Research On Intelligent Body Measurement Algorithm Based on AI Attitude Recognition

Zhou Wu *

School of Sino-German Robotics, Shenzhen University of Information Technology, Shenzhen, China, 518172

* Corresponding Author Email: wuz@sziit.edu.cn

Abstract. To solve the problems of difficult quantitative analysis of standing long jump movements, unclear factors affecting performance, and lack of personalized training programs, this paper conducts research based on AI human posture estimation technology to obtain key node coordinate data of athletes. To address the challenges of data noise and action stage segmentation, sliding window smoothing is used to process data, and dynamic ground baselines are constructed by combining foot nodes. Weighted fusion features are used to accurately determine takeoff and landing times, and the hovering process is quantified from dimensions such as center of gravity displacement and joint angles; To meet the needs of performance prediction and training optimization, a prediction model is constructed based on gradient boosting regression. The simulation results show a high accuracy in predicting the actual long jump performance of athletes. In the field of sports, the research findings of this article can be extended to other track and field events such as high jump, throwing, and sprinting, as well as the analysis of technical movements in ball sports such as basketball and football.

Keywords: Smooth Sliding Window, Lleave One Cross Validation, Gradient Boosting Regression.

1. Introduction

With the rapid development of artificial intelligence and computer vision technology, AI assisted intelligent body measurement systems have shown great potential for application in fields such as school sports health monitoring and scientific sports training. As a core project for evaluating human explosiveness, coordination, and physical function, standing long jump relies on manual observation and subjective judgment in its traditional assessment, which has problems such as inaccurate action analysis, insufficient quantification of performance influencing factors, and lack of personalized training programs. The current mainstream physical fitness testing systems tend to focus on recording performance, making it difficult to capture key technical details such as takeoff angle and joint coordination, and unable to provide targeted optimization solutions based on the physical characteristics of athletes, resulting in low training efficiency and limited performance improvement. How to achieve precise analysis of standing long jump movements, accurate identification of core influencing factors, and scientific development of personalized training through AI posture recognition technology has become a key issue in promoting the intelligence of physical testing and scientific training.

In recent years, scholars both domestically and internationally have conducted extensive research on posture recognition and motion analysis. The computer vision intelligent body measurement system developed by Gu Xuan et al. [1] achieves automated collection of body measurement data through key point capture, but its ability to dynamically analyze the motion process and predict performance is weak. Qi HX et al. [2] conducted experiments on the aerial jumping of humanoid robots using human motion capture technology, and verified the correlation analysis method between joint angles and aerial trajectories, providing technical reference for the biomechanical analysis of human jumping motion. Huang et al. [3] established an artificial intelligence evaluation model for standing long jump using medical biomechanical methods, clarifying the correlation between some physical fitness indicators and athletic performance, but did not form a complete closed loop of "analysis prediction training". The Otsu fast image segmentation method based on least squares fitting by Xu J D [4] provides technical support for feature extraction of motion images, but it has not been fully validated in the application of stage partitioning of motion time-series data. In addition, foreign scholars such as Chen X [5] used gradient boosting regression models to predict track and field performance, and their feature fusion strategy provided reference for the construction of the model in this paper; Kim J H et al. [6] identified the core influencing factors of sports performance through factor analysis and confirmed the effectiveness of dimensionality reduction for physical fitness indicators. Overall, existing research has made some progress in posture capture and single link analysis, but there are still shortcomings in precise segmentation of action stages, integration analysis of physical and technical parameters, and generation of personalized training plans. It is urgent to build an integrated analysis and optimization framework. The research team at Carnegie Mellon University (CMU) in the United States [7] used convolutional neural networks and supervised learning techniques to construct an open dataset using Caffe, which can effectively detect human behavior such as body movements, facial expressions, finger movements, etc. This two-dimensional pose estimation application adopts deep learning technology, which can simultaneously support the recognition of individuals or multiple targets. Its high robustness makes it the world's first two-dimensional pose estimation technology achieved through deep learning. Hu Jianlang et al. [8] developed a three-dimensional pose evaluation method based on a binocular zoom servo system and attention unit tracking mechanism to address issues such as non-standard and unsupervised motion postures. Wang Yuan [9] designed and trained a badminton player posture evaluation model based on the concepts of similarity and local evaluation. Yang Jun et al. [10] developed a monocular image recognition comparison guidance system using openpose technology.

This article focuses on the intelligent analysis and training optimization of standing long jump, and conducts the following research: Firstly, in response to the noise interference in motion data and the difficulty of dividing action stages, sliding window smoothing and median filtering are used for data preprocessing. Combined with the key nodes of the feet, a dynamic ground baseline is constructed, and the takeoff and landing times are accurately determined through weighted fusion features. The hovering process is quantified from dimensions such as center of gravity trajectory, joint angle, and posture features; Secondly, a performance prediction model is constructed based on gradient boosting regression, and the effectiveness of the model is verified by leaving one cross validation to achieve accurate prediction of athletes' performance.

2. Data preprocessing and attitude model establishment

2.1. Data preprocessing

We tracked and filmed multiple athletes who were performing standing long jump, and extracted their position information from each frame as the data source for this article. To address the challenges of noise interference in motion data and the division of action stages, sliding window smoothing and median filtering are first used for data preprocessing. A dynamic ground baseline is constructed by combining key foot nodes, and the takeoff and landing times are accurately determined through weighted fusion features. The hovering process is quantified from dimensions such as center of gravity trajectory, joint angle, and attitude features.

