Research Article

Design of Basketball Player Training Action Error Correction System Based on Convolutional Neural Network Algorithm

Hui Fang

Nanjing University of Posts and Telecommunications, Nanjing, Jiangsu 210023, China

Correspondence should be addressed to Hui Fang; 20140106@njupt.edu.cn

Received 23 January 2022; Revised 20 March 2022; Accepted 23 March 2022; Published 16 April 2022

Academic Editor: Naeem Jan

Copyright © 2022 Hui Fang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the development of industry and the progress of science and technology, more and more new technology is gradually applied to the movement of error correction. This not only relieves workers of unneeded burdens by making their task more straightforward and error-free, but it also improves production efficiency. Deep neural networks are one of these new technologies that have exploded in popularity in recent years, with applications in a variety of industries. Of course, the application in image recognition must not be less, image recognition technology based on deep neural network has become more mature, and the error rate of recognition is now much lower than human vision recognition. So at present, some industrial detection is gradually from human vision detection to computer vision detection. This study discloses a basketball action error correction method based on deep learning image recognition, which includes the following steps: receiving each frame of basketball image captured from the fitness video, recording the corresponding time of each frame of fitness image, and preprocessing each frame of fitness image; the preprocessed basketball image was fed into the human joint recognition model, and the human joint recognition model calculated each human joint in the fitness image and output its position coordinates. According to the coordinate position of each joint orderly line, the human skeleton diagram is obtained; the human skeleton diagram is compared and assessed in accordance with standard fitness action, and the nonstandard basketball image is generated to realize basketball action repair. A basketball action error correction system based on deep learning picture identification is also disclosed in the invention. The system and method are capable of efficiently addressing the difficult challenge of comparing fitness movements with and without music rhythm.

1. Introduction

In basketball teaching, teachers try to use their own long-term sports practice and many years of teaching experience, absorb other scientific knowledge, and strive to adopt the most effective methods and means, so that students through their own efforts, in the shortest possible time to learn a lot of knowledge, master more skills. Teachers, students, teaching content, teaching methods, and means and other factors influence and restrict each other. In teaching, students will produce wrong movements, and if these wrong movements are not found and corrected in time, wrong techniques will be formed, which will not only affect students’ mastery of correct techniques, but also may lead to accidents. Therefore, checking and correcting wrong actions are an indispensable part of basketball teaching. In teaching, the inspection and correction of students’ errors are similar to a doctor treating an illness, diagnosis of the cause, condition, prescription, appropriate treatment, and medicine to cure the ailment. Teachers in basketball should examine students’ grasp of skills at any time, identify new incorrect actions, analyze the causes, and take prompt and decisive action to correct mistakes to continuously enhance the quality of instruction. This study attempts to summarize and discuss four aspects, such as how to check the wrong action, the cause of the wrong action, the method of correcting the wrong action, and the problems that should be paid attention to when correcting the wrong action, to provide reference and basis for enriching the basketball teaching methods.

In basketball teaching, teachers should observe the situation of students’ mastery of technology at any time, to discover problems and take solutions to problems in time.
Some students’ wrong actions are more obvious, and teachers can usually only go by experience and see if they can be wrong. However, due to the complex structure of some technologies, individual differences in students, and other factors, more than two wrong actions may occur when learning a certain technology, and it is difficult to see the key to the wrong actions. At this point, the teacher must use certain methods to check the wrong actions and identify which is the main aspect of the mistake.

In teaching, the commonly used methods for checking wrong actions are as follows:

1. **Inquiry Method:**
   - inquiry method refers to the teacher asking students to complete the action of the self-feeling, through the students to complete the actual feeling and correct technology for comparison, and then find out the wrong method.

2. **Slow to Complete the Action Method:**
   - By having students slow down and repeat a certain action, the teacher observes and checks. For example, when students dribble and turn around, they run and jump. The teacher cannot see the structure of the action clearly, and it is difficult to determine the key to the error. At this point, let the students turn around in slow-speed dribble, and the teacher is easy to observe the details of each movement and find out the key to the wrong movement.

3. **Full Completion of the Action Method:**
   - By having students do their best to complete the technical movements according to the technical essentials (including the speed, strength, range, and route of the completed movements), the teacher observes and checks.

4. **Imitation Action Method:**
   - imitation action method refers to the method that students complete a certain action according to the technical requirements, and teachers observe and check it. For example, by asking the students to imitate the shooting action, the teacher can detect the mistakes of each action detail (basic standing posture, holding the ball, shooting force order, arm movement, wrist shaking, etc.).

