Hindawi
Scientific Programming
Volume 2022, Article ID 6447471, 13 pages
https://doi.org/10.1155/2022/6447471

Research Article

Research on Methods of Physical Aided Education Based on Deep Learning

Wei Su[1] and Jian Feng[2]

[1]*BasicEducationDepartment, ShangHaiCommunicationsPolytechnic, ShangHai200431, China*
[2]*Department of Physical Education of Tongji University, ShangHai 200092, China*

Correspondence should be addressed to Jian Feng; 91786@tongji.edu.cn

Received 15 December 2021; Revised 11 February 2022; Accepted 25 March 2022; Published 9 May 2022

Academic Editor: Hangjun Che

In order to better meet the training needs of sports and improve the standardization of sports training, an openpose-based sports posture estimation method and assisted training system are proposed, combining the basic structure and principle of openpose network. Firstly, the human posture estimation algorithm is constructed by combining with the openpose network; secondly, the overall framework, specific operation process, image acquisition, posture estimation, and other modules of the sports assistance system are designed in detail; finally, the openpose posture estimation method constructed above is validated. The results show that the value of the loss function obtained by the algorithm gradually stabilizes after 250 iterations. By using the COCO dataset as the training base and comparing it with the standard posture, it is found that the algorithm can correctly identify different badminton action postures, and the recognition rate can reach up to 94%. This shows that the algorithm is feasible and can be used for posture estimation and training of badminton sports movements.

1. Related Work

With the development of people's livelihood, ordinary people pay more and more attention to personal health, and the discussion about physical health and sports in the society is becoming more and more heated. However, most people do not master the standard motion posture, so that the best motion effect cannot be obtained, and even suffer unnecessary injuries during exercise. Therefore, it is necessary to process human movement recognition. In the past, people rely on assistant equipment to recognize human posture, so as to judge whether human movement is standard. With the mature of the machine learning algorithm and deep learning algorithm, researchers proposed diversified human movement recognition algorithms, including SVM classifiers, image processing, and deep neural networks. Furthermore, in human motion recognition technology, researchers also pioneered human body posture estimation technology, human motion recognition technology, and so on.

Human posture estimation has always been a popular research topic in academic research. For example, Amir Nadeem created the A-HPE method. There are four benchmark data sets, namely, significant profile detection, entropy Markov model, multidimensional cues from whole body profile, and robust body part model, used to detect human body parts. Its detection accuracy is significantly higher than that of traditional algorithms [1]. In addition, it can provide technical support for human-computer interaction. Poojitha Sing obtained data of various components of each parts of the human body by measuring the point cloud data of human posture in RGB images, which avoids the ambiguity of features and thus shows better detection performance [2]. Xinwei Li estimated the human joint moment by analyzing the dynamic human-computer interaction between the human elbow torque and the exoskeleton output [3]. Wei Quan et al. created an unsupervised learning algorithm based on a forward kinematics model of human skeleton, but the algorithm has not been tested. After the establishment of the human posture estimation algorithm, it is also necessary to introduce the integrated particle swarm optimization (PSO) for optimization. The advantage of the optimized algorithm is that

no pretraining data is required, and the posture estimation of the human body is more concise [4]. After that, this method is tested by a series of experiments. Many scholars have studied human motion recognition. For example, Xiaojun Zhang created a human motion recognition technology based on deep learning. The LSTM algorithm is used to optimize deep learning algorithms, which requires advanced smart wearables devices [5]. Bi Zhuo created a multimodal deep neural network model based on the joint cost function, which used MSR Action3D data sets to identify human motion processes. And, the overall application performance is excellent [6]. Liu Shuqin proposed a human posture estimation method based on discrete point 3D reconstruction algorithm. In this method, the data features are extracted using principal component analysis, and then the estimation of human posture is achieved by means of two-dimensional posture prediction [7]. Jalal Ahmad et al. proposed a 3D Cartesian approach to feature extraction, by which the features are made to contain rich feature information [8]. Licciardo Gian Domenico and others then proposed a posture estimation method of FCN, and the results showed that the method obtained an average accuracy of 96.77% for 17 posture recognition [9]. Combined with the above research, the purpose of this study is to build an auxiliary system that can be used for badminton training and try to realize the estimation of posture through the matching of key bone points of human body, so as to better assist the movement training of badminton lovers. The contribution of this study lies in the extraction of sports posture through in-depth learning, and then through similarity comparison, the standardization standard of sports action is constructed, which provides more accurate information reference for sports training.

2. Estimation of Human Key Bone Points Based on the Openpose Model

2.1. Basic Structure. Openpose model uses the multistage convolutional neural network for training and testing. The first 10 layers of VGGNet-19 are used to initialize the human body image and then fine-tune the initialized human body image. Finally, to input, a set of human body characteristics map F can be achieved. The predicted values for subsequent stages are related to their corresponding image features. Using the three consecutive 3×3 kernels to replace the 7×7 convolution kernel in the earlier output PAFs, which can not only ensure the receiving fields but also greatly reduce the amount of computation, so as to effectively improve the work efficiency of the network model. By referring to the DenseNet method, each output in the three convolution kernels is cascaded together. The network model can synchronously save high-level features and low-level features.

The network structure of the openpose model is shown in Figure 1.

In the first stage, the convolutional neural network generates a set of partial affinity field. In the subsequent stages, the prediction results of the previous stage are cascaded with the original graph feature F. So, more accurate prediction can be made [10]. $\varnothing^t$ represents the convolutional neural network at stage $t \le T_p$, and T_p represents the total PAF prediction stage.

After T_p iterations, take the latest PAF prediction stage as the first stage and repeat this process to predict the confidence map [11]. Here, ρ^t represents the convolutional neural network at stage T_p and $T_p + T_c$, and T_C represents the total prediction stage of the confidence map.

2.2. Estimation of Human Key Bone Points Based on Optimization Model Structure. The openpose model is created to recognize and estimate multihuman postures. Its innovation is reflected in three aspects: firstly, the human body vector inclined field PAFs is established to estimate the confidence map of human limb features, which is among the constrained bone points in the human pose model. The constraint relationship is strengthened by combining the human key points hot spot map. And, the classification of multihuman posture key points is simplified. Secondly, six stage layers are created. The next stage layer will re-estimate the human key point hot spot map and the confidence map of human limb features, which output from the previous stage layer. And, the estimation accuracy can be further improved. Thirdly, during the training process, the loss function of each stage is monitored to ensure that the overall loss is minimized. According to the test results, the openpose model has the advantages of high estimation accuracy, but also has the disadvantages of long estimation time.

For badminton, athletes' postures change quickly. The action frequency is higher than that of human body under normal circumstances. In order to track the change process of motion posture in real time, it is necessary to ensure the efficient operation of the estimation module and evaluation module in the human body posture evaluation system. Consider that the openpose model evaluates the head region according to the five bone points in the head, which have little impact on the body posture of badminton players. And, the estimated time is indirectly extended [12]. Therefore, taking the modified human posture model as reference, this paper created a new deep neural network model to estimate the two-dimensional coordinates of human skeleton points of badminton players in a single frame image. Its architecture is shown in Figure 2 [13].

