleration data with the same length. The final goal is to detect whether there is a coincidence between the swing and the swing. On the basis of this goal, we can make assumptions, regard the sequence length as t, and imagine the feeling of t as the requirement of the sequence convolution idea on the network. In short, t belongs to each result in the network output sequence. The appearance of TCN model verifies this idea. If you want to complete sequence perception, you need to complete it through causal convolution, because the two are closely related. At the same time, the scope of void perception is more extensive, including two points: one is to input the sequence length, and the other is to effectively control the extent of the network. At the same time, the design of residual convolution has a crucial advantage, that is, the final result of network training can be reached the highest, and the performance of the sequence network will not be degraded [15].

3.2. Design of tennis action recognition method based on TCN network

3.2.1. Training data processing

In order to ensure the diversity and credibility of the data, this study invited 15 participants with different tennis levels to participate in the data collection. These participants included 10 amateurs and 5 professionals with a wide range of tennis experience: Amateurs with 1 to 3 years of training experience, while professionals with more than 5 years of competition experience. Data collection is carried out on a unified hard earth field, and all environmental conditions (such as light, temperature and humidity) are consistent to avoid the influence of external factors on action performance.

During the experiment, the participants completed four types of swing-forehand, backhand, forehand and backhand. Each movement was repeated 20 times, and a total of 1200 movements were collected (15 participants × 4 movements × 20). In order to reduce the interference of fatigue on performance, each participant was given a 5-minute rest period after completing 10 sets of movements. All participants wore the MPU9250 sensor, which was mounted on the wrist in a fixed manner and the sampling frequency was set at 200 Hz to ensure high accuracy and consistency of data.

Through Gaussian filtering and sliding window processing (window length 4 seconds, window overlap rate 50%) on the collected original data, a time series of equal length is finally formed for subsequent model training and testing. In the segmentation process, it ensures that the adjacent sequence data has a coincidence degree of so% [16]. Different from the third chapter, there is no need to process, extract and select the time-series signals, but directly put the data into the TCN network for learning. Since the sliding window time width of this experiment is set to 4 seconds and the sampling frequency of the equipment is 200 Hz, each sample includes 800 sampling points, i.e., the T value is 800. Each action sample is an 800 × 3 matrix. As shown in **Figure 6** below.

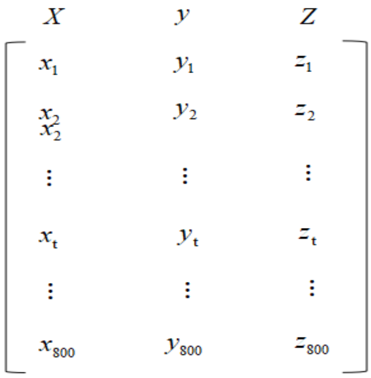

Figure 6. Schematic diagram of single sample input.

(*x*, *y*, *z*) in the figure represents the acceleration data values in the three directions *x*, *y* and *Z* at the *t*-th time in each sample. Since the sliding window time width of this experiment is set to 4 seconds and the sampling frequency of the equipment is 200 Hz, each sample includes 800 sampling points, i.e., the *T* value is 800. Each action sample is an 800 × 3 matrix. In all the data samples obtained, 80% of them are taken as training samples by random method, with a total of 4000 (1000 for each type of four actions); The remaining 20% is taken as the test sample, with a total of 1000 test samples (250 for each type of four actions). It is different from setting the labels of the four swing movements as 1, 2, 3 and 4 in the integrated learning. Here, the discrete one hot encoding is used to label the training data. The existence value of the single heat coding is that it is very useful for the error function, but the premise is that it is carried out in the form of cross entropy, otherwise the range within each label will become chaotic. The advantage of the discrete label is that it is more effective in parameter solution, and can help the model reduce the loss to a greater extent in calculation [17]. **Table 1** has listed the unique heat coding forms in the four swing movements in detail.

Table 1. Heat reading codes of four swing movements.

