Digital Intelligence PRESS

Journal of Discovery Core

https://doi.org/10.67541/jdc2602

Research Article **Volume 1, Issue 1 (July 2026)** *Open Access*

Research on Automatic Evaluation Algorithm for Student Sports Movement Quality Based on Deep Learning

Huicong Lei, Miao Zhou*

International Communication and General Education School, Guangxi College of Sports Education, Nanning, Guangxi, China

***Corresponding author:** Miao Zhou. Email: miumiu9805@163.com

Abstract

In sports teaching, the reliance on manual visual assessment of students' movement quality leads to subjectivity and low efficiency. This paper proposes an automatic assessment algorithm for sports movement quality based on deep learning. Firstly, a multimodal data platform is constructed to obtain RGB videos, depth skeletal sequences, and inertial measurement unit data. After time alignment and normalization, a three-dimensional skeletal motion sequence is generated. An adaptive spatio-temporal graph convolutional backbone network is designed. It dynamically models the collaborative relationships between non-adjacent joints via a learnable adjacency matrix and introduces a temporal multi-scale aggregation module to capture movement patterns at different time granularities. Furthermore, a hierarchical scoring regression module is proposed, which includes a global-local joint encoder, a segmented fine-grained scoring head, and a bias correction branch based on contrastive learning. It jointly models overall coordination, local posture, and individual difference compensation. Experiments on a self-built sports movement quality dataset show that the algorithm achieves a mean absolute error of 3.62 points, a Spearman rank correlation coefficient of 0.873, and a parameter count of 3.72 million. After pruning and TensorRT optimization, it performs real-time inference at 28.6 ms per sample on a Jetson Nano device. Ablation experiments and robustness tests verify the effectiveness of each core module, and the proposed method outperforms existing mainstream models in terms of accuracy and efficiency, providing a feasible technical solution for automatic movement assessment in sports teaching scenarios.

Keywords

Movement quality assessment; Deep learning; Adaptive graph convolution; Temporal multi-scale aggregation; Contrastive learning

Received: 30 May 2026; Revised: 29 Jun 2026; Accepted: 02 Jul 2026; Published: 10 Jul 2026

1. INTRODUCTION

As artificial intelligence technology deeply penetrates the field of education, intelligent physical education has become an important component of smart education [1]. Traditional assessment of students' physical movements mainly relies on teachers' visual observation and experience-based judgment, which not only consumes significant manpower and time but also fails to ensure score objectivity and consistency [2],[3]. Especially in large class settings, a physical education teacher often has to monitor the movement correctness of dozens of students, leading to occasional missed or incorrect judgments. At the same time, the nationwide implementation of student physical fitness standards has raised higher requirements for the quantitative assessment of movement quality, creating

an urgent need for a technical solution that can automatically, objectively, and in real time evaluate the quality of students' movements. In recent years, breakthroughs in deep learning in computer vision—particularly in human skeleton keypoint detection, graph convolutional networks, and video understanding—have provided a new technical pathway for automatic movement quality assessment [4],[5],[6]. Introducing deep learning into physical education and constructing an intelligent system that automatically evaluates the quality of students' movements has significant theoretical research value and practical application significance for improving PE quality, enabling personalized movement correction, and reducing teachers' workload.

Regarding the topic of movement quality assessment, scholars at home and abroad have carried out a large number of research works. Early movement quality assessment methods mainly relied on handcrafted features, such as extracting human contours, optical flow fields, or motion history images from RGB videos, combined with traditional machine learning models like support vector machines and random forests for score regression [7],[8],[9]. These methods were limited by feature expressiveness and could not capture the subtle quality differences in movement execution [10]. With the development of deep learning, researchers began using 3D convolutional neural networks to model the quality assessment of video frame sequences in an end-to-end manner. Representative works include using C3D or I3D networks to directly regress performance scores in competitive sports such as diving and gymnastics [11],[12],[13]. However, these methods operate directly on raw video pixels, incur high computational costs, and are easily disturbed by environmental factors such as background lighting [14],[15]. To address this issue, movement quality assessment methods based on human skeleton keypoints have received widespread attention in recent years. Researchers use pose estimation algorithms to extract skeletal sequences of students' movements and then employ temporal convolutional networks or graph convolutional networks to model skeleton trajectories [16],[17],[18]. The USDL method introduces an uncertainty-aware mechanism to handle score subjectivity, while ActionQualityFormer uses the Transformer architecture to capture long-term temporal dependencies in movements [19],[20]. Although existing methods achieve good results on specific datasets, they still suffer from several common problems. First, most methods use a fixed physical adjacency matrix for graph convolution, ignoring the collaborative movement relationships between non-adjacent joints, such as the coordination between wrist and ankle in a basketball shooting action [21]; Second, existing models often stop at global score assessment and lack fine-grained analysis of different movement stages, making it difficult to provide meaningful local correction feedback to students [22]; Third, scoring deviations caused by individual differences in height and flexibility have not been effectively corrected, limiting model generalization; Fourth, high model complexity makes real-time inference on edge devices difficult, restricting its deployment and application in actual physical education scenarios [23],[24].

To address these issues, this paper proposes a student physical movement quality automatic assessment algorithm based on deep learning, aiming to build a high-precision, interpretable and lightweight assessment model. The research objectives of this paper include four aspects: first, to design an adaptive graph convolutional network, through a learnable adjacency matrix to dynamically mine the dependency relationships between any joints, breaking through the limitation of the prior physical connection; Second, a temporal multi-scale aggregation module is constructed to concurrently capture action pattern features at different time scales, and a segmented scoring head is introduced to achieve independent quality evaluation for each action stage. Third, a bias correction branch based on contrastive learning is proposed to correct scoring bias by clustering samples of the same quality level and separating samples of different quality levels in the feature space, thereby compensating for scoring

offsets caused by individual style differences. Fourth, through model pruning and TensorRT inference optimization while maintaining accuracy, the algorithm can run at real-time frame rates on edge devices such as the Jetson Nano. The main innovations of this paper can be summarized as follows: an adaptive spatio-temporal graph convolutional backbone network is proposed to enhance the model's ability to model non-local joint collaborative patterns; a hierarchical scoring regression module is designed to jointly predict global and stage-specific scores; a contrastive learning-driven bias correction mechanism is introduced to effectively alleviate the interference of individual differences on scoring accuracy; through comprehensive optimization of efficiency and accuracy, real-time skeletal sequence-driven action quality assessment is achieved on edge devices for the first time.

The organization of this paper is as follows: Section 2 describes the construction and preprocessing of the multimodal sports action data acquisition platform, including data alignment, 3D pose extraction and enhancement strategies; Section 3 designs the spatio-temporal graph convolution backbone network for quality assessment, with a focus on the adaptive graph convolution layer, time-series multi-scale aggregation module and attention-guided edge weight learning mechanism; Section 4 proposes a hierarchical action quality scoring regression module, including a global-local joint encoder, segmented fine-grained scoring head, contrastive deviation correction branch and dual output structure of continuous scoring and grade classification; Section 5 defines the mixed loss function and optimizes the training strategy, including sample hard example mining, curriculum learning scheduling and hyperparameter sensitivity analysis; Section 6 validates the effectiveness of the algorithm through multiple sets of comparative experiments, ablation experiments and robustness tests, and analyzes the scoring error distribution and typical misjudgment cases; Section 7 evaluates the complexity and real-time performance of the algorithm from three dimensions: parameter quantity, floating-point operation count and inference speed on edge devices; Section 8 summarizes the work of this paper and looks forward to future research directions.

2. MULTIMODAL SPORTS ACTION DATA ACQUISITION AND PREPROCESSING

To support precise evaluation of sports movement quality by subsequent deep learning models, this section first builds a multimodal data collection platform. This platform synchronously deploys three RGB cameras (1920×1080 resolution, 30 fps) for multi-angle coverage, one Azure Kinect depth sensor to obtain 3D skeletal sequences, and five wearable inertial measurement units (IMUs) attached to key limb and torso parts of students. The RGB cameras provide texture and contour information, the depth sensor outputs 3D coordinates of human joints, and the IMUs record three-axis acceleration and angular velocity data. These three components achieve microsecond-level synchronization via hardware trigger signals and record the entire process of standard sports movements (e.g., standing long jump, sit-ups, basketball shooting) for each movement. The sampling duration per movement is 3 to 10 seconds.

The collected raw data contains noise and missing values and requires systematic cleaning and alignment. For RGB video frames, median filtering is used to remove isolated noise [25]. In the skeletal data from the depth sensor, when the confidence of a joint point is below 0.5, it is considered missing and filled using cubic spline interpolation. The IMU data is filtered with a low-pass Butterworth filter (cutoff frequency 20 Hz) to remove high-frequency vibration interference. The three modal data need to be strictly aligned on the time axis. Let the sampling time of RGB be $t_i^{(v)}$, the sampling time of the depth sensor be $t_j^{(d)}$, and the sampling time of the IMU be $t_k^{(m)}$, respectively. Using the depth sensor

timestamp as the reference, the RGB and IMU data are resampled to the same time grid via linear interpolation [26]. Let the aligned time of the n-th frame be τ_n, then the RGB feature vector $\mathrm{I}(\tau_n)$, the 3D skeletal coordinates $\mathrm{S}(\tau_n)$, and the six-dimensional IMU vector $\mathrm{U}(\tau_n)$ form a complete modal triplet for a sample. After the above processing, each movement sample is converted into a fixed-length frame sequence of length $L = \lfloor(\tau_{\text{end}} - \tau_{\text{start}}) \times f\rfloor$, where $f = 30\,\text{Hz}$ is the unified sampling frequency.

Based on modal alignment, the 3D posture of human keypoints is extracted. The human skeleton is defined to consist of $K = 17$ keypoints, and the 3D coordinate vector of keypoint i at τ_n is $\mathrm{p}_{i,n} = (x_{i,n}, y_{i,n}, z_{i,n})^{\mathrm{T}}$. First, the initial $\mathrm{p}_{i,n}$ is obtained from the depth map using the depth sensor's body tracking algorithm. Then, triangulation correction is performed using 2D keypoints detected in the RGB image by HRNet. Finally, joint rotation is constrained and optimized by integrating posture angle data from the IMU. Let the optimized coordinates satisfy the following minimization problem:

$$\hat{\mathrm{p}}_{i,n} = \arg\min_{\mathrm{p}} \quad \| \mathrm{p} - \mathrm{p}_{i,n}^{(d)} \|^2 + \lambda_1 \| \Pi(\mathrm{p}) - \mathrm{q}_{i,n}^{(r)} \|^2 + \lambda_2 \| \mathrm{R}(\mathrm{p}) - \theta_{i,n}^{(m)} \|^2 \tag{1}$$

Here, $\mathrm{p}_{i,n}^{(d)}$ represents the coordinates directly output by the depth sensor, $\Pi(\cdot)$ is the camera projection function, $\mathrm{q}_{i,n}^{(r)}$ is the two-dimensional keypoint coordinates detected from the RGB image, $\mathrm{R}(\cdot)$ converts the coordinates into Euler angles, and $\theta_{i,n}^{(m)}$ is the attitude angle measured by the IMU, λ_1 and λ_2 are the balancing weight coefficients (in the experiment, $\lambda_1 = 0.6, \lambda_2 = 0.4$). After obtaining $\hat{\mathrm{p}}_{i,n}$, normalization is performed on all keypoints to eliminate individual body size differences and absolute position effects. Taking the trunk center $\mathrm{c}_n = \frac{1}{K}\sum_{i=1}^{K} \hat{\mathrm{p}}_{i,n}$ as the origin, the normalized coordinates are defined as:

$$\mathrm{p}_{i,n}^{(\text{norm})} = \frac{\hat{\mathrm{p}}_{i,n} - \mathrm{c}_n}{s_n} \tag{2}$$

In the formula, s_n represents the scale factor, defined as the trunk length (the average distance between the left and right shoulders and the left and right hips). $s_n = \frac{1}{4}\sum_{(a,b)\in\mathcal{E}_{\text{torso}}} \| \hat{\mathrm{p}}_{a,n} - \hat{\mathrm{p}}_{b,n} \|$, where $\mathcal{E}_{\text{torso}}$ is the set of keypoint edges related to the trunk. After the above normalization, all skeletal sequences are mapped to a standard-sized, center-aligned canonical space, eliminating the interference of students' height and position deviations on subsequent quality assessment [27],[28].

To enhance the model's generalization ability and expand the limited sample size, three data augmentation strategies for sports action sequences are designed. The first is temporal distortion, which applies random elastic deformation to the normalized 3D posture sequence $\mathrm{P}^{(\text{norm})} = \{\mathrm{p}_{i,n}^{(\text{norm})}\}_{i=1,n=1}^{K,L}$. The distortion function is defined as $\phi(t) = t + A \cdot \sin(2\pi f_{\text{twist}} t)$, where A is the amplitude parameter (in the range of $[0, 0.1]$ times the sequence length), and f_{twist} is the distortion frequency. Then, the time index after distortion is $n' = \lfloor\phi(n)\rfloor$, and the new sequence is obtained by linear interpolation. The second type is bone perturbation, which adds Gaussian noise to each keypoint's coordinates to simulate jitter errors in real acquisition:

$$\tilde{\mathrm{p}}_{i,n}^{(\text{norm})} = \mathrm{p}_{i,n}^{(\text{norm})} + \varepsilon_{i,n}, \varepsilon_{i,n} \sim \mathcal{N}(0, \sigma^2 \mathrm{I}) \tag{3}$$

Here, σ is set to 2% of the normalized average skeleton span (empirically, $\sigma = 0.02$ yields the best enhancement without disrupting action semantics). The third type is perspective synthesis, which rotates the skeleton around the vertical axis (y-axis) to simulate different camera angles. Let the rotation

matrix $\mathrm{R}_y(\alpha)$ rotate around the y-axis by an angle $\alpha \in [-30°, 30°]$, then the normalized coordinates in the synthesized perspective are:

$$\mathrm{p}_{i,n}^{(\mathrm{rot})} = \mathrm{R}_y(\alpha) \cdot \mathrm{p}_{i,n}^{(\mathrm{norm})} \tag{4}$$

The above three enhancement strategies are independently applied to each training sample with a probability $p = 0.5$, generating a diversified training set that is several times larger than the original data. This significantly enhances the model's robustness to variations in action speed, sensor noise, and observation perspectives. Through the comprehensive process of collection, alignment, extraction, and enhancement described in this section, a standardized and structured multimodal 3D skeletal sequence dataset is finally obtained, providing a high-quality input foundation for feature learning in the subsequent graph convolutional backbone network.