Firstly, the VGG neural network is introduced into the improved human posture evaluation model, and then the evaluation of human posture is realized through two-stage processing of openpose. The basis of the VGG neural network in Figure 2 is the CNN. Convolutional neural network is used to extract image features. CNN network includes convolutional layer, pooling layer, full connected layer, and output layer.

The convolution operation is

$$x_j^l = f\left(\sum_{i \in M_j} x_j^{l-1} * k_{ij}^l + b_j^l\right), \tag{1}$$

where k represents the convolution kernel; l represents the number of layers; M_j represents the j th feature graph; and b represents the bias term.

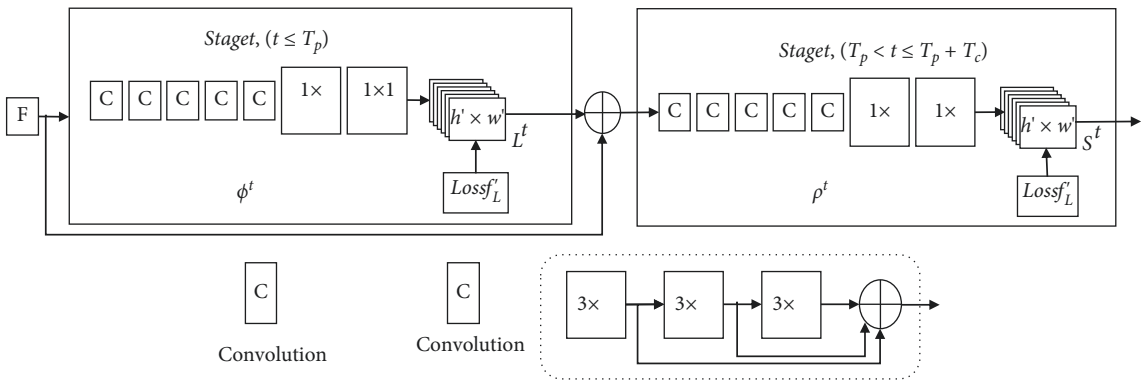

Figure 1: The network structure of the openpose model.

The calculation of the pooling layer is

$$x_j^l = f\left(\beta_j^l \text{down}\left(x_j^{l-1}\right) + b_j^l\right), \tag{2}$$

where down represents the lower sampling function and β and b represent the feature graph corresponding to each output, respectively.

The training process of CNN includes forward propagation and back propagation.

Among them, forward propagation can calculate corresponding actual output results after layer by layer transformation by inputting data (X_p, Y_p) to CNN, and the calculation formula of this process is as follows:

$$O_p = F_n\left(\cdots\left(F_2\left(F_1\left(X_p W_1\right)W_2\right)\right)\cdots W_n\right). \tag{3}$$

Back propagation is the calculation of the error between the actual output O_p and the target output Y_p, and then the error is back-propagated according to the principle of error minimization, and its weight is constantly adjusted.

The training process of CNN is as follows:

2.2.1. Back-Propagation Algorithm. In the process of forward propagation, the squared error cost function is used to measure the error. If the category is class c and the number of training samples is N, then E^N can be expressed as

$$E^N = \frac{1}{2}\sum_{n=1}^{N}\sum_{k=1}^{c}\left(t_k^n - y_k^n\right)^2. \tag{4}$$

In formula (4), t_k^n and y_k^n represent the target output of n th sample and the k-dimension of actual output, respectively.

In the process of back propagation, the sensitivity of base is used to represent the error of back propagation, which represents the change rate of error to the base b, and the expression is as follows:

$$\frac{\partial E}{\partial b} = \frac{\partial E}{\partial u}\frac{\partial u}{\partial b} = \delta. \tag{5}$$

In formula (5), since $\partial u/\partial b = 1$, $\partial E/\partial b = \partial E/\partial u = \delta$, which means that the sensitivity of a neuron's base b is equal to the derivative $\partial E/\partial u$ of error E with its all input u.

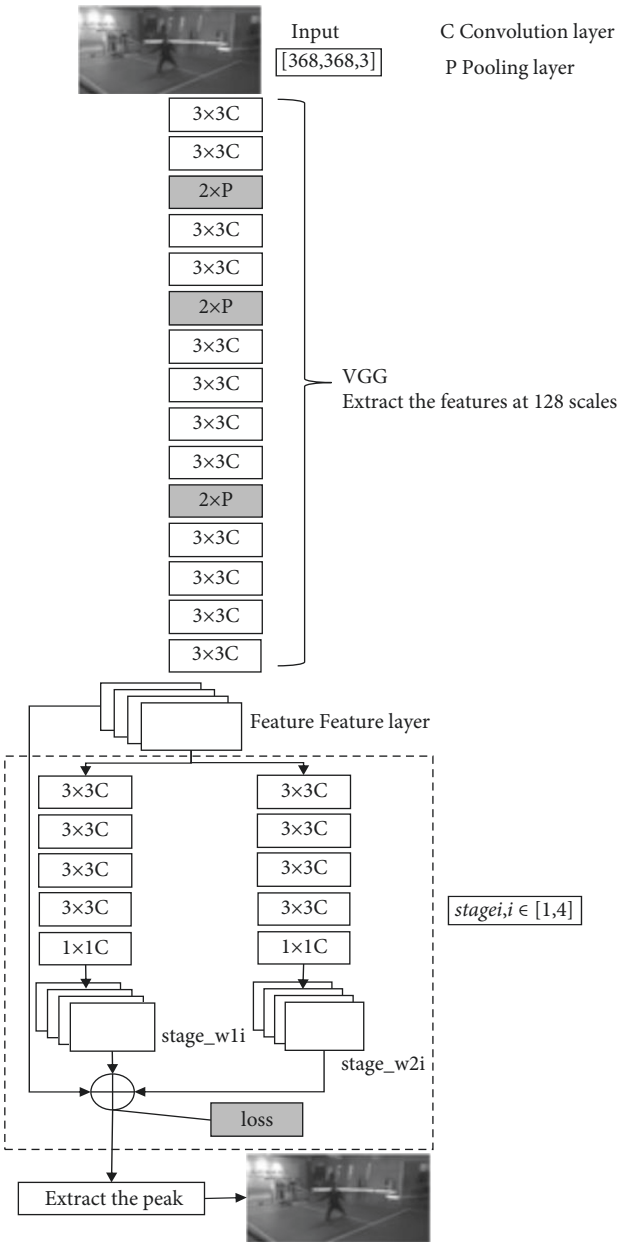

Figure 2: Estimation model structure.