5. **Decomposition to Complete the Action Method:**
   - For some complex techniques, have students break down into parts to complete the movement based on the structure of the technique, and have the teacher observe and check for the different parts. For example, the teacher divides the ball holding breakthrough technology into four technical links, such as stepping on the ground, turning shoulder, releasing the ball, and holding the ball to accelerate, and asks the students to complete the action one link at a time. The teacher observes and inspects a certain link completed by the students.

6. **Partial Observation Method:**
   - partial observation method refers to the method that students observe and inspect a certain link in the whole technical structure after completing a certain technology according to the technical key points. For example, a student’s aerial action to grab the rebound is always not good, and the teacher cannot be sure of the key to its error. At this time, the teacher can let the student repeatedly do the action to grab the rebound, and the teacher will repeatedly observe, analyze, and check the aerial action in the whole rebound technology.

7. **Comprehensive Observation Method:**
   - comprehensive observation method refers to the method of observing and checking the whole process of students completing a certain technology from beginning to end.

In the teaching of basketball skills, especially when learning new skills, it is easy to make wrong actions because the temporary neural connections of the cerebral cortex have not been formed yet, as well as various reasons mentioned above. After finding out the wrong actions and analyzing the causes of the mistakes, the teachers carefully study the methods to correct the mistakes and promote the development of differentiation and inhibition through repeated practice to avoid the interference of various qualitative factors, to achieve the purpose of correcting the wrong actions and establishing the right skills.

In teaching practice, the strategies used to correct incorrect actions vary and differ depending on the objects, technical qualities, and causes of incorrect actions. The following categories of procedures from the ways and means used to remedy mistakes are summarized in this study:

1. **Explanation and Demonstration Method:**
   - explanation and demonstration method is one of the most common methods for correcting mistakes. In interpretation, using the image interpretation (spoken dialects, technology formula, such as image and vivid language), Lenovo interpretation (appropriate metaphor, ask questions, provide clues, etc.) to guide students’ thinking and action of tip (in the presence of students’ complete action, timely warning, such as the pedal turn, jump, and wrist) causes the student to clear the wrong place and of their own fault. On this basis, the integration, decomposition, speed, etc., are demonstrated in different forms according to the technical characteristics, and the front, back, side, mirror image, etc., are demonstrated through different positions. In this way, through the stimulation of the first signal system, students can further understand the complete process and local process of technology, eliminate the traces of wrong concepts in their minds, and establish a profound and correct representation of technology.

2. **Representation Correction Method:**
   - representation correction method refers to the combination of the first and second signal systems through representation reproduction, imagination,
and self-suggestion of the perceived technology, to correct errors. The specific methods of representation correction method are as follows: 1. right and wrong contrast representation method: close your eyes, think about your wrong action and correct action from beginning to end, and then compare the two, where is wrong and how to do; 2. listen to the representation method: the teacher explained the main points of the method and technique and had to recall them in my mind again.

(3) Right and Wrong Contrast Method:
right and wrong contrast method will be wrong actions and correct actions through explanation or demonstration of mutual comparison, comparison, and analysis.

(4) Induction Method:
induction method refers to the use of simple practice means similar to the correct technique in the action structure, which is also called imitation exercise or auxiliary exercise. For example, when correcting some students’ errors in shooting actions, the un-armed imitation shooting exercises or induced exercises such as two-person throwing and wall throwing are often used so that students can be induced to understand the method of shooting. In the process of using induction exercise, if the teacher can combine hand demonstration, various signals and language stimulation, and reinforcement, the effect will be better.

(5) Restriction Law:
restriction law refers to the use of some restrictions to force students to complete the action in accordance with the intention of the teacher, so as to achieve the purpose of correcting wrong actions. For example, some students turn up and down, not horizontally. Teachers can use the restrictive method of holding a horizontal pole at the head of the student’s basic standing position to form an equal height, and students should not touch the horizontal pole when turning around, to help students overcome the mistakes of body ups and downs when turning around.

(6) Transformation Method:
for some techniques that are difficult to learn and require high physical function, attention is paid to correcting them by changing and simplifying some elements of the movement (strength, speed direction, route, etc.) according to the characteristics of the students.

