Research Article

A Basketball Training Posture Monitoring Algorithm Based on Machine Learning and Artificial Intelligence

Shihao Hou [ID],[1] **Bizhen Lian,**[1] **Wenhao Li,**[1] **and Hong Tang**[2]

[1]*China Basketball College, Beijing Sports University, Beijing 100084, China*
[2]*Sichuan University of Media and Communications, Chengdu 611745, Sichuan, China*

Correspondence should be addressed to Shihao Hou; 2020240860@bsu.edu.cn

To increase the widespread attention of human gesture recognition technology, this paper proposes a basketball pose recognition method based on unit action division. Initially, the human gesture recognition algorithm is introduced for the verification of various effects and gestures of basketball players by monitoring various actions of basketball and to obtain the data of limbs using different detectors for different basketball movements. A large amount of data collection work was carried out for the experiment, and the corresponding experimental scenarios were described in the experimental design for testing. The methods for data processing and data division presented in this work are used to process the collected data. A feature vector set that describes a particular action is acquired and used as a sample set. The sample set is then delivered to the classifier. The classifier is implemented here based on the already-existing Weka platform, and performance evaluation and analysis of various classifiers are implemented. The results show that the differential limb function category has a better recognition effect on BP. The average accuracy of upper limb function was 92.19%, the average recall rate was 92.19%, and the accuracy of lower limb was much higher. The average accuracy of the four algorithms was within the range of 96.99% to 99.19% for lower limb movements and 84.89% to 92.19% for upper limb movements. The BP prosthetic network is used to create separate classifiers, ensuring that each basketball move was more than 95% accurate and that the average accuracy per basketball move was much more accurate. As a result, the accuracy level reached up to 98.85%. The validity of the basketball gesture recognition method recognized by the authors is sufficient and reasonable.

1. Introduction

Motion sensors are called inertia sensors and include acceleration sensors (also known as acceleration meters) and angular velocity sensors, as well as IMU (inertia measurement units) and AHRS (magnetic sensor approach systems) combined in one, two, or three axes. Sensors are mainly used to detect and measure acceleration, deflection, impact, vibration, rotation, and multi-degree freedom of movement [1]. With the development of intelligent sensors, inertial sensors are becoming smaller, lighter, more accurate, and cheaper. For this reason, motion sensor modules are increasingly being used in relevant trend assessment research and commercial applications. In order to control the attitude of objects such as the mobile racing game interactive control

system, it is possible to simulate the spatial approach of the mobile phone, accurately restore the steering wheel of a racing car, and gain a more realistic driving experience [2]. By controlling object orientation, we can perform virtual simulations and visual experiences. In order to monitor people's physical condition, such as counting the movement steps of the Xiaomi Mi Band, it is possible to periodically analyze the change in acceleration in the human body while walking, and then calculate and display the number of human steps. This calculation plays a critical role in augmented reality [3]. In addition, human condition analysis have always been a hot field in college and university research projects, and motion sensing modules play an important role in body condition monitoring research. Image recognition technology for image retrieval is relatively

advanced, and gesture recognition accuracy is very high. However, the disadvantage of this method is that video surveillance has a dead angle, requires a large amount of equipment and a large workload, so it is not suitable for promoting and using the main idea of inertial sensor identification. The athletes wear a simple and portable data acquisition sensor and send the collected data to the real-time processing terminal to identify the position of athletes according to various position data. This kind of method has become one of the most popular image capture research approach at present, which makes up for the low environmental requirements, high recognition efficiency, basketball position recognition, etc., as shown in Figure 1. This is a human body positioning system based on a micro-inertial sensor.

In accordance with the results of the current study, the gathered data is processed using the data processing and data division methods outlined in this work, and a feature vector set that describes a particular action is obtained, constituting a sample set. Last but not least, the classifier receives the sample set. The classifier is implemented here using the Weka platform, which is already in place. Here, a number of classifiers' performance is compared, and their analysis is performed.

The remainder of this article is structured as follows: the literature review is presented in Section 2. The methods and approaches employed in this investigation are given in Section 3. The experimental findings are presented in Section 4. Finally, the study comes to an end, and Section 5 offers suggestions for future research directions.

2. Literature Review

For this research question, when Meng and Qiao. discussed solutions to common dribbling problems, summarized the common problems in dribbling training, these include touching the ball with the palm of the hand, running with the ball, double dribbling, unable to hold the ball when turning, insufficient ball protection when changing direction in front of the body, kicking the ball, and looking down at the ball while dribbling. The solutions are given, but most of them are macro guidelines such as "repeated practice" or "emphasis on certain technical principles", the details of technical training have not been discussed in depth [4]. Tang analyzed the technical movements of dribbling, which clearly pointed out the key features such as "raise the head, look at the glasses, and do not touch the ball with the palm" [5]. Ji . introduced three macro-problems related to dribbling technique learning: unclear understanding of their own quality, ignoring the importance of dribbling training, and lack of motivation to actively learn to dribble, and proposed a training method to improve the three-point dribbling technique: game competition method, mutual aid teaching method, and fun practice method, these three methods correspond to the psychological, interactive, and recreational utilization methods, in addition to the physical strengthening training, psychological triggering factors are the main development direction to improve the effect of basketball dribbling training [6]. Luo. have investigated the