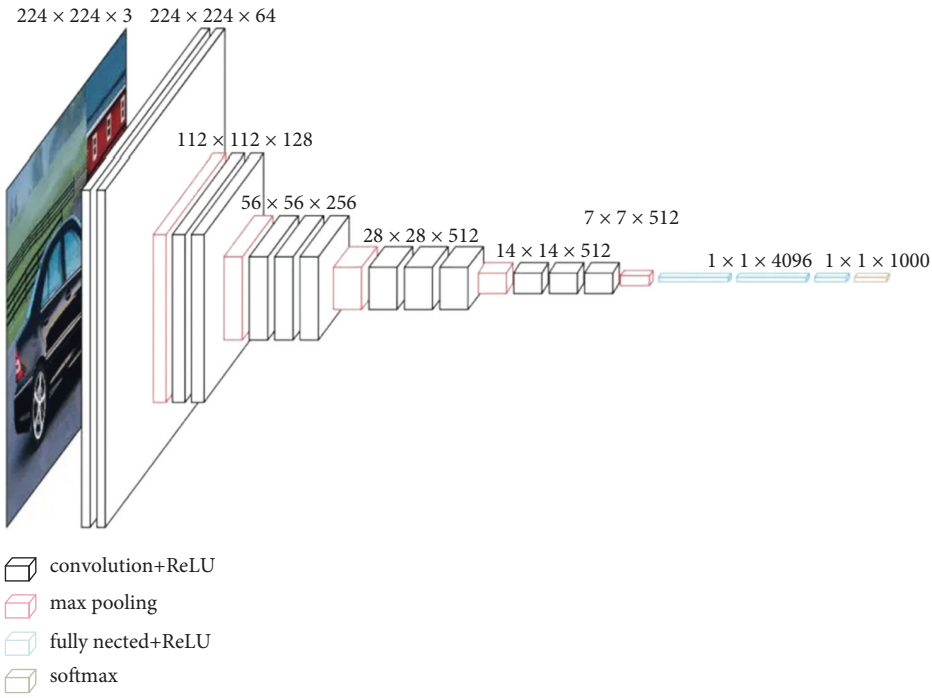

Figure 3: VGG model.

2.2.2. Weight Update. Weight update process of the convolutional layer: the calculation formula of weight update of this layer is the same as the calculation formula (1) of the convolution layer. The feature graph is input into a trainable convolution kernel for convolution operation, and a bias term is added. Finally, the output feature graph can be obtained through an activation function.

M_j represents the combination of input feature graphs. The corresponding convolution kernels of each output feature graph are different. Even though both output feature graph map_j and output feature graph map_k are obtained by convolution from input feature graph map_i, their corresponding convolution kernels are still different.

If there is a downsampling layer $l+1$ under each convolutional layer l, a pixel of the output feature map of the convolutional layer corresponds to the sensitivity D corresponding to one pixel in the downsampling layer. In order to effectively calculate the sensitivity of the convolutional layer l, the sensitivity map corresponding to the upsampling in downsampling will be used to upsample, so that the size of map is the same as the feature map size of the convolutional layer l. In addition, the sensitivity δ^l of the convolutional layer l can be obtained by multiplying the sensitivity by the parameter β. The calculation formula is

$$\delta_j^l = \beta_j^{l+1}\left(f'\left(u_j^l\right) \circ up\left(\partial_j^{l+1}\right)\right). \tag{6}$$

Here, up represents the upsampling operation, and $\circ$ represents the multiplication of each element. If the sampling factor during downsampling is n, upsampling is to replicate each pixel in horizontal and vertical directions, respectively, so as to achieve the upsampling size recovery goal. Upsampling can be realized by Kronecker product:

Figure 4: Input coordinate system distribution.

$$up(x) = x \otimes 1_{n\times n}. \tag{7}$$

On this basis, its sensitivity map can be obtained according to a given feature graph on the convolution layer. Firstly, the gradient of base b is calculated, that is, the sensitivity of all elements in the sensitivity map is summed, and the formula is

$$\frac{\partial E}{\partial b_j} = \sum_{u,v}\left(\delta_j^l\right)_{uv}. \tag{8}$$

According to the weight sharing feature, the gradient solution is carried out for the point through all the connections associated with the weight, and then the gradient is obtained and summed. The expression is

$$\frac{\partial E}{\partial k_{ij}^l} = \sum_{u,v}\left(\delta_j^l\right)_{uv}\left(p_i^{l-1}\right)_{uv}. \tag{9}$$

Here, $\left(p_i^{l-1}\right)_{uv}$ represents the block in x_i^{l-1} convolved with k_{ij}^l, namely, a unit input of convolution layer l.

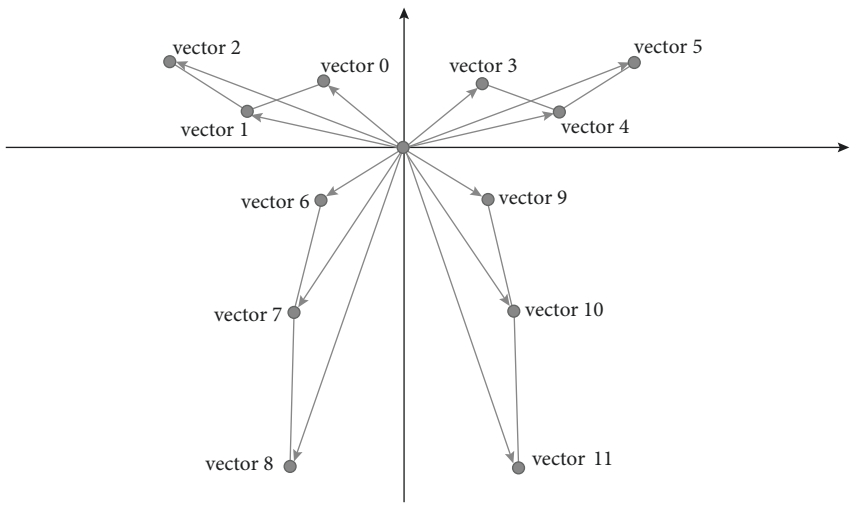

Figure 5: Converting the coordinates of the estimated bone points into vectors.

Formula (9) can be calculated by using the convolution function in MATLAB, and the following formula can be obtained:

$$\frac{\partial E}{\partial k_{ij}^{l}} = \text{rot180}\left(\text{conv2}\left(x_i^{l-1}, \text{rot180}\left(\delta_j^{l}\right), '\text{valid}'\right)\right). \tag{10}$$

Here, rot180 (δ_j^l) means to rotate it. After rotation, cross-correlation calculation can be carried out, and then the input is reversed.

Weight updating process of the downsampling layer: the weight updating process of downsampling layer is the same as the calculation formula (2) of the pooling layer. If the sensitivity map of the down sampling layer needs to be calculated, the updated values of parameters β and b can be calculated by using formula (8).