Swing mode	Original label	Single heat coding
Forehand attack	1	[1 0 0 0]
Backhand push	2	[0 1 0 0]
Forehand rubbing	3	[0 0 1 0]
backhand push	4	[0 0 0 1]

3.2.2. design of swing motion recognition model based on TCN

Figure 7 is a schematic diagram of a neural network for tennis swing motion recognition designed in this paper based on TCN sequential convolution idea. The model is mainly composed of time-series convolution layer and full connection layer. The length of input samples in the figure is *t*, and the dimension of each data in the sequence samples is *F*. it can be seen from the introduction of training samples that the values of *T* and *F* are 800 and 3 respectively. In order to fully perceive the information on the sequence, the model uses m one-dimensional convolution kernels of size D. After convolution of the three dimensions (*x*, *y* and *Z* axes) of the input sequence samples using one-dimensional convolution kernel, m time-series features are formed. The dimension of each time-series feature is consistent with the length of the input sequence, which is consistent with the characteristics of time-series perception. After that, the time-series feature sequence is input into the fully connected network and softmax to obtain the final prediction result.

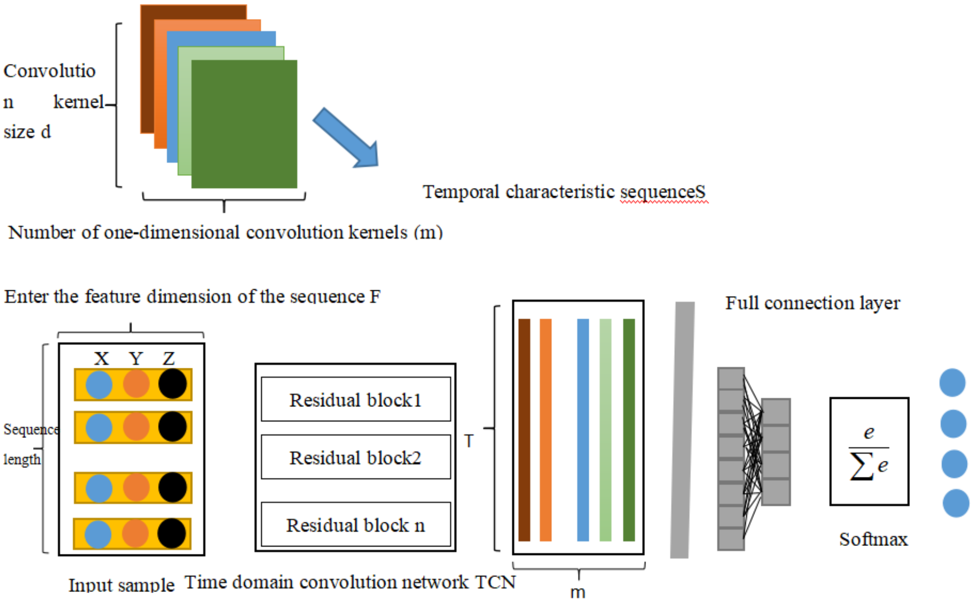
Convolutio n kernel size d
Temporal characteristic sequenceS
Number of one-dimensional convolution kernels (m)
Enter the feature dimension of the sequence F
X Y Z
Sequence length
Residual block1
Residual block2
Residual block n
T
Full connection layer
$\frac{e}{\sum e}$
Softmax
Input sample
Time domain convolution network TCN
m

Figure 7. Schematic diagram of algorithm network model.

The dilation rate values in **Table 2** were carefully chosen to ensure an effective balance between short-term and long-term temporal dependencies in the input sequences. Smaller dilation rates (e.g., 4) capture local features, while larger dilation rates (e.g., 48) extend the receptive field to encompass longer temporal patterns. This configuration allows the model to process both fine-grained and global motion characteristics critical for tennis swing recognition.

Table 2. Expansion rate of TCN layers.

Layer name	Expansion rate in cavity convolution
Input layer	-*
Hidden layer0	4
Hidden layer1	12
Hidden layer2	48
Hidden layer3	48
Hidden layer4	48
Hidden layer5	48
Hidden layer6	48
Hidden layer7	4
Output layer	-*

*No expansion.

Table 2 shows the design of some hidden layers in the TCN network model. A total of 8 hidden layers are used in the model. If measured by the residual blocks combined by two residual connections, it is equivalent to using 4 residual blocks. Through the above design, the receptive field of each dimension of the output layer can reach 800.