coordination of a whole body task (basketball free throw); among them, the shooting posture and the trajectory of the ball are used as independent and dependent variables, and the shooter's posture stability, posture movement, and ball release can be very reliable predictors. And obtained a functional relationship between shooting posture, ball release level, and hit rate [7]. Yang et al. studied foot position in basketball players with a history of tibial injuries, and concluded that foot abnormalities are common (80%) in basketball players with ankle pain, and these findings could be prevented and supported by footwear and orthopedic prescriptions. [8]. The influence of the offensive decision-making process on 1V1 basketball subsystem is analyzed, which is related to the relationship between defensive position and angle. After the video performance, digital analysis was conducted to determine the position of the attacker's and defender's feet and the trajectory of the participant's evacuation movement [9]. The use of inertial sensors to collect hand movement data to identify hand movements has led to effective interaction between doctors and computers [10]. A Human body positions to complete cognitive functions and can be used in human-computer interaction [11]. Inertial sensors are used to detect the strength of detecting and identifying cruciate ligament injuries, so as to reduce human knee joint injuries [12]. Acceleration sensors were used to monitor 24 classes in the gym and at home and obtain their activity information to calculate the energy consumption of the human body [13].

3. Methodology and Techniques

3.1. Classification Algorithm. Classification training is an important step in human gesture recognition, after data collection, feature extraction, and selection of different actions, the attribute vector describing the human posture information is obtained, and a large number of attribute vector sets are combined together, it constitutes the feature vector set of a single action. And this kind of research on gesture recognition through sensor data, generally, the statistical pattern recognition method is used, which requires a large number of sample sets for classifier training [14]. As a core part of human gesture recognition, classifier training needs to select different classification algorithms and evaluate them; finally, the optimal classifier is selected. There are many common classification algorithms, and four are listed below for detailed introduction [15].

3.1.1. Decision Tree (Decision Tree, DT). Often employed for data classification and regression, decision trees are supervised machine learning techniques. Its purpose as a decision tree is to organize complex problems into a hierarchical framework, making it a multilayer decision-making model. As a statistical theoretical model, a decision tree can be thought of as a tree structure made up of nodes and directed edges. Among them, the node type includes leaf nodes and internal nodes, the internal node represents the detection of a certain attribute of the multisample, and the extended branch represents a detection result. Leaf nodes represent a
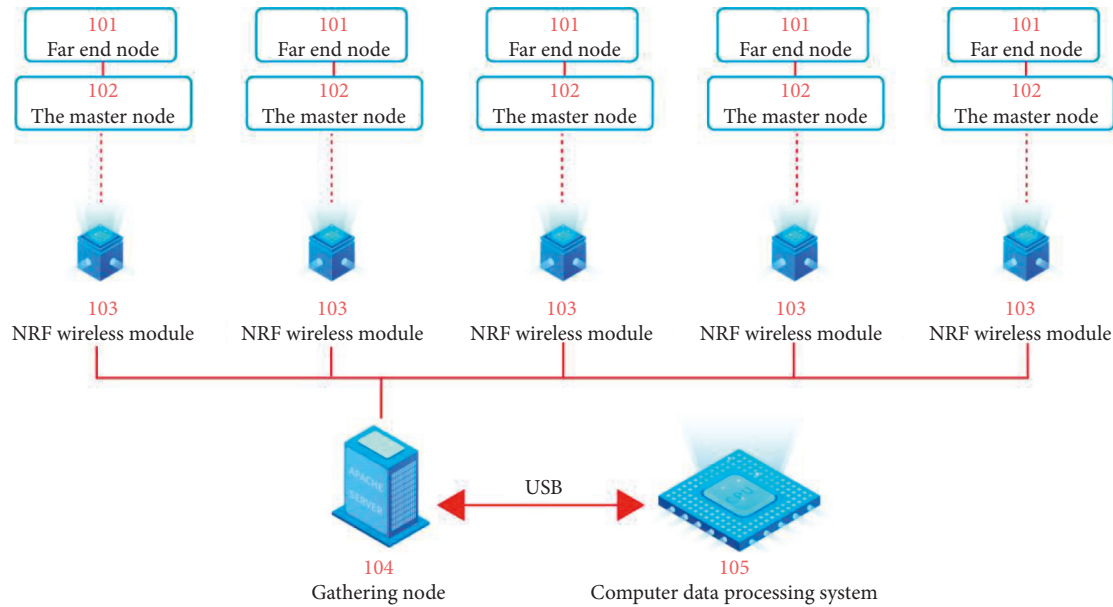
FIGURE 1: A human body attitude tracking system based on micro-inertial sensors.

specific classification result. Building a decision tree is a complex process, and feature selection and splitting are key steps in building a decision tree. One of them is feature selection, which has a specific information reference that is based mostly on an index connected to the feature. The more popular indicators include gain rate, information gain, entropy, etc. The purpose is to preserve the data pointing to each edge of the same kind as much as feasible for feature splitting, which can also be thought of as a subdivision approach where the distinct categories are screened out. Algorithms for building decision trees frequently used include C4.5, ID3, CART, etc. The decision tree creation procedure is straightforward, will not really take very long, and has a simple design principle [16].

3.1.2. Naive Bayes (NB). This method is a simple classification algorithm based on Bayes' theorem, on the basis of the given sample category, by calculating the probability of each category to which the sample to be classified belongs, according to this, the category with the highest probability is selected as the classification result, so as to realize the classification of the sample data. Naive Bayesian algorithm has a strong theoretical basis, it is derived from classical mathematical theory, and it is a relatively stable classifier. Since this method is easy to calculate, insensitive to missing data, it is often used in some uncomplicated classification problems. There are some limitations to the practicality of this method, which requires that the sample attributes are independent of each other, which is not common in practical applications. Furthermore, since the likelihood of events occurring in the context of the application of human gesture recognition is unclear, it is challenging to determine the prior probability of different classification actions, the Bayesian approach is not applicable [17].