The original key node coordinates of the athlete may have noise. The 5-frame sliding average method is used to smooth all horizontal and vertical coordinate fields, and the formula is as follows:

$$x_{smooth}(i)=\frac{1}{5}\sum_{k=-2}^{2} x\,(i+k) \tag{1}$$

Among them, $x(i)$ is the original coordinate of the i-th frame, $x_{smooth}(i)$ is the smoothed coordinate, and the edge frame is filled with the minimum period to avoid data loss. When athletes perform standing long jump, they will complete different movements at different stages of the standing long jump. The body data and characteristics of athletes are different when performing these different movements. Therefore, it is necessary to establish a body model for athletes at different stages of standing long jump in order to more accurately predict their long jump performance and provide more suitable training recommendations for athletes. According to the characteristics of standing long jump, it can be divided into 5 stages, and 5 body models can be established for athletes. Firstly, we selected 32 key nodes on the athlete as feature data collection points, as shown in Figure 1.

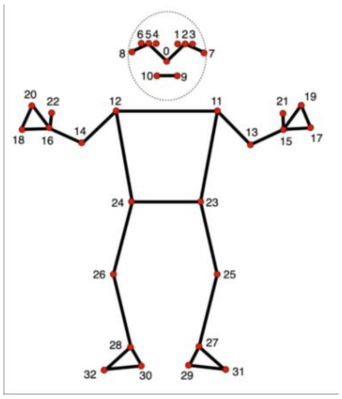

Figure 1. Key node diagram for athletes

2.2. Static stage model

BP neural network is back propagating, mainly composed of three parts: input layer, middle layer and output layer. The number of nodes in the input and output layers is relatively easy to determine, but the determination of the number of nodes in the hidden layer is a very important and complex problem.

Select 6 key foot nodes ($p\in\{27,28,29,30,31,32\}$) for athletes, calculate the comprehensive foot velocity for each frame, and quantify the foot movement status:

$$v_{foot}(f)=\frac{1}{6}\sum_{p\in foot}\sqrt{\left(\Delta\hat{x}_p(f)\right)^2+\left(\Delta\hat{y}_p(f)\right)^2} \tag{2}$$

Among them $\Delta\hat{y}_p(f)=\hat{y}_p(f)-\hat{y}_p(f-1)$. $\Delta\hat{y}_p(f)$ is the horizontal and vertical displacement difference of the p-th foot node in the f-th frame reflects the inter frame motion amplitude of the node.

Static stage determination: The standard deviation calculated for a sliding window with a window size of w=5. If the standard deviation within the window is less than the threshold, it is determined as a potential static segment; Select the longest potential stationary segment as the final stationary segment, where is the starting frame and is the ending frame.

2.3. Dynamic ground baseline model

To eliminate the influence of site tilt or image acquisition angle deviation on ground contact determination, 6 key foot nodes ($p\in\{27,28,29,30,31,32\}$) are still selected to construct a dynamic ground baseline.

Firstly, based on the stationary segment S, calculate the median Y coordinates of the left and right feet, and linearly interpolate to obtain the baseline for the first k=3 frames.

$$m_{left}=median\left(\hat{y}_p(f)\mid p\in\{27,29,31\}, f\in S\right) \tag{3}$$

$$m_{right}=median\left(\hat{y}_p(f)\mid p\in\{28,30,32\}, f\in S\right) \tag{4}$$

$$B_0(f)=m_{left}+\frac{m_{right}-m_{left}}{29}\cdot f \quad (f=0,1,\ldots,29) \tag{5}$$

Subsequent frames use a sliding window with a window size of w=30 to update the baseline with the median of the average vertical coordinates of the feet, ensuring that the baseline dynamically adjusts with the scene.

$$B(f) = median\left(\hat{y}_{foot}(f - w/2\colon f + w/2)\right) \tag{6}$$

Among them, $\hat{y}_{\mathrm{foot}}(\mathrm{f}) = \frac{1}{6}\sum_{\mathrm{p}\in\mathrm{foot}} \hat{y}_{\mathrm{p}}(\mathrm{f})$ is the average vertical coordinate of the foot.

2.4. Jump time determination model

The take-off moment is defined as the moment when the foot is completely off the ground and the center of gravity is accelerating upwards. Robust determination is achieved through four core features, dynamic thresholds, and weighted voting.

Foot to ground distance: measures the vertical distance between the overall foot and the dynamic ground baseline, reflecting the degree to which the foot is off the ground. A distance>0 indicates that the foot is above the ground, otherwise it is in contact or below the ground. The calculation formula is:

$$d_{foot}(f) = \max\left(0, B(f) - \hat{y}_{foot}(f)\right) \tag{7}$$

Among them, B(f) is the dynamic ground baseline of the f-th frame, representing the vertical position of the "ground" in that frame; $\hat{y}_{foot}(f)$ is the vertical coordinate mean of the six-foot keypoints (ankle, heel, toe) in frame f.

Vertical velocity of the center of gravity: Based on the vertical displacement difference of the human body's center of gravity, it reflects the direction and speed of the center of gravity's movement. Using the vertical coordinates of the left and right hips (23, 24) and the left and right shoulders (key nodes 11, 12), calculate the vertical coordinates of the center of gravity of the human body through weighted averaging. The formula is:

$$\hat{y}_{com}(f) = 0.6 \cdot \frac{\hat{y}_{23}(f) + \hat{y}_{24}(f)}{2} + 0.4 \cdot \frac{\hat{y}_{11}(f) + \hat{y}_{12}(f)}{2} \tag{8}$$

Among them, $\hat{y}_{23}(\mathrm{f})$ and $\hat{y}_{24}(\mathrm{f})$ are the vertical coordinates of the left and right hips, and $\hat{y}_{11}(f)\ and\ \hat{y}_{12}(f)$ are the vertical coordinates of the left and right shoulders. Due to the fact that the center of gravity of the human body is closer to the hip, a weight of 0.6 is assigned to the hip coordinates and a weight of 0.4 is assigned to the shoulder coordinates. The weighted average is more in line with the actual center of gravity position.