Through the study of the visual system, the researchers propose the application of an artificial neural network to simulate the human visual system for image recognition. A neural network can automatically learn this kind of matrix operator and then extract some recessive features from the image for classification recognition. The most representative convolutional neural network has made long enough progress in the field of image recognition to open a new chapter for image recognition, and the further development of convolutional neural network has injected new vitality into the field of image detection and other fields and promoted the progress of image recognition [1]. also brings new research upsurge to the convolutional neural network. However, the mathematical model structure of deep convolutional neural network itself [2, 3] is complex, and the training cost is still very high. When applying gradient descent algorithm to train parameters of deep coiling neural network, it is not only easy to fall into local optimal solution but also prone to gradient dispersion [4] as the number of layers increases in the process of back propagation. At the same time, with the development of network, the amount of network image data increases rapidly. The massive image data provide support for the training and testing of convolutional neural network model and also provide us with a lot of information. However, the rapid increase in image data has brought great burden to the recognition of massive image data using the convolutional neural network. How to construct simple and efficient convolutional neural network models [5–7] and efficiently analyze and process these image data using convolutional neural network has become a difficult and hot topic in various fields of research.

The following is a description for each section: Section 2 contains the literature review. Section 3 reviews the improved full convolutional neural network FC-VGGNet-plus. Section 4 described the experimental results of professional coaches. Finally, the study ends with the conclusion in Section 5.

2. Related Work
The term movement pattern was first put forward by Mr. Mark Verstegen, founder of EXOS (former AP, Athletes’ Performance). Movements are the scientific movement of the human body. Movements are regarded as the basic carrier of all athletic abilities, techniques, and tactics and ultimately determine athletic performance. Foreign scholars and researchers have recognized the research results of motion pattern earlier. In his book Motion-Functional Movement Training System, Gray Cook (2010) proposed that movement pattern is the fundamental factor determining whether an athlete is good or not. As the smallest component unit of all sports, it has become the cornerstone of sports training. According to Gary Cooke, the body needs multiple neuromuscular coordination to complete an action. This fixed coordination is called a motor program, and each motor program corresponds to a pattern of action. When the brain sends a signal to the body to move, the body quickly responds with a corresponding pattern of action. It makes the movement more automatic and maximizes the economic effect.

Convolutional neural networks (CNNs), developed on the basis of artificial neural networks (ANNs), are a major aspect of artificial intelligence research. An artificial neural network builds a similar network based on the structure and working mechanism of the biological neural network, so that machines can learn rules from a large amount of data and form a network just like human beings.
Frank Rosenblatt proposed the perceptron model \cite{8} and established the first mathematical model of an artificial neural network. Since then, the research of neural network has been in the theoretical stage. It was not until 1998 that LeCun from Bell LABS used the combination of convolutional network and back propagation \cite{9} to classify handwritten numbers, which became a successful application of neural network and set off the research upsurge of neural network. This network is the famous LeNet-5 \cite{10} convolutional neural network, which is considered by researchers to be a convolutional neural network in the real sense. After that, the neural network was in the incubation stage, and the growth of computer data processing capacity made researchers consider to promote the development of neural network by relying on the improvement of hardware performance. The advantages of graphic processing units (GPUs) \cite{11, 12} in intensive data processing and parallel processing accelerated image processing significantly. Dan Claudiu Ciresan and Jurgen Schmidhuber successfully run a 9-layer GPU neural network on a NVIDIA GTX 280 graphics processor, which makes the development of deep convolutional neural networks possible. The AlexNet \cite{13, 14} model proposed by Alex for the improvement of convolutional neural network structure has exploded the application of neural networks. The author uses ReLU \cite{15} activation function to replace sigmoid \cite{16} and the dropout \cite{17} technology to avoid overfitting. In addition, the maximum pooling method is applied to the lower sampling layer to build a deep convolutional neural network and win the champion of the image recognition contest, making CNN become the core algorithm model of image recognition and making the convolutional neural network development in a deeper direction. However, AlexNet is particularly deeply dependent on the network.

To solve the parameter problem of large-scale deep learning networks, MSRA’s Kaiming team proposed the ResNet structural model, and the author believes that most neural network models tend to saturate in accuracy as the depth of the network increases, mainly because of not only fitting, but also because some models themselves lead to an increase in training error \cite{18}. The basic idea of this model is to introduce layer-hopping technique, namely the concept of residual learning, which makes it possible to train neural networks with hundreds or even hundreds of layers. After proposing Inception V3, Christian Szegedy et al. studied a scheme to combine Inception V3 with ResNet proposed by He’s team by accelerating training through residual connection of ResNet structure. An Inception-ResNet V2 network is obtained, and a deeper Inception V4 \cite{19} model is designed, with faster network training speed and higher accuracy.

