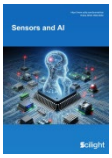
Scilight

Sensors and AI

https://www.sciltp.com/journals/sai



Article

A Tennis Motion Correction Approach Based on Ensemble Learning and MediaPipe

Yue Gao [1], Chuxin Cao [1], Xuzhen Wu [1], Yiyang Chen [1,*] and Hongtian Chen [2]

[1] School of Mechanical and Electrical Engineering, Soochow University, Suzhou 215301, China
[2] Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China
* Correspondence: yychen90@suda.edu.cn

How To Cite: Gao, Y.; Cao, C.; Wu, X.; et al. A Tennis Motion Correction Approach Based on Ensemble Learning and MediaPipe. *Sensors and AI* **2025**, *1*(1), 45–60.

Received: 1 July 2025
Revised: 11 August 2025
Accepted: 21 August 2025
Published: 28 August 2025

Abstract: This paper investigates human posture recognition methods in tennis sports and develops a tennis motion correction approach, which is capable of rectifying non-standard movements. Traditional tennis player posture detection methods suffer from several limitations, including insufficient robustness in complex backgrounds, high self-occlusion in tennis motions, and slow processing speeds for video-based action analysis. To address these issues, this paper proposes an integrated approach combining ensemble learning with the MediaPipe pose detection algorithm to address these challenges. This approach utilizes training data collected by an indoor motion capture system to train a tennis fundamental motion classification model based on Gradient Boosting Decision Trees (GBDT). MediaPipe is employed to perform human skeleton analysis, extracting eight key body joints. This paper evaluates tennis motions based on eight tennis-specific kinematic features and ultimately provides tailored corrective recommendations according to identified deficiencies. Experimental results demonstrate that this motion correction approach effectively delivers reasonable corrections for tennis players across different skill levels.

Keywords: intelligent sports training; human pose estimation; ensemble learning; MediaPipe; motion correction

1. Introduction

The advent of machine learning and artificial neural networks has driven significant advancements in motion capture systems, thereby propelling the extensive application of motion correction systems. Tennis is a sport that demands highly standardized movements, as the correctness of motions directly impacts hitting effectiveness and overall athletic performance. During the learning process, if various incorrect movements are not promptly corrected, they may not only hinder players' mastery of proper tennis techniques but could also lead to accumulated physical damage over time, such as skeletal injuries or muscle strains [1,2]. Traditional tennis instruction relies heavily on coaches' demonstrations and on-site guidance, where they visually assess students' postures based on personal experience [3]. Although this method has been widely used for an extended period, it suffers from several limitations:

(1) Subjectivity and lack of real-time feedback: Coaches' visual assessments tend to be inherently subjective and not real-time. Human observation is incapable of precisely quantify angular deviations in tennis motions, making it difficult to provide immediate feedback or accurate evaluation results.
(2) Inconsistency due to varying coach expertise: Different coaches may assess the same movement differently based on their individual experience and skill levels, leading to inconsistent guidance.
(3) Scalability issues in group training: In physical education class, a single coach ineffectively monitors and corrects the movements of multiple trainees simultaneously.

Current mainstreams of pose estimation methods are categorized into traditional feature-based approaches and deep learning-based approaches [4–9]. Traditional methods typically employ graph structures and deformable part models to design 2D human body part detectors. These methods model human joints as graph structures and incorporate kinematic constraints or geometric relationships between joints to optimize the graph model for pose estimation. For instance, Fischler et al. [10] proposed the Pictorial Structures model, which represents the human body as rigid parts connected by spring-like joints to form a tree-structured graph. However, this approach suffers from high computational complexity and poor performance in handling self-occlusion scenarios. P. Felzenszwalb et al. [11] employed a Deformable Part Model (DPM) that represents the human body using a root filter and multiple part filters, where parts are allowed elastic deformation relative to the root position through deformation cost constraints. However, this model similarly suffers from high computational complexity and demonstrates limited effectiveness in handling self-occlusion and complex backgrounds. In Zhang et al.'s work [12], local shape features were utilized for human pose estimation. However, this approach fails when body self-occlusion causes missing corner features, leading to joint localization errors. While traditional methods offer strong interpretability, their reliance on handcrafted features—primarily Histogram of Oriented Gradients (HOG) and SIFT features [13], and dependence on prior knowledge for skeletal length constraints prevents them from fully leveraging image information. Consequently, their accuracy degrades significantly when faced with appearance variations, viewpoint changes, occlusion, or cluttered backgrounds. Furthermore, many traditional approaches extract pose features from depth images. However, the requirement for specialized depth-sensing equipment makes these methods prohibitively expensive, limiting their applicability across diverse real-world scenarios.

Deep learning-based pose estimation methods [14–19] primarily leverage convolutional neural networks (CNNs) to automatically extract key human pose features from image data. Through data-driven feature learning, these approaches significantly improve both the accuracy and robustness of pose estimation. Researchers in [20,21] proposed a DNN-based pose estimation method that employs multiple neural networks to localize and analyze human keypoints frame-by-frame. While demonstrating strong performance on static images, this approach shows limitations when processing dynamic motion videos and remains susceptible to complex background interference. To enhance pose estimation, researchers in [22–24] developed a CNN-based method that extracts multi-scale features for keypoint detection, outputting probability heatmaps of keypoints. This method captures multi-scale, diverse human joint feature vectors across different receptive fields while comprehensively integrating contextual information for each feature. Although this CNN-based approach improves feature extraction and achieves higher precision through coordinate regression analysis of refined feature vectors, it fails to effectively model temporal continuity of motions. Researchers in [25,26] proposed a bottom-up approach utilizing Part Affinity Fields (PAFs) to associate body parts with individuals in images, achieving both high precision and real-time pose detection in still images. However, this method demonstrates slower processing speeds when handling video sequences. Researchers in [27,28] developed a multi-scale high-resolution network that maintains high-resolution representations for human pose estimation throughout deep networks. While these analytical methods show promising results for general pose estimation, they still exhibit limitations when applied to the precise motion analysis required for tennis techniques.

When performing pose recognition indoors, motion capture equipment can obtain human keypoint position information accurate to the millimeter level, which can reduce the impact of lighting and human self-occlusion on experimental results. However, motion capture equipment is bulky and inconvenient to transport, making the use of motion capture systems in outdoor scenarios unfeasible. When performing pose recognition outdoors, the human pose data obtained using ordinary cameras lacks sufficient accuracy, and experimental results are affected by changes in sunlight intensity and angle over time. This paper uses a combined approach using motion capture systems and Kinect cameras to balance the accuracy differences between indoor and outdoor environments while mitigating the impact of lighting conditions on experiments. By using GBDT to extract high-level motion features from raw pose keypoints and enhance the discriminative capability of temporal actions through gradient boosted tree ensembles, thereby improving the robustness of video-based pose estimation and detail capture performance. To address tennis-specific challenges including frequent self-occlusion and high real-time requirements, the MediaPipe pose recognition framework is utilized. This solution significantly improves both real-time performance and detection accuracy of motion analysis while maintaining recognition precision. The approach achieves fast response times and, when encountering occluded areas, can accurately predict the positions of obscured joints, minimizing their impact on final results.

The main contributions of this work include:

(1) Integration of indoor motion capture systems and outdoor cameras. By combining their respective advantages, the limitations of single-sensor solutions in cross-scenario applications are addressed, achieving consistent performance across indoor and outdoor environments.