3.1.3. Support Vector Machine (SVM). The support vector machine approach is founded on the idea of a minimal structure. This method, which can automatically identify the vector machine that achieves the ideal classification situation, is originally used to deal with two simple categories of classification situations. These classifications are linear; therefore, the method has a strong capacity for generalization. Support vector machines employ kernel tricks that allow them to be applied to nonlinear classification situations. A combination of several support vector machines is used to provide multimode recognition. The support vector machine approach is currently utilized widely in the fields of handwritten type recognition, image classification, text classification, and other areas [18].

3.1.4. Artificial Neural Network (ANN). This method imitates the biological neural network model using a machine learning technique. The complex network structure of the biological neural network model is made up of several interconnected neurons, each of which has a clear structure and fulfils a specific function. The hidden layer, sometimes known as the "hidden layer," includes additional layers between the input and output layers. It primarily handles the functions of message transmission, analysis, and trade-off. The output layer is used to output the obtained results to generate an output vector.

Neural networks are used in many fields, such as machine vision and speech recognition, based on the principles of neural networks, many methods have been optimized, such as learning vector quantization neural networks, self-organizing mapping neural networks, perceptron neural networks, and inverse neural networks [19].

3.2. Attitude Calculation Algorithm. Quaternion method and extended Kalman filter method are selected to achieve

data aggregation, which improves the accuracy of approximating calculation results. The principles of quaternions, Kalman filters, and extended Kalman filters are described in more detail below.

3.2.1. Quaternion.
The quartile is a simple hypercomplex number that describes the rotation of a solid. Quaternions are all composed of real numbers, and the definition form is shown in the following equation:

$$Q = w + xi + yj + zk. \tag{1}$$

If w, x, y, and z are real numbers and i, j, and k are three imaginary units, a quaternion can be expressed using (w, x, y, z).

Since only a single square can be used to determine the rotation of the rigid body, it is necessary to normalize the quaternion, whose normal shape is shown in

$$Q_{norm} = \frac{Q}{\sqrt{w^2 + x^2 + y^2 + z^2}}. \tag{2}$$

The differential equations for the quaternions can also be presented as

$$\dot{Q} = 0.5 \cdot Q \cdot p. \tag{3}$$

It is the quaternion with respect to time

$$p = 0 + \omega_x i + \omega_y j + \omega_z k. \tag{4}$$

Let $\dot{Q} = a + bi + cj + dk$, according to the complex number arithmetic, there are $ii = -1$, $ij = k$, $ji = \ldots$,. Bringing equation (1) and equation (4) into equation (3), we can obtain

$$\begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} = 0.5 \begin{bmatrix} -x\omega_x - y\omega_y - z\omega_z \\ w\omega_x - z\omega_y + y\omega_z \\ z\omega_x + w\omega_y - x\omega_z \\ -y\omega_x + x\omega_y \end{bmatrix}. \tag{5}$$

The new state square can be obtained by normalizing the unit quaternion used to determine the transition from a quarter of the original state to a position in the solid. The reform equation is as follows:

$$Q_{k+1} = Q_k + \Delta t \cdot \dot{Q}_k. \tag{6}$$

Since $\dot{Q}_k$ equation (7) can be obtained according to equation (3).

$$\dot{Q}_k = 0.5 \cdot Q_k \cdot p_k. \tag{7}$$

Substituting equation (7) into equation (6), we can obtain

$$Q_{k+1} = Q_k + 0.5 \cdot Q_k \cdot p \cdot \Delta t. \tag{8}$$

3.2.2. Kalman Filter.
Kalman filter has been widely used in the field of data fusion, it is a linear filter proposed by mathematician Kalman for discrete time-varying linear systems and can be used for state estimation of dynamic systems. The typical application is mainly used for the noise processing of data, which provides the optimal state estimation for the system, and its excellent performance is widely used in various engineering application fields.

For efficient autoregressive filters, Kalman filter can calculate the state of the system according to multiple observation data, and Kalman filter model includes system state transition equation and observation equation. Suppose x_k represents the state of the system at the kth moment, x_{k+1} is the state of the system at the next moment, and w_k is the noise vector of the system process. If it corresponds to the expected property of 0, the system state transition equation can be obtained, and its form is shown in

$$x_{k+1} = \varphi_k x_k + w_k. \tag{9}$$

Among them, φ_k is the state transition matrix of the system, and the state transition matrix represents the influence factor of the system state change from time k to time $k + 1$.