These specific dilation rates were determined based on both theoretical considerations and empirical testing. The exponential increase (4, 12, 48) ensures non-overlapping receptive fields across layers while maintaining computational efficiency. Ablation experiments confirmed that this configuration achieves the best trade-off between recognition accuracy and training time, outperforming alternative setups with smaller or excessively larger dilation rates.

3.2.3. Loss function design

In parameter estimation, the two most commonly used methods are the least square method and the maximum likelihood method. Currently, the mainstream loss functions mainly include the mean square error (MSE) loss function and the cross entropy error (CEE) loss function. For different application scenarios, the loss function needs to be designed as the objective function according to the specific task solution. In this paper, the method of cross entropy error loss function is used to estimate the parameters [18].

According to the definition of cross entropy, if p represents the actual probability distribution, q is the probability estimate calculated by the model, and q represents the cross entropy of p as follows:

$$H(p,q) = -\sum_{x} p(x) \log q(x) \tag{3}$$

According to the above formula, the parameter penalty term $\frac{\lambda}{2}\|\theta\|$, θ represents the model hyperparameters to be optimized) is added, and the cross entropy loss function applicable to this paper is derived by using the maximum likelihood method:

$$J(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{K} p_{ik} \log q_k(x_i;\theta) + \frac{\lambda}{2} \tag{4}$$

where, if N represents the number of samples, K represents the total number of categories of samples, and X_i is the i th sample point in the training sample set.

This paper uses the following algorithm to optimize the error of the loss function:

Step 1: determine the gradient size and direction of the loss function at the current position. The gradient vector grad formula for 111 is as follows:

$$grad = \frac{\partial}{\partial\theta} J(\theta) \tag{5}$$

Step 2: determine the step size, multiply the step size by the current gradient grad of the loss function, and calculate the current descent distance of the gradient, $\alpha \frac{\partial}{\partial\theta} J(\theta)$

Step 3: judge the size of all weights in the parameter θ vector, and stop the algorithm if the gradient descent distance is less than ε. Take the current θ as the final gradient vector, otherwise proceed to step 4.

Step 4: update 111 vector, and its update expression is as follows. And proceeds to the first step.

$$\theta = \theta - \alpha \frac{\partial}{\partial \theta} J(\theta) \tag{6}$$

Using the above calculation formula, assuming that the input vector is 111 and the real label is 222, the following formula can be obtained:

$$\frac{\partial}{\partial \theta} J(\theta) = X^T(X\theta - Y) \tag{7}$$

$$\theta = \theta - \alpha X^T(X\theta - Y) \tag{8}$$

where θ is the step size (i.e., the learning rate). There are many gradient descent algorithms [19]. According to the number of samples required for each iteration, they can be divided into random gradient descent C stochastic gradient descent, Stochastic Gradient Descent (SGD), batch gradient descent bgd and mini batch gradient descent Mini-Batch Gradient Descent (MBGD), as well as the new gradient descent methods evolved from them, such as Adam C adaptive motion estimation. Among them, Adam gradient descent algorithm has many advantages compared with other gradient descent algorithms: high calculation efficiency, easy to implement, less memory occupation, and the update step size is only related to the parameters and independent of the gradient size [20]. In addition, admin has no stable requirement for the objective function, that is, the loss function can handle the noise samples well with the change of time. This paper adopts Adam method.

4. Experimental data and analysis

4.1. Model performance analysis

The TCN model in this paper is implemented based on python programming under the deep learning framework keras.

The hyperparameters of the model designed in this chapter mainly include the size of one-dimensional convolution kernel, the number of residual blocks, the dimension of input samples, the value of random inactivation dropout, the learning rate, and the number of training times [21]. To ensure the performance of the model, the number of residual blocks is equivalent to the number of layers of the ordinary neural network; the convolution kernel should not be too large and the residual blocks should not be too large. They will increase the number of neural networks and the amount of computation, and also affect the convergence of the algorithm. In addition, when debugging the performance of the network model, it is found that the appropriate random inactivation rate can also improve the overfitting brought by the training samples and obtain better test results [22]. After repeated tests, the parameter settings finally adopted in this paper are shown in **Table 3** below:

Table 3. Training parameter settings.