(2) Ensemble learning for reduced annotation cost and enhanced motion detail capture. For the fast strokes and footwork in tennis, which are detail-oriented actions, the generalization ability of the model for subtle motions is improved through the combination of strong supervision from motion capture data and weak supervision from outdoor data.
(3) Enhanced MediaPipe framework for tennis-specific pose detection. While ensuring real-time performance, the prediction robustness under self-occlusion scenarios in tennis movements is enhanced.

The remaining of this paper is organized as follows. In Section 2, the proposed algorithm is detailed. Both the MediaPipe pose recognition algorithm and the Gradient Boosting Decision Trees (GBDT) algorithm are introduced. Section 3 presents research methodology, describing the composition of three subsystems in the tennis correction approach. Section 4 conducts the ablation experiments using the hyperparameters of GBDT and analyses the experimental results of tennis motion correction for trainees. Section 5 summarizes the experimental results and presents future prospects.

2. Problem Description

This section briefly introduces the research tasks in tennis motion correction methods and present the fundamental principles of MediaPipe and Gradient Boosted Decision Tree (GBDT) algorithms to facilitate the subsequent approach design.

2.1. Research Task

This paper aims to develop a tennis posture correction approach that reconstructs normalized 2D human keypoint coordinates from input videos of tennis trainees, computes relevant features, and compares them with standard tennis player movements to generate corresponding corrective guidance. Each fundamental tennis motion is decomposed into a series of basic sub-actions, including serve, forehand stroke, and backhand stroke. The MediaPipe algorithm is employed to identify the human skeleton in real-time motion capture videos. This allows for comparison with data from professional athletes collected by motion capture systems, thereby achieving motion correction. Given the extended learning curve in tennis training, our approach adapts to players at different skill levels, ensuring accurate corrective feedback for all trainees, thereby significantly improving training effectiveness. A complete tennis motion correction task in this paper involves the following steps:

(1) Data collection: After entering the data capture area at specified angles, tennis trainees perform serves, forehand strokes, and backhand strokes in sequence, striving to execute movements according to standard specifications. On-site cameras must capture and save video recordings of entire motion sequence.
(2) Keypoint detection: Upon completion of each motion, the system immediately applies the MediaPipe algorithm to identify the human skeleton and keypoints, obtaining normalized coordinates of keypoints.
(3) Motion quality evaluation: The system assesses the trainee's performance based on eight specialized features and ultimately provides corrective recommendations.

2.2. MediaPipe Algorithm

MediaPipe is a pose detection framework developed by Google for machine learning applications [29], specializing in efficient, low-latency multimedia data processing with particular emphasis on computer vision. It adopts a modular design and provides a series of ready-to-use solutions, including face detection, gesture recognition, and human pose estimation. In its human pose estimation module, the algorithm separates human subjects from the background in images or videos, detects and localizes human keypoints, thereby enabling the reconstruction of human postures, movements, or motion trajectories. MediaPipe supports both 2D and 3D pose estimation, and the standard 33-keypoint human body positions it predicts are shown in Figure 1. The 33 key points in Mediapipe are used to recognize the skeleton of tennis players to be detected in the motion correction subsystem.

2.3. Gradient Boosted Decision Tree

Gradient Boosting Decision Tree (GBDT) is an iterative decision tree algorithm [30]. This algorithm needs to build trees step by step and correct the errors produced by previous trees to provide new performance optimizations. The conclusions drawn by all the trees are eventually added together to form the final result. GBDT has strong generalization ability. It adapts to the characteristics of the dataset by applying hyperparameters and fine-tuning the model. It can automatically discover a variety of distinctive features and feature combinations. For example, through the tree structure, it can automatically capture complex rules such as "deduct points when shoulder-hip rotation > 45° and knee flexion < 120°".

The GBDT classifier is employed to assess the quality of movements, enabling it to learn combined features to determine whether a tennis stroke is standard. Assumed input: Training data $T = \{(x_1, y_1), (x_2, y_2), \cdots, (x_n, y_n)\}, x_i \in X \in R^n, y_i \in X \in R^n$. The loss function $L(y, f(x))$ is as

$$L(y, f(x)) = -\frac{1}{\mathrm{N}} \sum_{i=1}^{\mathrm{N}} \sum_{c=1}^{\mathrm{C}} \omega_c \cdot y_{i,c} \cdot \log(p_{i,c}) \tag{1}$$

where ω_c is the weight of action category; N is the number of samples; C is the number of action categories; $y_{i,c}$ is the true label of sample i in category c; $p_{i,c}$ is the probability predicted by the model that sample i belongs to category c, which must satisfy $\sum p_{i,c} = 1$.

The initial learner f_0 is established as shown as

$$f_0 = \underbrace{\arg\min}_{c} \sum_{i=1}^{m} L(y_i, c). \tag{2}$$

Calculate the negative gradient r_{ti} in

$$r_{ti} = \left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=f_{t-1}(x)}, \; t = 1,2,\cdots,T, \; i = 1,2,\cdots n, \tag{3}$$

where t represents the number of tree layers. The corresponding leaf node regions are R_{ij}, where $j = 1,2,\ldots,J$ and J denotes the number of leaf nodes.

For one possible leaf nodes, the optimal fitting values are calculated using

$$c_{tj} = \underbrace{\arg\min}_{c} \sum_{x_i \in R_{ij}} L(y_i, f_{t-1}(x_i) + c) \tag{4}$$

where c_{tj} is the minimum loss of R_{ij}.

The updated strong learner $f_t(x)$ is shown in

$$f_t(x) = f_{t-1}(x) + \sum_{j=1}^{J} c_{tj} I(x_i \in R_{ij}) \tag{5}$$

where I is the indicator function, when $x_i \in R_{ij}$, $I = 1$ in this Equation, otherwise $I = 0$.

The final GBDT learner obtained is shown in

$$f(x) = f_0(x) + \sum_{t=1}^{T} \sum_{j=1}^{J} c_{tj} I(x_i \in R_{ij}). \tag{6}$$

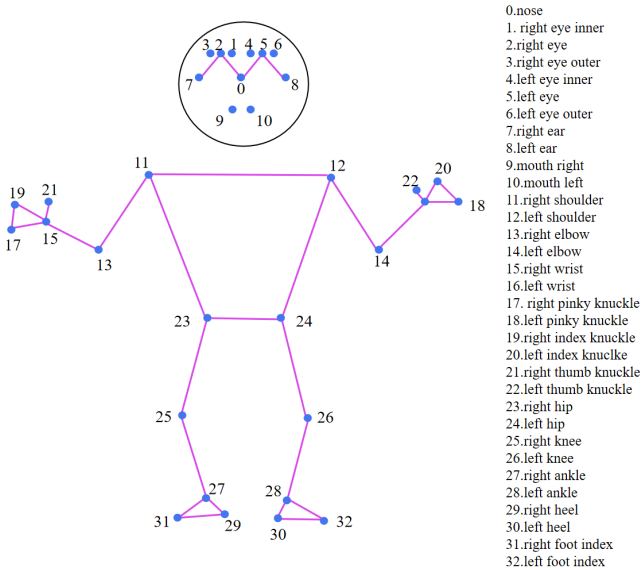


Figure 1. Standard 33-keypoint human body positions in Mediapipe.