In terms of the optimization algorithm of convolutional neural network, the gradient descent algorithm has become the core algorithm of the optimization algorithm of convolutional neural network since LeCun applied the combination of convolutional neural network and back propagation. Based on this, stochastic gradient descent (SGD) algorithm \cite{20, 21} was developed, which solves the problem of slow network training caused by the large amount of data operation of gradient descent algorithm. The Adagrad algorithm studied by Duchi et al. is based on a stochastic gradient descent algorithm that can automatically adjust the learning rate during the training process by using larger weight updates for lower frequency parameters and smaller weight updates for higher frequency parameters in order to better handle sparse data. Sutskever and his team proposed the stochastic gradient descent algorithm combined with momentum, which uses the momentum principle in physics to simulate the inertia of objects in motion and determines the current update direction by referring to the value after the last update during the gradient update. If the current update direction is the same or similar to the historical update direction, then the updating trend will be strengthened. If the updating direction is contrary to the historical updating direction, the updating trend will be weakened. In this way, the wave problem caused by the updating of the stochastic gradient descent algorithm solely relying on the current calculated gradient value can be avoided. The momentum algorithm is more stable and easy to jump out of the local optimal solution because it is updated by referring to the historical information.

3. Improved Full Convolutional Neural Network FC-VGGNet-Plus

The two cores of convolutional neural network are convolution and pooling. The random combination of convolution and pooling and the selection of parameters give CNN great flexibility, which also gives birth to many well-known classical networks, among which visual geometry group net (VGGNet) is a very successful and representative deep convolutional neural network model. This chapter deeply analyzes the structure and parameter design of VGGNet model, improves VGGNet model from the direction of full convolution, and designs a new convolutional neural network, fully connected (FC)-VGGNet-plus, to improve network training efficiency on the basis of ensuring accuracy.

We will preprocess the input image, and its processing process is shown in Figure 1. Its flow chart is shown in Figure 2.

3.1. VGGNet Model. VGGNet is one of the classic deep convolutional neural network models. It was proposed by the computer vision research group of Oxford University. The main feature of this model is to replace large convolution kernels with smaller ones, and VGGNet verifies the importance of depth through different depth combinations. Compared with AlexNet, VGGNet has the following improvements:

1. Local response standardization (LRN) layer is removed. The role of LRN in deep networks is not obvious. Such standardization does not improve performance on ILSVRC data sets, but leads to more memory consumption and calculation time.
2. A smaller 3 × 3 convolution kernel is used. AlexNet uses a large convolution kernel, such as one with
Therefore, compared with AlexNet, VGGNet has fewer parameters and higher efficiency.

(3) The pooled nucleus becomes smaller. In VGGNet, the pooled core is $2 \times 2$ and the stride is 2, while in AlexNet, the pooled core is $3 \times 3$ and the stride is 2. Such an improvement is mainly due to the problem of the number of parameters to be solved, the deeper the network means more parameters and harder to train. According to the author’s investigation, the convolution kernel with a size of $3 \times 3$ is sufficient to capture the variations of horizontal, vertical, and diagonal pixels due to the properties of convolutional neural networks. The use of large convolution kernel will bring an explosion of parameter number, and some parts of the image will be convolved many times, which may bring difficulties to feature extraction and convolve many redundant features. Therefore, $3 \times 3$ convolution is commonly used in VGGNet.

The network is divided into five parts of convolution, and each part of convolution has 2–4 convolution layers. After each part of convolution, the pooling operation is carried out in the mode of maximum pooling. The structure of the five-part convolution adopts the mode of 64-128-256-512-512, in which the number represents the number of convolution kernels at the convolution layer. That is, the number of feature maps output by the convolutional layer,
which can be considered as the number of features extracted by the convolutional layer. Thus, it can be seen that VGGNet is a deep convolutional neural network with a large number of parameters. The last three layers are fully connected, the first two layers are connected with 4096 channels, and the third layer is fully connected with 1000 channels. Finally, softmax is used for multi-category classification according to the proposed features, and each classification probability can be output. The classification probability with the largest is the category of the input image. In general, VGGNet adopts multiple groups of small convolutional kernels to replace large convolutional kernels, while increasing the number of network layers and adjusting network structure to improve network performance. Highlights of VGGNet include the following:

1. A large convolution kernel is decomposed into continuous small convolution kernels;
2. Parameters are reduced, the amount of calculation is reduced, and the depth is increased;
3. The simple and effective structure of AlexNet is inherited; and
4. Good performance makes it the preferred basic network for network transformation.