If the current downsampling layer is fully connected with the subsequent convolutional layer, the sensitivity map of te downsampling layer can be calculated by the BP algorithm, and the sensitivity can be calculated by back-propagation:

$$\delta_j^{l} = f'\left(u_j^{l}\right) \circ \text{conv2}\left(\delta_j^{l+1}, \text{rot180}\left(k_j^{l+1}\right), 'full'\right). \tag{11}$$

Here, δ_j^{l+1} represents the sensitivity reversely propagated to it by the next convolution layer of current downsampling layer, and rot180 (k_j^{l+1}) represents the rotated convolution kernel.

Then, the gradient of bias β and b is computed. The gradient calculation method of bias b is to add all elements in sensitivity map, and the calculation formula is the same as calculation formula (8) of the convolutional layer.

For the gradient calculation of bias β, the downsampled map in the forward propagation process should be obtained, and the expression of downsampled map is

$$d_j^{l} = \text{down}\left(x_j^{l-1}\right). \tag{12}$$

Thus, the gradient of β can be calculated as

$$\frac{\partial E}{\partial \beta_j} = \sum_{u,v} \left(\delta_j^{l} \circ d_j^{l}\right)_{uv}. \tag{13}$$

Through the above construction, the VGg model network structure of this study is obtained in Figure 3.

Through the above image processing and then combined with the two-stage pose estimation in Figure 2, the pose of human motion is obtained.

3. Human Posture Evaluation

3.1. Similarity Calculation. The above improvement is about how to estimate human posture and ensure the real-time and accuracy of estimation. After estimating a group of reliable skeletal points which can be referenced to the modified human pose model, how to identify human body posture according to human posture skeletal points has become the key of the human posture evaluation system. Considering that badminton belongs to the upper limb movement, the standard posture in various badminton sports is concentrated in the upper limb area. Therefore, on the basis of human skeletal point estimation, similarity is used to evaluate the similarity between the posture of badminton lovers and the standard badminton action library, so as to realize the objective evaluation of badminton action.

The human posture evaluation process consists of three steps: (1) convert the coordinates of the input bone points; (2) match with the standard posture library; (3) process and output the matching results.

A set of human bone point coordinates of a single frame image in the camera coordinate system is input into the human posture evaluation system, and the modified human posture model is used as the reference. In the human posture coordinate set index_T, each coordinate group has 13 points, as shown in Figure 4.

In the evaluation stage, the input coordinate system (image pixel coordinate system) should be transformed first, so as to prepare for the subsequent posture evaluation. In the coordinate conversion process, the camera's internal parameter matrix and external parameter matrix will be involved. Here, the former is fixed, and the latter depends on the location and angle of the camera lens. Therefore, in this regard, the camera's external matrix needs to be precalibrated

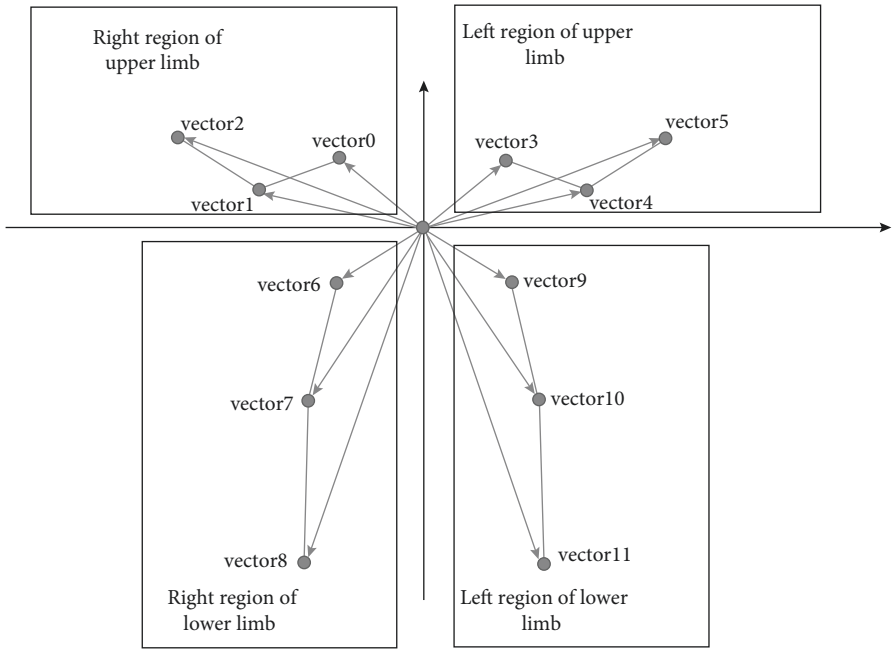

Figure 6: Human limbs partition diagram.

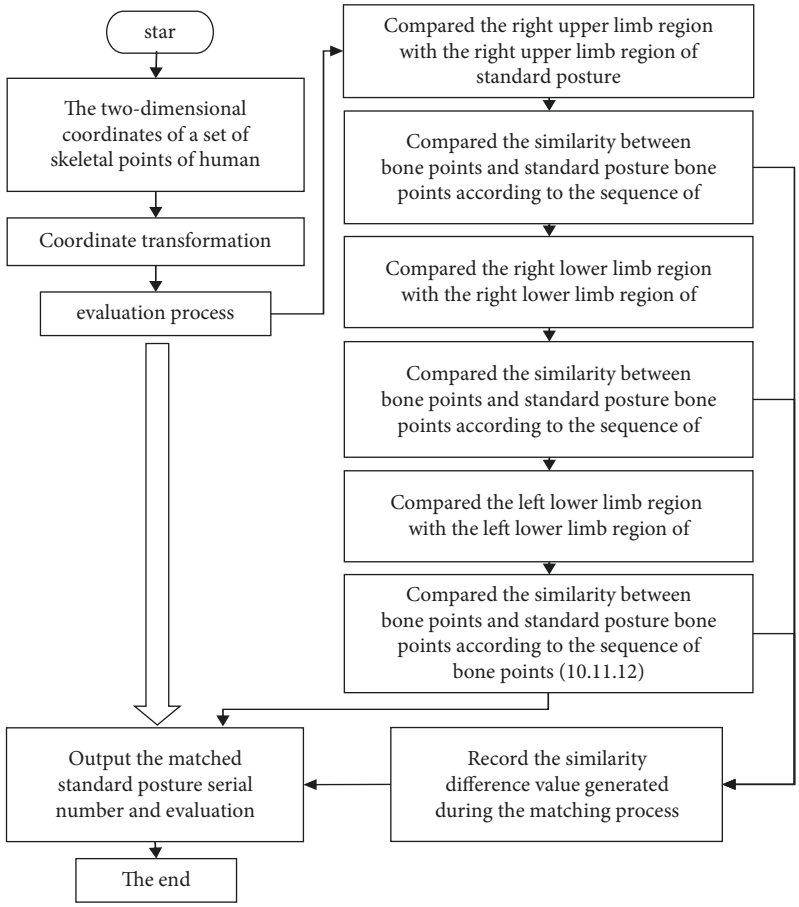

Figure 7: Human posture evaluation algorithm process.