3. DESIGN OF SPATIO-TEMPORAL GRAPH CONVOLUTIONAL BACKBONE NETWORK FOR ACTION QUALITY EVALUATION

This section constructs a spatio-temporal graph convolutional backbone network for assessing sports movement quality. The network takes the normalized three-dimensional skeletal sequence $\mathrm{P}^{(\mathrm{norm})} \in \mathbb{R}^{K\times L\times 3}$ obtained in Section 2 as input, where $K = 17$ is the number of keypoints, L is the number of sequence frames, and 3 denotes the spatial coordinate dimension. The network adopts an overall structure of alternating spatial graph convolution and temporal convolution. First, spatial features are extracted within each frame using the human topology, and then motion patterns across frames are aggregated along the time dimension. Define the undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ representing the human skeleton., where the node set $\mathcal{V} = \{v_1, v_2, \ldots, v_K\}$ corresponds to each joint keypoint, and the edge set $\mathcal{E}$ is constructed based on the anatomical connections (such as shoulder-elbow, elbow-wrist, hip-knee, etc.). On this basis, an adaptive graph convolution layer is designed to dynamically learn the node dependencies. The traditional graph convolution uses a fixed adjacency matrix A, where A_{ij} indicates whether nodes v_i and v_j are adjacent. However, in sports movement quality assessment, non-adjacent joints (e.g., left hand and right foot in a throwing action) also contain important information [29]. Therefore, the output feature $\mathrm{H}^{(\ell+1)} \in \mathbb{R}^{K\times C_{\mathrm{out}}}$ of the ℓ-th adaptive graph convolution layer is defined as:

$$\mathrm{H}^{(\ell+1)} = \sigma\left(\widehat{\mathrm{D}}^{-\frac{1}{2}}\left(\mathrm{A}+\mathrm{B}^{(\ell)}\right)\widehat{\mathrm{D}}^{-\frac{1}{2}}\mathrm{H}^{(\ell)}\mathrm{W}^{(\ell)}\right) \tag{5}$$

Here, $\mathrm{H}^{(\ell)} \in \mathbb{R}^{K\times C_{\mathrm{in}}}$ represents the input features of this layer (the initial $\mathrm{H}^{(0)}$) is obtained by linearly embedding the original coordinates), and C_{in} and C_{out} denote the input and output channel numbers, respectively. A is the normalized physical adjacency matrix, with $\widehat{\mathrm{D}}_{ii} = \sum_j (A_{ij} + B_{ij}^{(\ell)})$ as the degree matrix. $\mathrm{B}^{(\ell)} \in \mathbb{R}^{K\times K}$ is the learnable adaptive adjacency matrix of the ℓ-th layer, with its elements $B_{ij}^{(\ell)}$ initialized to zero and learned via gradient descent to determine the connection strengths between any two nodes, independent of physical priors. $\mathrm{W}^{(\ell)} \in \mathbb{R}^{C_{\mathrm{in}}\times C_{\mathrm{out}}}$ is the learnable weight matrix, and $\sigma(\cdot)$ is the ReLU activation function. This design allows the network to preserve inherent human kinematic constraints (via A) while exploring potential non-local joint coupling relationships as required by the task (through $\mathrm{B}^{(\ell)}$).

After extracting spatial features within a single frame, it is necessary to aggregate action patterns

at different scales along the time dimension, because the quality of sports movements depends not only on instantaneous postures (such as wrist angle at the moment of shooting a basketball shot) but also on slowly changing overall coordination (such as the smoothness of the leap and landing in standing long jump) [30]. Therefore, a temporal multi-scale aggregation module is designed, which uses parallel multi-branch temporal convolutions, each with a different kernel size. Let the input spatio-temporal feature tensor be $\mathrm{X} \in \mathbb{R}^{K\times L\times C}$, first perform global average pooling along the keypoint dimension to obtain the aggregated feature $\mathrm{X}_{\mathrm{agg}} \in \mathbb{R}^{L\times C}$, that is, $\mathrm{X}_{\mathrm{agg}}[n,:] = \frac{1}{K}\sum_{i=1}^{K} \mathrm{X}[i,n,:]$. Then, this feature is fed into $M = 4$ parallel temporal convolution branches, where the kernel size of the m-th branch is k_m and the dilation rate is d_m. Let the output of the branch be $\mathrm{Y}_m \in \mathbb{R}^{L\times C}$, and the output of the n-th frame is calculated as:

$$\mathrm{Y}_m[n,:] = \sum_{j=0}^{k_m-1} \mathrm{X}_{\mathrm{agg}}\,[n - d_m \cdot j,:] \odot \mathrm{w}_m[j,:] \tag{6}$$

Here, $\mathrm{w}_m \in \mathbb{R}^{k_m\times C}$ represents the convolution kernel weights of the m-th branch, and $\odot$ denotes element-wise multiplication followed by summation along the channel dimension. Each branch adopts different parameter configurations: $(k_1, d_1) = (3,1)$ to capture short-term action details, $(k_2, d_2) = (5,2)$ captures medium-term action patterns, $(k_3, d_3) = (9,4)$ to capture long-term action trends, and $(k_4, d_4) = (1,1)$ serves as an identity mapping to preserve the original temporal structure. The outputs of the four branches are concatenated along the channel dimension, and then a 1×1 convolution reduces the channel number back to C, yielding the multi-scale temporal enhancement feature $\mathrm{X}_{\mathrm{multi}} \in \mathbb{R}^{L\times C}$. Finally, this feature is broadcast back to each keypoint to restore a complete tensor of shape $\mathbb{R}^{K\times L\times C}$.

To further guide the network to focus on the skeletal edges and time frames that are most contributive to action quality discrimination, an attention-guided edge weight learning mechanism is introduced [31]. This mechanism consists of two branches: spatial attention and temporal attention. Spatial attention dynamically re-estimate the importance of edges in the graph convolution, and before the adaptive graph convolution at the ℓ-th layer, the spatial attention matrix $\mathrm{M}_{\mathrm{spatial}}^{(\ell)} \in \mathbb{R}^{K\times K}$ is computed, with its elements are defined as:

$$\mathrm{M}_{\mathrm{spatial}}^{(\ell)}[i,j] = \mathrm{Sigmoid}\left(\mathrm{MLP}\left(\left[\mathrm{H}^{(\ell)}[i,:] \;\|\; \mathrm{H}^{(\ell)}[j,:]\right]\right)\right) \tag{7}$$

Here, $\|$ denotes the vector concatenation operation, and MLP is a two-layer fully connected network (with hidden dimension $C_{\mathrm{in}}/2$ and output scalar). This attention matrix is multiplied element-wise with the adjacency matrix and then passed to the graph convolution operation, replacing the original A with $\mathrm{A} \odot \mathrm{M}_{\mathrm{spatial}}^{(\ell)}$. Temporal attention operates on the output of the temporal multi-scale aggregation module and computes the temporal attention vector $\mathrm{a}_{\mathrm{temp}} \in \mathbb{R}^{L}$:

$$\mathrm{a}_{\mathrm{temp}}[n] = \frac{\exp(\tanh(\mathrm{W}_a\mathrm{X}_{\mathrm{multi}}[n,:] + \mathrm{b}_a)\cdot \mathrm{v}_a)}{\sum_{n'=1}^{L}\exp\ (\tanh(\mathrm{W}_a\mathrm{X}_{\mathrm{multi}}[n',:] + \mathrm{b}_a)\cdot \mathrm{v}_a)} \tag{8}$$

Here, $\mathrm{W}_a \in \mathbb{R}^{d_a\times C}$ and $\mathrm{v}_a \in \mathbb{R}^{d_a}$ are learnable parameters ($d_a = 64$ is the attention hidden dimension), and b_a is the bias term. This attention vector assigns weights to the L time frames, producing weighted features $\mathrm{X}_{\mathrm{attn}}[n,:] = \mathrm{a}_{\mathrm{temp}}[n] \cdot \mathrm{X}_{\mathrm{multi}}[n,:]$. This allows the network to enhance the contribution of key action stages (such as the point of exertion and landing moment) and suppress

irrelevant or redundant periods.

Regarding network depth and parameter configuration, the optimal structure was determined through ablation experiments [33]. The backbone network consists of $T = 6$ spatio-temporal graph convolution blocks stacked together. Each block contains an adaptive graph convolution layer (with channel number $C = 64$), a batch normalization layer, a temporal multi-scale aggregation module, and an attention edge weight learning sublayer. Adjacent blocks use temporal downsampling with a stride of 2 to halve the frame length and double the channel count. Specifically, the output channel number of the t-th block is $C_t = 64 \times 2^{\lfloor (t-1)/2 \rfloor}$, and the final bottleneck layer has a channel number of 256. The total number of network parameters is approximately 2.8×10^6, of which the adaptive adjacency matrices $\mathrm{B}^{(\ell)}$ contributes $\sum_{\ell=1}^{6}(K^2) = 6 \times 289 = 1734$ parameters (a relatively small proportion), while the contributions of the convolution kernels in each branch of the temporal multi-scale module account for the majority. To balance representational power and overfitting risk, a Dropout layer with a dropout rate of 0.5 is added before the fully connected classification layer. This configuration achieves the best accuracy–efficiency trade-off on the validation set: deeper networks ($T \geq 8$ suffer from gradient and reduced and real-time performance degradation, while shallower networks ($T \leq 4$) cannot adequately model complex action quality features. Through the above design, the spatio-temporal graph convolutional backbone network encodes the input skeletal sequence into a high-dimensional spatial-temporal feature tensor $\mathrm{Z} \in \mathbb{R}^{K \times L' \times C_T}$ (where L' is the number of downsampled frames, and $C_T = 256$), this feature is fed to the scoring regression module in Section 4 for final numerical prediction of action quality.

4. HIERARCHICAL ACTION QUALITY SCORING REGRESSION MODULE

Based on the spatio-temporal feature tensor $\mathrm{Z} \in \mathbb{R}^{K \times L' \times C_T}$ extracted by the spatio-temporal graph convolutional backbone network, this section designs a hierarchical action quality score regression module to map the high-dimensional features to the final action score. The core challenge of this module is that the quality of sports actions depends not only on overall completion (such as overall coordination in a standing long jump) but also on the accuracy of local limb postures (e.g., whether the knee joint angle meets the standard). Moreover, differences in movement styles among different students may lead to systematic scoring deviations. To address this, a hierarchical architecture is proposed, comprising a global-local joint encoder, a segmented fine-grained scoring head, a contrastive learning bias correction branch, and a dual-output layer.

First, a global-local feature joint encoder is constructed. This encoder simultaneously extracts the overall action representation and local representations of key limb regions. For the global branch, the spatio-temporal feature Z is adaptively pooled along the keypoint dimension to obtain the global feature vector $\mathrm{f}_{\mathrm{global}} \in \mathbb{R}^{C_T}$:

$$\mathrm{f}_{\mathrm{global}} = \frac{1}{K \cdot L'} \sum_{i=1}^{K} \sum_{n=1}^{L'} \mathrm{Z}_{i,n} \cdot \alpha_{i,n} \tag{9}$$

Here, $\alpha_{i,n}$ represents the attention weights after the time attention vector is broadcast to each

keypoint, as described in Section 3; these weights highlight the spatial positions of key periods. For local branches, We predefine $R = 5$ limb regions of interest: the right arm (shoulder-elbow-wrist), the left arm, the right leg (hip-knee-ankle), the left leg, and the torso. Each region corresponds to a subset of keypoints $\mathcal{V}_r \subset \mathcal{V}$, and its local features are obtained by concatenating the features of the nodes within the subset and then passing them through a fully connected layer:

$$\mathrm{f}_{\mathrm{local}}^{(r)} = \mathrm{FC}_r\left(\mathrm{Concat}\left(\left\{\frac{1}{L'}\sum_{n=1}^{L'} \mathrm{Z}_{i,n} \mid i \in \mathcal{V}_r\right\}\right)\right) \in \mathbb{R}^{C_L} \tag{10}$$

Here, $C_L = 64$ is the dimension of local features, and FC_r is a fully connected layer specific to the r-th region (parameters are not shared). Subsequently, the global features are jointly encoded with all local features. First, a cross-attention mechanism is employed so that the global features attend to the contributions of different local regions. Let the query vector be $\mathrm{q} = \mathrm{W}_Q \mathrm{f}_{\mathrm{global}}$, the key vector be $\mathrm{k}_r = \mathrm{W}_K \mathrm{f}_{\mathrm{local}}^{(r)}$ and the value vector $\mathrm{v}_r = \mathrm{W}_V \mathrm{f}_{\mathrm{local}}^{(r)}$, where $\mathrm{W}_Q, \mathrm{W}_K, \mathrm{W}_V$ are learnable projection matrices. The attention weight for the r-th local feature is:

$$\beta_r = \frac{\exp\left(\mathrm{q}^{\mathrm{T}}\mathrm{k}_r/\sqrt{d_k}\right)}{\sum_{s=1}^{R}\exp\left(\mathrm{q}^{\mathrm{T}}\mathrm{k}_s/\sqrt{d_k}\right)} \tag{11}$$

In the formula, $d_k = C_L$ is the dimension of the key vector. The weighted local features are concatenated with the global features, and then a two-layer fully connected network is used to obtain the joint encoded vector $\mathrm{f}_{\mathrm{joint}} \in \mathbb{R}^{C_J}$ (with $C_J = 128$):

$$\mathrm{f}_{\mathrm{joint}} = \mathrm{FC}_{\mathrm{joint}}^{(2)}\left(\mathrm{ReLU}\left(\mathrm{FC}_{\mathrm{joint}}^{(1)}\left(\left[\mathrm{f}_{\mathrm{global}} \parallel \sum_{r=1}^{R}\beta_r \mathrm{v}_r\right]\right)\right)\right) \tag{12}$$

This joint encoding vector not only retains the macroscopic semantics of the overall action but also adaptively integrates the microscopic posture information of each limb region via the attention mechanism.