Let z_k denote the measured value of the system at time k, v_k and the measurement equation can be expressed as shown in

$$z_k = H_k x_k + v_k. \tag{10}$$

For a system noise covariance matrix, process noise and measurement noise are two independent forms of signal, so equations (11)–(13) are satisfied.

$$E\left[w_K w_I^T\right] = \begin{cases} C_K, & i = K, \\ 0, & I \neq K, \end{cases} \tag{11}$$

$$E\left[v_K v_I^T\right] = \begin{cases} R_K, & i = k, \\ 0, & i \neq K, \end{cases} \tag{12}$$

$$E\left[w_K v_I^T\right] = 0. \tag{13}$$

In the equation, C is the covariance matrix of process noise W, and R is the covariance matrix of measurement noise V. Suppose the system computation was defined at that time t_k, this estimate is a priori based on all known conditions. Denoting this prior estimate by $\hat{x}_k^-$, where $\wedge$ denotes the estimated value, representing the optimal prior estimate. Assuming that the error covariance matrix about $\hat{x}_k^-$ has been obtained, the estimated error is defined as shown in equations (14) and (15).

$$e_k^- = x_k - \hat{x}_k^-, \tag{14}$$

$$\begin{aligned} P_K^- &= E\left[e_K^- e_k^{-T}\right], \\ &= e\left[(x_k - \hat{x}_K^-)(x_k - \hat{x}_k^-)^T\right]. \end{aligned} \tag{15}$$

In most cases, the prediction process has no previous measurement, so if the process noise is 0, the initial calculation is also 0, and the associated error covariance matrix is the system state X covariance matrix. Based on preliminary estimates $\hat{x}_k^-$, the prior estimate of the system can be optimized by the measured value z_k of the system, in

equation (16), the measured values and prior estimates of the noisy system are fused.

$$\widehat{x}_k = \widehat{x}_k^- + K_k\left(z_k - H_k\widehat{x}_k^-\right), \tag{16}$$

where $\widehat{x}_k$ is the updated estimate and K_k is the fusion factor of the data. Among them, $z_k - H_k\widehat{x}_k^-$ is called the residual, which means the difference between the predicted value and the measured value, if this is 0, it means the prediction matches the measurement.

In equation (16), K_k is a key quantity in Kalman filter. The main problem now is to find a special fusion factor K_k to make the update process optimal; here we use the minimum mean square error estimate as the calculation criterion, and the optimal solution is obtained as shown in

$$K_k = P_K^- H_K^T\left(H_k P_K^- H_k^T + R_K\right)^{-1}. \tag{17}$$

This special K_k can make the equation get the smallest mean square estimation error, and this special K_k is called the Kalman gain. The Kalman gain is obtained, so that the optimally estimated covariance matrix can be obtained in the following equation:

$$P_k = P_k^- - K_K H_k P_k^- - P_k^- H_K^T K_K^T + K_K\left(H_k P_k^- H_K^T\right)K_K^T. \tag{18}$$

Bringing the Kalman gain into equation (18) can simplify the posterior covariance equation, and into equations (18) and (19) can be obtained as

$$P_k = \left(I - K_k H_k\right)P_k^-. \tag{19}$$

Equation (19) is relatively simple, this equation is often used in practical applications, but it should be noted that equation (19) only holds when the optimal Kalman gain is used.

The means of fusing the measured values at time k has been given as shown in equation (16), and the measurement value Z_{K+1} at the next moment also needs these two values for fusion calculation. The update estimate of this state can be obtained through the state transition matrix. Ignoring the noise vector to get the state estimate because this noise vector is expected to be 0 and has nothing to do with other noise vectors, so equation (20) can be obtained.

$$\widehat{x}_{k+1}^- = \varphi_k\widehat{x}_k. \tag{20}$$

The error covariance matrix associated with $\widehat{x}_{k+1}^-$ can be obtained by

$$e_{k+1}^- = \varphi_k e_k + w_k. \tag{21}$$

In equation (21), since w_k is the process noise at the previous moment, it is irrelevant to e_k, so P_{k+1}^- can be expressed as the following equation:

$$P_{k+1}^- = \varphi_k P_k \varphi_k^T + C_k. \tag{22}$$

As shown above, the five equations (16), (17), (19), (20), and (22) constitute the autoregressive process of the Kalman filter, it includes three stages: prediction, correction, and update. First, in the prediction stage of the system, the prediction of the system state can be completed by equation (20), and the prediction of the system state covariance matrix can be completed by equation (22). The state estimation of the system can be obtained through the prediction stage, and the observed value of the system can be obtained by measuring equipment. It is known that there are process noise and measurement noise in the state estimation and observation of the system, respectively. In order to obtain a relatively accurate estimated value, it is necessary to combine the two signal quantities to calculate, this is conducted during the calibration phase of the system. Kalman gain, as the key quantity of Kalman filter, can be obtained by equation (17). Finally, the optimal estimated value of the system is finally obtained by equation (16). Finally, for the update stage of the system, we use the current state covariance matrix of the system to update the covariance matrix of the next stage by equation (19), which is iteratively used for the next filtering stage [20].

4. Results and Analysis

4.1. Data Division. The data decomposition is divided into two steps. The first step is to decompose the motion state data according to the properties of motion data distribution in two states, and the samples obtained include instantaneous and continuous operations. Since continuous movements are composed of several continuous unit movements, in the second stage of data division, movements of upper and lower limb units are generated for continuous movements according to the nature of the limb angle changes in the movement process.

(1) The data distribution of each sensor can be calculated, and the state of each motion can be determined by the threshold. The principles of the distribution-based data sharing approach are shown in Figures 2–5. Figure 2 shows the leg angular velocity distribution and the running state curve while walking, and Figure 4 shows the arm angular velocity distribution and the running state. In Figures 2 and 4, the abscissa is time, the ordinate is angular velocity distribution and running state, the red line is the running state curve, and the blue line is the angular velocity distribution curve.