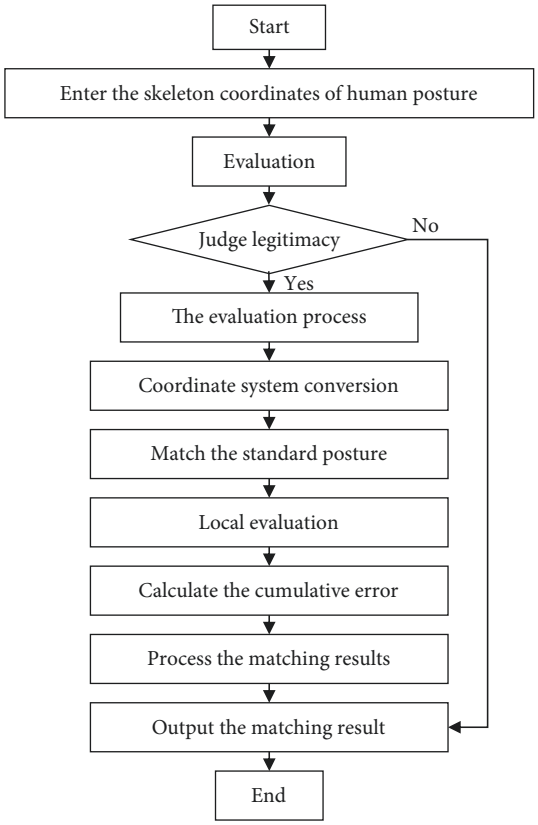

Figure 8: Human posture evaluation algorithm process.

to ensure the validity of the external matrix. Although the above process is feasible, it is difficult to operate in practice. So, a new coordinate transformation method is proposed in this paper, that is, (a) convert from an image pixel coordinate system to a rectangular coordinate system with the neck point as the origin in the human bone point; (b) transform from the rectangular coordinate system with the neck point as the origin to the polar coordinate system with the neck point as the origin and determine the angle between the other 12 points in the polar coordinate system and the positive x axis.

The coordinate transformation step (a) solves the matching problem of human posture and standard posture caused by different positions, and step (b) solves the uncertainty of human posture evaluation caused by individual body size difference.

After the coordinate transformation is completed, 12 included angle values of the positive X-axis and the vector [0, 11] are obtained, respectively, as shown in Figure 5 [14].

The calculation process of coordinate transformation is as follows:

(i) Input the human bone point coordinate $\{a_0, a_1, \ldots, a_{12}\}$ in the image pixel coordinate system and establish the rectangular coordinate system with a_0 as the origin

(ii) The vector set $\{t_0, t_1, \ldots, t_{11}\}$ is established after the coordinates $\{a_1, \ldots, a_{12}\}$ of the remaining 12 human bone points subtracts a_0

Table 1: Structural parameters of the optimization model.

Training data set	Human images in the COCO data set and acquiring the badminton player images
Input number of images	10
Maximum iteration	200000
Basic learning rate	5×10^{-5}
Initial value of weight value	5×10^{-4}
Initial value of weight correction	0.01
The optimizer	Stochastic gradient descent
Loss function	$\text{loss} = \sum_{i=1}^{4} k_i \cdot (\text{stage\_}w1_i - \text{pafmap})^2 + (\text{stage\_}w2_i - \text{heatmap})^2/\text{batch\_size} \times 2$ Pafmap human bone point grayscale of the preset image Heatmap human bone point grayscale of the preset image

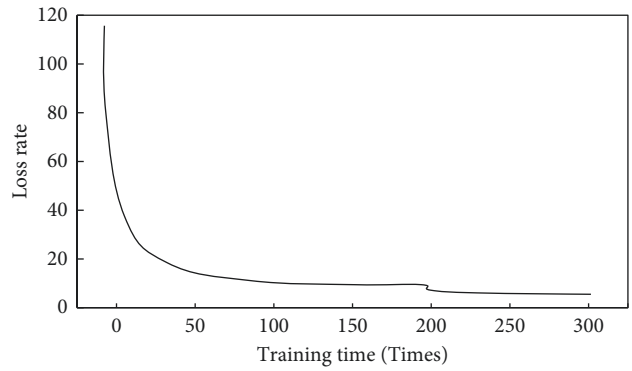

Figure 9: Loss curve of retraining optimization model.

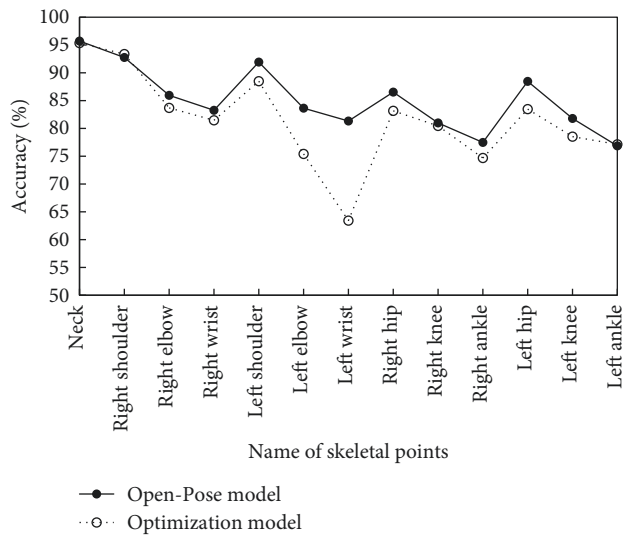

Figure 10: Model estimation accuracy of each bone point in the modified human posture model.

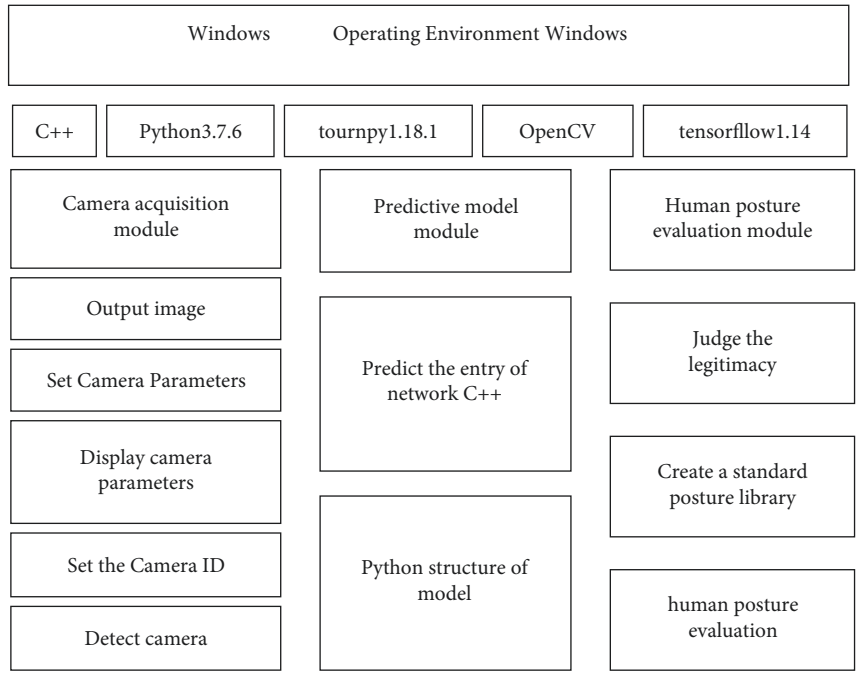

Figure 11: Software architecture of the human posture evaluation system.