The GBDT schematic diagram is shown in Figure 2, r_n denotes the residual error, while ω_n signifies the weight. Through the hierarchical splitting of GBDT decision tree, it achieves intersections of pose features which are more interpretable. By loading the labeled standard action positive sample videos of professional tennis players and the common tennis error action negative sample videos, and training the model after feature extraction, the final output features are the tennis action categories determined by the judgment: such as serving, forehand stroke, and backhand stroke.

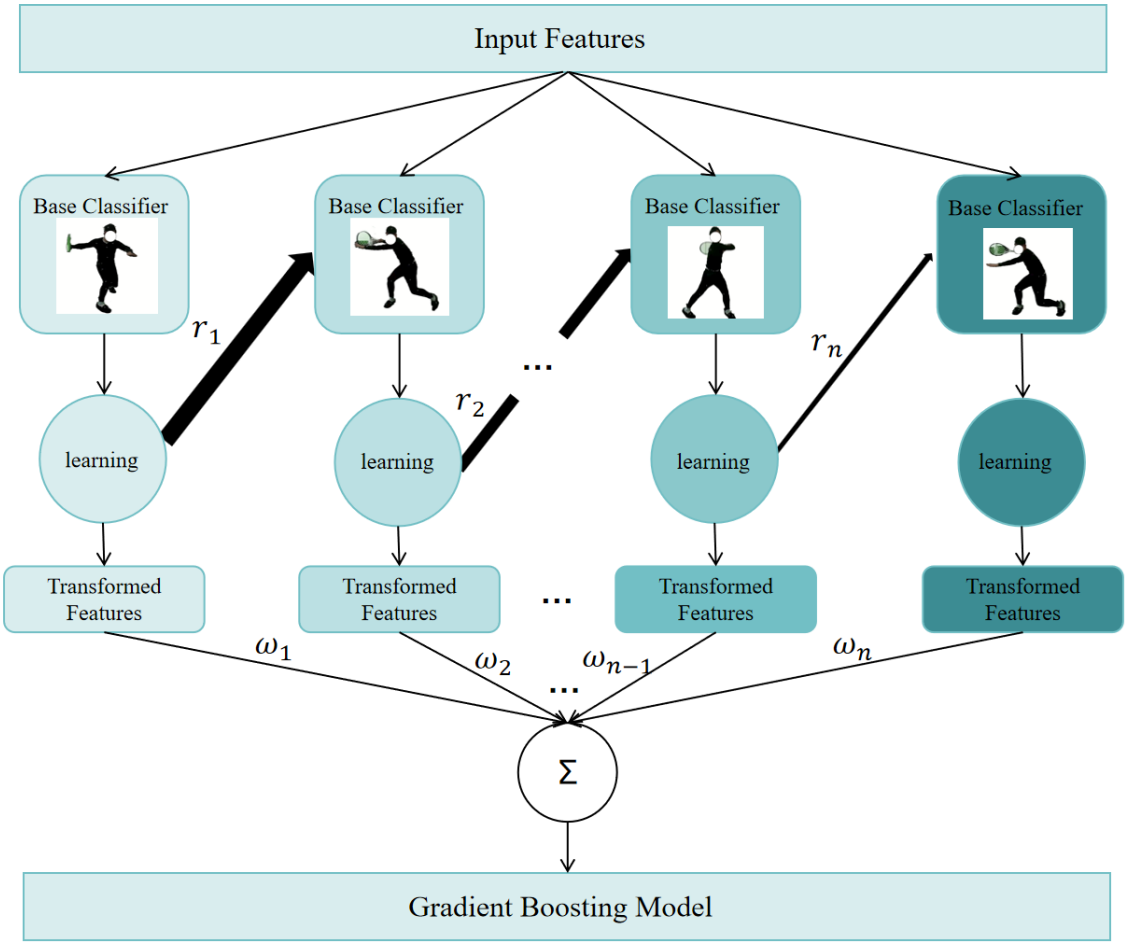


Figure 2. Gradient Boosted Decision Tree schematic diagram.

3. Research Methodology

This section presents a novel tennis motion correction approach which consists of three subsystems: a standard motion library subsystem, motion real-time collection subsystem, and motion correction subsystem.

3.1. Research Framework Design

The implementation framework of tennis motion correction approach established in this paper is shown in Figure 3. In the standard motion library subsystem, the Nokov motion capture system is used to collect data on six basic tennis postures from multiple professional tennis players, and the MediaPipe pose recognition algorithm is employed to identify and extract key features of each motion to construct a standard tennis motion dataset. In the motion real-time collection subsystem, Kinect cameras are used to capture and store real-time training videos of trainees for tennis motion detection. In the motion correction subsystem, an ensemble learning approach is employed to construct a keyframe matching model. Based on the established standard tennis motion dataset, the approach trains an ensemble learning model to perform frame-by-frame pose estimation on the trainee's real-time motion videos. This process completes the extraction of key kinematic features from tennis motions. The extracted features are matched with keyframes from the standard motion database subsystem and output both assessment results and specific corrective suggestions.