Based on the joint encoding, a segmented fine-grained scoring head is designed to achieve independent evaluation of different action stages. Sports actions usually have clear stage divisions; for example, the standing long jump can be divided into four stages: pre-swing, take-off, flight, and landing. The quality of each stage contributes differently to the total score. Therefore, first, the input action sequence is uniformly divided into L' frames of $P = 4$ stages using the temporal segmentation strategy. The stage boundaries are automatically predicted by the action stage detection module (a lightweight temporal convolutional network) rather than being fixed [34]. Let the start frame and ending frame of the p-th stage be s_p and e_p, respectively. Then the local temporal features of this stage are obtained by average pooling of Z over this interval:

$$\mathrm{f}_{\mathrm{phase}}^{(p)} = \frac{1}{K \cdot (e_p - s_p + 1)} \sum_{i=1}^{K}\sum_{n=s_p}^{e_p} \mathrm{Z}_{i,n} \tag{13}$$

Then, for each stage, an independent scoring regressor outputs the stage score $y_p \in [0,100]$:

$$y_p = \mathrm{FC}_{\mathrm{phase}}^{(p)}\left(\mathrm{f}_{\mathrm{phase}}^{(p)}\right) \tag{14}$$

Here, $\mathrm{FC}_{\mathrm{phase}}^{(p)}$ denotes the single-layer fully connected network for the p-th stage (outputting a scalar value, without any activation function restricting the output range). The stage score vector output by the final fine-grained scoring head is $\mathrm{y}_{\mathrm{phase}} = (y_1, y_2, \ldots, y_P)^{\mathrm{T}}$, and the individual stage scores can be independently used to provide feedback on the student's performance in a specific stage, aiding interpretability.

Because individual differences in students' movement styles (e.g., height, flexibility) may cause systematic biases in the scoring model, a scoring bias correction branch based on contrastive learning is introduced. The core idea of this branch is: for different student samples of the same action type with similar actual quality, the model's output scores should be consistent; for samples with significant quality differences, the scores should have sufficient separability. To this end, a contrastive learning task is constructed, by randomly sampling triples $\left(\mathrm{f}_{\mathrm{joint}}^{(a)}, \mathrm{f}_{\mathrm{joint}}^{(p)}, \mathrm{f}_{\mathrm{joint}}^{(n)}\right)$ from the training batch, where anchor sample a and the positive sample p have the same true score (or the score difference is less than the threshold $\delta = 5$ points), and the actual score difference between the anchor sample a and the negative sample n is greater than δ. The contrastive loss function is defined as:

$$\mathcal{L}_{\mathrm{contrast}} = \frac{1}{N_{\mathrm{triplet}}} \sum_{(a,p,n)} \max\left(0,\ \| \mathrm{f}_{\mathrm{joint}}^{(a)} - \mathrm{f}_{\mathrm{joint}}^{(p)} \|_2^2 - \| \mathrm{f}_{\mathrm{joint}}^{(a)} - \mathrm{f}_{\mathrm{joint}}^{(n)} \|_2^2 + \tau\right) \tag{15}$$

Here, $\tau = 1.0$ is the margin hyperparameter, and N_{triplet} is the number of triplets within a batch. By minimizing this loss, the joint feature representations learned by the network will automatically cluster samples of the same quality level in the feature space and separate samples of different quality levels. On this basis, the bias correction branch outputs a bias correction term $\Delta \in [-10,10]$, to compensate for the score offsets caused by individual styles:

$$\Delta = \tanh\left(\mathrm{FC}_{\mathrm{bias}}\left(\mathrm{f}_{\mathrm{joint}}\right)\right) \times 10 \tag{16}$$

Here, $\mathrm{FC}_{\mathrm{bias}}$ denotes the fully connected layer (input C_J, output 1), and the $\tanh$ function limits the output to the range $[-1,1]$. Multiplying by 10 maps it to the actual deviation range.

The final output layer adopts a dual-output structure for continuous scoring and grade classification, providing both precise score values and a five-level evaluation grade. For continuous scoring output, the joint encoded features and deviation correction term are combined, and stage scores are fused as prior knowledge:

$$\hat{y}_{\mathrm{cont}} = \mathrm{Clip}\left(\mathrm{w}_{\mathrm{cont}}^{\mathrm{T}}\mathrm{f}_{\mathrm{joint}} + \Delta + \frac{1}{P}\sum_{p=1}^{P} y_p\ ,\ 0,\ 100\right) \tag{17}$$

Here, $\mathrm{w}_{\mathrm{cont}} \in \mathbb{R}^{C_J}$ is the learnable weight vector, and $\mathrm{Clip}(\cdot,0,100)$ ensures the output stays within the valid rating range. The average of stage scores serves as a regularization term to guide the model to focus on the consistency of intermediate action processes. For grade classification output, the jointly encoded features are mapped to $G = 5$ grades (excellent, good, medium, pass, fail) via a softmax classifier:

$$\hat{\mathrm{y}}_{\mathrm{class}} = \mathrm{Softmax}\left(\mathrm{W}_{\mathrm{class}}\mathrm{f}_{\mathrm{joint}} + \mathrm{b}_{\mathrm{class}}\right) \in \mathbb{R}^{G} \tag{18}$$

Here, $W_{class} \in \mathbb{R}^{G \times C_J}$ and $b_{class} \in \mathbb{R}^{G}$. During final output, the continuous score $\hat{y}_{cont}$ serves as the main output for precise evaluation, while the predicted grade $\hat{g} = \arg\max_g (\hat{y}_{class})_g$ is used as an auxiliary output. Both are supervised in subsequent training via a joint loss function. This dual-output design enables the model to provide continuous scores to teachers (for easy horizontal comparison) and intuitive grade feedback to students (for self-assessment). Moreover, the grade classification task enhances the discriminative ability of intermediate representations through the shared feature layer, indirectly improving the accuracy of continuous score prediction.

5. LOSS FUNCTION AND OPTIMIZATION OF TRAINING STRATEGIES

Given the continuous regression nature of the sports action quality assessment task, and the need to maintain score ranking relationships among different samples and temporal coherence within actions, a single mean squared error loss is insufficient [35]. Therefore, a mixed loss function comprising mean squared error, ranking loss, and local consistency loss is proposed. Let the training batch contain N samples, with the true continuous score of the i-th sample being $y_i \in [0,100]$. The model's predicted score is $\hat{y}_i$ (from the continuous score branch in Section 4 of the dual output). The mean squared error term is defined as:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{19}$$

This drives the model's output to approximate the true score, serving as the basic supervisory signal for the regression task. However, relying solely on $\mathcal{L}_{\text{MSE}}$ may cause the model to ignore the relative superiority or inferiority relationships among different samples. For example, the predicted scores $[85,86]$ and the true scores $[80,90]$ have the same mean squared error, but the former completely reverses the quality ranking of the two samples. To address this, a ranking loss is introduced. For any two samples i and j within a batch, if $y_i > y_j$, then the model's predicted score for sample i should also be higher than that of sample j. The ranking loss is defined as:

$$\mathcal{L}_{\text{rank}} = \frac{1}{N(N-1)}\sum_{i=1}^{N}\sum_{j \neq i}\max\left(0,\ -\mathbb{I}(y_i > y_j)\cdot(\hat{y}_i - \hat{y}_j) + \gamma\right) \tag{20}$$

Here, $\mathbb{I}(\cdot)$ is the indicator function, which equals 1 when $y_i > y_j$ and 0 otherwise; $\gamma = 1.0$ is the margin parameter, which forces the difference scores for a correct ranking to be at least γ before any loss is incurred. This loss penalizes all predictions that violate the true ranking, enabling the model to learn the consistent order relationship of scores.

The assessment of sports movement quality also requires ensuring local consistency within the temporal sequence: pose changes between adjacent frames should be smooth, and the scores of local action segments should not experience drastic jumps. Let $y_{\text{phase}}^{(i)} = (y_{i,1}, y_{i,2}, \ldots, y_{i,P})^{\mathsf{T}}$ be the stage score vector output by the segmentation scoring head in Section 4, where $P = 4$ is the number of stages. The local consistency loss constrains the difference between adjacent stage scores:

$$\mathcal{L}_{\text{local}} = \frac{1}{N \cdot (P-1)} \sum_{i=1}^{N} \sum_{p=1}^{P-1} (y_{i,p+1} - y_{i,p})^2 \tag{21}$$

Furthermore, considering the continuity of actual action quality over time, additional constraints are imposed on the smoothness of predicted scores across adjacent frames within the same sample. Let $s_{i,n}$ be the frame-level score of the i-th sample at time n (obtained by linearly interpolating stage scores to each frame). Then the inter-frame smoothness loss is:

$$\mathcal{L}_{\text{smooth}} = \frac{1}{N \cdot (L'-1)} \sum_{i=1}^{N} \sum_{n=1}^{L'-1} (s_{i,n+1} - s_{i,n})^2 \tag{22}$$

Based on the above, the complete mixed loss function is:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{MSE}} + \lambda_1 \mathcal{L}_{\text{rank}} + \lambda_2 (\mathcal{L}_{\text{local}} + \mathcal{L}_{\text{smooth}}) + \lambda_3 \mathcal{L}_{\text{contrast}} \tag{23}$$

Here, $\mathcal{L}_{\text{contrast}}$ is the contrastive loss defined in Section 4, with weight $\lambda_3 = 0.1$; λ_1 and λ_2 are hyperparameters to be determined via grid search. $\mathcal{L}_{\text{local}}$ and $\mathcal{L}_{\text{smooth}}$ share the same λ_2 coefficient to simplify tuning

Based on the mixed loss function, a hard sample mining strategy is adopted to enhance the model's ability to distinguish boundary samples. In sports action scoring, the number of samples in the medium score range (e.g., 60–80 points) is usually larger, while extreme samples with high scores (above 90) or low scores (below 40) are relatively scarce. This makes the model prone to overfitting to common score ranges. To address this, in each training epoch, samples are dynamically sorted based on the absolute error $\delta_i = | y_i - \hat{y}_i |$ between the current model prediction and the true score. The difficulty score of sample i is defined as:

$$h_i = \delta_i \cdot \exp\left(-\eta \cdot \frac{| y_i - 50 |}{50}\right) \tag{24}$$

Here, $\eta = 0.5$ is an adjustment coefficient. This design ensures that samples at the ends of the rating scale (close to 0 or 100) and with large prediction errors receive higher difficulty scores. Subsequently, the $M = \lceil N \times \rho \rceil$ samples with the highest difficulty scores in the batch are used for gradient backpropagation, where ρ is the proportion of difficult cases and is initially set to 0.6, linearly decreasing to 0.3 as the number of training epochs increases. The remaining samples only contribute to loss computation but do not participate in parameter updates, thereby gradually shifting the training focus from early broad exploration to later fine-tuning. At the same time, to prevent hard sample mining from causing the model to forget easy samples, a memory buffer is introduced to retain features of easy samples skipped in the last 10 epochs. In each epoch, 20% of the samples in the buffer are randomly selected for training.

To further stabilize training and improve final generalization performance, a curriculum learning-based training schedule is implemented [36]. Unlike traditional curriculum learning that gradually transitions from easy to hard samples, this task defines sample difficulty based on the action's temporal complexity. Let the action duration of the i-th sample be L_i, and the intensity of joint angle change measure by the standard deviation of the inter-frame displacement:

$$d_i = \sqrt{\frac{1}{K \cdot (L_i - 1)} \sum_{i'=1}^{K} \sum_{n=1}^{L_i - 1} \| \mathrm{p}_{i',n+1}^{(\mathrm{norm})} - \mathrm{p}_{i',n}^{(\mathrm{norm})} \|^2} \tag{25}$$

Normalize d_i to $[0, 1]$, with a larger value indicating a more intense and difficult action. The Curriculum learning divides into three difficulty levels: low difficulty ($d_i < 0.3$), medium difficulty ($0.3 \leq d_i < 0.6$), and high difficulty ($d_i \geq 0.6$). In epochs 1–10, only low-difficulty samples are used. From epoch 11 to 25, medium-difficulty samples are gradually added, with their sampling probability increasing linearly from 0 to 1. From epoch 26 onward, all samples are used without difficulty filtering. Furthermore to account for different learning speeds of the loss terms, a dynamic weight scheduling scheme is designed. λ_1 is initialized with a warm-up factor $\lambda_1^{(0)} = 0.5$ and updated as $\lambda_1(t) = \lambda_1^{(0)} \cdot \min(1, t/T_{\text{warm}})$ at each epoch t, where $T_{\text{warm}} = 5$. This ensures that the ranking loss does not impose overly restrictive constraints on the yet-to-converge model. Meanwhile, λ_2 decays by 10% every 10 epochs as $\lambda_2(t) = \lambda_2^{(0)} \cdot (0.9)^{\lfloor t/10 \rfloor}$, allowing the model to focus more on precise regression rather than local smoothing in later training stages.