(iii) Apply formula (14) to solve the included angle value between each vector in the vector set and the positive X-axis and establish the included angle set $\{\theta_0, \theta_1, ..., \theta_{11}\}$ [15]:

$$\theta_i \begin{cases} \cos^{-1}\left(\dfrac{\vec{t}_i . \vec{y}}{\|\vec{t}_i\| . \|\vec{y}\|}\right) t_{iy} \geq 0 \\ \\ \cos^{-1}\left(\dfrac{\vec{t}_i . \vec{y}}{\|\vec{t}_i\| . \|\vec{y}\|}\right) + 180 t_{iy} < 0 \end{cases}, \quad (14)$$

where θ represents the cosine angle.

3.1.1. Matching with the Standard Posture. The human posture evaluation algorithm divides the human body into four regions, as shown in Figure 6 [16].

Coordinate transformation is performed for the other points in the human posture model relative to the neck points to adjust the corresponding serial number. The right region of the upper limb is composed of the right elbow, right shoulder, and right wrist. The coordinate serial number after adjustment is 0, 1, and 2. The left region of the upper limb is composed of left elbow, left shoulder, and left wrist. The coordinate serial number after adjustment is 3, 4, and 5. The right region of lower limbs is composed of right knee, right hip, and right ankle, and the coordinate serial number after adjustment is 6, 7, and 8. The left region of the lower limbs is composed of the left knee, left hip, and left ankle. The coordinates after adjustment are 9, 10, and 11.

The posture evaluation of each small area in all the regions is to compare the posture to be evaluated with the candidate standard posture of the previous stage. And, the accumulative error is calculated. If the accumulative error does not exceed the allowable error of the stage, the standard posture is included in the candidate standard posture set.

3.2. Human Posture Assessment Process. Combined with the above analysis, the evaluation of human posture is mainly divided into the following steps:

For the right upper limb area (including the right shoulder, the right elbow, and the right wrist), the three vectors between bones and neck, as well as the angle of the positive x axis can be solved, respectively. So, the right upper limb regional similarity sets can be established. Then, the absolute values of similarity degree with corresponding standard attitude are solved, respectively. Finally, the similar standard postures are filtered with the predetermined error values.

For the right lower limb region (including right hip, right knee, and right ankle), the three vectors between the skeleton and the neck, as well as the angle of the positive X axis, are solved, respectively, to establish the regional similarity set of the right lower limb. Then, the absolute values of similarity degree with corresponding standard attitude are solved, respectively. Combining with the predetermined error value, the similar standard postures can be filtered.

For the left upper limb area (including the left shoulder, the left elbow, and the right wrist), the three vectors between bones and neck as well as the angle of the positive x axis can be solved, respectively, so as to set up the similarity sets of the left lower limb region. Then, the absolute values of the similarity degree with the corresponding standard attitude

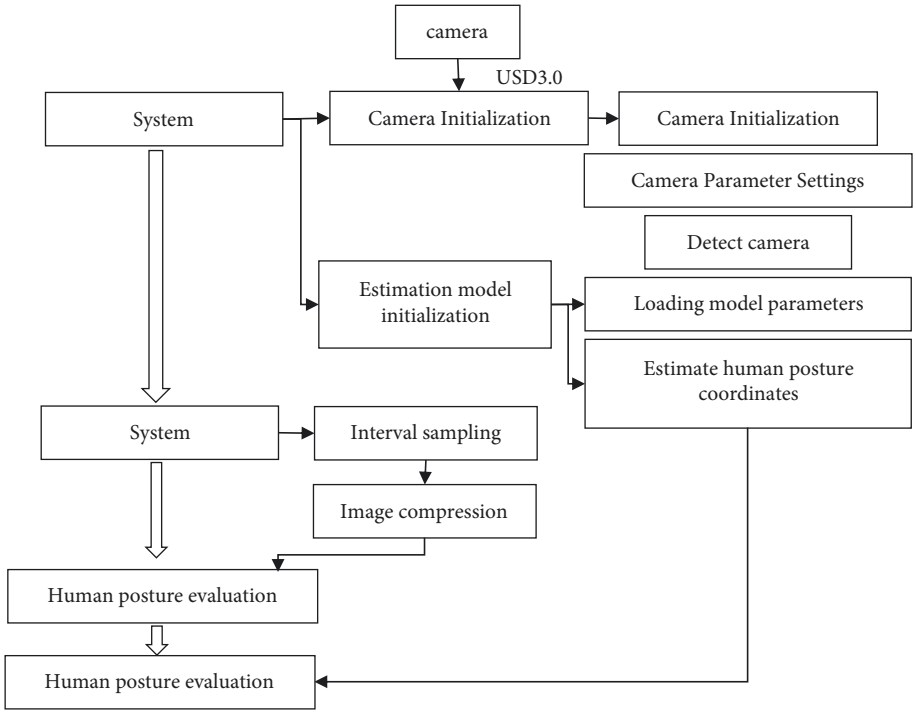

Figure 12: System running process.

Table 2: Camera hardware parameters.

The parameter name	Recommended parameter
Interface	USB 3.0
Pixel	More than 8 million pixels
Frame rate	More than 60 frames
The focal length	Below 3.1 mm

are solved, respectively. Finally, the similar standard postures are filtered with the predetermined error values.

After the above screening process is completed, the human obtained standard posture is the evaluation result. And, in this process, the cumulatively determined similarity is the evaluation value.

The above process is shown in Figure 7 [17, 18].

To determine the similarity difference in the above matching process, it can be weighted according to the influence degree of different regions on human posture. But this may lead to coupling problem, which means that two different representative values of human posture tend to be consistent after the completion of weighting. Therefore, this paper finally decided to directly output the bone point, evaluation value, and matching standard posture serial number at the end of the matching [19].

Considering that when badminton players hold the racquet with their right hand, their left hand is mainly used for coordination to maintain balance. Therefore, the algorithm in this paper cancels the matching of the left upper limb region. Meanwhile, the weight of the other three regions is optimized. Specifically, the weight of the right upper limb region is the largest, the right lower limb and the left lower limb region are second.

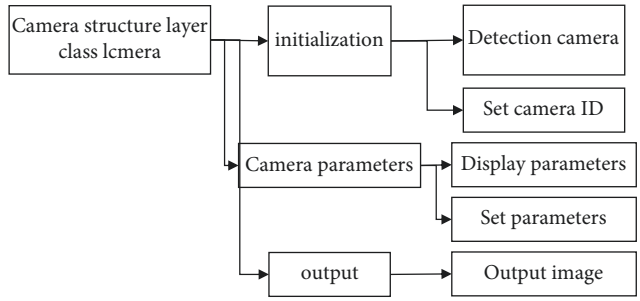

Figure 13: Software interface layer of the camera capture module.