Parameter name	**Parameter value**
Convolution kernel	4
Residual block	4
Enter sample dimension	3
Time series length	800
Random inactivation rate	0.45
optimizer	Adam
Learning rate	0.01
Activation function	ReLU

In addition, some parameters in the paper, such as the learning rate and the number of iterations, are also verified in the experiment.

In the experiment, the selection of the activation function is first determined. Secondly, three kinds of activation functions are selected, namely sigmoid, tanh and relu [23]. **Figure 8** lists the function convergence state in detail. From the comparison results of the three functions, it can be seen that sigmoid has the highest convergence of the objective function, reaching 0.40, tank follows closely, reaching 0.32, and relu is the smallest, reaching 0.23. It can be seen that the convergence of this function is better.

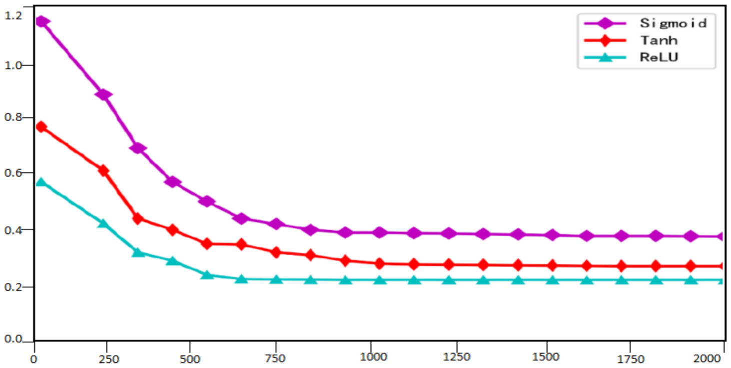

Figure 8. Convergence of objective function.

However, the initialization value of the network model is not without disadvantages. It has a great impact on the network performance. For example, setting the learning rate value can affect its two major factors. First, the convergence of the objective function is not balanced. If the price learning rate value is set too small, its convergence will be in an increasingly slow state, which will ultimately affect its efficiency and waste its resources. Second, the minimum convergence value of the objective function cannot be reached, which leads to the failure to reach the standard of F value and the edge wandering state [24].

In this paper, three learning rates (0.01, 0.008 and 0.005 respectively) are set for comparative experiments, and the experimental results are shown in **Figure 9**. Through the comparative experimental results, it is found that when the learning rate is 0.01, the network can quickly reach the highest recognition accuracy, so the learning rate is set to 0.01.

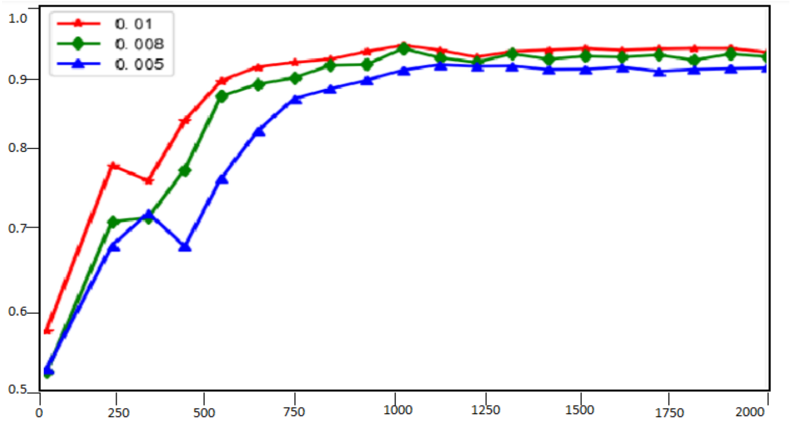

Figure 9. change curve of average accuracy under different learning rates.