3.2. Structural Design of FC-VGGNet-Plus. VGGNet model has achieved excellent performance in image recognition, but the depth of the network and the huge number of parameters limit the training speed of the network model. At the same time, the depth of the complex model may lead to other problems, such as gradient dispersion when using gradient descent back propagation to correct parameters. However, the number of parameters is mainly consumed in the last three fully connected layers. Although the convolutional part in front is very deep, the number of parameters consumed is not large. The problem caused by model parameter confusion can be handled by modifying the structure of the VGGNet model and replacing the last full connection with full convolution or even deleting the full connection. Due to the large amount of calculation and memory consumption of VGGNet, this study is based on VGGNet-B level network optimization design. Figure 3(a) shows the structure of VGGNet-B network model, which contains five convolution parts and the full connection of the sixth part. Each convolution part contains two convolution layers and a pooling layer. For the convolution neural network model, with the increase in the number, the larger the computation, the more demanding the platform is, but we want to build an efficient training speed model with guaranteed accuracy, so in this study, after analyzing the research of VGGNet-B, the network structure design improvements are made as shown in Figure 3.

The perceptron unit is shown in Figure 4. Input is on the left. Suppose there are three inputs a1, A2, and A3, which are finally output on the right after calculation and transformation in the middle part.

The corresponding calculation formula is as follows:

\[ h_{W,b}(x) = f(W^T x) = f \left( \sum_{i=1}^{3} W_i x_i + b \right), \]

(1)

where W1, w2, w3 are weights and b is bias. The function of activation function is to add nonlinear factors to the model, so that more characteristics can be represented, because without activation function, the transformation of neural network multilayer superposition operation is linear transformation, and the linear model has poor characterization ability, so it is necessary to introduce such nonlinear activation function. The perceptron unit is the most basic component of neural network.

A neural network model is formed when multiple units are combined and have a hierarchical structure.

Inputs are input in the left side and passed through the middle hidden layer to output in the right side, and the corresponding calculation formula is as follows:

\[
\begin{align*}
    a_i^{(2)} &= f(W_i^{(1)} x_1 + W_i^{(2)} x_2 + W_i^{(3)} x_3 + b_i^{(1)}), \\
    a_i^{(2)} &= f(W_i^{(1)} x_1 + W_i^{(2)} x_2 + W_i^{(3)} x_3 + b_i^{(2)}), \\
    a_i^{(3)} &= f(W_i^{(1)} x_1 + W_i^{(2)} x_2 + W_i^{(3)} x_3 + b_i^{(3)}), \\
    h_{W,b}(x) &= a_i^{(3)} = f(W_i^{(1)} a_i^{(2)} + W_i^{(2)} a_i^{(2)} + W_i^{(3)} a_i^{(2)} + b_i^{(3)}).
\end{align*}
\]

The structure contains an input layer, an output layer, and a hidden layer in the middle. Similarly, it can be extended to have 2, 3, 4, 5, . . . . In the convolutional neural network, the hidden layer is composed of the convolution layer and the down-sampling layer. The following will introduce how the multilayer convolutional neural network carries out parameter propagation and parameter training.

Random gradient descent is used to calculate the gradient value of individual samples J by randomly selecting a small amount of input samples, and the average value to estimate the overall gradient value J is calculated. If the number of samples is large enough, the estimated value and the actual value will approach the same, that is,

\[
\frac{\sum_{j=1}^{m} \nabla J(x)}{m} = \frac{\sum_{j=1}^{m} \nabla J(x)}{n} = \nabla J.
\]

(3)

It can be obtained from the following formula:

\[
\nabla J \approx \frac{1}{m} \sum_{j=1}^{m} \nabla J(x).
\]

(4)

The stochastic gradient descent algorithm uses the gradient descent method to calculate the derivative of the error to the weight value layer by layer and update the weight and bias of the network layer by layer. The concrete implementation steps are as follows.

The first step is to find the partial derivative of the cost function J with respect to \(a_L\):

\[
\delta_L^{(a)} = \frac{\partial}{\partial a_L} J = -(t - a_L).
\]

(5)

In the second step, the error is transmitted from A to Z at layer \(L + 1\):
The third step is to transfer the error from \( L + 1 \) layer to \( L \) layer:

\[
\frac{\partial a^{(l+1)}}{\partial z^{(l+1)}} = a^{(l+1)}(1 - a^{(l+1)}). \tag{6}
\]

The fourth step is to find the partial derivatives of the \( l \) layer with respect to \( a \) and \( z \):

\[
\delta_L^{(z)} = \frac{\partial f}{\partial z^{(l)}} = \frac{\partial f}{\partial a^{(l)}} \frac{\partial a^{(l)}}{\partial z^{(l)}} = \delta_1^{(a)} a^{(l)}(1 - a^{(l)}). \tag{8}
\]

4. Experimental Results of Professional Coaches

The frame release detection algorithm is used to automatically acquire the free-throw shot frame through the frame release model. The experiment obtains the frame release, respectively, through the track analysis of the basketball coach and two ordinary students to verify the effectiveness of the algorithm. Some sample data are shown in Figures 5 and 6.