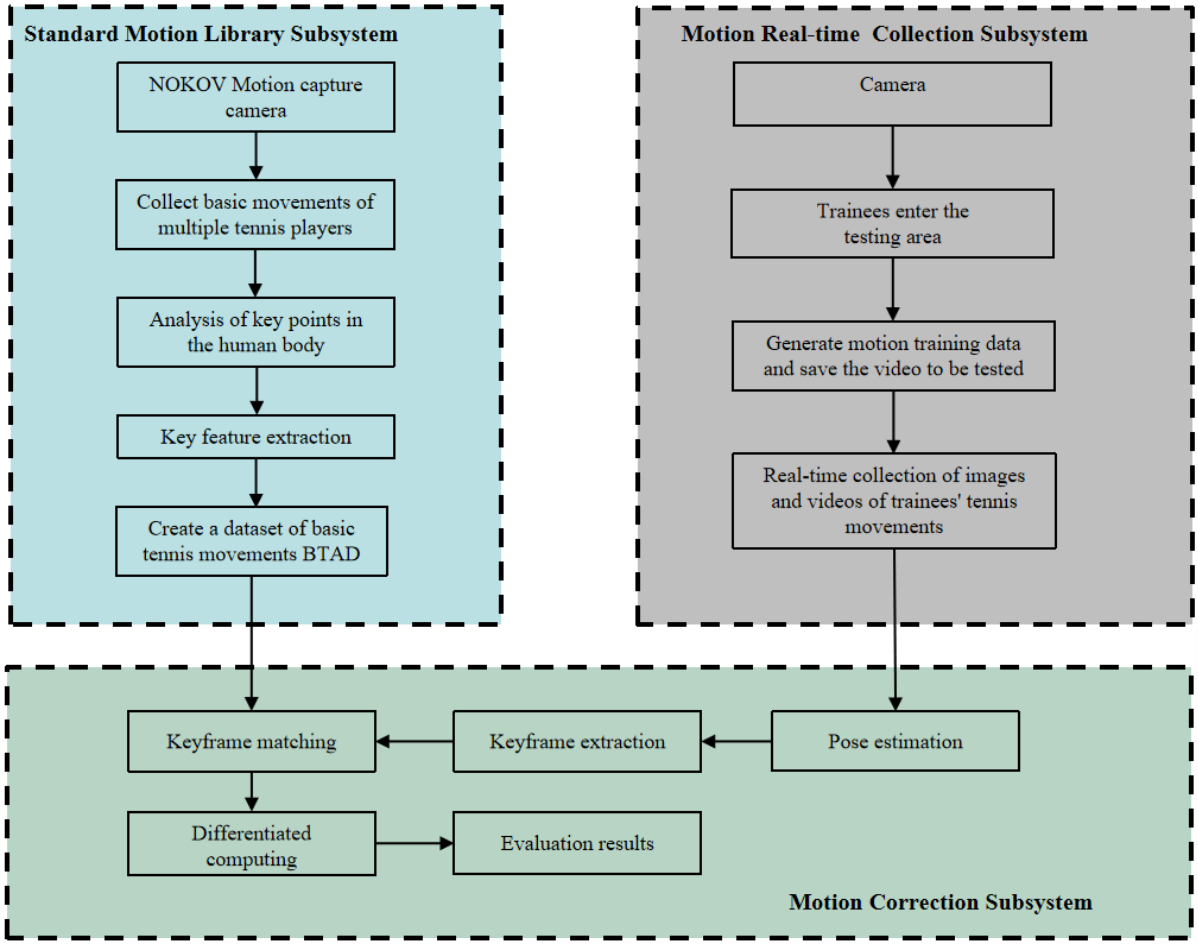


Figure 3. Implementation framework of tennis motion correction approach.

3.2. Standard Motion Database

NOKOV Optical 3D Motion Capture System as shown in Figure 4a, is a high-precision motion tracking technology solution. It consists of 12 multi-view NOKOV high-speed infrared motion capture cameras as shown in Figure 4b. This system captures the 3D coordinates of reflective markers, as shown in Figure 4c. When a tennis player wears a motion capture suit, the system's built-in efficient data processing engine ensures the immediate processing and 3D feedback of the reflective markers' data, as illustrated in Figure 5. It captures the standard movements of professional tennis players without delay.

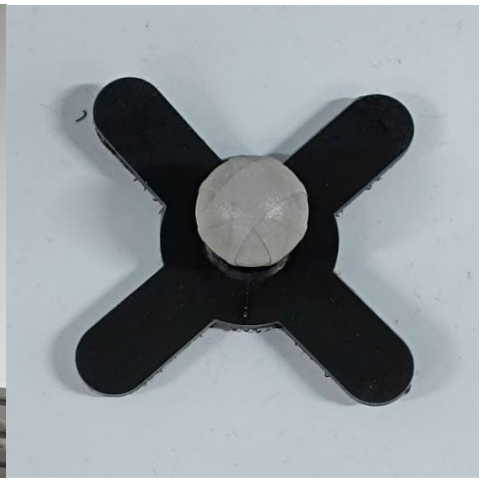
XYZ-axis coordinate variation curves of 53 points captured by the NOKOV motion capture system are shown in Figure 5a, where one curve represents the position changes attached to one body point. The 53 key points in the Nokov motion capture system are used to recognize the human body key points of professional athletes during the production process of the BTAD dataset.The illustration for 53 body points and their corresponding color in Figure 5a is given in Figure 5b.

Compared with the posture recognition of simple actions such as handshaking and standing, the tennis movement pattern is complex, and the degree of self-occlusion is high, making its analysis more challenging. To make the tennis motion correction approach designed in this paper more valuable for use, reflective markers were attached to the standard locations of motion capture markers as shown in Figure 6a. The basic tennis actions of a professional tennis player from our university's Physical Education Department as shown in Figure 6b were captured to create the Basic Tennis Action Datasets (BTAD).

To enhance the generalization ability and robustness of dataset, the basic tennis actions of eight athletes of different genders, heights, and weights from our university's Physical Education Department were captured. Each athlete performed 15 repetitions of three basic tennis actions: serving, forehand stroke, and backhand stroke. The final dataset comprises a total of 720 s of various tennis action video images and 10,800 frames. Using the BTAD dataset as the target dataset, it was divided into training and testing datasets in 7:3 ratio. The structure table of the BTAD dataset is presented in Table 1.

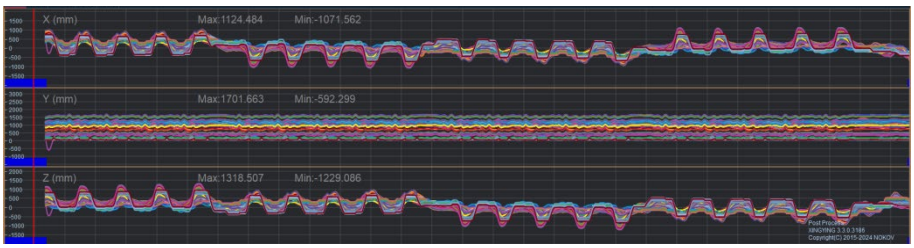
(**a**) NOKOV Optical 3D Motion Capture System

(**b**) NOKOV high-speed infrared motion capture camera

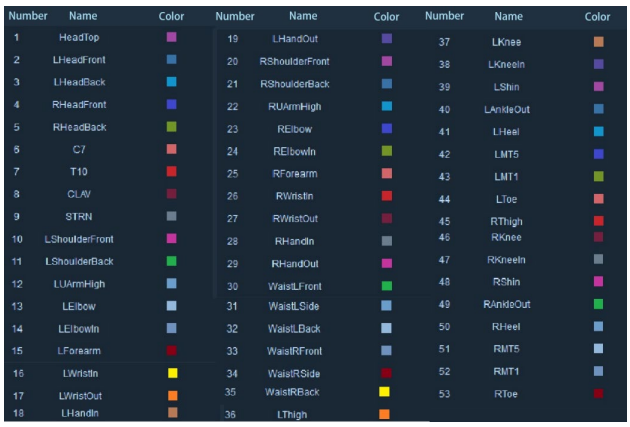


(**c**) Reflective markers

Figure 4. NOKOV optical 3D motion capture system.



(**a**) Coordinate variation curves of 53 points



Number	Name	Color	Number	Name	Color	Number	Name	Color
1	HeadTop		19	LHandOut		37	LKnee	
2	LHeadFront		20	RShoulderFront		38	LKneeIn	
3	LHeadBack		21	RShoulderBack		39	LShin	
4	RHeadFront		22	RUArmHigh		40	LAnkleOut	
5	RHeadBack		23	RElbow		41	LHeel	
6	C7		24	RElbowIn		42	LMT5	
7	T10		25	RForearm		43	LMT1	
8	CLAV		26	RWristIn		44	LToe	
9	STRN		27	RWristOut		45	RThigh	
10	LShoulderFront		28	RHandIn		46	RKnee	
11	LShoulderBack		29	RHandOut		47	RKneeIn	
12	LUArmHigh		30	WaistLFront		48	RShin	
13	LElbow		31	WaistLSide		49	RAnkleOut	
14	LElbowIn		32	WaistLBack		50	RHeel	
15	LForearm		33	WaistRFront		51	RMT5	
16	LWristIn		34	WaistRSide		52	RMT1	
17	LWristOut		35	WaistRBack		53	RToe	
18	LHandIn		36	LThigh				

(**b**) Color labeling of curves

Figure 5. Nokov software (version 3.4.0.4088) export image.

Table 1. BTAD dataset structure table.