Figure 10 shows the test results of the TCN network on the test data set, incorporating both the mean recognition rates and their standard deviations across 10 independent test runs [25]. It is evident from the figure that the TCN network achieves high recognition accuracy overall, with the highest mean accuracy observed in "Backhand rubbing" (94.96% ± 0.015) and the lowest mean accuracy in "Backhand push" (93.87% ± 0.03). These results confirm the findings discussed in Chapter 3, where it was noted that "Backhand push" has weaker swing arm amplitude and strength compared to the other movements, making it more challenging to distinguish. Additionally, the low standard deviations across all motion types (mostly below 0.01) indicate the model's stable performance across different test runs, further reinforcing its reliability in recognizing tennis swing actions [26]. **Figure 11** shows the comparison between the performance of TCN network and the above machine learning algorithm in the average accuracy. From the figure, it can be seen that the integrated learning has reached the highest recognition accuracy in the forehand attack and backhand rubbing, and the TCN network has the highest recognition accuracy in the backhand rubbing and forehand rubbing [27]. However, it is difficult to judge the performance of the integrated learning in Chapter 3 and the TCN network in this chapter from the accuracy rate. However, it can be seen that the TCN network is relatively stable when identifying the four actions (the average accuracy rate is 94.73%, and there will be no ups and downs like ensemble learning [28]. The author of this paper believes that the advantage of TCN network over machine learning algorithms such as ensemble learning lies not only in the feature expression of learning differences, but also in the ability of nonlinear fitting. In addition, TCN network directly perceives time-series samples. Compared with machine learning methods, TCN does not lose time-series correlation information, so TCN can better integrate the four swing movements are stably distinguished.

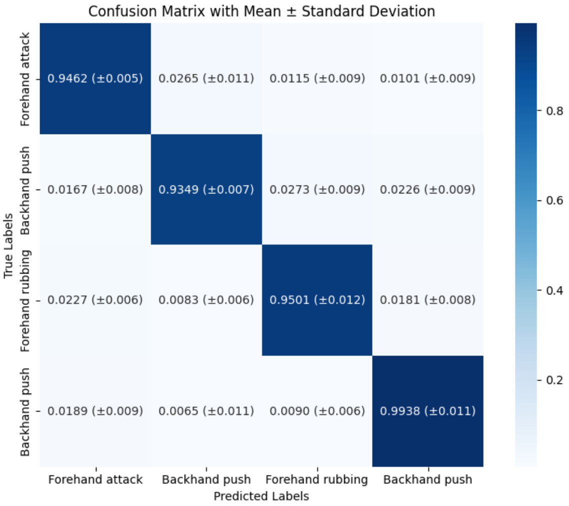

Figure 10. Confusion matrix of TCN network.

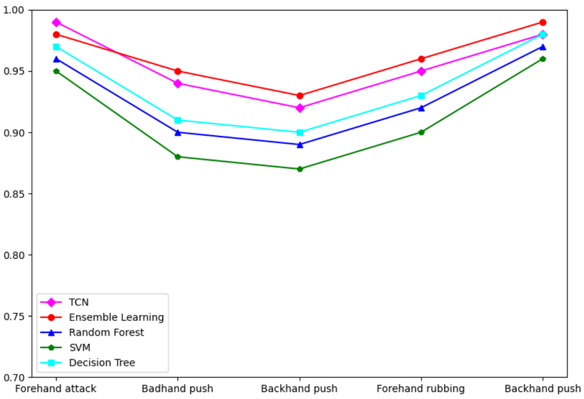

Figure 11. Performance comparison between TCN network and the aforementioned machine learning algorithm.

This chapter systematically introduces the construction of TCN network and tennis action recognition model based on TCN sequential convolution, and analyzes the performance and characteristics of TCN network through experiments, as well as its comparison with traditional machine learning algorithms. In terms of algorithm performance, TCN has no absolute advantage, but thanks to the advantages of time-series convolution network in feature learning and time-series expression, its stability is obviously due to other machine learning algorithms and integrated machine learning algorithms. In addition to the algorithm performance, TCN eliminates the feature engineering in machine learning and avoids the manual feature extraction. TCN algorithm is more versatile and has higher practical value.

However, despite these advantages, the TCN model also has notable limitations that warrant further investigation. Firstly, the performance of the TCN model may decline when applied to data from players not represented in the training dataset [29]. This is particularly evident in cases where the players exhibit significantly different playing styles or levels of expertise.

Secondly, the model's ability to accurately recognize actions under varying swing speeds and styles remains a challenge. While the current model performs well under controlled conditions, variations in swing speed, such as rapid or slow movements, can lead to reduced recognition accuracy [30]. Similarly, unique or unconventional swing styles may not be effectively captured by the TCN's learned features. Incorporating training data that reflects these variations and refining the feature extraction process could improve the model's robustness.