The basketball is positioned after each frame of the picture. Through extensive analysis of free-throw trajectory, one of the video units is listed below. The comparison list of the release angle and entry angle of the 20 groups of free throws is shown in Table 1.
As shown in Table 1, the error between the measured entry angle and the theoretical angle is between 2.8% and 7.5%. Due to the neglect of external forces such as air resistance, it is approximately considered as a parabolic motion only receiving gravity, so there is an error. However, the error is within the acceptable range, and it is still valid for us to analyze the release angle.

After the correction training, the average score of each mode test was significantly improved compared with that before the correction training, and the average score of most movement modes was more than 2.5 points, indicating that the athletes’ various movement modes were effectively improved. The difference between the left and right sides decreased and the total score was almost the same, indicating that the ability of balance and symmetry of the left and right sides of the body was greatly improved. Test analysis of each mode before and after training is shown in Table 2.

Since the initial screening results of movement patterns such as left hurdle stepping, left straight lunge squat, and bilateral shoulder flexibility were good, there was no change in the mean value after the correction training, so the results of the measurement before and after were the same, so the paired-samples t test was not conducted. The results of paired-samples t test before and after the other modes are shown in Table 2. The results of the over-top squat mode, the right hurdle stepping mode, and the trunk stable push-up mode after the correction training are all higher than the results before the training, with $P < 0.05$, showing a significant difference. The results of right active straight knee lifting and right rotation stabilization after training were higher than those before training, but $0.05 < P < 0.01$, and there was no significant difference. The results of right straight lunge squatting and left active straight knee lifting after training were higher than those before training, but both $P > 0.01$. The
reason is not difficult to find that the scores of several movement modes without significant differences have all exceeded 2.5 points in the early stage, indicating that athletes perform well in this movement mode, so it cannot be directly reflected in the paired-samples t test.

After the correction training, eight basketball players were subjected to a second FMS, and the statistical analysis of the scores of each item is shown in the following table.

It can be seen from Table 3 that the 8 athletes have all scored more than 16 points in screening, and the scores of each screening mode are generally more than 2 points, without 0 points, indicating that all athletes can complete the screening content. In the screening results, there was no longer a discrepancy of more than two points between the left and right sides, showing that symmetry and coordination on both sides of the body had greatly improved after functional motor training. It can be seen from Table 3 that the mean total scores of all screening modes measured after training (M = 17.75) are significantly higher than the mean total scores of all screening modes measured before training (M = 15.25) (P = 0.000), which indicates that FMS scores of 8 basketball players are significantly improved after functional motor training. The corresponding basketball action mode has been improved effectively.

Table 1: Comparison list of release angle and entry angle.

| Number | High Angle | Theoretical entry angle | Actual basket entry angle | Error (%) |
|--------|------------|--------------------------|----------------------------|-----------|
| 1      | 2.34       | 60.25                    | 42.61                      | 45.66     | 6.60   |
| 2      | 2.35       | 58.23                    | 41.56                      | 44.61     | 6.80   |
| 3      | 2.34       | 61.5                     | 43.05                      | 46.58     | 7.50   |
| 4      | 2.34       | 61.67                    | 43.35                      | 46.78     | 7.30   |
| 5      | 2.35       | 65.23                    | 45.56                      | 47.94     | 4.90   |
| 6      | 2.34       | 58.73                    | 40.51                      | 43.47     | 6.80   |
| 7      | 2.35       | 64.4                     | 45.27                      | 47.43     | 4.50   |
| 8      | 2.35       | 58.23                    | 41.56                      | 46.61     | 6.80   |
| 9      | 2.34       | 61.45                    | 43.23                      | 46.01     | 6.00   |
| 10     | 2.34       | 63.49                    | 44.64                      | 46.51     | 4.00   |
| 11     | 2.34       | 60.35                    | 40.73                      | 44.06     | 7.50   |
| 12     | 2.34       | 65.33                    | 46.74                      | 48.12     | 2.80   |
| 13     | 2.33       | 58.32                    | 39.71                      | 42.61     | 6.80   |
| 14     | 2.35       | 66.14                    | 47.18                      | 49.31     | 4.30   |
| 15     | 2.34       | 63.43                    | 44.54                      | 46.51     | 4.20   |
| 16     | 2.35       | 64.47                    | 45.27                      | 47.33     | 4.30   |
| 17     | 2.35       | 64.52                    | 45.22                      | 47.35     | 4.40   |
| 18     | 2.35       | 58.23                    | 41.56                      | 44.61     | 6.80   |
| 19     | 2.35       | 58.77                    | 41.53                      | 44.68     | 7.00   |
| 20     | 2.34       | 60.31                    | 40.69                      | 42.97     | 5.30   |