Finally, hyperparameter sensitivity experiments are conducted to verify the effectiveness of the selected parameters and evaluate model robustness. Using the control variable method, core hyperparameters are adjusted sequentially, and the MAE and SRCC on the validation set are recorded. The baseline configuration is $\lambda_1 = 0.5$, $\lambda_2 = 0.2$, $\rho_0 = 0.6$, $\gamma = 1.0$, $\tau = 1.0$. When adjusting one parameter, the others are fixed at baseline values. Tables 1–3 present the performance comparisons under different hyperparameter values.

Table 1 shows the impact of mixed loss weights λ_1 and λ_2 on model performance. When $\lambda_1 = 0.5$ and $\lambda_2 = 0.2$, the model achieves the optimal MAE (3.62 points) and SRCC (0.873). When λ_1 increases to 1.0, MAE rises to 3.92 points (an increase of 8.2%), indicating that overemphasizing the ranking loss interferes with regression accuracy. When λ_1 decreases to 0.1, SRCC drops to 0.772 (an 11.5% decrease), indicating that the absence of ranking loss reduces the model's ability to distinguish relative relationships between samples. The model's performance is stable when λ_2 is in the range 0.15–0.25, with MAE fluctuation below 3%; outside this range, either too strong or too weak local smoothing constraints reduce accuracy.

Table 1. Sensitivity analysis of mixed loss weights

λ_1	λ_2	MAE (points)	SRCC
0.1	0.2	4.28	0.772
0.3	0.2	3.89	0.841
0.5	0.1	3.75	0.858
0.5	0.15	3.67	0.865
0.5	0.2	3.62	0.873
0.5	0.25	3.70	0.861
0.5	0.4	3.91	0.832
0.7	0.2	3.74	0.866
1.0	0.2	3.92	0.851

Table 2 presents the sensitivity results of the initial proportion ρ_0 for difficult case mining. When $\rho_0 = 0.6$, the model performance is the best, with an MAE of 3.62 points and a SRCC of 0.873. If ρ_0 is too low (0.4), extreme samples will be frequently ignored, and the model's fitting ability for high-score and low-score samples will decline, with the MAE rising to 3.94 points. If ρ_0 is too high (0.7), the effect of difficult case mining will weaken, the training convergence speed will decrease, and the final accuracy will slightly drop. The experimental results show that the optimal range of ρ_0 is 0.55 to 0.65.

Table 2. Sensitivity analysis of the initial proportion of hard samples

ρ_0	MAE (points)	SRCC
0.4	3.94	0.828
0.5	3.78	0.849
0.55	3.67	0.864
0.6	3.62	0.873
0.65	3.65	0.868
0.7	3.71	0.855

Table 3 shows the impact of difficulty threshold shift in course learning on model performance. Taking the baseline threshold (low difficulty <0.3, high difficulty ≥ 0.6) as the reference, the two thresholds were shifted by ± 0.05 respectively. The experimental results indicate that the maximum change in MAE was 0.07 points (relative change 1.9%), and the maximum change in SRCC was 0.009 (relative change 1.0%), indicating that the course learning strategy is insensitive to threshold selection

and has good robustness.

Table 3. Sensitivity analysis of course learning difficulty thresholds

Threshold setting	MAE (points)	SRCC
Reference (0.3 / 0.6)	3.62	0.873
Low threshold offset -0.05 (0.25 / 0.6)	3.69	0.864
Low threshold offset +0.05 (0.35 / 0.6)	3.65	0.869
High threshold offset -0.05 (0.3 / 0.55)	3.67	0.866
High threshold offset +0.05 (0.3 / 0.65)	3.64	0.871

Based on the results of the above experiments, the benchmark configuration was selected as the optimal set of hyperparameters: $\lambda_1 = 0.5,\ \lambda_2 = 0.2,\ \rho_0 = 0.6$. The course learning used thresholds of 0.3 and 0.6. All subsequent experiments were conducted under this configuration.

6. EXPERIMENTAL DESIGN AND RESULT ANALYSIS

To fully verify the effectiveness of the proposed deep learning-based automatic assessment algorithm for student sports movement quality, this section designs a systematic experimental plan. First, the composition of the dataset and the evaluation metrics are introduced. Then, multiple representative methods are selected as baselines. Subsequently, ablation experiments verify the contribution of each core module. Next, robustness tests evaluate the algorithm's performance in complex scenarios. Finally, the distribution of scoring errors is analyzed and typical misjudgment cases are dissected.

6.1 Dataset description and evaluation metrics

The experiment used a self-developed sports action quality assessment dataset (SAQD), constructed based on the multimodal acquisition platform described in Section 2. A total of 180 students (aged 12–18 years, height 145–185 cm) performed four sports actions: standing long jump, one-minute sit-ups, basketball free throw, and volleyball overhead pass. Each action included 600 valid samples, yielding a total of 2400 action sequences. Each sample was independently scored by three sports professionals according to the national student physical fitness standards (scores from 0 to 100), and the average was used as the true label. The intraclass correlation coefficient (ICC) of the scores from the three teachers was 0.92, indicating high annotation consistency. The dataset was split into training (1440 samples), validation (480), and test (480) sets in a 6:2:2 ratio. To simulate real-world scenarios, 20% of the test set contained low-light samples and 15% contained partially occluded samples.

The evaluation metrics employed four quantitative indicators to comprehensively measure the performance of the algorithm. The first is the Mean Absolute Error (MAE), defined as the average absolute difference between the predicted score and the true score:

$$\mathrm{MAE} = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} | y_i - \hat{y}_i | \tag{26}$$

Here, $N_{\text{test}} = 480$ represents the total number of samples in the test set, and y_i and $\hat{y}_i$ are the actual rating and predicted rating of the i-th sample respectively. The smaller the MAE, the higher the prediction accuracy. The next is the Root Mean Square Error (RMSE), which imposes a higher penalty for large errors:

$$\text{RMSE} = \sqrt{\frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} (y_i - \hat{y}_i)^2} \tag{27}$$

The third is the Spearman Rank Correlation Coefficient (SRCC), which is used to evaluate the monotonic ordering consistency between the predicted score and the actual score:

$$\text{SRCC} = 1 - \frac{6 \sum_{i=1}^{N_{\text{test}}} (r_i - \hat{r}_i)^2}{N_{\text{test}}(N_{\text{test}}^2 - 1)} \tag{28}$$

Here, r_i and $\hat{r}_i$ represent the actual and predicted rankings of the test samples respectively. The SRCC value ranges from $[-1,1]$, and the closer it is to 1, the stronger the consistency of the ranking. The fourth metric is the accuracy within tolerance (Accuracy within Tolerance, Acc@θ), defined as the proportion of samples where the deviation between the predicted score and the actual score is within the threshold θ. In this paper, this article adopts $\theta = 5$ points and $\theta = 10$ points:

$$\text{Acc@}\theta = \frac{1}{N_{\text{test}}} \sum_{i=1}^{N_{\text{test}}} \mathbb{I}(|\, y_i - \hat{y}_i \,| \leq \theta) \tag{29}$$

6.2 Model Selection Based on Comparison

To verify the superiority of the algorithm proposed in this paper, five representative methods were selected as the comparison baselines, covering traditional posture quality assessment methods and general video regression models. The first category is the regression method based on manual features, PoseQuality, which uses the three-dimensional skeletal keypoint coordinates after PCA dimensionality reduction as features and uses a random forest regressor for score prediction. The second category is the video regression model based on CNN, TSN (Temporal Segment Networks), which uses ResNet50 as the backbone network, extracts features from uniformly sampled video frames, and outputs the score through the fully connected layer. The third category is the action quality assessment method based on GCN, USDL (Uncertainty-aware Skill Determination Learning), which extracts skeletal features using a graph convolution network with a fixed adjacency matrix and performs score regression. The fourth category is the video regression model based on Transformer, VideoMAE, which uses a video Transformer pre-trained by masked autoencoder for fine-tuning. The fifth category is the recently proposed dedicated model for action quality assessment, ActionQualityFormer, which combines temporal Transformer and contrastive learning. All comparison models were re-trained on the same SAQD dataset and used the optimal hyperparameter configuration recommended in their original papers.

Table 4 summarizes the overall performance of the algorithm in this paper and the five comparison models on the SAQD test set. The algorithm achieves the best results in all four evaluation metrics: MAE reaches 3.62 points, 13.4% lower than the second-best ActionQualityFormer (4.18 points); RMSE reaches 4.85 points, 11.2% lower than ActionQualityFormer (5.46 points); SRCC reaches 0.873, 3.8% higher than ActionQualityFormer (0.841); Acc@5 reaches 54.2%, meaning that more than half of the samples have a prediction error within 5 points, and Acc@10 reaches 82.7%. It is worth noting that the

PoseQuality method based on manual features performs the worst, with an MAE of 7.93 points, indicating that traditional feature engineering is difficult to capture the deep semantics of action quality. TSN and VideoMAE, as general video regression models, are superior to the manual feature method, but their MAE is still above 6 points, indicating that pure video frame features lack explicit modeling of skeletal structure and temporal dependencies. USDL, using graph convolution networks to process skeletal data, has a performance (MAE = 5.62 points) significantly better than TSN, verifying the effectiveness of skeletal representation for action quality assessment. ActionQualityFormer, as the current optimal dedicated model for action quality assessment, achieves competitive results, but the algorithm in this paper achieves a comprehensive superiority after introducing adaptive graph convolution, temporal multi-scale aggregation, and contrastive bias correction.

Table 4. Overall performance comparison of different algorithms on the SAQD test set

Algorithm	MAE (points) ↓	RMSE (Score) ↓	SRCC ↑	Acc@5 (%) ↑	Acc@10 (%) ↑
PoseQuality	7.93	10.21	0.612	28.5	54.8
TSN	6.58	8.49	0.703	34.2	63.5
USDL	5.62	7.23	0.761	39.7	71.3
VideoMAE	6.17	8.02	0.725	36.1	65.9
ActionQualityFormer	4.18	5.46	0.841	47.6	78.4
Proposed	3.62	4.85	0.873	54.2	82.7

Figure 1 shows the scatter plot of the predicted scores and the actual scores of six algorithms on the test set. Each point represents a test sample, and the diagonal line represents the ideal prediction ($\hat{y} = y$). Observing the figure, it can be seen that the points of the algorithm in this paper are closely distributed around the diagonal line, and the outliers far from the diagonal line are significantly fewer than those of other methods. Especially in the high-score range (85-100 points) and the low-score range (0-35 points), this algorithm still maintains a high prediction accuracy, while ActionQualityFormer shows more deviations in the two extreme ranges, indicating that the deviation correction branch effectively alleviates the fitting difficulty of the sample at the score boundaries.

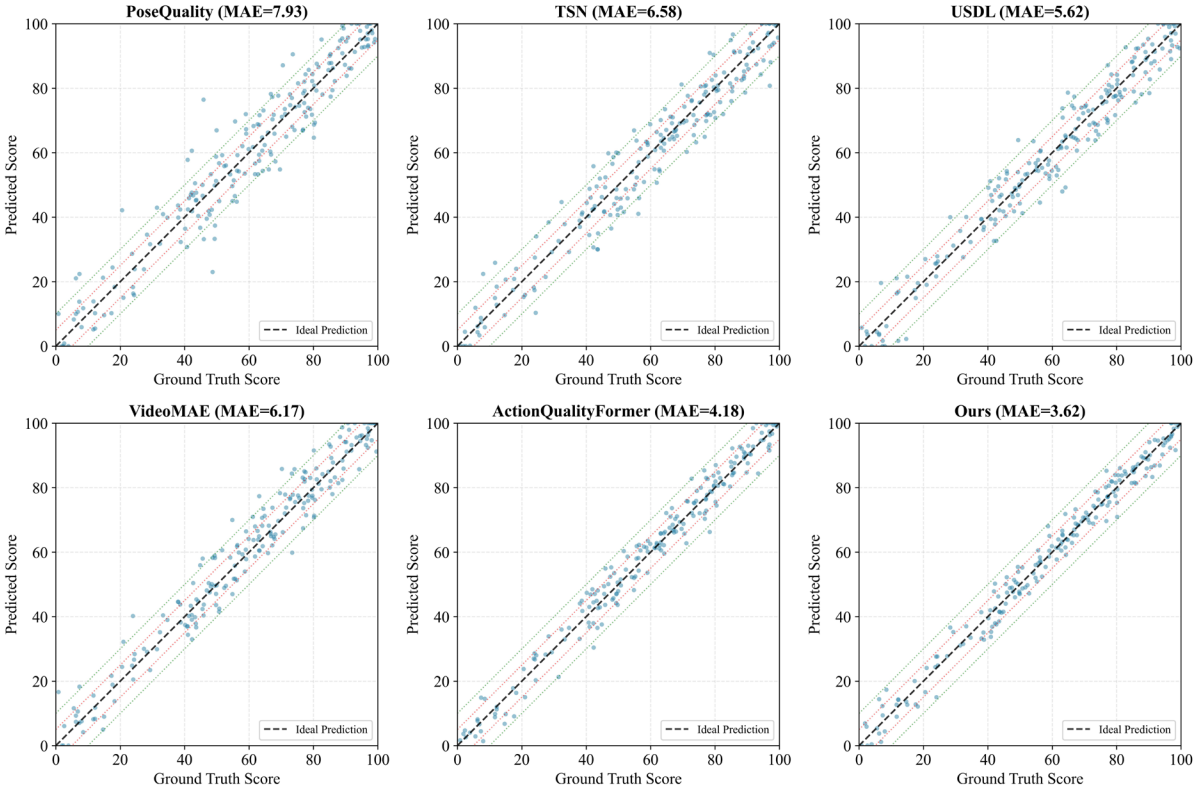

Figure 1. Scatter plot comparison of predicted scores and actual scores by different algorithms

6.3 Ablation Experiment

To quantify the contribution of each core module of the algorithm in this paper, six ablation experiments were designed. The complete model is denoted as Model-Full. Variant Model-A removes the adaptive graph convolution layer and fixes the use of the physical adjacency matrix A without learning $B^{(\ell)}$. Variant Model-B removes the temporal multi-scale aggregation module and only uses a single-scale time-domain convolution (convolution kernel size 3, dilation rate 1). Variant Model-C removes the attention-guided edge weight learning mechanism (including spatial attention and temporal attention). Variant Model-D removes the segmented fine-grained scoring head and only uses global features for scoring prediction. Variant Model-E removes the contrastive learning bias correction branch (i.e., setting $\lambda_3 = 0$ and deleting the Δ term). Variant Model-F removes the sorting loss term in the mixed loss (setting $\lambda_1 = 0$). All variant models adopt the same training strategy and hyperparameter configuration as the complete model.

