Hindawi
Mobile Information Systems
Volume 2021, Article ID 4485589, 11 pages
https://doi.org/10.1155/2021/4485589

Research Article

Artificial Intelligence and Neural Network-Based Shooting Accuracy Prediction Analysis in Basketball

Hongfei Li[1] **and Maolin Zhang**[2,3]

[1]*Shanxi Agricultural University, Jinzhong 030801, China*
[2]*Hoseo University, Asan 31499, Republic of Korea*
[3]*Business College of Shanxi University, Taiyuan 030031, China*

Correspondence should be addressed to Maolin Zhang; zhangmaolin@bcsxu.edu.cn

Received 13 April 2021; Revised 30 April 2021; Accepted 12 May 2021; Published 3 June 2021

Academic Editor: Jianhui Lv

In order to improve the accuracy of shooting in basketball. A shooting accuracy prediction method based on the convergent improved resource allocating network (CIRAN) online radial basis function neural network (RBFNN) is proposed, and the RBFNN learning algorithm is improved. Through the collection of shooting motion images, feature point extraction, and edge contour feature extraction, the shooting motion trajectory is obtained. Using the online neural network based on the CIRAN learning algorithm to predict the accuracy of shooting, this method analyzes the radial basis function (RBF) network. Based on the RBF analysis, the number of network layers and the number of hidden layer neurons are adjusted and optimized. In order to improve the prediction accuracy of shooting in basketball, a method based on. Through the analysis, it can be known that the accuracy of both the traditional RBFNN and the CIRAN-based online neural network for the prediction of shooting accuracy is above 70%. The prediction accuracy of the online neural network for shooting is higher than that of the traditional one. This is mainly because the online update function of the learning algorithm can better adjust the corresponding structure with the development of the game and has a better generalization ability. In addition, because the CIRAN learning algorithm introduces the hidden layer neuron deletion strategy, its network structure is simpler than that of the traditional one, the number of hidden layer neurons is less, and the running time required is less, which can better meet the real-time requirements and provide a more scientific method for basketball training.

1. Introduction

With the development of computer image processing technology, embedded digital image and video information analysis methods are used to carry out image analysis and feature extraction of sports, establish a feature analysis model of sports images, and improve the ability of feature identification and movement correction of sports. In basketball, the accuracy of shooting determines the key to scoring. It is necessary to study the extraction of basketball players' shooting motion trajectory, combined with the image feature analysis method of basketball shooting, to reconstruct and quantitatively track the basketball players' shooting motion trajectory [1], establishing the image analysis model of basketball players' shooting motion trajectory and improving the calibration ability of basketball players' shooting motion. The research on the extraction method of basketball players' shooting motion trajectory has attracted great attention. Based on Harris corner detection, a method for extracting characteristics of lower-upper extremity action in basketball [2], this method first maps the spatial distribution of the pixel gray level in the upper extremity action area of the image, using Gaussian mixture model standards and normalized athletes and the contours of the lower-upper limbs of the strong smash, the Harris corner detection method is used to carry out the affine invariant closed area enhancement processing on the continuous motion images of the athletes, and the corner detection of the contours of the upper limbs of the athletes is performed to complete the lower-upper limbs of the

basketball smash. The action feature extraction method, however, has low accuracy in extracting the action features of the lower-upper limbs of basketball players. The volleyball player's motion trajectory optimization recognition method based on chaos theory [3] is based on the background difference principle to detect the player's motion trajectory, and the particles of the color histogram are used. Filtering for dynamic tracking, fusion with chaos theory to reconstruct the phase space of the athlete's motion trajectory, the chaotic invariant representing the athlete's motion trajectory is extracted from the reconstructed phase space, the motion trajectory with three-dimensional space characteristics is converted into a one-dimensional motion trajectory, and the optimized recognition of the volleyball player's trajectory is completed. However, this method has a low accuracy in predicting the trajectory of volleyball players. To solve the above problems, this paper extracts the basketball player's shooting motion trajectory based on the block growth optimization algorithm, extracts the edge contour feature of the collected basketball player's shooting motion trajectory image, establishes the image fusion model of the basketball player's shooting motion trajectory, and extracts the image feature of the shooting motion trajectory, and the corner points are marked to realize the extraction of the basketball player's shooting motion trajectory. Finally, the simulation experiment analysis is carried out to show the superior performance of this method in improving the ability of basketball player's shooting motion trajectory extraction.