Figure 3 shows the angular velocity curve of foot walking, and Figure 5 shows the radial angular velocity curve of transition. As shown in Figure 3, walking is a continuous activity consisting of a large number of unit activities. However, in Figure 5, the change of arm angular velocity during transmission is not intermittent but transient. Therefore, the running state partition can produce the data of the basketball movement state.

(2) Division of various unit actions: During continuous movement, the movement of the unit can be divided according to the operational data because the movement of the leg and arm has a continuous cycle, and its continuous change can be clearly observed from time to time. The angular velocity data were used as a reference for dividing the data, as it was
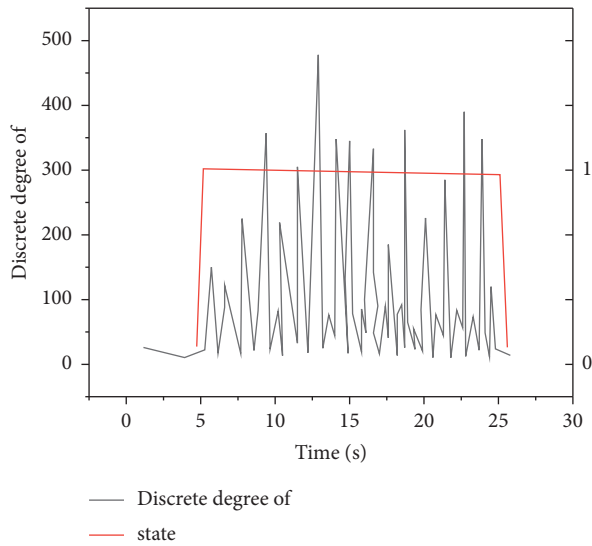
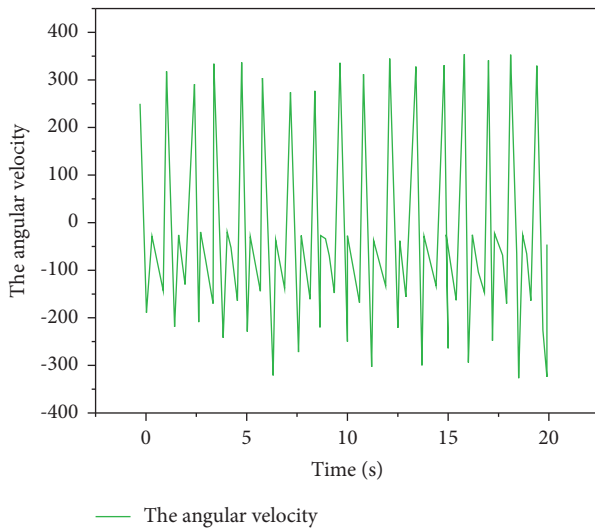FIGURE 2: Velocity distribution and the motion state during walking.

FIGURE 3: The velocity of the legs during walking.

FIGURE 4: Velocity dispersion and the motion state in the process of passing the ball.

FIGURE 5: Arm velocity during passing.

found by comparison that the angular velocity data were the easiest to understand when determining angular changes during rigid body motion. Figure 6 depicts how the leg's angle changes while you walk. The angle curve generated without the use of a Kalman filter technique is represented by the red line. The angle curve changes periodically, but the angle value fluctuates dramatically over time. The angle curve derived by the Kalman filter technique is represented by the blue line, and both sides are essentially equal with a fluctuation of 0 degrees.

Figures 7–10 show the comparison of an angle and calf angle during walking, respectively. As shown in Figures 7 and 9, there are many noise signals of a signal. On the other hand, since the signal curves in Figures 8 and 10 are relatively uniform, the division of unit operation based on angle can reduce the complexity of implementation.
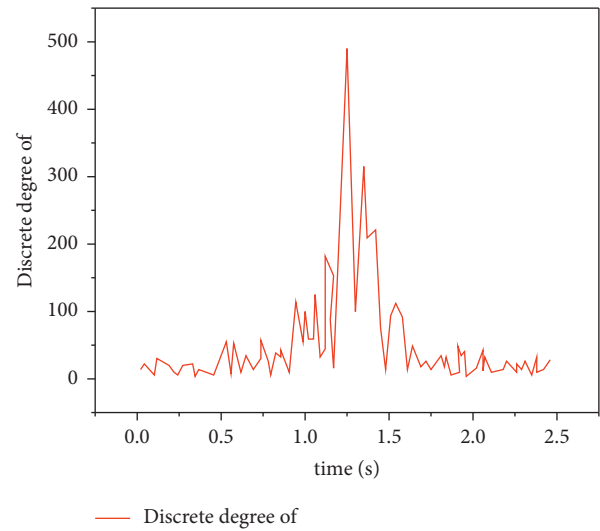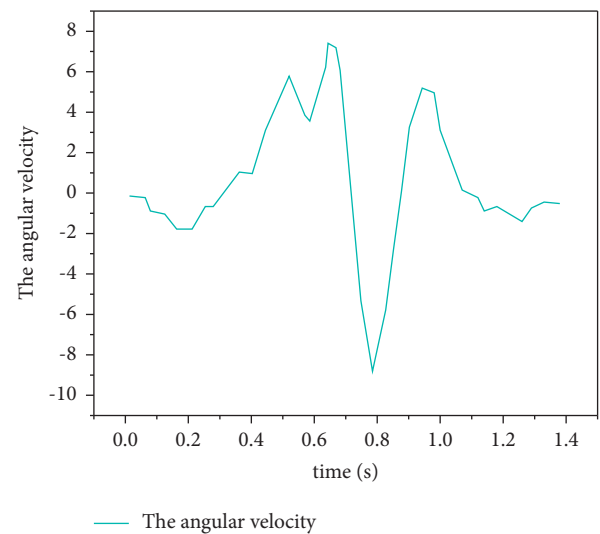
4.2. Experimental Design. Data were collected from eight male examiners who walked, ran, jumped, dribbled standing up, dribbled standing up, ran, caught the ball, passed the ball, and caught the ball, with each movement repeated 50 times. In the process of sampling inspection, each inspector executes the prescribed action according to the provisions, and the inspector records the number of operations. In Tables 1 and 2, samples taken by each tester during data collection are statistically analyzed.