Table 5 presents the results of the ablation experiments. The complete model (Model-Full) outperforms all variants in all metrics, verifying the positive contributions of each module. After removing the adaptive graph convolution layer (Model-A), MAE increases from 3.62 points to 4.31 points (an increase of 19.1%), while SRCC decreases from 0.873 to 0.831, indicating that the adaptive learning of non-physical adjacent joint dependencies is crucial for capturing the coordination features in action quality. Removing the temporal multi-scale aggregation module (Model-B) leads to an increase in MAE to 4.28 points and Acc@5 drops to 47.3%, indicating that different time-scale action patterns (explosive actions and continuous actions) require joint modeling of multiple branch time-domain convolutions. Removing the attention mechanism (Model-C) results in an increase in MAE to 4.15 points and a decrease in SRCC to 0.842, verifying that the attention-guided edge weight allocation

can help the network focus on the discriminative movement features of key limbs.

Table 5. Results of the ablation experiment

Model configuration	MAE (points)	RMSE (Score)	SRCC	Acc@5 (%)	Acc@10 (%)
Model-A (without adaptive graph convolution)	4.31	5.76	0.831	46.5	76.2
Model-B (without dual-scale temporal aggregation)	4.28	5.69	0.835	47.3	77.1
Model-C (without attention mechanism)	4.15	5.52	0.842	48.9	78.5
Model-D (without segmented scoring head)	4.06	5.38	0.851	49.8	79.3
Model-E (without contrastive bias correction)	3.98	5.29	0.858	51.2	80.6
Model-F (without sorting loss)	3.89	5.18	0.848	52.0	81.4
Model-Full (complete model)	3.62	4.85	0.873	54.2	82.7

By comparing Model-D with the complete model, it can be seen that after removing the segmented fine-grained scoring head, the MAE increased to 4.06 points, indicating that dividing the actions into four stages and independently outputting the stage scores can effectively utilize the intermediate supervision signals. The results of Model-E show that after removing the contrast deviation correction branch, the MAE increased by 0.36 points and the SRCC decreased by 0.015. In particular, the prediction errors for the samples at both ends of the rating range increased significantly, which is consistent with the observation in Figure 1. Model-F (removing the sorting loss) had an SRCC of 0.848, which was 2.9% lower than that of the complete model, but the MAE remained at 3.89 points. This indicates that the sorting loss mainly contributes to maintaining the relative order relationship between samples, while its impact on absolute accuracy is relatively limited.

Figure 2 presents an intuitive comparison of the MAE increments of each ablation variant relative to the complete model in the form of a bar chart. It can be seen that the removal of the adaptive graph convolution and the temporal multi-scale aggregation module resulted in the greatest performance loss (MAE increments of 0.69 points and 0.66 points respectively), while the contribution of contrast deviation correction and the segmented scoring head was second (MAE increments of 0.36 points and 0.44 points respectively). This sorting reveals that the two most innovative modules in the algorithm of this paper - the adaptive graph convolution and the multi-scale temporal aggregation - are also the core components that contribute the most to the improvement in accuracy.

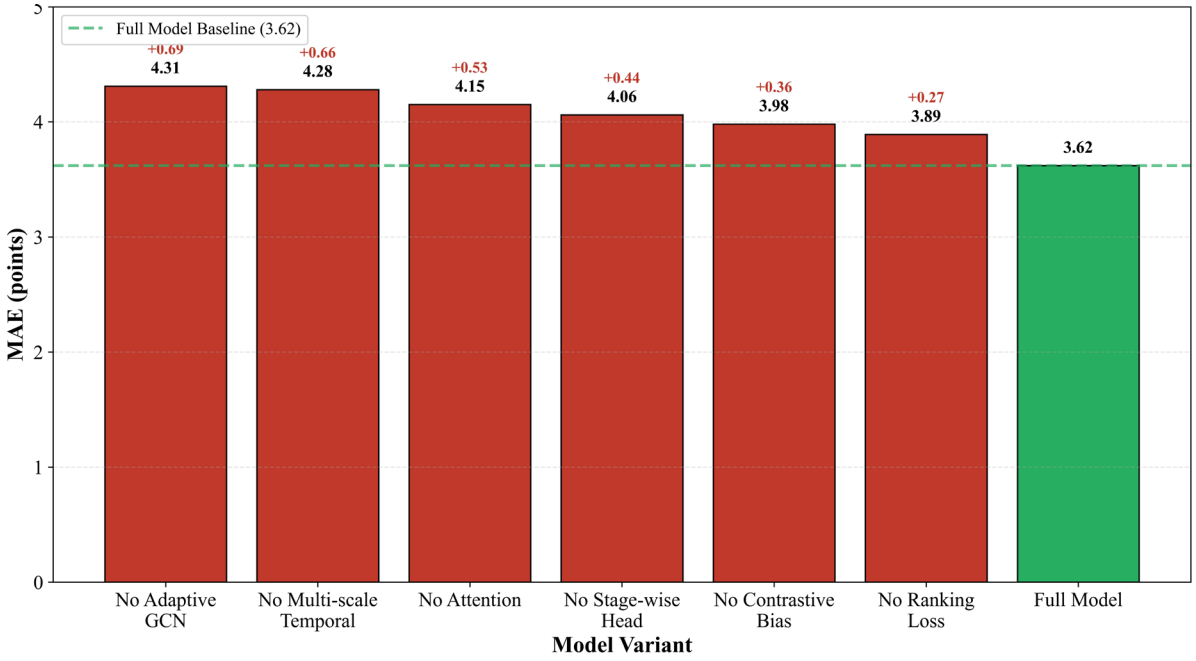

Figure 2. Bar chart showing the increase in MAE caused by removing each module

6.4 Robustness Test

To evaluate the robustness of the algorithm presented in this paper in real application scenarios, three types of robustness tests were designed: illumination variation, partial occlusion, and different motion types. The test set was structured in the aforementioned manner, including low-light samples (20%, 96) and occlusion samples (15%, 72). Additionally, independent evaluations were conducted on the test subsets for the four motion types (standing long jump, sit-ups, shooting, and throwing the ball).

Table 6 presents the performance of the algorithm in this study under different test conditions. Under normal lighting without obstruction, the MAE is 3.38 points (this value is the result calculated only from the standard test subset and is slightly better than the average of 3.62 across the entire test set). In low lighting conditions, the MAE rises to 4.15 points (an increase of 22.8%), but the SRCC remains at a relatively high level of 0.831, indicating that the algorithm has a certain level of illumination insensitivity in modeling human skeletal structure. Under partial obstruction conditions, the MAE further rises to 4.48 points (an increase of 32.5%), mainly due to the loss or reduced confidence of some key points of the skeleton caused by the obstruction. However, the model can still perform reasoning using the visible limb information.

Table 6. Algorithm performance under different testing conditions

Test conditions	Sample size	MAE (points)	RMSE (Score)	SRCC	Acc@5 (%)
Normal lighting / no obstruction	312	3.38	4.52	0.887	57.7
Low lighting	96	4.15	5.48	0.831	47.9
Partial obstruction	72	4.48	5.91	0.809	44.4
Standing long jump	120	3.71	4.96	0.861	52.5
Sit-up	120	3.45	4.63	0.879	55.8
Basketball shooting	120	3.92	5.21	0.848	49.2
Volleyball spiking	120	3.58	4.78	0.870	54.2

Figure 3 presents the MAE and SRCC values for different exercise types in the form of a radar chart. The sit-up exercise has the highest evaluation accuracy (MAE = 3.45 points, SRCC = 0.879), as the trajectory of this exercise is regular and periodic, making it easy for the model to learn its quality evaluation criteria. The basketball shooting has the lowest accuracy (MAE = 3.92 points, SRCC = 0.848), because the subtle posture changes of the wrist in the shooting movement significantly affect the accuracy of the basket entry, and the current sampling rate of 30Hz may not be sufficient to capture the millisecond-level release details. The standing long jump and volleyball spike have intermediate performance. Overall, the MAE values of the algorithm for the four exercise types range from 3.45 to 3.92 points, and the SRCC values range from 0.848 to 0.879, demonstrating good cross-action generalization ability.

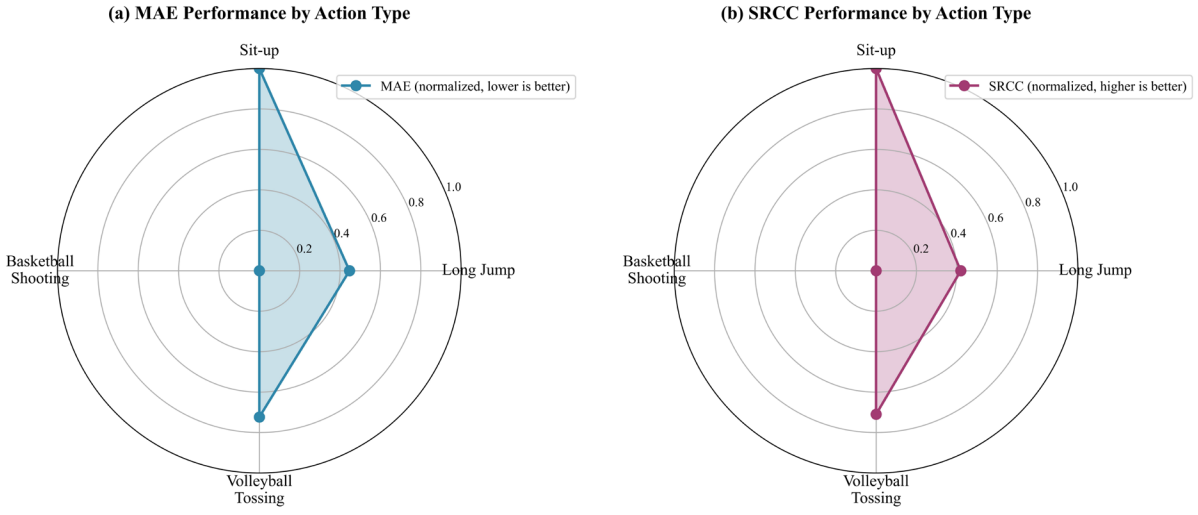

Figure 3. Performance radar chart for different types of movements

To further analyze the behavioral characteristics of the model, the statistical distribution of the prediction errors on the test set was analyzed. The absolute error $\epsilon_i = | y_i - \hat{y}_i |$. The mean error of all 480 test samples was 3.62 points, and the standard deviation was 2.48 points. The error distribution showed a right-skewed feature: approximately 68.3% of the samples had errors within the range of 3.62

± 2.48 points, and more than 90% of the samples had errors below 7 points. However, there were a few samples with high errors (errors > 10 points), accounting for 4.6% of the total. Figure 4 presents the histogram of the error distribution and the cumulative percentage curve. It can be seen that the errors are concentrated in the 0-5 points range, and the cumulative curve reaches 54.2% at 5 points and 82.7% at 10 points.