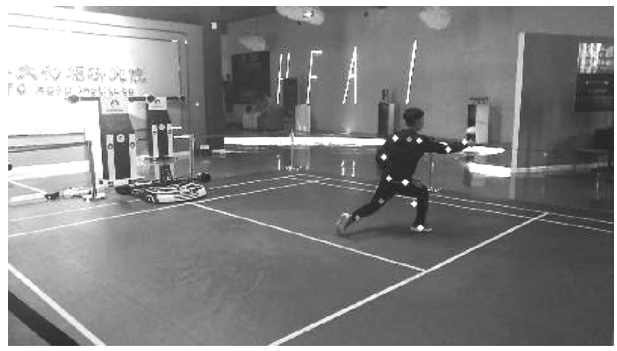

Figure 14: Output of stage 4. (a) Confidence map of human limb features. (b) Bone point heat map.

4. Human Posture Evaluator and Evaluation

Figure 8 shows the design idea of human posture evaluation algorithm [20].

Figure 15: Evaluation effect display.

(1) Convert the human bone point coordinate system into the rectangular coordinate system. The vector of each human bone point coordinates to the origin and the angle between the included vector, and the positive x axis is solved.

(2) Determine the priority of each bone point. According to the size of priority, the margin calculation is made between the selected bone points one by one and the corresponding bone point difference of all posture model in the candidate posture set. Finally, output the cumulative difference value, and after inputting all bone points, step (3) can start.

(3) Solve the accumulative error. The posture model with the minimum accumulated error is found from the candidate posture set. The output results include the posture model serial number and the accumulative error.

4.1. Matching Result Processing. Through the human posture evaluation, the matched standard human posture serial number and the similarity with the matched standard human posture can be obtained.

The processing procedure of matching results is as follows:

If the output is "−1", "0," and "1", it means that the left area of upper limb "fails to match," which is the key area of badminton player's posture matching. Therefore, it can be judged that the athlete's posture in this frame image does not conform to any posture in the standard library, which means that the athlete's posture in this frame image is not standard.

If other information is output, the matching result is obtained, and the higher output standard human posture serial number is, the better the matching result is.

5. Method and System Verification

To verify the above method and system, this study attempts to build a badminton posture evaluation system to verify the above methods.

5.1. Method Verification

5.1.1. Data Sources and Training. To verify the above method and system, part of the video image data is selected as the basic data set for verification. Image data are obtained from badminton video, and images collected by camera and human images in COCO data set are used as training data set. The settings of training parameters are listed in Table 1.

Images in the COCO data set are equipped with human limb grayscale images and human bone point grayscale images. The image collected by the camera can become a suitable training data set only after a series of processing. The process is as follows:

(1) Normalize the collected image to ensure that the pixel value of the image is in the range of [−0.5,0.5]

(2) Mark the pixel value of each human limbs as 0.5 and save it as the human limbs grayscale

(3) Mark the pixel value of each human bone point as 0.5 and save it as the human bone point grayscale

In the first training, the model is trained with COCO data set, which ensured that the optimized model can accurately estimate the general human posture. In the subsequent training process, there is no need to use the initial weight, only need to read the weight parameters of the first training. And, the collected images are adopted to carry out training, so as to further improve the evaluation accuracy.

Only reasonably setting the basic learning rate can effectively prevent the problem of excessive learning rate.

Figure 16: Frames 138 and 139 (from left to right and top to bottom).

Therefore, the basic learning rate set in this paper is equal to 5e-5.

5.1.2. Loss Function Curve. After the first training based on COCO data set is completed, the collected images are used for subsequent training. The loss situation after training is shown in Figure 9.

It can be seen that, in the course of multiple training, the loss keeps decreasing trend as a whole. And, the gradient descent tends to be gentle, which finally approaches the optimal solution.

5.1.3. Accuracy and Timeliness of Skeletal Keypoint Estimation. The estimation accuracy of traditional openpose model and structure-optimized openpose model for each skeletal point is statistically analyzed, and the specific data are shown in Figure 10.

It can be seen from the figure that the estimation accuracy of optimization model is slightly lower than that of the openpose model, and the estimation accuracy of each skeletal point in the left limb is lower than that in the right limb.

5.2. Application Verification. To further verify the feasibility of the above algorithm, an experimental system is set up for verification.

5.2.1. Overall Architecture. The human posture evaluation system consists of camera acquisition module, human posture evaluation module, and prediction model module.

Table 3: Records of the frames to be measured and the measured frames.

Video name	Frames to be measured (frame)	Measured frames (frame)	Detection rate
Video 1mmc1	35	23	0.66
Video 2mmc2	34	32	0.94
Video 3mmc3	192	124	0.65
Video 4mmc4	54	33	0.61
Video 5mmc5	23	17	0.74
Video 6mmc6	49	37	0.76

The output result of the system is the matching result and matching loss of human posture and standard posture library in the current frame image. The matching result refers to the highest standard posture with the human posture matching degree in the current frame image, and the matching loss indicates the similarity between the human posture and the standard posture. The overall framework of the human posture evaluation system is shown in Figure 11 [21].

5.2.2. System Operation Process. The operation mechanism of human posture evaluation system is shown in Figure 12 [22].

5.2.3. Camera Acquisition Module. Camera acquisition module includes two parts, namely, hardware parameter and software interface. Among them, the key of hardware parameters is to correctly set the placement angle of the camera and reasonably determine the camera parameters. Combined with the above analysis results, the camera should be placed on the left side of the badminton net and on the right side of the badminton player. In addition, the best height is 1.2 m. The relevant parameters of the camera are listed in Table 2.

The key of the software interface is to make use of the camera interface layer to make the driver compatible, as shown in Figure 13 [23, 24].

ICmera, the base class of this module, stores one worker function and four detection functions.

Above all, the number and ID of cameras used in the human posture assessment system are determined, and the initial deployment is completed. Then, according to the site environment and the requirements for the evaluation, the camera resolution, frame rate, and other parameters are debugged. Therefore, the module sets up two function interfaces, namely, showParam and setParam. Finally, the function work is used to eliminate invalid information in the image information collected, such as resolution, width, height, and so on. The collected image is converted into a unified cv: Mat format.

The base class ICmera is used for compatibility of driver modules of other cameras, and the subsequent evaluation process adopts the form of ICmera. It can be seen that the human posture evaluation system is not sensitive to the camera model. The camera parameters must meet the setting requirements so that the driver can be set by inheriting the base class. If the evaluation is not effective, the function of ICmera can be called to debug the current camera parameters.