Number	Attribute	Statistical Value	Supplementary Explanation
1	Number of athletes	8	4 males and 4 females
2	Height range	158–178 cm	The average height is 173.2 cm
3	Weight range	48–78 kg	The average weight is 63.5 kg
4	Action category	Serve, forehand, backhand	15 times per person for each category
5	Total Frames	10,800	120 Hz

The key to establishing the dataset is to obtain the sequence information of human keypoints in the standard videos. Therefore, in the program of this paper, the MediaPipe human pose estimation algorithm is utilized to predict the keypoints of the human body in the dataset videos and output the sequence of human keypoints in the videos. In the dataset, taking the completion of a forehand stroke as an example, considering every frame in the complete stroke video of the same professional athlete to be of equivalent importance would lead to redundant information, as adjacent frames typically contain similar spatial and motion information in actual sports activities. Equally considering each frame would also result in the loss of temporal feature extraction. On the other hand, selecting keyframes at fixed time intervals to identify spatial and motion information would lead to the loss of local dynamic information. Moreover, keyframes selected at fixed intervals are not representative of coherent actions in terms of semantics and lack contextual information. Therefore, keyframes are selected based on the characteristics of the forehand stroke: Since a forehand stroke involves the player first rotating the body to draw the racket back and then swinging the racket to hit the ball, keyframes can be identified based on the extreme positions of human body keypoints. Taking a right-handed grip as an example, the player needs to spread both hands during the backswing, which ends when the distance between the right wrist and the body center is maximized. Thus, the frame corresponding to this moment is selected as one of the keyframes. During the forward swing, the player swings the racket above the shoulder, which ends when the distance between the left and right wrists is minimized and the distance from the body center is maximized. Therefore, the frame corresponding to this moment is selected as another keyframe. After selecting the keyframes, the angles of the thighs and knees of the lower body are calculated. The extraction of action keyframes is shown in Figure 7.

3.3. Real-Time Motion Acquisition

The real-time motion acquisition subsystem is the core data input module of tennis motion correction approach. It is mainly responsible for collecting the trainee's motion posture data and synchronously storing the tennis training video to be detected. To reduce the self-occlusion of swing action and facilitate real-time action acquisition of trainee, the Kinect camera is deployed in the manner shown in Figure 8. It is placed about 3 m away from the data acquisition area on the right side of trainee parallel to the net, with the camera tilted downward by about 5 degrees. The data acquisition area should be the clean background middle area collected by the camera, which is conducive to camera focusing and acquisition.

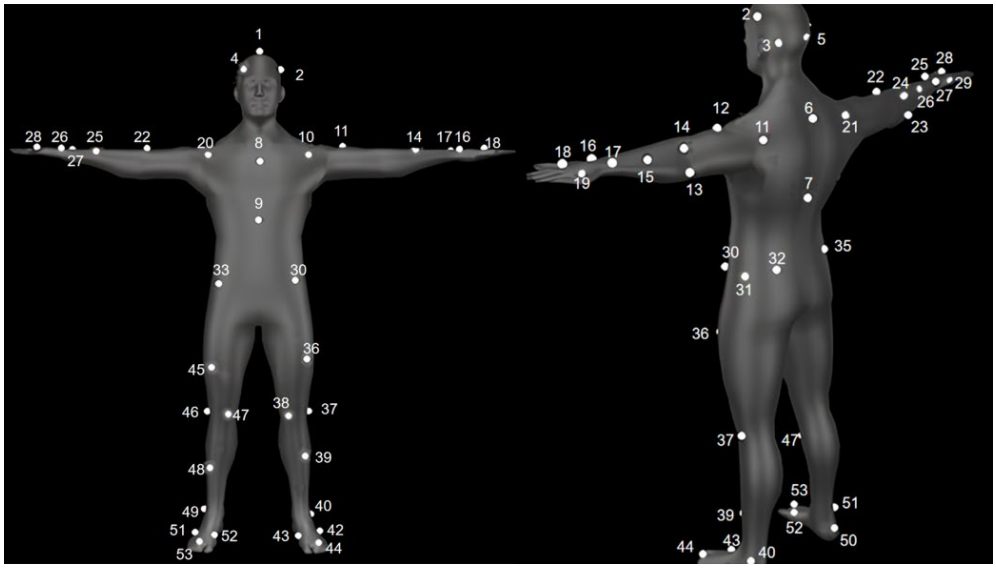
After the deployment of tennis motion correction approach, the trainee is instructed to enter the acquisition area facing the net. Then the evaluation is started. The camera continuously collects video data from the test area and saves it to a folder, preparing for the subsequent comparison of differences between the standard tennis posture and the posture to be corrected.

3.4. Motion Correction

The motion correction subsystem comprises two core components: a pose detector based on MediaPipe and an motion evaluation model based on a gradient boosting classifier.

3.4.1. Pose Detection

In this paper, the MediaPipe Pose model is used to detect human body keypoints in the input video on a per-frame basis, outputting the 3D coordinates of 33 keypoints. These 3D coordinates are normalized based on the width and height of original image, ensuring that the normalized coordinates of each keypoint (x, y, z) fall within the range of (0~1). The time point of current frame is also recorded. The eight core keypoints obtained by the Pose model are shown in Table 2. Both the detection confidence threshold and the tracking confidence threshold are set to 0.7.



(**a**) The standard locations for motion capture markers

(**b**) Athletes wearing motion capture suits with reflective markers

Figure 6. Standard 53-keypoint human body positions in Nokov.

Figure 7. Key frame extraction for forehand stroke.

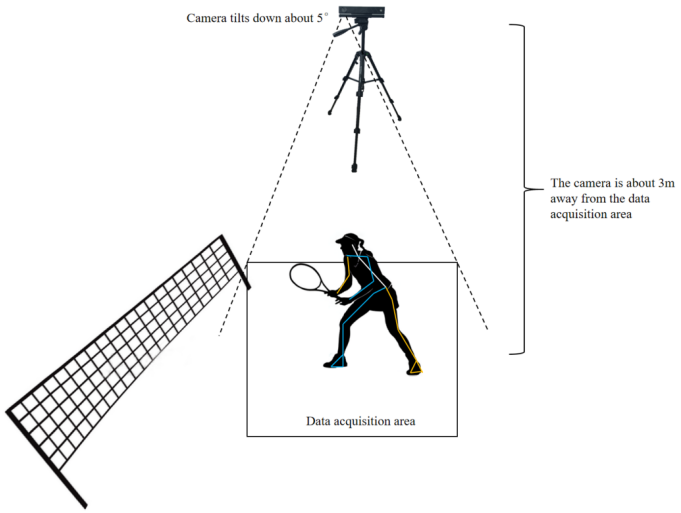


Figure 8. Deployment diagram for real-time motion acquisition on site.

Table 2. The eight core keypoints obtained by the Pose model.

Number	Joint Name	MediaPipe Pose Name	Example of Normalized Coordinates
1	Left shoulder	left_shoulder	(0.35, 0.60)
2	Right shoulder	right_shoulder	(0.65, 0.60)
3	Left elbow	left_elbow	(0.30, 0.75)
4	Right elbow	right_elbow	(0.70, 0.75)
5	Left wrist	left_wrist	(0.25, 0.85)
6	Right wrist	right_wrist	(0.75, 0.85)
7	Left hip	left_hip	(0.40, 0.90)
8	Right hip	right_hip	(0.60, 0.90)

Using the eight core keypoints obtained, the following eight kinematic features of tennis action are calculated, as shown in Table 3.

Table 3. Kinematic features of tennis actions.