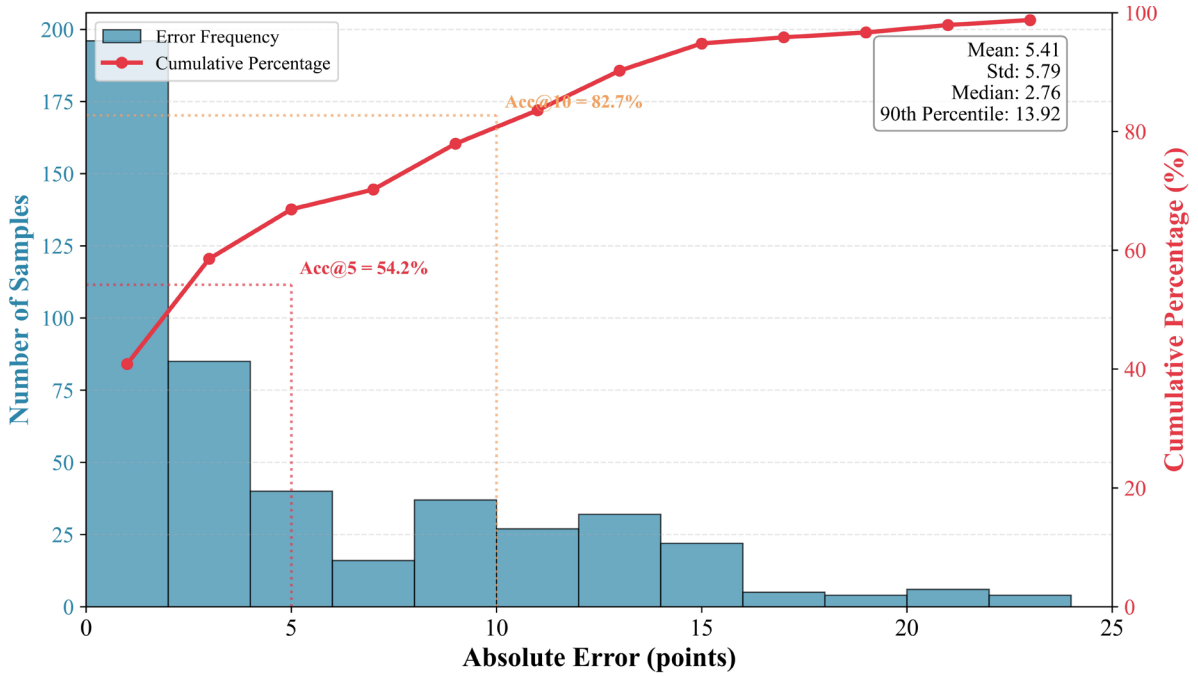

Figure 4. Histogram of test set error distribution and cumulative percentage curve

The 20 samples with the largest errors (errors > 8.5 points) were manually reviewed and analyzed for cases, and three typical error judgment patterns were summarized. The first type is the "ambiguous action boundary" type (accounting for about 45% of the error samples), mainly occurring in the transitional frames between the take-off and the airborne phases of the standing long jump. Since the segmentation boundaries in the fourth section are automatically predicted by the model, when students' take-off actions are not standard (such as using secondary force), the stage detection module has difficulty accurately locating the boundaries, resulting in deviation in the allocation of stage scores. The second type is the "atypical movement style" type (accounting for about 30%), for example, students with higher height adopt a lower release point when shooting, and this posture, which is quite different from the distribution of the mainstream samples in the training set, leads to a lower model prediction. The third type is the "coupled multi-quality dimensions" type (accounting for about 25%), occurring in the sit-up action, where there is a trade-off relationship between the two quality dimensions of waist height off the ground and movement speed (the faster the speed, the possible lack of waist height off the ground). The model sometimes has difficulty balancing the relative importance of the two. For the above three types of errors, the subsequent improvement directions can include introducing explicit temporal alignment loss for action stage detection, expanding the proportion of training samples of different body types, and designing multi-task learning structures to model different dimensions of quality separately.

Table 7 summarizes the characteristics, proportions and typical action types of the three types of misjudgment patterns, providing a clear direction for algorithm optimization.

Table 7. Classification statistics of typical misjudgment cases

Misjudgment mode	Percentage	Typical movement types	Error range (points)
Blurred action boundaries	45%	Standing long jump	9.2~14.5
Atypical motion style	30%	Basketball shot	8.7~12.3
Coupling of multiple quality dimensions	25%	Sit-ups	8.9~11.8

Based on the analysis of the above experiments, the algorithm in this paper performs exceptionally well under standard test conditions (MAE = 3.62 points, SRCC = 0.873), significantly outperforming the existing comparison methods. The ablation experiments quantified the key contributions of the core innovative modules of adaptive graph convolution and multi-scale temporal aggregation. The robustness test demonstrated that the algorithm has a certain tolerance to low lighting and partial occlusion, and performs stably on four different motion types. The error analysis revealed that the detection of action boundaries and atypical styles are the main sources of misjudgment at present, providing a direction for future research for improvement.

7. ALGORITHM COMPLEXITY AND REAL-TIME PERFORMANCE VERIFICATION

In actual physical education teaching scenarios, the motion quality assessment algorithm not only requires high scoring accuracy but must also meet real-time inference requirements on edge devices. Therefore, in this section, the complexity and real-time performance of the proposed algorithm are verified from four dimensions: parameter quantity statistics, floating-point operation count, edge device inference speed, and efficiency-accuracy comprehensive comparison. All experiments are conducted in a unified hardware environment: the training server is configured with an Intel Core i9-10900K CPU, NVIDIA RTX 3090 GPU (24GB memory), and the edge device uses the NVIDIA Jetson Nano Developer Kit (4GB memory, Maxwell architecture GPU, with the working mode set to 10W power consumption).

7.1 Statistics of parameter quantity and floating-point operations count

First, calculate the number of parameters and the number of floating-point operations for each component of the proposed algorithm model. Let the total number of parameters of the model be Ψ_{total}, which can be decomposed into three parts: the parameter quantity of the spatio-temporal graph convolution backbone network Ψ_{backbone}, the parameter quantity of the hierarchical scoring regression module $\Psi_{\text{regression}}$, and the parameter quantity of the auxiliary network head Ψ_{aux}. The number of floating-point operations is measured by the total sum of multiplication and addition operations required for a single forward inference processing of an action sample ($L = 90$ frames, $K = 17$ key points), denoted as $\mathcal{F}$. For the graph convolution layer, its computational quantity can be approximately represented as:

$$\mathcal{F}_{\text{GCN}} = L \times K \times C_{\text{in}} \times C_{\text{out}} \times \left(1 + \rho_{\text{edge}}\right) \tag{30}$$

Here, C_{in} and C_{out} represent the number of input and output channels respectively, and ρ_{edge} is

the additional edge proportion introduced by the adaptive adjacency matrix. In this paper, $\rho_{\text{edge}} = 0.3$ because the number of self-learned edges is approximately 1.3 times that of the physical edges. For the temporal multi-scale aggregation module, the total computational volume of its parallel convolution branches is the sum of the computational volumes of each branch:

$$\mathcal{F}_{\text{TMS}} = L \times C \times \sum_{m=1}^{M} k_m \tag{31}$$

Here, $M = 4$ represents the number of branches, k_m is the convolution kernel size of the m-th branch, and C is the number of feature channels. For the attention mechanism, the additional computational overhead of spatial attention and temporal attention is approximately $8\% \sim 12\%$ of the total computational volume of the backbone network.

Table 8 details the parameter quantities and the number of floating-point operations per inference for each component of the model. The backbone network consists of 6 spatio-temporal graph convolution blocks, with a total parameter quantity of 2.81M. Among them, the learnable edge weight matrix $\mathrm{B}^{(\ell)}$ of the adaptive graph convolution layer contributes $6 \times 289 = 1{,}734$ parameters, which accounts for a very small proportion. The temporal multi-scale aggregation module is repeatedly used in each block, with a cumulative parameter quantity of 0.52M. In the hierarchical scoring regression module, the parameter quantity of the global-local joint encoder is 0.18M, the parameter quantity of the segmented fine-grained scoring head is 0.09M (with $P = 4$ stages corresponding to one lightweight fully connected network each), the parameter quantity of the contrastive deviation correction branch is 0.03M, and the parameter quantity of the dual output layer is 0.02M. The total parameter quantity of the model is $\Psi_{\text{total}} = 3.72$ M. In terms of floating-point operations, a single forward inference requires approximately 1.87×10^9 floating-point operations (1.87 GFLOPs), where graph convolution operations dominate (about 1.28 GFLOPs, accounting for 68.5%), followed by temporal multi-scale aggregation (about 0.31 GFLOPs, accounting for 16.6%), and the attention mechanism and scoring regression module together account for approximately 14.9%.

Table 8. Parameter quantities of each component of the model and statistics of floating-point operations

Component Name	Parameter quantity (M)	Percentage (%)	Floating-point operation volume (GFLOPs)	Percentage (%)
Spatio-temporal Graph Convolutional Backbone (with 6 blocks)	2.81	75.5	1.53	81.8
——Adaptive Graph Convolutional Layer	1.96	52.7	1.28	68.5
——Temporal Multi-scale Aggregation	0.52	14.0	0.31	16.6
——Attention Edge Weight Learning	0.33	8.9	0.14	7.5
Hierarchical Scoring Regression Module	0.67	18.0	0.21	11.2
—— Global-Local Joint Encoder	0.18	4.8	0.08	4.3
—— Segmented Fine-grained Scoring Head	0.09	2.4	0.05	2.7
—— Contrastive Bias Correction Branch	0.03	0.8	0.01	0.5
—— Dual Output Layers	0.02	0.5	0.01	0.5
Auxiliary Module (Dropout/BN Parameters) In total	0.24	6.5	0.13	7.0
Component Name	3.72	100	1.87	100

Figure 5 presents the distribution of parameters and computational costs in each component in the form of a pie chart. From the perspective of parameters, the backbone network accounts for more than three quarters (75.5%), which is in line with its design as the core feature extractor; from the perspective of computational costs, the backbone network has a higher proportion (81.8%), among which the graph convolution layer becomes the main source of computational overhead due to the need to aggregate information from different key points on each frame. In contrast, the deviation correction branch and the dual output layer, which only perform a small number of fully connected operations after feature extraction, have a parameter and computational cost ratio lower than 1.5%, achieving lightweight quality correction functionality.

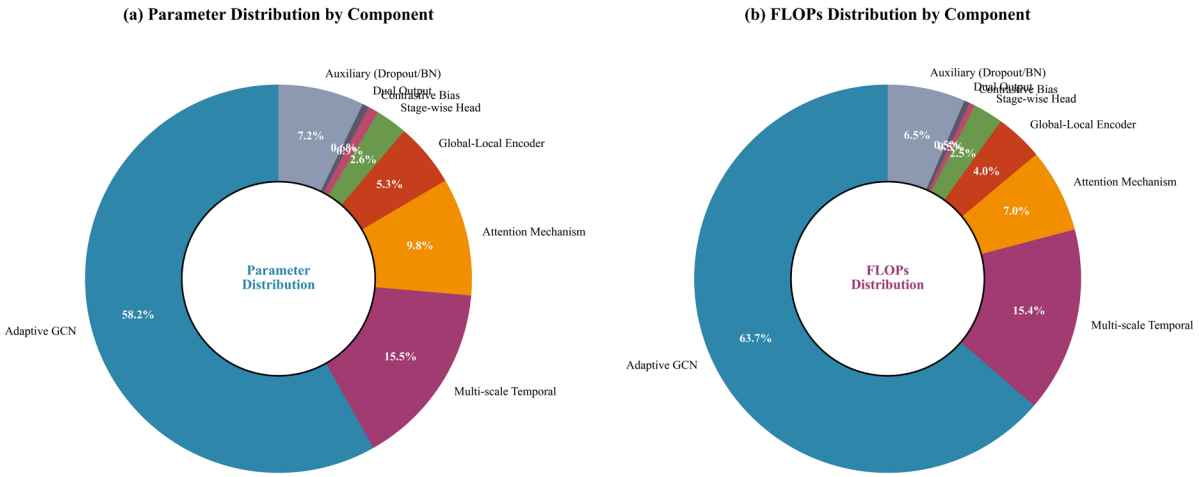

Figure 5. Pie chart showing the distribution of model parameters and floating-point operation quantities by component

7.2 Edge Device Inference Speed Test

The trained model is deployed to the Jetson Nano edge device, and inference acceleration and FP16 half-precision quantization are adopted using TensorRT. During the test, different frame lengths of action sequences (30 frames, 60 frames, 90 frames, 120 frames, corresponding to 1 second, 2 seconds, 3 seconds, 4 seconds of action duration) are input respectively. 100 inference runs are performed under each configuration and the average time consumption is taken. The single sample inference delay is defined as τ_{inf}, and the processing frame rate (throughput) is $\text{FPS} = 1/\tau_{\text{inf}}$. To meet the real-time requirements, it is usually expected that the processing frame rate is not lower than the action acquisition frame rate (30 fps), that is, the single sample inference delay needs to be less than 33.3 milliseconds.

Table 9 reports the inference performance of the proposed algorithm on Jetson Nano. For the standard input of 90 frames (3 seconds of action), the inference delay of the original model with FP32 precision is 74.2 milliseconds, corresponding to a frame rate of 13.5 fps, which does not meet the real-time requirements. After using TensorRT FP16 optimization, the inference delay is reduced to 41.3 milliseconds (frame rate 24.2 fps), still slightly lower than 30 fps. Further combining model pruning (removing low-contribution convolution kernels with parameter volume less than 0.01M, with a pruning rate of 15%) and layer fusion optimization, the inference delay is reduced to 28.6 milliseconds, the frame rate reaches 35.0 fps, exceeding the real-time threshold. For shorter 30-frame input, the delay after optimization is only 12.4 milliseconds (frame rate 80.6 fps), which can meet the higher frequency real-time feedback requirements. For long sequences of 120 frames input, the delay after optimization is 52.7 milliseconds (frame rate 19.0 fps), slightly lower than the real-time requirement. At this time, an inter-frame sliding window strategy can be adopted, and only the most recent 90 frames are evaluated.