5.2.4. Effect Display. Bone point hot spot map: the bone point hot spot map output is shown in Figure 14.

The evaluation effect achieved by the human posture evaluation system in the test stage is shown in Figure 15.

Effect display: the human posture evaluation algorithm proposed in this paper is used to match successive single frame images. The frames 138 to 139 are successfully matched to the standard posture. The evaluation effect of these 8 frames is shown in Figure 16.

The analysis of Figure 15 shows that first, "Frame i: matching failure," which means that the image in frame I failed to match the standard posture. Second, "Frame i: ending stage A, matching standard posture serial number B, matching loss X," which indicates that the serial number of bone point at the exit of frame *i* matching is A. It successfully matched with standard posture serial number B, and the loss value of the two is *X*.

5.2.5. Detection Rate. The human posture evaluation system is used to evaluate the posture of 6 videos. The number of the frames to be tested and the measured frames in each video are shown in Table 3.

6. Conclusion

To sum up, through the above design, the application of the openpose neural network in the actual sports is realized, so as to provide a new reference method for the accurate training of sports. The innovation of this paper is the accuracy improvement of attitude estimation. At the same time, through the collection of badminton movements, the real-time estimation of badminton posture movements is realized, which provides a reference way for the application of this method.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding this work.

References

[1] A. Nadeem, J. Ahmad, and K. Kim, "Automatic Human Posture Estimation for Sport Activity Recognition with

Robust Body Parts Detection and Entropy Markov model," *Multimedia Tools and Applications*, vol. 80, no. 14, pp. 1–34, 2021.

[2] P. Sing, R. Lakshmi, and R. Raajan, "Point cloud human posture estimation using single RGB image," *Materials Today Proceedings*, vol. 33, pp. 3907–3911, 2020.

[3] X. Li, S. Liu, Y. Chang, S. Li, Y. Fan, and H. Yu, "A human joint torque estimation method for elbow exoskeleton control," *International Journal of Humanoid Robotics*, vol. 17, no. 3, p. 17, 2020.

[4] W. Quan, J. Woo, Y. Toda, and N. Kubota, "Human posture recognition for estimation of human body condition," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 23, no. 3, pp. 519–527, 2019.

[5] X. Zhang, "Application of human motion recognition utilizing deep learning and smart wearable device in sports," *International Journal of System Assurance Engineering and Management*, vol. 12, pp. 1–9, 2021.

[6] Bi. Zhuo and W. Huang, "Human action identification by a quality-guided fusion of multi-model feature," *Future Generation Computer Systems*, vol. 116, pp. 13–21, 2021.

[7] S. Liu, "Discrete pointvv 3D reconstruction algorithm based human pose estimation," *Microprocessors and Microsystems*, vol. 82, Article ID 103806, 2021.

[8] J. Ahmad, A. Israr, and K. Kim, "Human posture estimation and sustainable events classification via pseudo-2D stick model and K-ary tree hashing," *Sustainability*, vol. 12, no. 23, 9814 pages, 2020.

[9] L. Gian, A. Russo, A. Naddeo, and N. Cappetti, "A resource constrained neural network for the design of embedded human posture recognition systems[J]," *Applied Sciences*, vol. 11, no. 11, p. 4752, 2021.

[10] W. Kim, J. Sung, D. Saakes, C. Huang, and S. Xiong, "Ergonomic postural assessment using a new open-source human pose estimation technology (OpenPose)," *International Journal of Industrial Ergonomics*, vol. 84, Article ID 103164, 2021.

[11] K. Fujiwara and K. Yokomitsu, "Video-based tracking approach for nonverbal synchrony: a comparison of Motion Energy Analysis and OpenPose," *Behavior Research Methods*, vol. 53, no. 6, pp. 2700–2711, 2021.

[12] L. Needham, M. Evans, D. Cosker, and S. Colyer, "Can markerless pose estimation algorithms estimate 3D mass centre positions and velocities during linear sprinting activities?" *Sensors*, vol. 21, no. 8, p. 2889, 2021.

[13] P. Trettenbrein and E. Zaccarella, "Controlling video stimuli in sign language and gesture research: the OpenPoseR package for analyzing OpenPose motion-tracking data in R," *Frontiers in Psychology*, vol. 12, 2021.

[14] E. Jensen, V. Lugade, J. Crenshaw, E. Miller, and K. Kaufman, "A principal component analysis approach to correcting the knee flexion axis during gait," *Journal of Biomechanics*, vol. 49, no. 9, pp. 1698–1704, 2016.

[15] M. Elhoseny, A. Tharwat, A. Hassanien, and H. Aboul, "Bezier curve based path planning in a dynamic field using modified genetic algorithm," *Journal of Computational Science*, vol. 25, pp. 339–350, 2018.

[16] N. Alluri, S. Selvarajan, A. Chandrasekhar, B. Saravanakumar, J. Jeong, and S. Kim, "Piezoelectric BaTiO 3/alginate spherical composite beads for energy harvesting and self-powered wearable flexion sensor," *Composites Science and Technology*, vol. 142, pp. 65–78, 2017.

[17] I. Takeda, A. Yamada, and H. Onodera, "Artificial Intelligence-Assisted motion capture for medical applications: a comparative study between markerless and passive marker motion capture," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 24, no. 8, pp. 864–873, 2020.

[18] Y. Tsai, L. Hsu, Y. Hsieh, and S. Lin, "The real-time depth estimation for an occluded person based on a single image and OpenPose method," *Mathematics*, vol. 8, no. 8, p. 1333, 2020.

[19] J. Yang, T. Ishikawa, T. Tokoro, T. Nakamura, I. Kijiya, and T. Okayasu, "Effect evaluation of drainage condition and water content on cyclic plastic deformation of aged ballast and its estimation models," *Transportation Geotechnics*, vol. 30, Article ID 100606, 2021.

[20] N. Wang, J. Ma, D. Jin, and B. Yu, "A special golden curve in human upper limbs' length proportion: a functional partition which is different from anatomy," *BioMed Research International*, vol. 2017, pp. 1–6, 2017.

[21] U. Kristianto, B. Michae, G. Gurme, and S. Manu, "Using probabilistic fault tree analysis and Monte Carlo simulation to examine the likelihood of risks associated with ballasted railway drainage failure," *Transportation Research Record*, vol. 2675, no. 6, pp. 70–89, 2021.

[22] N. Nakano, T. Sakura, K. Ueda et al., "Evaluation of 3D markerless motion capture accuracy using OpenPose with multiple video cameras," *Frontiers in Sports and Active Living*, vol. 2, p. 50, 2020.

[23] N. Cappetti and M. Di, "Study of the relationships between articular moments, comfort and human posture on a chair," *Work*, vol. 68, no. s1, pp. S59–S68, 2021.

[24] T. Ito, K. Ayusawa, E. Yoshida, and H. Kobayashi, "Evaluation of active wearable assistive devices with human posture reproduction using a humanoid robot," *Advanced Robotics*, vol. 32, no. 12, pp. 635–645, 2018.