Calculate the vertical velocity through the frame difference of the vertical coordinate of the center of gravity, and the formula is:

$$v_{com,y}(f) = \Delta\hat{y}_{com}(f) \tag{9}$$

Due to the downward direction of the y-axis in the pixel coordinate system, if $\mathrm{v}_{\mathrm{com,y}}(\mathrm{f}) < 0$, it indicates that the center of gravity is moving upwards; Otherwise, it indicates a downward movement of the center of gravity.

Footstep horizontal acceleration: Calculate the inter frame difference of foot horizontal velocity, reflecting the change in acceleration in the horizontal direction. The calculation formula is:

$$v_{foot,x}(f) = \frac{1}{6}\sum_{p\in foot}\left(x_p(f) - x_p(f-1)\right) \tag{10}$$

$$a_{foot,x}(f) = \Delta v_{foot,x}(f) \tag{11}$$

Ground contact ratio: For each foot node p, if the absolute value of the difference between its vertical coordinates $\hat{y}_{\mathrm{p}}(\mathrm{f})$ and the dynamic ground baseline B(f) is less than the foot ground distance threshold T_{d}, then the node is considered to be in contact with the ground.

Ground clearance threshold: $\mathrm{T}_{\mathrm{d}} = 3 \cdot \mathrm{std}(\hat{y}_{\mathrm{foot}}(\mathrm{f}) \mid \mathrm{f} \in \mathrm{S})$, where std is the standard deviation of the average vertical coordinates of the foot during the stationary phase. The threshold is set at 3 times the fluctuation of the stationary segment because the slight shaking of the foot during stationary will not exceed this range, and only when actively lifting off the ground will it significantly exceed this threshold.

Center of gravity velocity threshold: $T_v = 2 \cdot mean\big(|v_{com,y}(f)| \mid f \in S\big)$ calculate the mean absolute value of the center of gravity vertical velocity during the stationary phase S and multiply it by 2 as the threshold. When at rest, the velocity of the center of gravity is close to 0. When taking off (with the center of gravity moving upwards, the velocity is negative and the absolute value is large), it will significantly exceed twice the absolute value of the average velocity at rest, in order to distinguish between active takeoff and static shaking.

Horizontal acceleration threshold: $\mathrm{T_a} = 1.5 \cdot \max\big(|\mathrm{a_{foot,x}(f)}| \mid \mathrm{f} \in \mathrm{S}\big)$ calculate the maximum absolute value of the horizontal acceleration of the foot during the stationary phase, and multiply it by 1.5 as the threshold. When the acceleration is close to 0 at rest, the acceleration during takeoff will far exceed 1.5 times the maximum static acceleration, in order to capture the horizontal acceleration characteristics before takeoff.

Contact ratio threshold: $\mathrm{T_r} = \frac{0.5 \cdot \mathrm{mean(Number\ of\ contact\ nodes|f \in S)}}{6}$ calculate the average number of contact nodes during the stationary phase, take 50% of it, and divide by the total number of nodes 6 to obtain the ratio threshold. When stationary, most of the foot nodes are in contact with the ground, with a relatively high proportion; The contact ratio during takeoff will be lower than the ratio corresponding to 50% of the static average contact number, in order to determine whether the foot is significantly off the ground.

Weighted fusion judgment logic: Normalize the four features (positive feature min max normalized to [0,1], contact ratio reverse normalized), and assign weights based on biomechanical importance: Foot ground clearance distance (0.35): directly reflects the core state of "ground clearance"; Vertical velocity of center of gravity (0.25): reflects the take-off trend of "upward acceleration"; Foot horizontal acceleration (0.20): Quantify the strength of the pedaling force; Ground contact ratio (0.20): Assist in verifying the degree of ground clearance. The formula for calculating the comprehensive score is:

$$S(f) = 0.35 \cdot S_d(f) + 0.25 \cdot S_v(f) + 0.20 \cdot S_a(f) + 0.20 \cdot S_r(f) \tag{12}$$

Among them, $S_d(f)$、$S_v(f)$、$S_a(f)$、$S_r(f)$ are the normalized feature scores, respectively. The frame with the highest comprehensive score within the candidate range of 40 frames after the start frame of the movement is the take-off moment.

2.5. Landing time determination model

The landing moment is the moment when the foot returns to contact the ground and the center of gravity moves downwards. Based on the take-off determination model framework, it is adapted in reverse, while reusing core parameters such as the stationary segment S and dynamic baseline ($B(f)$):

Foot proximity distance: The formula for determining whether the foot is close to the ground is as follows:

$$d_{land}(f) = B(f) - y_{foot}(f) \tag{13}$$

Threshold value $T_{land,h} = 1.5 \cdot std\{y_{foot}(f) \mid f \in S\}$, taking 1.5 times the stationary segment fluctuation as the threshold can accurately capture small distance changes at the moment of contact with the ground. When $d_{land}(f) < T_{land,h}$, it is judged as "close to the ground".

Downward velocity of the center of gravity: Determine whether the center of gravity is moving downward, using the vertical velocity of the center of gravity $v_{com,y}(f)$, with a threshold value of $T_{land,v} = 2 \cdot mean\{|v_{com,y}(f)| \mid f \in S\}$ (consistent with the takeoff threshold value). When $v_{com,y}(f) > T_{land,v}$, it is judged as "downward motion".