4.3. Results Analysis and Discussion. Basketball games typically end with the player's upper and lower limbs moving generally, which must be taken into account independently. Consequently, a different classifier was created to distinguish between upper and lower limb movements, employing a combination of the two to identify and assess the athlete's
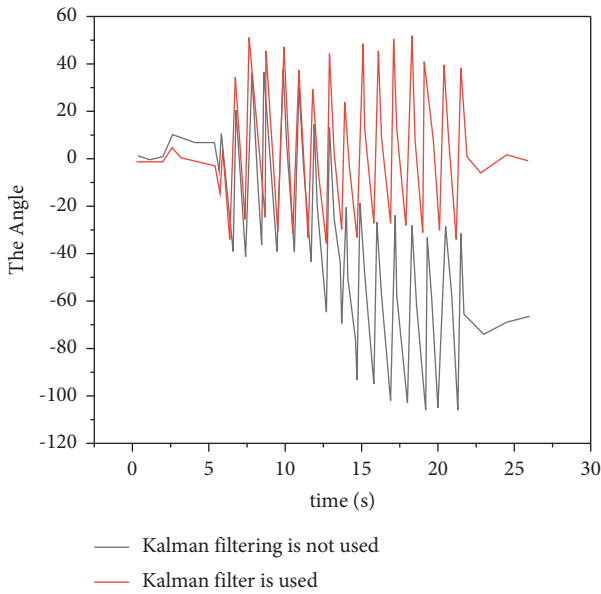
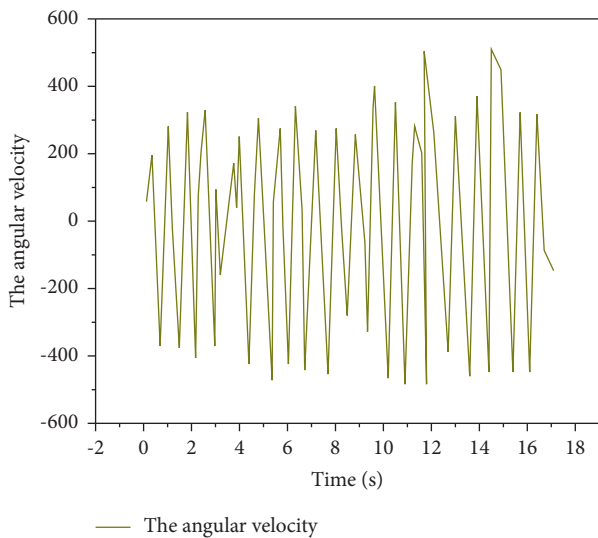FIGURE 6: Comparison of the Kalman filter effect for walking angle change.

FIGURE 7: Forearm angular velocity when dribbling.

FIGURE 8: Forearm angle when dribbling.

FIGURE 9: Calf angular velocity when walking.
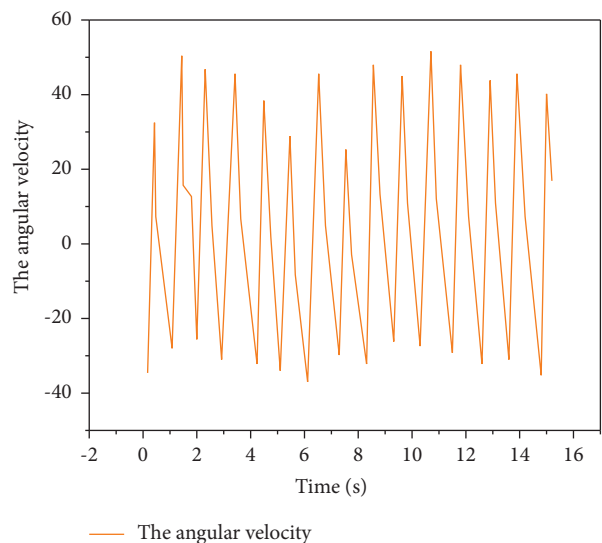
FIGURE 10: Calf angle when walking.

actions. Basketball position classification includes the properties of various classifiers. According to Tables 3 and 4, the entire testing procedure is carried out on the Weka platform using the 10 fold cross verification approach.