At present, in the field of video-based analysis of moving human bodies, most of the research results are mainly on the discussion of motion behavior recognition, but there is no in-depth study on motion behavior prediction. However, in the process of real life, the prediction of human motion behavior based on videos has more practical value than behavior recognition [4]. For example, in many crowded public places, video-based human motion behavior prediction technology can be used to predict possible criminal behaviors in surveillance videos, and prompt criminal behavior prediction alerts the public security department to facilitate timely actions by the public security department corresponding solutions. In this way, it can not only reduce the manpower, material, and financial resources that the public security department spends on security investigations but also effectively prevent sudden crimes. In the field of sports, a comprehensive set of human motion behavior prediction technology in videos can accurately obtain the game data of some excellent teams during related sports training and tailor a sports behavior prediction discriminator for each athlete [5]. Through the discriminator, the coach can distinguish the difference between the movement made by the exercise and the standard movement, so as to adjust the training intensity in a targeted manner. In the course of the basketball game, there are three sports behaviors taken by players throughout the game, namely, shooting, passing, and dribbling. Basketball game is a team sport with multiple players; players must cooperate with each other in order to win the game. During the basketball game, players will be affected by various factors such as players or opponents when they choose related sports behaviors. Therefore, the process of basketball player behavior prediction has a high degree of complexity, which brings a certain degree of difficulty to the establishment of the behavior prediction mathematical model [6]. Combining this feature, we can also regard the prediction of basketball behavior as a nonlinear problem. In the prediction of solving nonlinear problems, artificial neural networks are widely used in the field of nonlinear system modeling due to their own self-adaptive and nonlinear characteristics. In this regard, this article combines the advantages of artificial neural networks, proposes a prediction method based on the CRIAN algorithm, and elaborates its implementation in detail.

The main contribution of this paper is to propose a CIRAN-based online RBFNN shooting prediction method; the accuracy of shooting prediction is above 90%. The rest of the paper is organized as follows. Section 2 summarizes domestic and foreign research work in the analysis of movements. Section 3 introduces shooting image collection and motion trajectory extraction optimization. Experimental results are reported in Section 4, and finally, Section 5 concludes this paper.

2. Related Work

At present, many universities and research institutions at home and abroad have carried out research on the introduction of digital video technology into auxiliary sports. Well-known research units abroad include the Media Analysis Laboratory of the Massachusetts Institute of Technology, the Digital Video Multimedia Laboratory of Columbia University, the School of Engineering and Applied Sciences, the University of Rochester, the Department of Computer Science at the University of Texas at San Antonio, the University of Delft in the Netherlands, Multimedia Analysis Laboratory and Microsoft Asia Research Institute, American Research Institute, Mitsubishi Electric American Research Institute, and Singapore Institute of Information and Communication Research. Domestic research institutions mainly include Advanced Human-Machine Communication Laboratory, Institute of Computing Technology, Chinese Academy of Sciences, State Key Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Institute of Digital Media, Peking University, and Visual Intelligence Interface Laboratory, Harbin Institute of Technology.

According to the complexity of the current research objects, the existing literature can be divided into two levels: the analysis of single-person individual sports and the analysis of multiperson team sports.

The current research on individual sports is mainly reflected in the recognition and analysis of athletes' movements. In 1996, Pennsylvania State University in the United States developed the "computer graphics for the improvement of springboard diving" system to help diving coaches and athletes to strengthen their understanding of the entire body posture during diving; some researchers in National Chiao Tung University analyzed the motion trajectory of a tennis player which is used to judge the volley or the baseline ball in a

tennis match [7]; Tsinghua University's "video-based diving posture analysis system" uses target detection and tracking technology to extract sports targets from diving videos, and to compare sports targets, video synthesis is performed; Roh et al. proposed an action recognition method based on the curvature scale space template [8–10] and applied it to the player's action recognition in a tennis match. The actions of ice hockey and football players are analyzed and studied using the histogram of oriented gradient (HOG) and hidden Markov model (HMM) to identify the direction of the players' movement [11]; Su et al. proposed a method of recognizing periodic motion [12]; Zhong et al. proposed an appearance-based method [13] to identify and label players' shoulder swings in tennis matches (overshoulder swing) and a series of actions such as foreside swing and backside swing. This method has been further expanded in the follow-up work. Combining the position information of the players and the ball, an action based on reasoning ideas is proposed. Ramasso et al. used the TBM (transferable belief model) [14] to identify back jump, pole vault, triple jump, and other actions in track and field competitions; Roh et al. recognized various postures [15]; Min et al. obtained the dance trajectory by tracking some key points of the body parts of the color bud dancer [16] and realized the automatic analysis of dance movements; Tong et al. realized the recognition of the four swimming styles in the competition video [17].

At present, there are many research studies on the contour tracking and extraction of basketball shooting motion video images, and the relative research has also produced certain results. Based on the Surendra background difference, a basketball shooting action video image contour tracking extraction method [18–20], this method first uses the Surendra background subtraction method to establish a basketball shooting action background model, giving players a complete shooting action and obtaining dynamic motion area; based on this, the contour tracking and extraction of the video image of the basketball shooting action is completed. This method is relatively simple, but there is a problem of large limitations of the method. The video image contour tracking and extraction method of basketball shooting action is based on visual analysis [21, 22]. This method first detects the edge contour of the shooting image and gives the dynamic feature segmentation threshold of the basketball shooting action, which is used as the basis to complete the basketball shooting action. The video image contour tracking extraction method has high marking efficiency, but when the current method is used for marking, the dynamic pixel information characteristics of basketball shooting actions cannot be given, and there is a problem of low contour tracking extraction accuracy. A basketball shooting action video image contour tracking extraction method based on figure contour feature extraction [23, 24] first extracts the dynamic figure edge contour feature points of the shooting action and uses the bright spot model diffraction method to achieve visual penetration. According to the technical characteristics of basketball shooting action, the contour tracking and extraction of basketball shooting action video image can be completed. This method has high marking accuracy, but there is a problem that the marking process is more cumbersome.