Ground contact ratio: Determine whether the foot has returned to contact, using the contact ratio $r_{contact}(f)$, with a threshold value of $T_{land,r} = T_r$ (consistent with the take-off threshold). When $r_{contact}(f) > T_{land,r}$, it is determined that the proportion of contact nodes meets the standard.

Physical constraint: To avoid misjudging a slight drop after takeoff as landing, a stagnation time constraint of "$\mathrm{t_{down}} > \mathrm{t_{up}} + 5$" is set to ensure that the landing frame is within a sufficiently long gap after the takeoff frame.

Weighted fusion judgment logic: Normalize three core features (reverse min max normalization is used for foot ground proximity distance, forward min max normalization is used for center of gravity vertical velocity and ground contact ratio, and weight allocation is based on biomechanical importance: foot ground proximity distance (0.4) directly reflects the "ground proximity" core state; Vertical velocity of the center of gravity (0.3), reflecting the landing trend of "downward movement"; The ground contact ratio (0.3) assists in verifying the degree of foot contact with the ground, and the comprehensive score calculation formula is:

$$S_{land}(f) = 0.40 \cdot S_d(f) + 0.30 \cdot S_v(f) + 0.30 \cdot S_r(f) \tag{14}$$

Among them, $S_d(f)$、$S_v(f)$、$S_r(f)$ are the normalized scores of foot proximity to the ground, vertical velocity of the center of gravity, and ground contact ratio, respectively.

2.6. Stagnation stage motion model

The hovering stage is the core process of the standing long jump, which involves expanding the body and completing the movement. It requires three types of indicators: center of gravity trajectory, joint angle, and posture characteristics, to achieve quantitative description from three dimensions: overall motion trend, local joint posture, and body balance control.

Center of gravity trajectory parameters: The center of gravity is the core reference point of human motion, and its trajectory directly reflects the aerial effect of long jump. It is quantified by the following parameters:

Horizontal displacement: Calculate the change in the horizontal coordinates of the center of gravity between the takeoff time t_{up} and the landing time t_{down}, using the following formula:

$$\Delta x_{com} = \hat{x}_{com}(t_{down}) - \hat{x}_{com}(t_{up}) \tag{15}$$

Among them, $\hat{x}_{com}(f)$ is the horizontal coordinate of the center of gravity in frame f (obtained by weighted average of hip and shoulder nodes), which reflects the "long-range potential" of long jump. The larger the value, the farther the center of gravity moves horizontally.

Maximum airborne height: Calculate the difference between the maximum vertical coordinate of the center of gravity during the hovering phase and the vertical coordinate of the center of gravity during takeoff. The formula is:

$$h_{com} = \max\big(\hat{y}_{com}(f) \mid f \in [t_{up}, t_{down}]\big) - \hat{y}_{com}(t_{up}) \tag{16}$$

Among them, $\hat{y}_{com}(f)$ is the vertical coordinate of the center of gravity in the f-th frame, which reflects the height potential of long jump. The larger the value, the higher the center of gravity is lifted.

Trajectory angle: The arctangent of the ratio of maximum aerial height to horizontal displacement is used to reflect the steepness of the center of gravity trajectory. The formula is:

$$\theta_{com} = \arctan\left(\frac{h_{com}}{\Delta x_{com}}\right) \times \frac{180^{\circ}}{\pi} \tag{17}$$

The larger the trajectory angle, the higher the proportion of height during the airborne process; The smaller the angle, the higher the proportion of distance, which can intuitively reflect the "focus direction" of the long jump trajectory.

Joint angle parameter: Joint angle is a direct reflection of the "local motion posture" of the human body. Taking the right knee joint as an example (the linkage of the right hip, right knee, and right ankle is the core of the long jump suspended body), the vector dot product method is used to calculate the angle.

Vector construction: Using the right knee (node 26) as the joint center, construct two vectors to describe the relative position of the thigh and calf:

1) Thigh vector $\overrightarrow{BA}$: pointing from the right hip (node 24) to the right knee, coordinates are:

$$\overrightarrow{BA} = (\hat{x}_{24}(f) - \hat{x}_{26}(f), \hat{y}_{24}(f) - \hat{y}_{26}(f)) \tag{18}$$

2) Calf vector $\overrightarrow{BC}$: pointing from the right ankle (node 28) to the right knee, coordinates are:

$$\overrightarrow{BC} = (\hat{x}_{28}(f) - \hat{x}_{26}(f), \hat{y}_{28}(f) - \hat{y}_{26}(f)) \tag{19}$$

3) Angle calculation: Using the relationship between vector dot product and cosine of angle, calculate the right knee joint angle $\theta_{knee}(f)$, formula:

$$\theta_{knee}(f) = \arccos\left(\frac{\overrightarrow{BA}\cdot\overrightarrow{BC}}{|\overrightarrow{BA}|\cdot|\overrightarrow{BC}|}\right) \times \frac{180°}{\pi} \tag{20}$$

Among them, $\overrightarrow{BA} \cdot \overrightarrow{BC}$ is the dot product of vectors, and $|\overrightarrow{BA}| \cdot |\overrightarrow{BC}|$ is the modulus of vectors.

Similarly, the right hip joint angle $\theta_{hip}(f)$ can be calculated to construct a vector with the right shoulder, right hip, and right knee as nodes. The final output includes the range and average value of joint angles during the idle phase, with the former reflecting the "range of motion" of the joints and the latter reflecting the "stability and habitual trend" of the posture.