Number	Feature Name	Calculation Method	Function
1	Shoulder hip rotation angle	get_rotation_angle()	Assess the degree of body twisting and determine whether the core force is sufficient when hitting the ball
2	Balance score	_balance_score()	Quantify the body stability during hitting, reflecting lower limb control ability
3	Forehand hitting elbow angle	calculate_angle()	Determine whether the elbow position is correct when hitting the forehand ball
4	Forehand hitting wrist joint angle	calculate_angle()	Evaluate wrist force posture and its impact on face control
5	Backhand stroke elbow angle	calculate_angle()	Analyze the degree of elbow extension during backhand stroke
6	Backhand stroke wrist angle	calculate_angle()	Detecting the locked state of the wrist during backhand stroke
7	Wrist movement speed	_get_wrist_speed()	Quantify swing acceleration and affect hitting power
8	Knee curvature	_get_knee_bend()	Assess the level of lower limb energy storage and its impact on the efficiency of force transmission

The function "calculate_angle" is implemented in Python: calculate_angle (*a, b, c*), which can calculate the angle θ between three points $a = (a_x, a_y)$, $b = (b_x, b_y)$, $c = (c_x, c_y)$ with b as the center. The equations are as follows:

$$\theta = \begin{cases} 360° - |\alpha|, \text{if } |\alpha| > 180° \\ |\alpha|, \text{otherwise} \end{cases} \tag{7}$$

$$\alpha = \left(\arctan2(c_y - b_y, c_x - b_x) - \arctan2(a_y - b_y, a_x - b_x)\right) \times \frac{180°}{\pi}. \tag{8}$$

In the function, the points *a*, *b*, and *c* need to be converted to NumPy arrays first. In Equation (7), taking the absolute value can yield an angle within the range of (0~360°), and the final result obtained is the smallest angle between two points, which is $\theta \leq 180°$. In Equation (8), α represents the rotation angle from $\overrightarrow{ba}$ to $\overrightarrow{bc}$.

The function "_get_rotation_angle" is implemented in Python and is capable of computing the body rotation angle. It involves four key points: left_shoulder = (x_{ls}, y_{ls}), right_shoulder = (x_{rs}, y_{rs}), left_hip = (x_{lh}, y_{lh}), right_hip = (x_{rh}, y_{rh}). The calculation process is divided into the following three steps:

$$(x_s, y_s) = \left(\frac{x_{ls} + x_{rs}}{2}, \frac{y_{ls} + y_{rs}}{2}\right) \tag{9}$$

$$(x_h, y_h) = \left(\frac{x_{lh} + x_{rh}}{2}, \frac{y_{lh} + y_{rh}}{2}\right) \tag{10}$$

$$\beta = \text{calculate\_angle}(\text{shoulder\_center}, \text{hip\_center}, \text{right\_hip}). \tag{11}$$

In Equations (9) and (10), the midpoints of shoulders and hips are calculated using the method of averaging. In Equation (11), β represents the body rotation angle. When $\beta = 0°$, it indicates that the line connecting the shoulders is completely parallel to the line connecting the hips, and the body has not rotated. The larger the value of β, the greater the trunk rotation angle, and the more fully the body twists, which allows for better force generation during a swing.

The function "_get_knee_bend" is implemented in Python to calculate the degree of knee bending. This paper select four key points: left_hip = (x_{lh}, y_{lh}), right_hip = (x_{rh}, y_{rh}), left_knee = (x_{lk}, y_{lk}) and right_knee = (x_{rk}, y_{rk}). Additionally, to reduce the differences in ankle positions caused by toeing off during a swing and to better mitigate the accuracy drop due to self-occlusion during the swing, virtual points virtual_lpoint = $(x_{lk}, y_{lk} - 0.1)$ and virtual_rpoint = $(x_{rk}, y_{rk} - 0.1)$ were introduced. These points are located 10% below the knees in the image frame, based on gravity.

$$\gamma_L = \text{calculate\_angle}(\text{left\_hip}, \text{left\_knee}, \text{virtual\_lpoint}) \tag{12}$$

$$\gamma_R = \text{calculate\_angle}(\text{right\_hip}, \text{right\_knee}, \text{virtual\_rpoint}) \tag{13}$$

$$\gamma = \min(\gamma_L, \gamma_R) \tag{14}$$

In Equation (12), γ_L represents the calculated left knee bending angle. The calculate_angle calculates the angle between left_hip and virtual_lpoint with left_knee as the center. Similarly, with right_knee as the center, the right knee bending angle γ_R between right_hip and virtual_rpoint can be calculated using Equation (13). In Equation (14), the minimum value between γ_L and γ_R better represents the degree of leg bending in the human body. The minimum value between γ_L and γ_R is taken as the knee bending angle γ. The ideal range of γ during the human body's energy storage phase is 120° to 140°. When $\gamma < 110°$, it affects the rapid movement of human body, and when $\gamma > 150°$, it is usually considered insufficient force exertion by the human body.

The function "_get_wrist_speed" is implemented in Python to calculate the two-dimensional instantaneous velocity v_{wrist} of hand in the screen coordinate system. This paper selects two joint points: left_wrist = (x_{lw}, y_{lw}) and right_wrist = (x_{rw}, y_{rw}). The calculation process is divided into five steps as follow:

$$\triangle d_{lw} = \sqrt{(x_{lw}^{current} - x_{lw}^{prev})^2 + (y_{lw}^{current} - y_{lw}^{prev})^2} \tag{15}$$

$$\triangle d_{rw} = \sqrt{(x_{rw}^{current} - x_{rw}^{prev})^2 + (y_{rw}^{current} - y_{rw}^{prev})^2} \tag{16}$$

$$v_{lw} = \frac{\triangle d_{lw}}{\triangle t} \tag{17}$$

$$v_{rw} = \frac{\triangle d_{rw}}{\triangle t} \tag{18}$$

$$v_{wrist} = \max(v_{lw}, v_{rw}) \tag{19}$$

in Equation (15), $\triangle d_{lw}$ represents displacement of left wrist over $\triangle t$, $x_{lw}^{current}$ is the normalized x-axis coordinate of left wrist at the end of action, x_{lw}^{prev} is the normalized x-axis coordinate of left wrist at the start of action, and y-axis coordinates are similar. In Equation (16), $\triangle d_{rw}$ represents the displacement of right wrist over $\triangle t$, $x_{rw}^{current}$ is the normalized x-axis coordinate of right wrist at the end of action, x_{rw}^{prev} is the normalized x-axis coordinate of right wrist at the start of action, and y-axis coordinates are similar. In Equation (17), v_{lw} represents the speed of left wrist over $\triangle t$; In Equation (18), v_{rw} represents the wrist speed of right hand during $\triangle t$. In Equation (19), the maximum value between v_{lw} and v_{rw} better represents the wrist speed of racket-holding hand, allowing for the determination of swing speed based on the wrist speed. The maximum value between v_{lw} and v_{rw} is used to determine the wrist speed v_{wrist}. However, it is important that the wrist speed is determined based on screen coordinate system, and the actual physical speed should also consider the size of video pixels.