Table 2: Test analysis of each mode before and after training.

|                      | Mean (M) | The correlation coefficient | 95% confidence interval | T   | df | P     |
|----------------------|----------|-----------------------------|-------------------------|-----|----|-------|
|                      |          |                             |                         |     |    |       |
| Overhead the squat   | 2.00     | 2.625                       | 0.898                   | −1.05768 | −0.19232 | −3.416  | 7   | 0.010 |
| Step over hurdles L  | 2.50     | 2.500                       | 0.655                   | −0.94687 | −0.05313 | −2.646  | 7   | 0.033 |
| Step over the hurdles R | 1.750 | 2.250                       | 0.655                   | −0.94687 | −0.05313 | −2.646  | 7   | 0.033 |
| Straight lunge squat L | 2.62   | 2.625                       | 0.354                   | −1.13197 | 0.13197  | −1.871  | 7   | 0.104 |
| Shoulder flexibility L | 2.87   | 2.875                       | 0.354                   | −0.80768 | 0.05768  | −2.049  | 7   | 0.080 |
| Shoulder flexibility R | 3.00   | 3.000                       |                          |     |    |       |
| Lift leg with straight knee | 2.75 | 2.875                       | 0.655                   | −0.42058 | 0.17058  | −1.000  | 7   | 0.351 |
| Lift your legs straight with your knees | 2.37 | 2.750                       | 0.447                   | −0.80768 | 0.05768  | −2.049  | 7   | 0.080 |
| Steady push-ups with torso | 2.12 | 2.750                       | 0.218                   | −1.05768 | −0.19232 | −3.416  | 7   | 0.011 |
| Rotary stability L   | 2.50     | 2.750                       | 0.577                   | −0.63700 | 0.13700  | −1.528  | 7   | 0.170 |
| Rotary stability R   | 2.50     | 2.875                       | 0.378                   | −0.80768 | 0.05768  | −2.049  | 7   | 0.080 |
Table 3: Total score of each mode after training.

| No. | Overhead the squat | Hurdle shangbu | Straight lunges | Shoulder flexibility | Straighten | Steady push-ups with torso | Rotational stability | Total |
|-----|--------------------|----------------|-----------------|----------------------|------------|---------------------------|--------------------|-------|
| A   | 3                  | 3              | 3               | 3                    | 3          | 3                         | 3                  | 20    |
| B   | 3                  | 3              | 2               | 3                    | 3          | 3                         | 2                  | 19    |
| C   | 3                  | 2              | 2               | 3                    | 3          | 3                         | 2                  | 18    |
| D   | 1                  | 2              | 3               | 3                    | 3          | 3                         | 3                  | 18    |
| E   | 3                  | 3              | 4               | 3                    | 3          | 2                         | 3                  | 17    |
| F   | 2                  | 3              | 3               | 2                    | 3          | 3                         | 3                  | 17    |
| G   | 3                  | 2              | 2               | 3                    | 2          | 3                         | 2                  | 17    |
| H   | 1                  | 2              | 1               | 3                    | 1          | 3                         | 2                  | 16    |
5. Conclusion

Functional action training has a considerable influence on enhancing the particular action mode of basketball players, according to a meta-analysis of functional action training on improving specific action mode of basketball players, which lays a theoretical foundation for the subsequent experiment of functional action training in improving basketball action mode in this study. FMS score has the same effect as the evaluation of basketball-specific action mode, which can reflect the problems existing in the basketball action mode of screening objects to a certain extent. The establishment of tailored functional action training can successfully improve the low-score action mode, according to the screening results. Based on the results of the screening, the functional movement training was carried out with full reference to the characteristics of basketball and previous injuries. The majority of the training content is focused on core stability and force and quantity training in the knee and ankle areas.

Data Availability

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

Conflicts of Interest

The author declares that he has no conflicts of interest.

References

[1] B. P. Rimal, D. P. Van, and M. Maier, "Mobile edge computing empowered fiber-wireless access networks in the 5G era," *IEEE Communications Magazine*, vol. 55, no. 2, pp. 192–200, 2017.