Attitude feature parameters: Attitude features describe the control effect of the overall posture of the human body, taking the forward tilt angle and arm swing amplitude as examples, quantifying body balance and motion coordination:

Body forward tilt angle: Construct a torso vector (from shoulder to hip) and a vertical vector, and reflect the degree of forward/backward tilt of the body through the angle between the vectors

Trunk vector $\overrightarrow{trunk}$: pointing from the center of the shoulder (midpoint of left and right shoulder nodes 11 and 12) to the center of the hip (midpoint of left and right hip nodes 23 and 24), coordinates are:

$$\overrightarrow{trunk} = \left(\hat{x}_{shoulder}(f) - \hat{x}_{hip}(f), \hat{y}_{shoulder}(f) - \hat{y}_{hip}(f)\right) \tag{21}$$

Vertical vector $\overrightarrow{vertical} = (0, -1)$, representing the vertical upward direction.

Body forward tilt angle $\theta_{lean}(f)$:

$$\theta_{lean}(f) = \arccos\left(\frac{\overrightarrow{trunk}\cdot\overrightarrow{vertical}}{|\overrightarrow{trunk}|\cdot|\overrightarrow{vertical}|}\right) \times \frac{180°}{\pi} \tag{22}$$

The smaller the angle, the more forward leaning the body (beneficial for horizontal distance); The larger the angle, the more vertical/backward the body is, which is beneficial for maintaining height or balance.

Arm swing amplitude: Calculate the distance between the left and right arms from the wrist to the shoulder, and take the maximum value to reflect the maximum amplitude of arm swing (assisting body balance and momentum transfer):

Left arm amplitude $L_{left}(f)$: Euclidean distance from left wrist (node 15) to left shoulder (node 11);

Right arm amplitude $L_{right}(f)$: Euclidean distance from the right wrist (node 16) to the right shoulder (node 12);

The maximum swing amplitude of the arm, $L_{arm}(f) = \max\left(L_{left}(f), L_{right}(f)\right)$, ultimately outputs the range and average value of this amplitude during the hovering phase. The range reflects the "dynamic change degree" of the arm swing, and the average value reflects the "overall trend" of the arm posture.

3. Predictive model

Using a supervised learning framework, the problem of predicting grades is transformed into a regression task, and a mapping relationship from feature space to grade space is constructed.

The detection of takeoff and landing moments is a key step in feature extraction, and the calculation of takeoff angle takes into account the time averaged effect of velocity vectors. Calculate the average horizontal and vertical velocities within a window of 3 frames before and after the takeoff point, and then use the arctangent function to calculate the angle.

The model selection has undergone a systematic comparative analysis. The candidate models include linear regression, support vector regression, random forest, and gradient boosting regression tree. Through preliminary experiments, gradient boosting regression trees exhibit the best predictive performance.

The gradient boosting regression tree iteratively constructs a weak learner (regression tree), fits the prediction residuals (the difference between the actual value and the predicted value) of the previous round of the model in each iteration, and finally weights and sums the prediction results of all weak learners to obtain the final predicted value. Its core advantage lies in the ability to capture nonlinear relationships between features and minimize prediction errors through gradient descent optimization strategies.

3.1. Feature set construction

We will set seven core features in the long jump of athletes, including the idle time (T_f), takeoff angle (θ_t), height (h), weight (w), explosive power (P_e), muscle fat ratio (R_{mf}), and skeletal muscle index (SMI). The final dataset is defined as:

$$X = [T_f, \theta_t, h, w, P_e, R_{mf}, SMI] \tag{23}$$

The calculation methods for each feature data are as follows:

Flight time: $T_f = \frac{Landing\ frame - Starting\ frame\ skipping}{frame\ rate}$ (frame rate set to 30 frames per second);

Jump angle: $\theta_t = \arctan\left(\frac{\overline{v_{y,\mathrm{tk}}}}{\overline{v_{x,\mathrm{tk}}}}\right) \times \frac{180°}{\pi} (\overline{v_{x,\mathrm{tk}}}$ 、 $\overline{v_{y,\mathrm{tk}}}\}$ are the average horizontal and vertical velocities of the three frames before and after takeoff respectively. The y-axis in the pixel coordinate system is downward. Therefore, $\overline{v_{y,\mathrm{tk}}}$ is taken as negative to ensure the rationality of the angle;

Skeletal muscle index: $SMI = \frac{\text{Skeletal muscle weight(kg)}}{\left(\frac{\text{height(cm)}}{100}\right)^2}$;

Explosive power: $P_e = \frac{\text{Power output at the moment of takeoff(W)}}{\text{weight(kg)}}$ (reflecting the leg power output per unit weight);

Muscle fat ratio: $R_{mf} = \frac{\text{muscle mass percentage(\%)}}{\text{body fat percentage(\%)}+1}$.

3.2. Data Standardization

Using StandardScaler to perform Z-score normalization on all features "to eliminate the influence of dimensional differences (such as height in cm, weight in kg, angle in °) on the selection of tree model splitting thresholds. The normalization formula is:

$$x_{scaled} = \frac{x-\mu}{\sigma} \tag{24}$$

Among them, μ is the feature mean, and σ is the feature standard deviation; For example, for the "weight" feature, if the mean weight in the training set is 50.2kg and the standard deviation is 12.3kg, then a certain athlete weighs w=21kg, and after standardization, it is $w_{\text{scaled}} = \frac{21-50.2}{12.3} \approx -2.37$.

3.3. Dataset partitioning

Using Leave One Out Cross Validation (LOOCV) to evaluate model performance.