Finally, practical deployment of the TCN model in real-world scenarios faces additional challenges. The reliance on high-quality sensor data means that variations in sensor placement or environmental factors, such as outdoor lighting and noise, could impact model performance. Moreover, deploying the model on edge devices requires optimizing it for low-latency and resource-constrained environments. Future work should focus on developing lightweight versions of the model and investigating techniques to maintain performance under suboptimal conditions.

4.2. Statistical significance testing

To strengthen the comparative analysis between the TCN model and traditional machine learning methods, statistical significance testing was conducted. A one-way ANOVA test was performed to evaluate the overall differences in accuracy across four methods: TCN, Random Forest, SVM, and KNN. The ANOVA results showed a significant F-statistic ($p < 0.01$), indicating that the accuracy differences between these methods are statistically significant. Further pairwise *t*-tests revealed that TCN significantly outperformed Random Forest ($p < 0.05$), SVM ($p < 0.01$), and KNN ($p < 0.01$).

Figure 12 shows the boxplot visualization of accuracy distributions further highlights TCN's superior performance and stability, with a higher mean accuracy (94.7%) and smaller variance compared to traditional methods. These findings provide strong statistical evidence for the advantages of the TCN model in recognizing tennis swing motions.

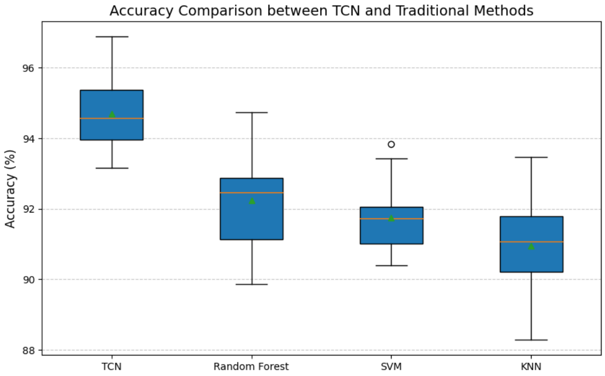

Figure 12. Accuracy comparison between TCN and traditional methods.

4.3. Parameter sensitivity analysis

In addition to the optimization of the learning rate, understanding the sensitivity of other key parameters, such as convolution kernel size and the number of residual blocks, is crucial for improving the model's performance and efficiency. To investigate their effects, we conducted a series of experiments varying these parameters while keeping all other settings constant.

4.3.1. Convolution kernel size

The kernel size determines the receptive field of the convolution operation, directly impacting feature extraction. We tested kernel sizes ranging from 2 to 8 and observed its effect on recognition accuracy and training time. The results, as shown in **Figure 13**, indicate that a kernel size of 4 achieved the best trade-off between accuracy and computational efficiency. Smaller kernels (e.g., size 2) resulted in lower accuracy due to insufficient context capture, while larger kernels (e.g., size 8) increased training time significantly without notable accuracy gains.

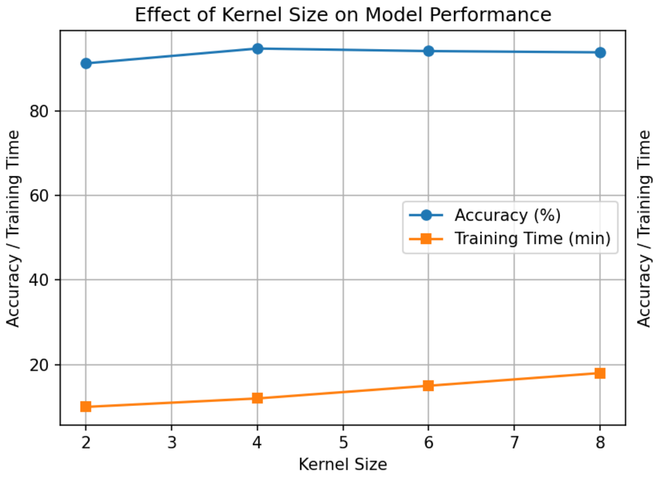

Figure 13. Effect of kernel size on model performance.

4.3.2. Number of residual blocks

Residual blocks contribute to model depth and representation capacity. Experiments were conducted with 2, 4, 6, and 8 residual blocks. The findings, illustrated in **Figure 14**, show that while increasing the number of blocks improves accuracy, the marginal benefits diminish after 4 blocks. Models with more than 6 blocks experienced longer training times and occasional overfitting, suggesting 4 residual blocks as the optimal configuration for this task.