Tables 3 and 4 show that different limb function categories have a better identification effect on BP through the artificial neural network. The accuracy of the lower limb movement was 99.19% and the average retraction rate was 99.19%. The average accuracy of the four algorithms ranged from 96.99%–99.19% for lower limb movements and 84.89%–92.19% for upper limb movements, such as dripping, walking, running, and so on, are dripping state. BP prosthesis network was used to construct motion classifiers for upper and lower limbs, with the abscissa representing motion type and the ordinate representing accuracy, as shown in Figure 11. The identification of basketball motion position was completed.
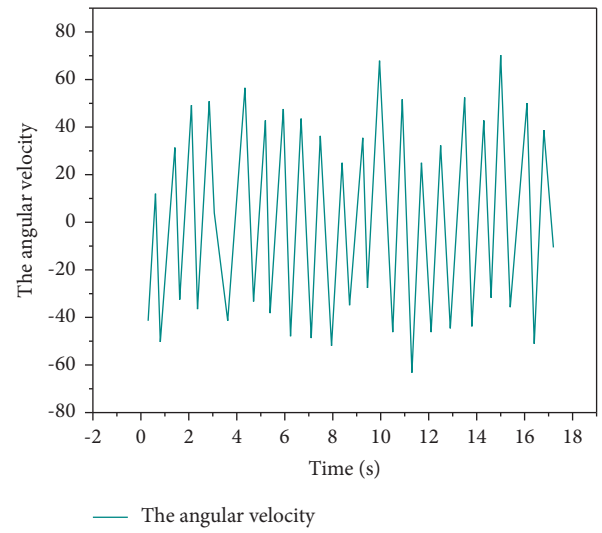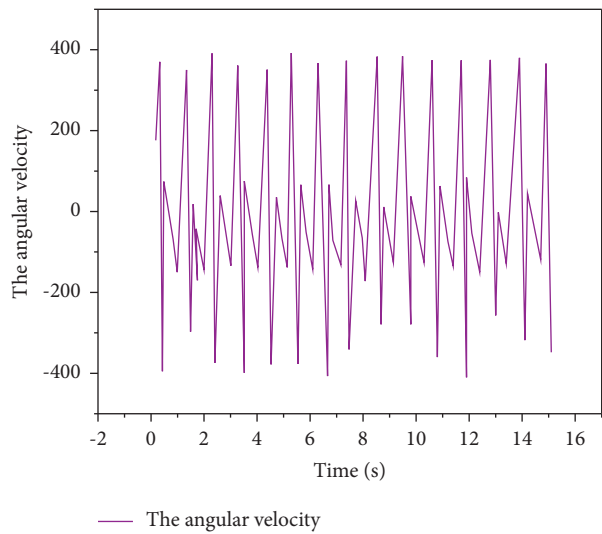
TABLE 1: Number of data collection (samples) for each tester.

Behavior	Number of actions
Walking	50
Running	50
Jumping	50
Stand dribble	50
Walking dribble	50
Running dribble	50
Shoot	50
Passing the ball	50
Catch the ball	50

TABLE 2: Movements of different upper and lower limbs.

Different upper and lower limbs	Varying actions	Varying number of actions
Arm movement	Standing dribble	400
	Walk dribble	400
	Running dribble	400
	Shoot	400
	Pass the ball	400
	Catch the ball	400
Leg movement	Walk	400
	Run	400
	Jump	400
	Shoot	400
	Running dribble	550
	Dribble	450

TABLE 3: Upper limb movements classified using various classification algorithms.

Behavior	C4.5 accuracy	C4.5 recall	SVM accuracy	SVM recall	BN accuracy	BN recall	BP-ANN accuracy	BP-ANN recall
Catch the ball	0.939	0.954	0.965	0.976	0.895	0.929	0.965	0.978
Pass the ball	0.910	0.932	0.976	0.990	0.936	0.956	0.983	0.980
Shoot	0.925	0.909	0.988	0.994	0.958	0.915	0.979	0.982
Walk dribble	0.887	0.863	0.931	0.918	0.791	0.945	0.933	0.918
Standing dribble	0.786	0.767	0.817	0.569	0.778	0.701	0.905	0.877
Running dribble	0.782	0.802	0.691	0.862	0.804	0.757	0.875	0.899

TABLE 4: Movements of the upper limbs classified using several classification methods.

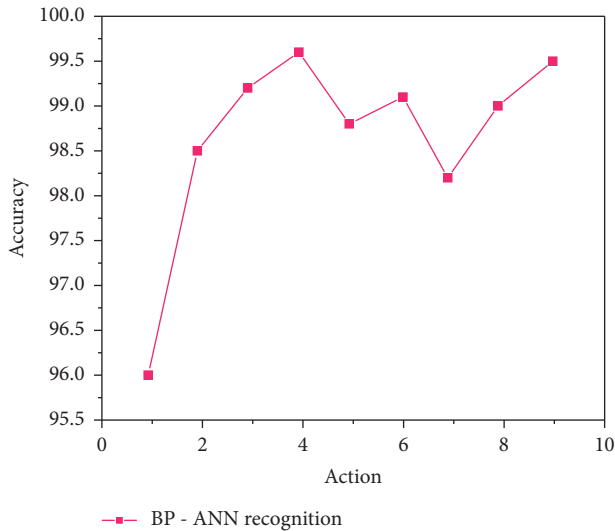| Behavior	C4.5 accuracy	C4.5 recall	SVM accuracy	SVM recall	BN accuracy	BN recall	BP-ANN accuracy	BP-ANN recall
Jump	0.997	0.995	0.982	0.998	0.978	0.975	0.984	0.998
Run	0.983	0.980	0.997	0.978	0.945	0.974	0.996	0.984
Walk	0.971	0.970	0.987	0.996	0.980	0.957	0.992	0.993

Figure 11: BP-ANN recognition results of basketball poses.