3.4. Steps of Gradient Boosting Regression Tree Algorithm(GBRTA)

Step 1: Initialize the base learner (constant model):

The initial model is a constant predictor that directly predicts the mean of the target variable (standing long jump score S) in the training set, ensuring minimal initial residuals. The formula is:

$$F_0(x) = \arg\min_{c} \sum_{i=1}^{n} L(y_i, c) \tag{25}$$

Among them, n is the number of training set samples (n=7 in each LOOCV); y_i is the actual score of the i-th sample (unit: m); $L(y_i, c) = \frac{1}{2}(y_i - c)^2$ is the squared loss function; C is a constant, and after solving, we obtain $c = \frac{1}{n}\sum_{i=1}^{n} y_i$ (i.e., the average performance of the training ensemble). For example, if the average score of 7 athletes in a certain round of LOOCV training set is 1.62m, then the initial model $F_0(x) = 1.62$.

Step 2: Iteratively build a weak learner (regression tree):

The gradient boosting regression tree algorithm constructs M regression trees through M rounds of iterations, with each iteration optimizing the residuals of the previous round. In the paper, M=100 was determined through preliminary experiments, and the specific iteration process is as follows:

Calculate the negative gradient (residual) of the mth round.

For the squared loss function $L(y_i, F(x_i)) = \frac{1}{2}(y_i - F(x_i))^2$, its negative gradient with respect to $F(x_i)$ is the residual (the gradient direction is the direction in which the loss increases, and the negative gradient direction is the direction in which the loss decreases), the formula is:

$$r_{im} = -\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)}\bigg|_{F(x)=F_{m-1}(x)} = y_i - F_{m-1}(x_i) \tag{26}$$

Among them: m is the current iteration round ($1 \le m \le M$); $F_{m-1}(x_i)$ is the prediction score of the i-th sample by the m-1th round model; r_{im} is the residual of the i-th sample in the mth round (i.e. "actual score - previous round predicted score"). For example, if the m-1 round model predicts a sample score of 1.50m, but the actual score is 1.65m, then the m-th round residual of the sample is $r_{im} = 1.65 - 1.50 = 0.15$m.

Build the $\boldsymbol{m}$ regression tree $\boldsymbol{h_m(x)}$**.**

Using the residual r_{im} as the "new target variable", fit a regression tree with the training set X, and divide the feature space into J non overlapping leaf node regions $R_{1m}, R_{2m}, \ldots, R_{Jm}$. The splitting rule of the regression tree is based on minimizing the mean square error (MSE). For each possible splitting point t of each feature, calculate the MSE of the left and right subtrees after splitting:

$$\mathrm{MSE}(t) = \frac{1}{n_L}\sum_{x_i \in L_t} (r_{im} - \overline{r_{Lt}})^2 + \frac{1}{n_R}\sum_{x_i \in R_t} (r_{im} - \overline{r_{Rt}})^2 \tag{27}$$

Among them, L_t and R_t are the left and right subtree sample sets divided by the splitting point t, respectively; n_L and n_R are the sample sizes of the left and right subtrees, respectively; $\overline{r_{Lt}}$ and $\overline{r_{Rt}}$ are the mean residuals of the left and right subtrees, respectively. Select the splitting point with the minimum MSE for node splitting until the tree depth reaches the preset value or the number of leaf node samples is less than the threshold.

Calculate the optimal output value of the leaf node of the $\boldsymbol{m}$ tree.

For each leaf node region R_{jm}, calculate the mean of the residuals in that region as the optimal output value (the optimal solution under squared loss) for that node, using the formula:

$$\gamma_{jm} = \arg\min_{\gamma} \sum_{x_i \in R_{jm}} L(y_i, F_{m-1}(x_i) + \gamma) = \frac{1}{n_{jm}}\sum_{x_i \in R_{jm}} r_{im} \tag{28}$$

Among them, n_{jm} is the number of samples at the jth leaf node of the mth tree, and γ_{jm} is the output value of that node. For example, if a leaf node on the m-th tree contains 3 samples with residuals of 0.12, 0.15, and 0.09, then the optimal output value of that node is $\gamma_{jm} = \frac{0.12+0.15+0.09}{3} = 0.12$.

Learning rate decay (to prevent overfitting)

To avoid excessive impact of a single tree on the final prediction, a learning rate $\eta(\eta = 0.1)$ is introduced to attenuate the output of the mth tree. The updated model is:

$$F_m(x) = F_{m-1}(x) + \eta \cdot h_m(x) \tag{29}$$

Among them, $h_m(x) = \sum_{j=1}^{J} \gamma_{jm} \cdot I(x \in R_{jm})$ is the indicator function, where I=1 when $x \in R_{jm}$, otherwise I=0). For example, if the m-1th round model predicts a value of 1.50m, the output of the mth tree for this sample is 0.12, and the learning $\eta = 0.1$, then the m-th round prediction value $F_m(x) = 1.50 + 0.1 \times 0.12 = 1.512$.

Step 3: Generate the final prediction model:

After completing M rounds of iterations, all weak learners (regression trees) are weighted and summed to obtain the final prediction model of GBRT:

$$F(x) = F_0(x) + \eta \cdot \sum_{m=1}^{M} h_m(x) \tag{30}$$

For the new sample, input the standardized feature X_{test}, and the above model can output the predicted value $\hat{y}_{\text{test}} = F(X_{\text{test}})$ of its standing long jump performance.

4. Simulation and Analysis

4.1. Prediction and judgment of takeoff and landing

In order to verify the performance of the model, we selected two athletes who were performing standing long jump for tracking and shooting. Based on the 32 key nodes of their bodies shown in Figure 1, we extracted the coordinate information of each frame's 32 key nodes. Table 1 shows the partial landmark information of athlete 1; Table 2 shows partial coordinate information of Athlete 2, where columns (0_X, 0_Y) represent the coordinate information of Key Node 0, columns (1_X, 1_Y) represent the coordinate information of Key Node 1, and so on.