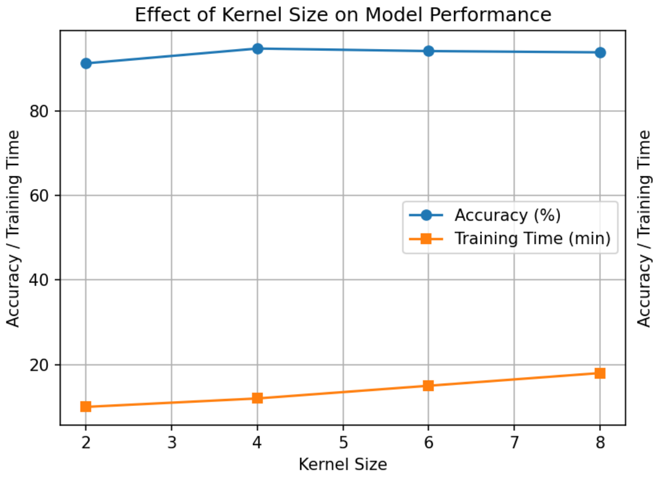

Figure 14. Effect of residual blocks on model performance.

5. Conclusion

Only with the continuous improvement of sensor technology can the field of human motion recognition develop so rapidly. In addition to the main research methods based on vision, sensor-based approaches also hold significant potential for exploration. This paper focuses on the recognition of tennis swing motions using a time-series convolution network (TCN), combining advancements in sensor technology with deep learning methodologies. The research process involved extensive literature review, identification of existing challenges, and the construction of a recognition model capable of classifying four tennis swing actions. The proposed model simplifies the feature extraction process compared to traditional machine learning methods, achieving high recognition accuracy while demonstrating strong stability. Despite the promising results, this study has certain limitations, such as the reliance on controlled experimental conditions and a relatively homogeneous dataset. These constraints leave room for improvement and further innovation. Future work could explore expanding the model's applicability beyond tennis to other racket sports like badminton and table tennis, adapting it to different motion dynamics and equipment. Additionally, employing transfer learning methods may enhance the model's ability to generalize across varied sports or tasks with minimal data. Finally, integrating multi-sensor data, such as gyroscopes or electromyography (EMG), could improve the precision and robustness of motion recognition in more complex and uncontrolled environments. Addressing these aspects will not only refine the current approach but also pave the way for broader applications in sports science and human motion analysis.

Ethical approval: Not applicable.

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

References

1. Khanam S, Sharif M, Cheng X, et al. Suspicious action recognition in surveillance based on handcrafted and deep learning methods: A survey of the state of the art. Computers and Electrical Engineering, 2024, 120(PC):109811-109811.
2. Wu X. Application of intelligent trajectory analysis based on new spectral imaging technology in basketball match motion recognition. Optical and Quantum Electronics, 2023, 56(3).
3. Yawen P, Yi N. Dance video motion recognition based on computer vision and image processing. Applied Artificial Intelligence, 2023, 37(1).
4. Zhonghua S, Tianyi W, Meng D. Combining channel-wise joint attention and temporal attention in graph convolutional networks for skeleton-based action recognition. Signal, Image and Video Processing, 2022, 17(5):2481-2488.
5. Shehzad F, Khan M, Asfand M, et al. Two-stream deep learning architecture-based human action recognition. Computers, Materials & Continua, 2022, 74(3):5931-5949.
6. Kau, L. A Multi-Scale Human Action Recognition Method Based on Laplacian Pyramid Depth Motion Images. In Proceedings of the 2020 ACM Multimedia Asia Conference (MMAsia '20), pp. 33-38. ACM, 2020
7. Kalantarian H., Sarrafzadeh M. An Indoor Human Action Recognition Method Based on Spatial Location Information. Proceedings of the 2015 IEEE International Conference on Pervasive Computing and Communication Workshops (PerCom Workshops), 2015:1-6.
8. Jin X, Tang P, Houet T, et al. Sequence Image Interpolation via Separable Convolution Network. Remote Sensing, 2021, 13(2):296.
9. Harry J. Time series models with univariate margins in the convolution-closed infinitely divisible class. Journal of Applied

Probability, 2020, 53(1):43-18.