The function "_get_balance_score" is implemented in Python, which calculates the human balance score ξ. This paper select the center points of left and right shoulders as shown in Fomula (9); the center points of left and right hip joints derived from Equation (10); and the left and right ankles, left_ankle = (x_{la}, y_{la}) and right_ankle = (x_{ra}, y_{ra}). There are four key points in total, and the calculation process is divided into three steps as follow:

$$L_{alignment} = |x_s - x_h| \tag{20}$$

$$L_{ankle} = |x_{la} - x_{ra}| \tag{21}$$

$$\xi = 1 - \min\left(\frac{L_{alignment}}{L_{ankle}}, 1\right). \tag{22}$$

In Equation (20), L alignment represents the horizontal offset between center of shoulders and center of hips when a person stands, reflecting the degree of trunk tilt. In Equation (21), L_{ankle} is the horizontal distance between two ankles, indicating the width of support base. A smaller ratio of $L_{alignment}$ to L_{ankle} indicates a more balanced trunk position. In Equation (22), ξ represents the human body balance score, with a range from 0 to 1, where 1 indicates perfect balance and 0 indicates complete imbalance.

3.4.2. Motion Assessment and Suggestion

At the end of each training epoch, the loss value on the dataset was calculated and recorded. The cross-entropy loss function, as shown in Equation (1), was employed. This function evaluates the model's performance by measuring the difference between the model's predicted values and the true values. Figure 9 illustrates how the loss function varies with the training epochs for different combinations of learning_rate and max_depth on the same dataset. It shows that the model gradually converges during training. When the learning_rate is set to 0.2 and max_depth is set to 7, the loss function decreases most rapidly, indicating the best fitting effect of the GBDT model. When the number of trees reaches 125, the loss function converges and becomes stable.

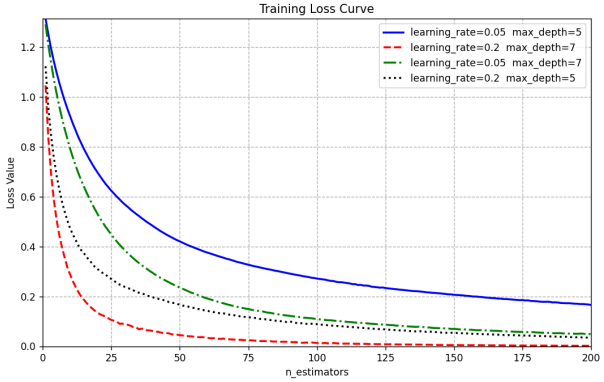


Figure 9. Training Loss Curve.

4. Results and Analysis

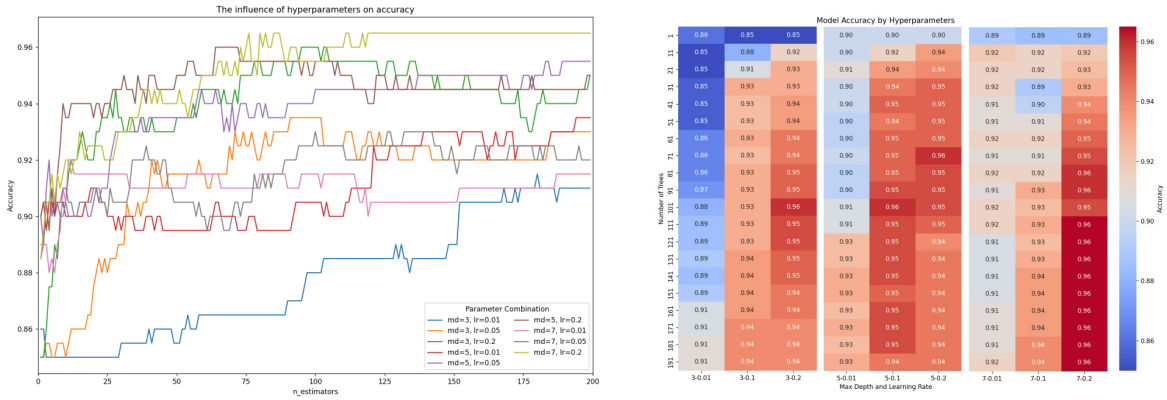
The hardware facilities and software environment used in the tennis motion correction approach of this paper are shown in Table 4.

Table 4. Hardware Facilities and Software Environment.

	Name	Model
Hardware Facilities	CPU	Intel i5 13490f
	GPU	RTX 4060 8G
	RAM	32 GB
Software Environment	Operating System	Windows 11
	Development Environment	Python 3.11
	Dependency Libraries	Mediapipe and OpenCV

4.1. GBDT Hyper Parameter Ablation Study

This paper conducts ablation study on three hyper parameters in the model, namely the number of trees (n_estimators), maximum depth of trees (max_depth), and learning rate (learning_rate), to study their impact on classification accuracy. The results of ablation study are shown in Figure 10a. Different combinations of hyper parameters are selected for training over 200 generations. When trained up to 125 generations, the model with a learning rate of 0.2 and a tree depth of 7 achieves the highest accuracy and tends to stabilize. However, the model with a learning rate of 0.2 and a tree depth of 5 has the fastest increase in accuracy in the early stages of training when the number of training generations is small, but later on, due to overfitting, the accuracy shows a downward trend. To more intuitively present the experimental results, this paper also draws an ablation study heatmap, as shown in Figure 10b. The heatmap shows the coherent changes in model accuracy under different combinations of max_depth and learning_rate. The heatmap illustrates that changes in accuracy are coherent with parameters. The combination of max_depth = 7 and learning_rate = 0.2 generally achieves better and more stable performance in most cases, with higher model prediction accuracy.



(**a**) Result of ablation study (**b**) Heatmap of ablation study

Figure 10. Result of ablation study under different hyperparameters.

4.2. Results of Tennis Stroke Correction Experiment

As shown in Figure 11, considering the impact of various uncertain factors in real scenarios, this paper selects a tennis court with soft lighting to conduct experimental verification of motion correction approach. The figure displays test interface for forehand stroke. The test video is demonstrated in Supplementary Material section. The skeleton and keypoints of trainee's body are identified by Mediapipe. Based on the identified skeleton and keypoints, the feature angles of tennis stroke are calculated, and correction approaches for forehand stroke process are provided through suggestions. Trainees are able to adjust their tennis actions in accordance with the provided suggestions.

Figure 11. Forehand stroke test.

5. Conclusions and Future Work

This paper proposes a tennis stroke correction approach based on ensemble learning, which effectively identifies and corrects non-standard actions in tennis. A tennis basic action classification model is successfully constructed to reduce labeling costs by using Gradient Boosting Decision Tree (GBDT). In addition, this motion correction approach integrates the advantages of indoor motion capture systems and outdoor Kinect cameras to improve the applicability and accuracy of motion capture. Mediapipe is used to analyze the human skeleton in real-time tennis action videos captured by the outdoor Kinect camera, enhancing the real-time nature of tennis posture detection and the ability to handle self-occlusion. By extracting eight key joints of the human body and combining them with eight tennis action features, the actions are evaluated, and reasonable correction suggestions are provided based on deficiencies of actions. Experiments show that the poposed motion correction approach provides effective action guidance for tennis trainees at different learning stages. In terms of model optimization, this paper systematically compares and verifies hyperparameters of GBDT model through ablation experiments.

In the future work, the tennis motion correction approach will be advanced by deeply integrating the Dynamic Time Warping (DTW) algorithm with deep learning. This integration forms a more efficient and accurate temporal model, significantly boosting correction accuracy and real-time capabilities.

Supplementary Materials

The additional data and information can be downloaded at: https://media.sciltp.com/articles/others/25082514223840 91/video.mp4.