Table.1. Key Node Coordinate Information of Athlete 1

frame number	0_X	0_Y	1_X	1_Y	2_X	2_Y
0	308.81	198.51	309.58	190.21	311.42	190.31
1	306.87	198.18	308.58	190.31	310.68	190.64
2	305.95	197.57	308.1	190.31	310.29	190.69
3	305.49	197.43	307.85	190.46	310.1	191.01

Table.2. Key Node Coordinate Information of Athlete 2

frame number	0_X	0_Y	1_X	1_Y	2_X	2_Y
0	316.23	320.86	314.18	314.09	314.25	314.1
1	315.49	323.31	314.05	316.68	314.25	316.75
2	315.28	324.84	313.93	318.01	314.24	317.95
3	315.46	325.88	314.08	319.04	314.53	318.86

Applying the algorithm presented in this article to analyze two athletes, the ankle height variation curves and their derivatives and acceleration curves are obtained as shown in Figures 2 and 3. The two images clearly display the recognition results of takeoff and landing moments.

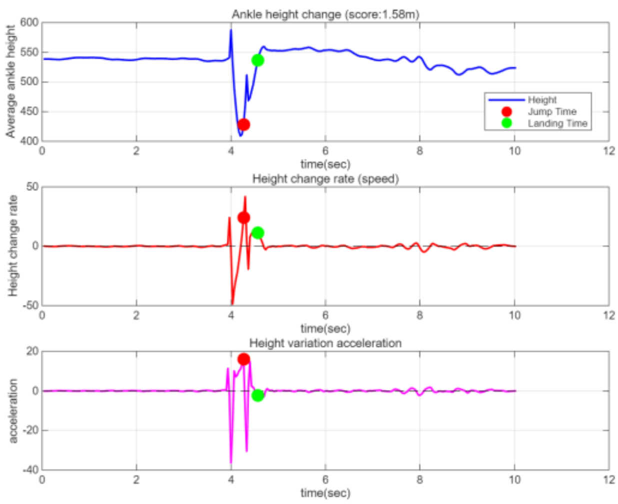

Figure 2. Athlete 1 ankle height variation curve

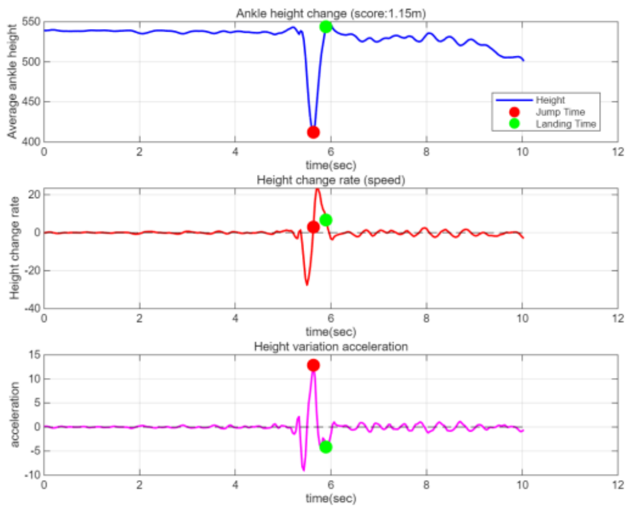

Figure 3. Athlete 2 ankle height variation curve

It can be clearly observed from Figures 2 and 3 that the takeoff moment (marked in red) corresponds to the peak point of the acceleration curve, while the landing moment (marked in green) corresponds to the moment when the height returns to the reference level. The specific values are shown in Table 3:

Table.3. Comparison of Key Motion Time Parameters for Standing Long Jump

analytical metrics	Jump time (frames/second)	Landing time (frames/second)	Stagnation time (frames/second)
Athlete 1	127/4.23	136/4.53	9/0.30
Athlete 2	169/5.63	177/5.90	8/0.27

The results showed that the hovering time of athlete 1 (0.30 seconds) was 0.03 seconds longer than that of athlete 2 (0.27 seconds). According to the law of projectile motion, when the takeoff speed is similar, a longer hovering time means a longer jumping distance. Therefore, the actual long jump performance of athlete 1 (1.58m) is better than that of athlete 2 (1.15m).

4.2. Analysis of the Motion Process during the Stagnation Stage

Quantify the motion process of Athlete 1 and Athlete 2 during the hovering phase (from takeoff to landing), calculate five key indicators based on the following algorithm, and conduct comparative analysis.

Kinematic index algorithm:

Body center of gravity height: using the double hip point weighted average method:

$$h_{\mathrm{com}}(n) = \frac{y_{23}(n)+y_{24}(n)}{2} \tag{31}$$

Body forward tilt angle: based on the angle between the shoulder hip line vector and the vertical direction:

$$\theta_{\mathrm{body}}(n) = \arccos\left(\frac{\vec{V}_{\mathrm{body}}\cdot\vec{V}_{\mathrm{vertical}}}{\|\vec{V}_{\mathrm{body}}\|\cdot\|\vec{V}_{\mathrm{vertical}}\|}\right) \times \frac{180}{\pi} \tag{32}$$

Knee joint angle: calculated using the three-point method (taking the right knee as an example):

$$\theta_{\mathrm{knee}}(n) = \arccos\left(\frac{(P_{23}-P_{25})\cdot(P_{27}-P_{25})}{\|P_{23}-P_{25}\|\cdot\|P_{27}-P_{25}\|}\right) \times \frac{180}{\pi} \tag{33}$$

Limb coordination: Calculate the angle between feature vectors:

$$\theta_{\mathrm{coord}}(n) = \arccos\left(\frac{\vec{V}_1\cdot\vec{V}_2}{\|\vec{V}_1\|\cdot\|\vec{V}_2\|}\right) \times \frac{180}{\pi} \tag{34}$$

Hand movement amplitude: Euclidean distance calculation:

$$d_{\text{hand}}(n) = \sqrt{(x_{22}(n) - x_{20}(n))^2 + (y_{22}(n) - y_{20}(n))^2} \tag{35}$$

Analysis of the hovering process of Athlete 1: According to the above algorithm, the range of changes in body center of gravity is 54.39 pixels, body angle is 50.7 °~74.2 °, knee angle is 94.4 °~158.7 °, limb coordination index is 22.8 °~82.3 °, and hand movement amplitude is 1.6~9.2 pixels. The results indicate that Athlete 1 has good vertical takeoff ability and aerial posture control ability.