10. Kourogi M, Ishikawa T, Kurata T. A Method of Pedestrian Dead Reckoning Using Action Recognition. IEEEION Position, Location and Navigation Symposium. IEEE, 2019, 83(5):62-73.
11. Charalampous K, Gasteratos A. On-line deep learning method for action recognition. Pattern Analysis & Applications, 2016, 19(2):337-354.
12. Charalampous K, Gasteratos A. On-line deep learning method for action recognition. Pattern Analysis & Applications, 2021, 44(10):33-38.
13. Moussa MM, Hamayed E, Fayek MB, EL Nemr HA. An enhanced method for human action recognition. Journal of Advanced Research, 2015, 6(2):163-169.
14. Vajda T. Action recognition based on Fast Dynamic-Time warping method5th International Conference on Intelligent Computer Communication and Processing. IEEE, 2021, 1043(2):022032 12pp.
15. Zhang J, Liu Z. A Method of Abnormal Action Recognition in Variable Scenarios. Journal of Image and Graphics, 2009, 14(10):2097-2101.
16. Pei X, Fan H, Tang Y. Action recognition method of spatio-temporal feature fusion deep learning network. Infrared and Laser Engineering, 2015, 47(2):203007.
17. Estevam V, Laroca R, Menotti D, et al. Tell me what you see: A zero-shot action recognition method based on natural language descriptions. Multimedia Tools and Applications, 2021, 83: 28147-28173.
18. Dong X, Tan L, Zhou L, et al. An Action Recognition Method Based on Deformable Convolution Network. Journal of Physics Conference Series, 2021, 11(10):15-23.
19. Choi Y., Tang J., Jang S., Kim S. User Customizable Hit Action Recognition Method Using Kinect. Journal of Korea Multimedia Society, 2015, 18(4):557-565
20. He W, Liu B, Xiao Y. Multi-View Action Recognition Method Based on Regularized Extreme Learning Machine. 2017 IEEE International Conference on Computational Science and Engineering (CSE) and IEEE International Conference on Embedded and Ubiquitous Computing (EUC), Guangzhou, China, 2017, pp. 854-857.
21. Lu Y, Li Y, Yang S, et al. A Human Action Recognition Method Based on Tchebichef Moment Invariants and Temporal Templates. 2012 4th International Conference on Intelligent Human-Machine Systems and Cybernetics, Nanchang, China, 2012, pp. 76-79.
22. Liu Z, Miao Z, Huo Y. A realtime human action recognition method based on single view key poses in sports video. 6th International Conference on Wireless, Mobile and Multi-Media (ICWMMN 2015), Beijing, 2015, pp. 210-213.
23. Jin X. SEMG Action Recognition Method Based on WPKPCA and SVM. Application of Electronic Technique, 2017, 43(2):1532-1538.
24. Guo M., Wang Z. A Feature Extraction Method for Human Action Recognition Using Body-Worn Inertial Sensors. Proceedings of the 2015 IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2015:576-581.
25. Ruan R.Q., Liu X.Q., Wu X. Action Recognition Method for Multi-Joint Industrial Robots Based on End-Arm Vibration and BP Neural Network. Proceedings of the 2021 6th International Conference on Control and Robotics Engineering (ICCRE), 2021:971-976.
26. Kim H Y, Song Y K, Cho J S. Fight Action Recognition Method using AI Deep Learning. Journal of Institute of Control Robotics and Systems, 2021, 27(7):482-489.
27. Wang L., Zhang J., Pfab C., et al. A Temporal–Spatial Attention-Based Action Recognition Method for Intelligent Fault Diagnosis. ISA Transactions, 2021, 111:319-328.
28. Qiao Y., Zhou L., Wang Y., et al. Action Recognition Method and Apparatus. Chinese Patent CN101200349A, 2008.
29. Zhang S., Wang J., Liang Z., et al. Action Recognition Method, Apparatus and Device and Storage Medium. European Patent EP3907653A1, 2021.
30. Feng J., Zhang L. Research of Human Action Recognition Method Based on Random Dropout Convolutional Neural Network. Journal of Test and Measurement Technology, 2011, 58(3):346-357.