Author Contributions

Y.G.: Writing—original draft, Investigation, Conceptualization. C.C.: Writing-Validation, Software, Methodology, review & editing, Investigation. X.W.: Writing—review & editing. Y.C.: Writing-review & editing, Supervision. H.C.: Resources & supervision. All authors have read and agreed to the published version of the manuscript.

Funding

This work is supported in part by the National Natural Science Foundation of China (NSFC) under Grant 72201186, 52205027, 62303308, in part by the Natural Science Foundation of Jiangsu Province under Grant BK20220481, in part by the Natural Science Foundation of the Jiangsu Higher Education Institutions of China under Grant 22KJB410002, in part by China Postdoctoral Science Foundation under Grant 2023M731450, in part by Basic Science Center Program of NSFC under Grant 62388101, in part by Shanghai Pujiang Program under Grant 23PJ1404700, in part by Open Research Project of the State Key Laboratory of Industrial Control Technology under Grant ICT2024B15, in part by Joint Research Fund of Shanghai Academy of Spaceflight Technology under Grant USCAST2023-22, and in part by Supported by the Open Fund of Intelligent Control Laboratory under Grant 2024-ZKSYS-KF03-01.

Data Availability Statement

We have uploaded the program to the Github website. Here is the website address: https://github.com/ccx031/Motion-correction.

Conflicts of Interest

The authors declare no conflict of interest.

References

1. Mei, Z. 3D Image Analysis of Sports Technical Features and Sports Training Methods Based on Artificial Intelligence. *J. Test. Eval.* **2022**, *51*, 189–200.
2. Kulkarni, K.M.; Shenoy, S. Table Tennis Stroke Recognition Using Two-Dimensional Human Pose Estimation. In Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Nashville, TN, USA, 19–25 June 2021; pp. 4571–4579.
3. Kwon, S.; Letuchy, E.M.; Levy, S.M.; et al. Youth Sports Participation Is More Important among Females than Males for Predicting Physical Activity in Early Adulthood: Iowa Bone Development Study. *Int. J. Environ. Res. Public Health* **2021**, *18*, 1328.
4. Zhang, J.; Tao, D. Empowering Things with Intelligence: A Survey of the Progress, Challenges, and Opportunities in Artificial Intelligence of Things. *IEEE Internet Things J.* **2021**, *8*, 7789–7817.
5. Andriluka, M.; Iqbal, U.; Insafutdinov, E.; et al. PoseTrack: A Benchmark for Human Pose Estimation and Tracking. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
6. Shotton, J.; Fitzgibbon, A.; Cook, M.; et al. Real-time human pose recognition in parts from single depth images. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 1297–1304.
7. Girdhar, R.; Gkioxari, G.; Torresani, L.; et al. Detect-and-Track: Efficient Pose Estimation in Videos. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018.
8. Andriluka, M.; Pishchulin, L.; Gehler, P.; et al. 2D Human Pose Estimation: New Benchmark and State of the Art Analysis. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014.
9. Chen, Y.; Shen, C.; Wei, X.-S.; et al. Adversarial PoseNet: A Structure-Aware Convolutional Network for Human Pose Estimation. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017.
10. Fischler, M.A.; Elschlager, R.A. The Representation and Matching of Pictorial Structures. *IEEE Trans. Comput.* **1973**, *100*, 67–92.
11. Forsyth, D. Object Detection with Discriminatively Trained Part-Based Models. *Computer* **2014**, *47*, 6–7.
12. Zhang, C.-L.; Li, Y.; Wu, J. Weakly supervised foreground learning for weakly supervised localization and detection. *Pattern Recognit.* **2023**, *137*, 109279.
13. Patel, C.I.; Labana, D.; Pandya, S.; et al. Histogram of Oriented Gradient-Based Fusion of Features for Human Action Recognition in Action Video Sequences. *Sensors* **2020**, *20*, 7299.
14. Luo, M.; Du, B.; Zhang, W.; et al. Fleet Rebalancing for Expanding Shared e-Mobility Systems: A Multi-Agent Deep Reinforcement Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 3868–3881.
15. Qiao, Y.; Li, K.; Lin, J.; et al. Robust Domain Generalization for Multi-modal Object Recognition. In Proceedings of the 2024 5th International Conference on Artificial Intelligence and Electromechanical Automation (AIEA), Shenzhen, China, 14–16 June 2024; pp. 392–397.
16. Yang, Y.; Yang, R.; Pan, L.; et al. A lightweight deep learning algorithm for inspection of laser welding defects on safety vent of power battery. *Comput. Ind.* **2020**, *123*, 103306.
17. Chen, Y.; Zhang, F.; Wang, G.; et al. An Active Contour Model Based on Fuzzy Superpixel Centers and Nonlinear Diffusion Filter for Instance Segmentation. *IEEE Trans. Instrum. Meas.* **2025**, *74*. https://doi.org/10.1109/TIM.2025.3573369
18. Cheng, C.; Zhang, H.; Sun, Y.; et al. A cross-platform deep reinforcement learning model for autonomous navigation without global information in different scenes. *Control Eng. Pract.* **2024**, *150*, 105991.
19. Cao, J.; Gao, Y.; Wang, C. A Novel Four-Step Algorithm for Detecting a Single Circle in Complex Images. *Sensors* **2023**, *23*, 9030.
20. Dang, Q.; Yin, J.; Wang, B.; et al. Deep learning based 2D human pose estimation: A survey. *Tsinghua Sci. Technol.* **2019**, *24*, 663–676.
21. Wang, G.; Zhang, F.; Chen, Y.; et al. An Active Contour Model Based on Local Pre-piecewise Fitting Bias Corrections for Fast and Accurate Segmentation. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 1–13.
22. Munea, T.L.; Jembre, Y.Z.; Weldegebriel, H.T.; et al. The Progress of Human Pose Estimation: A Survey and Taxonomy of Models Applied in 2D Human Pose Estimation. *IEEE Access* **2020**, *8*, 133330–133348.
23. Wang, G.; Li, Z.; Weng, G.; et al. An overview of industrial image segmentation technology using deep learning models. *Intell. Robot.* **2025**, *5*, 143–180.
24. Gao, Y.; Cheng, X.; Chen, Y.; et al. Data-driven propagation and recovery of supply-demand imbalance in a metro system. *Control Eng. Pract.* **2025**, *161*, 106339.

25. Cao, Z.; Hidalgo, G.; Simon, T.; et al. OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 172–186.
26. Wu, P.; Tian, E.; Tao, H.; et al. Transfer Learning-Motivated Intelligent Fault Diagnosis Framework for Cross-Domain Knowledge Distillation Learning Systems. *Neural Netw.* **2025**, *190*, 107699.
27. Sun, K.; Xiao, B.; Liu, D.; et al. Deep High-Resolution Representation Learning for Human Pose Estimation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 5686–5696.
28. Li, Z.; Zhang, F; Wang, G.; et al. An Active Contour Model Based onKullback-Leibler Divergence and Morphologyfor lmage Segmentation with Edge Leakage. *Signal Process.* **2026**, *238*, 110143.
29. Latreche, A.; Kelaiaia, R.; Chemori, A.; et al. Reliability and validity analysis of MediaPipe-based measurement system for some human rehabilitation motions. *Measurement* **2023**, *214*, 112826.
30. Konstantinov, A.V.; Utkin, L.V. Interpretable machine learning with an ensemble of gradient boosting machines. *Knowl.-Based Syst.* **2021**, *222*, 106993.