Table 9. Jetson nano edge device inference performance

Input frame length	Corresponding duration (seconds)	Optimization method	Inference delay (ms)	Frame rate (FPS)	Is it meeting the requirement of real-time (≥ 30 FPS)
30	1.0	Original FP32 TensorRT FP16	28.3	35.3	✓
30	1.0	Pruning + TensorRT	14.8	67.6	✓
30	1.0	FP32 original TensorRT FP16	12.4	80.6	✓
60	2.0	Pruning + TensorRT	52.6	19.0	✗
60	2.0	FP32 original TensorRT FP16	29.5	33.9	✓
60	2.0	Pruning + TensorRT	21.8	45.9	✓
90	3.0	FP32 original TensorRT FP16	74.2	13.5	✗
90	3.0	Pruning + TensorRT	41.3	24.2	✗
90	3.0	Optimization method	28.6	35.0	✓
120	4.0	Original FP32 TensorRT FP16	96.5	10.4	✗
120	4.0	Pruning + TensorRT	68.9	14.5	✗
120	4.0	FP32 original TensorRT FP16	52.7	19.0	✗

Figure 6 presents the comparison chart of inference latency for three optimization configurations (FP32 original, TensorRT FP16, pruning + TensorRT) under different input frame lengths. It can be observed that the pruning + TensorRT combination achieves approximately 60% to 65% latency reduction compared to the original FP32 model across all input lengths. Moreover, as the input frame length increases, the absolute latency savings become more significant. Additionally, we also tested the inference latency when the batch size is 4. Under 90 frames of input, the batch processing latency is 85.3 milliseconds, with an average latency of 21.3 milliseconds per sample. The real-time performance is further improved in the batch processing scenario.

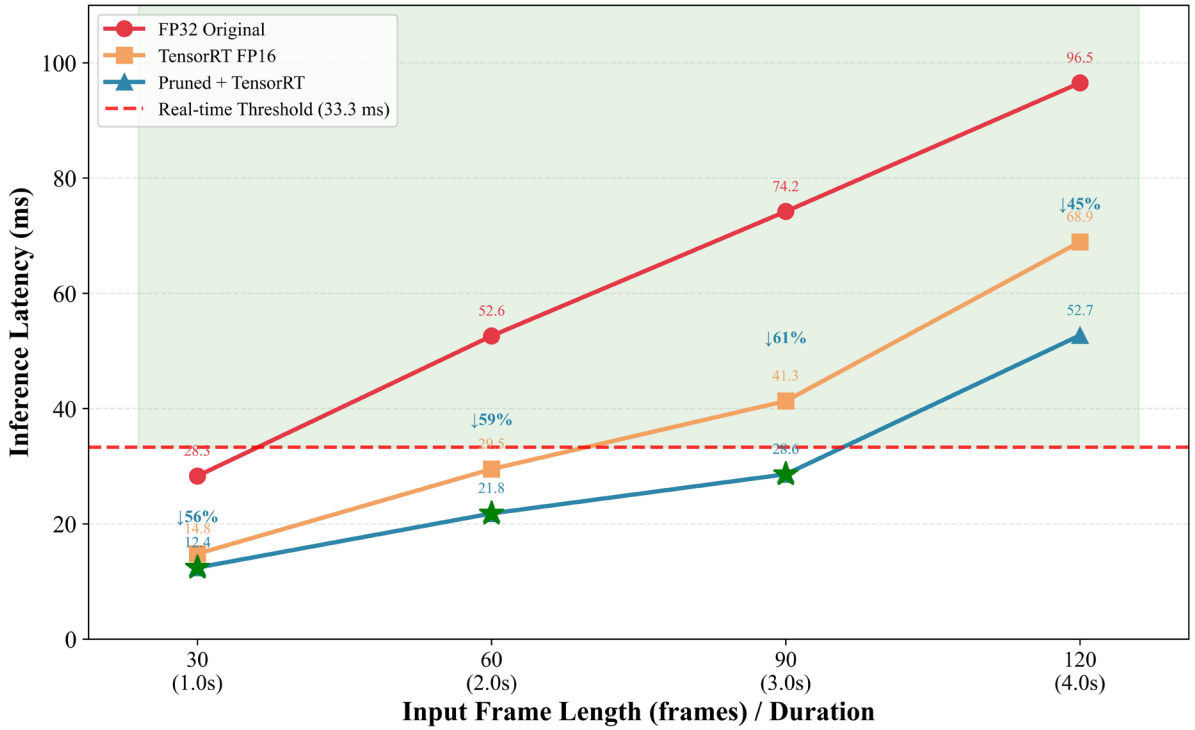

Figure 6. Relationship between inference delay and input frame length under different optimization methods

7.3 Comprehensive efficiency-accuracy comparison with the comparative model

Finally, the proposed algorithm is compared with the contrast model in Section 6.2 in terms of efficiency and accuracy. The comprehensive efficiency index "precision-speed trade-off coefficient" $\eta = \mathrm{SRCC}/(\tau_{\mathrm{inf}} \times \Psi_{\mathrm{total}})$ is defined. This index takes into account the sorting correlation, inference delay (in milliseconds), and model parameter quantity (in millions). The larger the value, the higher the comprehensive efficiency. All models are implemented on Jetson Nano using the optimal inference acceleration scheme (each model uses TensorRT FP16, the proposed algorithm additionally uses pruning, and the contrast model performs pruning optimization as recommended in its original paper). The input frame length is uniformly set to 90 frames.

Table 10 presents the comparison results of the six models in terms of accuracy (MAE and SRCC), parameter quantity, inference delay, and comprehensive efficiency coefficient. The proposed algorithm achieves an SRCC of 0.873 with 3.72M parameters and an inference delay of 28.6 milliseconds, with an MAE of 3.62 points. Compared with ActionQualityFormer (with 5.84M parameters, 35.2 milliseconds delay, and an SRCC of 0.841), the proposed algorithm reduces the parameter quantity by 36.3%, lowers the inference delay by 18.7%, and increases the SRCC by 3.8%. Compared with USDL (the graph convolution baseline model), the proposed algorithm only increases the parameter quantity by 0.47M (about 14.5%), but the SRCC increases from 0.761 to 0.873 (an increase of 14.7%), demonstrating the significant accuracy gain brought by the adaptive graph convolution and multi-scale aggregation module. Compared with the general VideoMAE (with a parameter quantity of up to 87.5M and a delay of 142.7 milliseconds), the proposed algorithm's deployment feasibility on Jetson Nano is several orders of magnitude higher, making it more suitable for edge computing scenarios.

Table 10. Comprehensive efficiency-precision comparison of different models on jetson nano (90 frames input)

Model	MAE (points)	SRCC	Parameter quantity (M)	Inference delay (ms)	Overall efficiency η (×10⁻³)
PoseQuality	7.93	0.612	0.03	3.7	5.51
TSN	6.58	0.703	24.9	68.5	0.41
USDL	5.62	0.761	3.25	24.8	9.44
VideoMAE	6.17	0.725	87.5	142.7	0.06
ActionQualityFormer	4.18	0.841	5.84	35.2	4.09
Proposed	3.62	0.873	3.72	28.6	8.21

Although PoseQuality has extremely low inference latency (3.7 ms) and the smallest number of parameters (0.03M), its SRCC is only 0.612, indicating insufficient accuracy for actual teaching assessment requirements. The proposed algorithm achieves the highest SRCC under the premise of moderate parameter quantity and inference latency, with the comprehensive efficiency coefficient η reaching 8.21×10^{-3}, ranking first among all models with "available accuracy" (SRCC > 0.8).

In summary, the proposed algorithm maintains high accuracy (MAE = 3.62 points, SRCC = 0.873) while being optimized through pruning and TensorRT. After optimization, it can run in real-time on Jetson Nano at a speed of 28.6 milliseconds per sample (35 FPS), with the parameter size controlled at 3.72M. The overall efficiency is superior to existing mainstream action quality assessment models and meets the requirements for edge deployment in actual sports teaching scenarios.

8. CONCLUSION

This paper addresses the actual demand for automatic assessment of students' movement quality in physical education teaching scenarios, proposing an end-to-end assessment algorithm based on deep learning. The entire paper conducts a systematic study around six core aspects: multi-modal data acquisition and preprocessing, adaptive spatiotemporal graph convolution backbone network, hierarchical scoring regression module, mixed loss function and training strategy optimization, experimental verification, and complexity analysis. At the data level, a multi-modal acquisition platform including RGB cameras, depth sensors, and wearable inertial units was constructed. Time series alignment, three-dimensional pose extraction and normalization processing were carried out, and three types of data augmentation strategies such as temporal distortion, skeletal perturbation and perspective synthesis were designed to provide high-quality skeletal sequence inputs for subsequent model training. At the network architecture level, an adaptive graph convolution layer based on human topology was designed, breaking the limitation of fixed adjacency matrix and achieving dynamic modeling of the collaborative relationship between non-physically adjacent joints; a temporal multi-scale aggregation module was introduced to capture the pattern features of actions at different time granularities through parallel convolution branches; combined with the attention-guided edge weight learning mechanism, the network automatically focuses on the most critical limb connections and temporal stages for quality discrimination. At the scoring regression level, a hierarchical architecture including a global-local joint

encoder, segmented fine-grained scoring head, contrastive learning bias correction branch and dual output layers was proposed to achieve comprehensive quality modeling from overall coordination to local posture and individual deviation compensation. At the training optimization level, a mixed loss function integrating mean squared error, ranking loss and local consistency loss was designed, along with sample difficult case mining and curriculum learning scheduling strategies, effectively improving the model's ability to distinguish boundary samples and generalization performance.

Through a large number of experiments on the self-built sports movement quality assessment dataset, the algorithm achieved an average absolute error of 3.62 points and a Spearman rank correlation coefficient of 0.873, significantly outperforming five comparison methods such as PoseQuality, TSN, USDL, VideoMAE and ActionQualityFormer. Ablation experiments quantitatively verified the positive contributions of each core module, including adaptive graph convolution, temporal multi-scale aggregation, attention mechanism, segmented scoring head, contrastive bias correction and ranking loss. The removal of adaptive graph convolution and multi-scale temporal aggregation modules led to an increase of 0.69 points and 0.66 points in the average absolute error, highlighting the crucial role of these two as core innovative components. Robustness tests showed that the average absolute error was 4.15 points in low-light conditions and 4.48 points in partial occlusion conditions, and the algorithm maintained stable performance in four different sports types such as standing long jump, sit-ups, basketball shooting and volleyball spiking, demonstrating excellent cross-scenario and cross-action generalization capabilities. In terms of algorithm complexity, the total parameter quantity of the model was controlled at 3.72 million, and the single inference floating-point operation count was 1.87 billion. After pruning and TensorRT optimization, it could achieve a processing speed of 35 frames per second with a latency of 28.6 milliseconds per sample on the Jetson Nano edge device, meeting real-time requirements. The comprehensive accuracy-efficiency comparison showed that the algorithm achieved the best overall efficiency coefficient among all available precision levels (Spearman rank correlation coefficient greater than 0.8).

The main advantages of the algorithm in this paper are as follows: First, the adaptive graph convolution mechanism can dynamically adjust the connection weights between joints for different movement types. For example, it automatically strengthens the coupling relationship between the wrist and shoulder in the shooting action, and strengthens the coordination relationship between the hip, knee and ankle in the standing long jump. This data-driven modeling approach significantly improves the scoring accuracy. Secondly, the segmented fine-grained scoring head not only outputs the overall score but also provides independent scores for different action stages such as pre-swing, take-off, flight, and landing, offering students actionable local improvement directions and enhancing the interpretability of the model. Thirdly, the deviation correction branch effectively alleviates the scoring system deviation caused by individual differences such as student height and flexibility, making the model more stable on different body types of test samples. Fourthly, the pruned and quantized model has the ability of real-time inference at the edge, enabling automatic scoring within the sports venue without relying on cloud computing power, reducing deployment costs and network latency. The applicable boundaries of the algorithm mainly lie in: the current version relies on three-dimensional skeletal key points output by depth sensors, and in scenarios with only ordinary RGB cameras, additional posture estimation algorithms need to be deployed; for long sequences with action duration exceeding 4 seconds, the single inference delay will rise to 52.7 milliseconds, although it can be handled through sliding window strategies, the real-time performance will be somewhat reduced; moreover, the algorithm has been trained for four basic sports actions, and for action types with significantly different complexity, it needs to be retuned.

Looking to the future, the research in this paper can be further expanded in the following directions. First, the introduction of weak supervision and self-supervised learning. The current algorithm relies on three sports teachers to independently annotate the score of each sample, which incurs high annotation costs and cannot completely eliminate subjectivity. In the future, contrastive learning and metric learning could be explored to train quality assessment models using only action category labels or pairwise comparison labels, significantly reducing reliance on detailed scoring annotations. Specifically, a self-supervised pre-training task based on ranking consistency could be designed, first learning temporal structure representations of actions from large-scale unlabeled skeletal sequences, and then fine-tuning on small labeled datasets. Second, improving cross-action generalization. The current model trains or fine-tunes separately for different action types, failing to fully exploit shared quality assessment knowledge across actions. In the future, research on meta-learning or domain generalization methods can be conducted to train a quality assessment basic model that can quickly adapt to new action types with only a small number of new action samples. For example, by constructing an action-independent motion quality representation space, the common quality dimensions such as “coordination”, “standardization”, and “coherence” can be decoupled for learning. Third, deep integration and complementarity of multimodal information. The current algorithm mainly relies on skeletal keypoint sequences, whereas RGB images contain subtle cues like muscle tension and facial expressions, as well as acceleration details in IMU data that have not been fully utilized. In the future, a multimodal fusion Transformer could be designed to adaptively fuse skeletal structure, texture, and inertial information at the feature level, further improving robustness in occluded and fast-moving scenarios. Fourth, enhancing interpretability of assessment feedback. Although the segmented scoring head outputs stage scores, it has not yet identified specific problematic limbs or joints for individual students. In the future, by combining attention map visualization and skeletal heatmap technology, the system could annotate specific problem areas such as “insufficient takeoff angle” or “off-center landing” while providing a score, enabling truly intelligent error correction guidance. Fifth, online learning and personalized adaptation. Different students have different physical conditions and sports habits, and static models cannot adapt to each individual. In the future, an online incremental learning framework could be designed, performing lightweight parameter updates based on a small amount of student feedback or consistency signals from repeated actions after deployment. This would enable the assessment standard to gradually align with each student’s specific situation, evolving from general to personalized assessment. In conclusion, the method proposed in this paper provides a new technical pathway for intelligent assessment of sports movements. Subsequent research will continue to advance in the directions of weakly supervised learning, cross-action generalization, multimodal fusion, and explainable feedback, with the goal of truly deploying the algorithm in daily physical education scenarios.