Analysis of the hovering process of Athlete 2: The same algorithm calculation shows that the amplitude of the body's center of gravity change is 49.12 pixels, the range of body angle change is 45.2 °~59.4 °, the range of knee angle change is 73.9 °~160.9 °, the range of limb coordination index change is 27.5 °~105.8 °, and the amplitude of hand movements is 1.2~7.9 pixels. The data shows that its technical stability and coordination need to be improved.

Comparative analysis: As shown in the table 4, through algorithmic quantification analysis, Athlete 1 outperforms Athlete 2 in all indicators, especially in terms of the amplitude of changes in body center of gravity, the range of changes in body angle, and limb coordination.

Table.4. Comparison of kinematic indicators during the hovering phase

Kinematic indicators	Athlete 1	Athlete 2	variance analysis
Change amplitude of body center of gravity (pixels)	54.39	49.12	Athlete 1 has a 10.7% increase
Range of body angle variation(°)	50.7~74.2	45.2~59.4	Athlete 1 has a wider range
Range of knee angle variation(°)	94.4~158.7	73.9~160.9	Athlete 1 changes more smoothly
Lower limb coordination angle(°)	22.8~82.3	27.5~105.8	Athlete 2 has poorer coordination
Arm swing angle(°)	18.6~101.7	4.1~117.9	Athlete 2 has a larger swing
Hand movement amplitude (pixels)	1.6~9.2	1.2~7.9	Athlete 1 moves more fully

Conclusion: Through algorithmic quantification analysis, Athlete 1 exhibits better technical characteristics during the hovering phase, including stronger vertical takeoff ability, better aerial posture adjustment ability, more comprehensive lower limb techniques, and better limb coordination. These algorithmic quantified features together explain the reasons for its better performance (1.58mvs1.15m).

5. Conclusions

This article is based on AI human posture recognition technology, focusing on the analysis of standing long jump movements, factors affecting performance, and performance prediction, forming a complete technical closed loop. By using sliding window smoothing and dynamic ground baseline models, the landing time of athletes can be accurately determined, and quantitative analysis can reveal their hovering performance, joint extension, and landing buffering. Factor analysis and correlation testing indicate that physical indicators such as basal metabolism and weight are significantly positively correlated with academic performance, while arm swing amplitude is negatively correlated. The performance prediction model based on gradient boosting regression explained 88.1% of the performance variation, with an accuracy of 95%. It predicted the athlete's performance to be 1.231 meters, verifying the accuracy of the model.

Acknowledgments

This article has received support from the Education Science 2024 Planning Project of Shenzhen (yb24014) and the Key Research Area Support Research Project (Science and Technology) of Shenzhen University of Information Technology (SZIIT2024KJ044).

References

[1] Gu Xuan, Wang Chuanwei, Xu Hua, etc Student Intelligent Body Measurement System Based on Computer Vision [J]. Information Technology and Informatization, 2023, 10: 186-193.

[2] Qi H X, Chen X C, Yu Z G, etc. Vertical Jump of a Humanoid Robot with CoP-Guided Angular Momentum Control and Impact Absorption[J]. IEEE Transactions on Robotics, 2023, 39(4): 3154 - 3166.

[3] T Yokozawa, D Kumagawa, H Arakawa, Y Katsumata, R Akagi. A Biomechanical Analysis of the Relationship between the Joint Powers during the Standing Long Jump and Maximum Isokinetic Strength[J]. International Journal of Sport and Health Science, 2019, 17: 13-24.

[4] Xu J, Li C, Wang Y. An Improved Otsu Segmentation Algorithm Based on Least Squares Fitting for Medical Images [J]. Journal of Digital Imaging, 2020, 33 (4): 892-905.

[5] Chen X, Yang Y, Zhang J. Predicting Elite Sprinters' 100m Performance Using Gradient Boosting Regression and Biomechanical Features [J]. Journal of Sports Sciences, 2022, 40 (18): 2011-2020.

[6] Kim J H, Lee D H, Park S Y. Identifying Key Physiological and Biomechanical Factors Influencing Jump Performance in Collegiate Athletes Using Exploratory Factor Analysis [J]. Journal of Sports Science & Medicine, 2020, 19 (3): 452-460.

[7] Cao Z, HIDALGO G, Simon T, et al. Realtime multi-person 2D pose estimation using part affinity fields[J]. IEEE transactions on pattern analysis & machine intelligence, 2021,43(1):172-186.

[8] Hu Jianlang, Luo Yarong, Guo Chi A visual servo robot for motion posture evaluation [J]. Surveying Bulletin, 2020521 (8): 36-41.

[9] Wang Yuan Human motion pose estimation and recognition based on deep neural networks [D]. University of Electronic Science and Technology of China, 2020.

[10] Yang Jun, Zhang Sujun, Zhang Chuanghao, Huang Jingjing Comparative Study on Human Action Recognition Based on OpenPose [J]. Sensors and Microsystems, 2021, 40 (1): 11-14.