[2] K. Kumar, J. Liu, Y.-H. Lu, and B. Bhargava, "A survey of computation offloading for mobile systems," *Mobile Networks and Applications*, vol. 18, no. 1, pp. 129–140, 2013.

[3] M. R. Rahimi, J. Ren, C. H. Liu, A. V. Vasilakos, and N. Venkatasubramanian, "Mobile cloud computing: a survey, state of art and future directions," *Mobile Networks and Applications*, vol. 19, no. 2, pp. 133–143, 2014.

[4] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Computing*, vol. 8, no. 4, pp. 14–23, 2009.

[5] A. Waqas, Z. M. Yusof, and A. Shah, "A security-based survey and classification of cloud architectures, state of art and future directions," in *Proceedings of the International Conference on Advanced Computer Science Applications Technologies*, pp. 284–289, IEEE Computer Society, Kuching, Malaysia, 2013.

[6] X. H. You, C. X. Wang, J. Huang et al., "Towards 6G wireless communication networks: vision, enabling technologies, and new paradigm shifts," *Science China Information Sciences*, vol. 64, no. 1, pp. 1–74, 2020.

[7] Cisco, *Cisco Visual Networking index: Global mobile Data Traffic Forecast Update, 2017-2022*, Cisco Public Information, 2017.

[8] S. He, W. Huang, J. Wang, J. Ren, Y. Huang, and Y. Zhang, "Cache-enabled coordinated mobile edge network: opportunities and challenges," *IEEE Wireless Communications*, vol. 27, no. 2, pp. 204–211, 2020.

[9] H. Guo, J. Liu, and J. Zhang, "Computation offloading for multi-access mobile edge computing in ultra-dense networks," *IEEE Communications Magazine*, vol. 56, no. 8, pp. 14–19, 2018.

[10] M. Kamel, W. Hamouda, and A. Youssef, "Ultra-dense networks: a survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 4, pp. 2522–2545, 2016.

[11] N. Abbas, Y. Zhang, A. Taberkordi, and T. Skeie, "Mobile edge computing: a survey," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 450–465, 2018.

[12] S. J. Chang, "Development and analysis of AIS applications as an efficient tool for vessel traffic service," in *Proceedings of the IEEE TECHNO-OCEAN’2004*, vol. 4, pp. 2249–2253, Kobe, Japan, November 2004.

[13] W. D. Hillis, "Co-evolving parasites improve simulated evolution as an optimization procedure," *Physica D: Nonlinear Phenomena*, vol. 42, no. 1-3, pp. 228–234, 1990.

[14] C. D. Rosin and R. K. Belew, "New methods for competitive coevolution," *Evolutionary Computation*, vol. 5, no. 1, pp. 1–29, 1997.

[15] K. O. Stanley and R. Miikkulainen, "Competitive coevolution through evolutionary complexification," *Journal of Artificial Intelligence Research*, vol. 21, pp. 63–100, 2004.

[16] K. C. Tan, T. H. Lee, Y. J. Yang, and D. S. Liu, "A cooperative coevolutionary algorithm for multiobjective optimization," in *Proceedings of the 2004 IEEE International Conference on Systems, Man & Cybernetics*, vol. 1-7, pp. 1926–1931, The Hague, Netherlands, October 2004.

[17] A. B. Amor, M. Abid, and A. Meddeh, "A privacy-preserving authentication scheme in an edge-fog environment," in *Proceedings of the Proceedings of IEEE/ACS International Conference on Computer Systems & Applications*, pp. 1225–1231, Hammamet-Tunisia, 2017.

[18] B. Danev and S. Capkun, "Transient-based identification of wireless sensor nodes," in *Proceedings of the International Conference on Information Processing in Sensor Networks*, pp. 25–36, San Francisco, 2009.

[19] T. Kohno, A. Broido, and K. C. Claffy, "Remote physical device fingerprinting," *IEEE Transactions on Dependable and Secure Computing*, vol. 2, no. 2, pp. 93–108, 2005.

[20] B. Danev, D. Zanetti, and S. Capkun, "On physical-layer identification of wireless devices," *ACM Computing Surveys*, vol. 45, no. 1, pp. 1–29, 2012.

[21] B. Danev, H. Luecken, and S. Capkun, "Attacks on physical-layer identification," in *Proceedings of the Proceedings of ACM Conference on Wireless Network Security*, pp. 89–98, Hoboken, New Jersey, USA, 2010.