Abbreviations

RGB, Red Green Blue;

IMU, Inertial Measurement Unit;

HRNet, High-Resolution Network;

PCA, Principal Component Analysis;

GCN, Graph Convolutional Network;

MLP, Multilayer Perceptron;

ReLU, Rectified Linear Unit;

C3D, Convolutional 3D;

I3D, Inflated 3D;

TSN, Temporal Segment Networks;

VideoMAE, Video Masked Autoencoder;

USDL, Uncertainty-aware Skill Determination Learning;

MAE, Mean Absolute Error;

RMSE, Root Mean Square Error;

SRCC, Spearman Rank Correlation Coefficient;

Acc, Accuracy;

ICC, Intraclass Correlation Coefficient;

GFLOPs, Giga Floating Point Operations per Second;

FPS, Frames Per Second;

TensorRT, Tensor RunTime;

FP32, 32-bit Floating Point;

FP16, 16-bit Floating Point;

BN, Batch Normalization;

SAQD, Sports Action Quality Dataset;

CPU, Central Processing Unit;

GPU, Graphics Processing Unit.

Supplementary Material

Not applicable.

Appendix

Not applicable.

Ethics approval and consent to participate.

This study did not involve human participants, animal subjects, or any data requiring ethical approval. Therefore, ethics approval and consent to participate are not applicable.

Acknowledgements

The authors would like to thank the editors of this journal and all the anonymous reviewers who provided valuable comments on this work.

Competing interests

The authors declare that they have no financial or personal relationships that may have inappropriately influenced them in writing this article.

Author contributions

All authors have read and agreed to the published version of the manuscript. The author's contributions are specified as follows: **H.L.:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Data Curation, Writing – original draft, Writing – review & editing, Visualization, Supervision, Project administration. **M.Z.:** Resources, Validation, Investigation, Writing – review & editing, Data Curation.

Funding information

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Data availability

The data that support the findings of this study are available upon request from the corresponding authors, **M.Z.**

Disclaimer

The views and opinions expressed in this article are those of the authors and are the product of professional research. It does not necessarily reflect the official policy or position of any affiliated institution, funder, agency, or that of the publisher. The authors are responsible for this article's results, findings, and content.

Declaration of AI and AI-assisted Technologies in the Writing Process

During the writing of this article, the author used DeepSeek for spelling and grammar checking. After using this tool, the author reviewed and edited the content as needed and assumes full responsibility for the final published content.

REFERENCES

[1] Hu, Z., Liu, Z., & Su, Y. (2024). AI-driven smart transformation in physical education: Current trends and future research directions. *Applied Sciences*, *14*(22), 10616. https://doi.org/10.3390/app142210616

[2] Sun, R., Liu, Y., Li, H., & Yim, J. (2026). A Study on the Factors Influencing User Experience of AI Pose Recognition Feedback Systems in Ballet-Class Contexts. *Applied Sciences*, *16*(7), 3431. https://doi.org/10.3390/app16073431

[3] Van Maarseveen, M., Leenhouts, J., de Witte, A., Flux, E., Van Doorn, H., & van der Kamp, J. (2025). Enhancing affordance perception in pre-service physical education teachers: effects of content knowledge, motor experience and visual experience programs. *Frontiers in Sports and Active Living*, *7*, 1583448. https://doi.org/10.3389/fspor.2025.1583448

[4] Feng, L., Zhao, Y., Zhao, W., & Tang, J. (2022). A comparative review of graph convolutional networks for human skeleton-based action recognition. *Artificial Intelligence Review*, *55*(5), 4275-4305. https://doi.org/10.1007/s10462-021-10107-y

[5] Liu, Y., Zhang, H., Li, Y., He, K., & Xu, D. (2023). Skeleton-based human action recognition via large-kernel attention graph convolutional network. *IEEE Transactions on Visualization and Computer Graphics*, *29*(5), 2575-2585. https://doi.org/10.1109/tvcg.2023.3247075

[6] Liu, F., Wang, C., Tian, Z., Du, S., & Zeng, W. (2025). Advancing skeleton-based human behavior recognition: multi-stream fusion spatiotemporal graph convolutional networks. *Complex & Intelligent Systems*, *11*(1), 94. https://doi.org/10.1007/s40747-024-01743-2

[7] Zhang, C., Xu, Y., Xu, Z., Huang, J., & Lu, J. (2022). Hybrid handcrafted and learned feature framework for human action recognition: C. Zhang et al. *Applied Intelligence*, *52*(11), 12771-12787. https://doi.org/10.1007/s10489-021-03068-w

[8] Sakaa, B., Elbeltagi, A., Boudibi, S., Chaffaï, H., Islam, A. R. M. T., Kulimushi, L. C., ... & Wong, Y. J. (2022). Water quality index modeling using random forest and improved SMO algorithm for support vector machine in Saf-Saf river basin. *Environmental Science and Pollution Research*, *29*(32), 48491-48508. https://doi.org/10.1007/s11356-022-18644-x

[9] Adugna, T., Xu, W., & Fan, J. (2022). Comparison of random forest and support vector machine classifiers for regional land cover mapping using coarse resolution FY-3C images. *Remote Sensing*, *14*(3), 574. https://doi.org/10.3390/rs14030574

[10] Pang, Y., Zhang, K., & Li, F. (2025). Explainable quality assessment of effective aligned skeletal representations for martial arts movements by multi-machine learning decisions. *Scientific Reports*, *15*(1), 323. https://doi.org/10.1038/s41598-024-83475-4

[11] Freire-Obregón, D., Lorenzo-Navarro, J., Santana, O. J., Hernández-Sosa, D., & Castrillón-Santana, M. (2023, July). An x3d neural network analysis for runner's performance assessment in a wild sporting environment. In *2023 18th International Conference on Machine Vision and Applications (MVA)* (pp. 1-5). IEEE. https://doi.org/10.23919/mva57639.2023.10215918

[12] Bratta, C. (2024). Exploring Sex Differences in Diving Performance Analysis. https://hdl.handle.net/20.500.14242/190721

[13] Gan, Q. (2025). *Sports Motion Analysis: From Competition Videos to Data-Driven Interpretations* (Doctoral dissertation, Institut Polytechnique de Paris). https://doi.org/10.70675/5ae10807zc807z4f8dzb4d3z832fd95de4cb

[14] Yang, L., & Li, Y. (2026). Swimming action recognition algorithm based on improved C3D and attention-residual network. *International Journal of Information and Communication Technology*, *27*(20), 40-62. https://doi.org/10.1504/ijict.2026.10076718

[15] Dong, K., Feng, X., & Dong, L. (2025, September). Dynamic Thermal Gesture Recognition Algorithm Based on C3D. In *2025 5th International Conference on Artificial Intelligence, Automation and High Performance Computing (AIAHPC)* (pp. 126-133). IEEE. https://doi.org/10.1109/aiahpc66801.2025.11290620

[16] Lovanshi, M., & Tiwari, V. (2024). Human skeleton pose and spatio-temporal feature-based activity recognition using ST-GCN. *Multimedia Tools and Applications*, *83*(5), 12705-12730. https://doi.org/10.1007/s11042-023-16001-9

[17] Filtjens, B., Vanrumste, B., & Slaets, P. (2022). Skeleton-based action segmentation with multi-stage spatial-temporal graph convolutional neural networks. *IEEE Transactions on Emerging Topics in Computing*, *12*(1), 202-212. https://doi.org/10.1109/tetc.2022.3230912

[18] Xu, J., Liu, F., Wang, Q., Zou, R., Wang, Y., Zheng, J., ... & Zeng, W. (2024). Enhancing human behavior recognition with spatiotemporal graph convolutional neural networks and skeleton sequences. *EURASIP Journal on Advances in Signal Processing*, *2024*(1), 60. https://doi.org/10.1186/s13634-024-01156-w

[19] Zhou, C., Huang, Y., & Ling, H. (2022). Uncertainty-driven action quality assessment. *arXiv preprint arXiv:2207.14513*. https://doi.org/10.48550/arXiv.2207.14513

[20] Lei, Q., Yao, L., Zhang, H., & Du, J. (2025). Skeletal Spatio-Temporal Decoupling Transformer for Long-Duration Action Quality Assessment. *Knowledge-Based Systems*, 114672. https://doi.org/10.1016/j.knosys.2025.114672

[21] Zhu, S., Chen, J., & Su, Y. (2024). Spatio-temporal articulation & coordination co-attention graph network for human motion prediction. *Signal Processing*, *223*, 109551. https://doi.org/10.1016/j.sigpro.2024.109551

[22] Shao, D., Zhao, Y., Dai, B., & Lin, D. (2020). Finegym: A hierarchical video dataset for fine-grained action understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 2616-2625). https://doi.org/10.1109/cvpr42600.2020.00269

[23] Shuvo, M. M. H., Islam, S. K., Cheng, J., & Morshed, B. I. (2022). Efficient acceleration of deep learning inference on resource-constrained edge devices: A review. *Proceedings of the IEEE*, *111*(1), 42-91. https://doi.org/10.1109/jproc.2022.3226481

[24] Sharma, D., & Sarkar, S. (2022). Enabling inference and training of deep learning models for AI applications on IoT edge devices. In *Artificial Intelligence-based Internet of Things Systems* (pp. 267-283). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-030-87059-1_10

[25] Shah, A., Bangash, J. I., Khan, A. W., Ahmed, I., Khan, A., Khan, A., & Khan, A. (2022). Comparative analysis of median filter and its variants for removal of impulse noise from gray scale images. *Journal of King Saud University-Computer and Information Sciences*, *34*(3), 505-519.

https://doi.org/10.1016/j.jksuci.2020.03.007

[26] Gao, Y., Dang, C., Zhu, J., Xie, Y., Hu, Y., Yan, C., ... & Li, X. (2025). A real-time 3D modeling method for buildings driven by IMU and RGB-D fusion. *International Journal of Digital Earth*, *18*(1), 2506496. https://doi.org/10.1080/17538947.2025.2506496

[27] Lee, H., & Ryu, J. (2025). Toward Efficient Generalization in 3D Human Pose Estimation via a Canonical Domain Approach. *IEEE Access*. https://doi.org/10.1109/access.2025.3566109

[28] Zhang, J., Tu, Z., Weng, J., Yuan, J., & Du, B. (2024). A modular neural motion retargeting system decoupling skeleton and shape perception. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *46*(10), 6889-6904. https://doi.org/10.1109/tpami.2024.3386777

[29] JunZhang, C., & GuanLi, Y. (2026). A hybrid framework combining adaptive graph learning and global temporal attention for skeleton-based action recognition. *Scientific Reports*. https://doi.org/10.1038/s41598-026-49915-z

[30] Iorga, A., Jianu, A., Gheorghiu, M., Crețu, B. D., & Eremia, I. A. (2023). Motor coordination and its importance in practicing performance movement. *Sustainability*, *15*(7), 5812. https://doi.org/10.3390/su15075812

[31] Kilic, U., Karadag, O. O., & Ozyer, G. T. (2025). AGMS-GCN: Attention-guided multi-scale graph convolutional networks for skeleton-based action recognition. *Knowledge-Based Systems*, *311*, 113045. https://doi.org/10.1016/j.knosys.2025.113045

[32] Singh, S. K., Bharti, B. K., Yadav, A. N., & Dwivedi, A. K. (2025). Optimized Microwave Ablation with a Novel Applicator: Integration of Taguchi Neural Networks for Enhanced Predictive Accuracy of Ablation Zone. *IEEE Journal on Multiscale and Multiphysics Computational Techniques*. https://doi.org/10.1109/jmmct.2025.3589163

[33] Tekin, M., Yurdal, M. O., Toraman, Ç., Korkmaz, G., & Uysal, İ. (2025). Is AI the future of evaluation in medical education?? AI vs. human evaluation in objective structured clinical examination. *BMC medical education*, *25*(1), 641. https://doi.org/10.1186/s12909-025-07241-4

[34] Hu, K., Shen, C., Wang, T., Xu, K., Xia, Q., Xia, M., & Cai, C. (2024). Overview of temporal action detection based on deep learning. *Artificial Intelligence Review*, *57*(2). https://doi.org/10.1007/s10462-023-10650-w

[35] Zahan, S., Hassan, G. M., & Mian, A. (2024). Learning sparse temporal video mapping for action quality assessment in floor gymnastics. *IEEE Transactions on Instrumentation and Measurement*, *73*, 1-11. https://doi.org/10.1109/tim.2024.3398072

[36] Wang, Y., Yue, Y., Lu, R., Han, Y., Song, S., & Huang, G. (2024). Efficienttrain++: Generalized curriculum learning for efficient visual backbone training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *46*(12), 8036-8055. https://doi.org/10.48550/arXiv.2405.08768

APA Citation

Lei, H., & Zhou, M. (2026). Research on Automatic Evaluation Algorithm for Student Sports Movement Quality Based on Deep Learning. *Journal of Discovery Core*, *1*(1), 19-54. **https://doi.org/10.67541/jdc2602**