5. Conclusion

This paper discusses a basketball position recognition method based on action division in the field of artificial intelligence using a position tracking algorithm. The first step is to determine the position of the basketball, ready to identify the classified action of the basketball and then the data sharing part. The first step is to analyze the basketball, and then summarize the data sharing method according to the analysis results. Finally, a classification method is developed to generate relevant key data according to different fluctuation characteristics of upper and lower limbs of the basketball movement, and complete the identification of the basketball movement. This paper analyzes different classifiers, identifies the position of the basketball, compares the performance data classification of different body movements, establishes the classification algorithm suitable for training, and analyzes the recognition effect from the accuracy of two perspectives. The experiment shows that the method proposed by the author has practical significance in recognizing basketball gestures. As far as the standard of basketball dribbling posture is concerned, there is no objective data to support it both at home and abroad, for example, the degree to which the head bows will affect the ball carrier's judgment of the situation on the field. Subsequent research should use causality to determine which posture can bring better actual combat effects in real situations, and use this as a new generation of bad posture judgment criteria.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

[1] Y. Yang, "Smart community security monitoring based on artificial intelligence and improved machine learning algorithm," *Journal of Intelligent and Fuzzy Systems*, vol. 38, no. 1, pp. 1–13, 2020.

[2] C. Guo, "Prediction and Evaluation Model of Physical Training for Volleyball Players' Effect Based on Grey Markov Theory," *Scientific Programming*, vol. 2021, pp. 1–8, Article ID 61470322021.

[3] Y. Liu and H. Jing, "A Sports Video Behavior Recognition Using Local Spatiotemporal Patterns," *Mobile Information Systems*, vol. 2022, Article ID 4805993, 2022.

[4] L. Meng and E. Qiao, "Analysis and Design of Dual-Feature Fusion Neural Network for Sports Injury Estimation Model," *Neural Computing & Applications*, vol. 2021, pp. 1–13, 2021.

[5] J. Tang, "An Action Recognition Method for Volleyball Players Using Deep Learning," *Scientific Programming*, vol. 2021, Article ID 3934443, 2021.

[6] R. Ji, "Research on basketball shooting action based on image feature extraction and machine learning," *IEEE Access*, vol. 8, pp. 138743–138751, 2020.

[7] L. Luo, "A Sports Digital Training System Based on Middle and Bottom Visual Information," *Mobile Information Systems*, vol. 2022, Article ID 6765954, 2022.

[8] B. Yang, B. Cheng, Y. Liu, and L. Wang, "Deep Learning-Enabled Block Scrambling Algorithm for Securing Telemedicine Data of Table Tennis Players," *Neural Computing & Applications*, vol. 2021, pp. 1–14, 2021.

[9] F. Liu and W. Zhang, "Basketball Motion Posture Recognition Based on Recurrent Deep Learning Model," *Mathematical Problems in Engineering*, vol. 2022, pp. 1–7, 2022.

[10] Y. Egi, "Basketball self training shooting posture recognition and trajectory estimation using computer vision and Kalman filter," *Journal of Electrical Engineering*, vol. 73, no. 1, pp. 19–27, 2022.

[11] G. Saini, Seema, and K. Mor, "Machine learning and prophecy of behavior: a breakthrough in artificial intelligence," *IOP Conference Series: Materials Science and Engineering*, vol. 1099, no. 1, Article ID 012026, 2021.

[12] D. Ö. Kaplan, "Evaluating the effect of 12 Weeks football training on the posture of young male basketball players," *Journal of Education and Training Studies*, vol. 6, no. 10, pp. 47–53, 2018.

[13] G. N. Moawad, J. Elkhalil, J. S. Klebanoff, S. Rahman, N. Habib, and I. Alkatout, "Augmented realities, artificial intelligence, and machine learning: clinical implications and how technology is shaping the future of medicine," *Journal of Clinical Medicine*, vol. 9, no. 12, p. 3811, 2020.

[14] F. Kilinç, H. Yaman, and E. Atay, "Investigation of the effects of intensive one-sided and double-sided training drills on the postures of basketball playing children," *Journal of Physical Therapy Science*, vol. 21, no. 1, pp. 23–28, 2009.

[15] H. Ren and X. Wang, "Application of wearable inertial sensor in optimization of basketball player's human motion tracking method," *Journal of Ambient Intelligence and Humanized Computing*, vol. 2021, pp. 1–15, 2021.

[16] M. Grabara, "A comparison of the posture between young female handball players and non-training peers," *Journal of Back and Musculoskeletal Rehabilitation*, vol. 27, no. 1, pp. 85–92, 2014.

[17] F. M. Verhoeven and K. M. Newell, "Coordination and control of posture and ball release in basketball free-throw shooting," *Human Movement Science*, vol. 49, pp. 216–224, 2016.

[18] W. Zhu, "Classification accuracy of basketball simulation training system based on sensor fusion and Bayesian algorithm," *Journal of Intelligent and Fuzzy Systems*, vol. 39, no. 4, pp. 5965–5976, 2020.

[19] P. Yao, "Real-time Analysis of Basketball Sports Data Based on Deep Learning," *Complexity*, vol. 2021, pp. 1–11, Article ID 9142697, 2021.

[20] M. Nakai, Y. Tsunoda, H. Hayashi, and H. Murakoshi, "Prediction of basketball free throw shooting by open pose," in *Proceedings of the JSAI International Symposium on Artificial Intelligence*, pp. 435–446, Springer, Cham, November. 2018.