3. Shooting Image Collection and Motion Trajectory Extraction Optimization

3.1. Shooting Image Collection in Basketball. In order to achieve the extraction of the basketball player's shooting motion trajectory, the video sensor image tracking method is used to collect the image of the basketball player's shooting motion trajectory; the edge contour feature extraction of the collected basketball player's shooting motion trajectory image and the fuzzy decision method are used for state recognition and action trajectory planning. The grid segmentation method is used to divide the basketball sports video images collected by the video into feature blocks, α is the angle between the projection speed direction and the horizontal direction, that is, the shooting angle, v is the shooting point of the shot speed, g is the acceleration of gravity, the basketball is thrown at $t = 0$, and $f(x, y)$ is the trajectory function of the basketball which is defined as follows:

$$f(x, y) = x \tan \alpha - \frac{x^2 g}{2v^2 \cos \alpha}. \tag{1}$$

In a single scale, the pixel space of the basketball motion image feature collection is defined as follows:

$$a = \frac{1}{\nabla f(x, y)} \left(\frac{\partial f}{\partial y} i - \frac{\partial f}{\partial x} j \right), b = \frac{1}{\nabla f(x, y)} \left(\frac{\partial f}{\partial y} i + \frac{\partial f}{\partial x} j \right). \tag{2}$$

Using the adaptive weighting method for threshold modulation, the threshold of image grid segmentation is M, the initial value of the characteristic points of the basketball flight trajectory is calculated, and the edge pixel feature decomposition method is used for image template matching, and the number of template pixel blocks of the image is obtained as

$$B(x_n, y_n) = \prod_{x_i \in n} \prod_{m=1}^{M} \beta_m g\left(x_{ij}, y_{ij} \mid \mu_m, \sigma_m^2\right). \tag{3}$$

The position conversion set in basketball shooting is h_c, and the target configuration θ_g is unknown. The multiscale wavelet decomposition method [25] is used to segment the gray value in the image. The segmentation threshold meets $\nabla_x = [1, -1]$ and $\nabla_y = [1, -1]^T$ and produces the high-frequency part of the basketball shooting image. $y = [\nabla_x h', \nabla_y h']$ represents the low-frequency component of the image pixel value. The image feature is collected through normalization processing, and the image collection result obtained is

$$\min_{x,m} \lambda \| x \otimes m - y \|_2^2 + \frac{\|x\|_1}{\|x\|_2} + \delta \|m\|_1. \tag{4}$$

When the output pixel feature set of basketball shooting meets the constraint condition $k > 0$, $\sum_i m_i = 1$, the collected motion image can better reflect the shooting angle information and flight trajectory characteristics.

3.2. Feature Points' Extraction. On the basis of optimizing the collection of the image, the characteristic corner points of the basketball shooting trajectory are extracted from the image. In the state of motion, the state equation of the scattering model of the basketball shooting trajectory is defined as follows:

$$S(x) = J(x)t(x)\frac{v' \times \varepsilon_0}{\eta} + A(1 - t(x)). \tag{5}$$

Among them, $J(x)$ is the strength of the shot force in a fixed-point shot, x is the edge pixel sequence of the collected original image, v' is the motion function of the multicontour viewpoint switching of the shooting action, ε_0 is the standard deviation of the Gaussian function, and η is the direction of the filter.

In shooting, the Monte Carlo mathematical expectation of the edge corners of the image is extracted, and the pixel value extracted from the characteristic corners of the image in the basketball shooting motion is defined as follows:

$$\tau(Z; D_X) = \sum_{i>j} \left| d_{ij}(Z) - d_X(x_i, x_j) \right|^2. \tag{6}$$

Among them, $d_{ij}(Z)$ is the fitness judgment, which is a Euclidean distance; $d_X(x_i, x_j)$ represents the three-dimensional coordinate component value in sports shooting and fixed-point shooting.

When the basketball is flying in the air, the trajectory deviation will occur. Due to the influence of wind resistance and other factors, the attenuation coefficient of the flight trajectory is $e^{-\delta d(x)}$. Assuming that $A(1 - e^{-\delta d(x)})$ represents the edge pixel error of the shooting motion trajectory, the pixel value of the edge contour corner point is defined as follows:

$$p_t = C_t \sum_{x_i \in n} m\left(\|x_i\|^2 \right). \tag{7}$$

Among them, C_t is a normalized constant. The statistical characteristic analysis of basketball shooting angles and the calculation of characteristic corner points are performed through the spatial adaptive correction method. The number of image pixels is defined as follows:

$$G(x) = \frac{S(x) - A}{\max(t(x), t_0)}. \tag{8}$$

Among them, $G(x)$ is the degree of similarity of gray values, and $\max(t(x), t_0)$ is the maximum visual deviation of the basketball caused by changes in shooting mechanics and other factors.

The number of information features with respect to the basketball player's shooting motion image is $x.k$. The discrete pixel sequence reconstruction method is used to reconstruct the three-dimensional basketball player shooting motion image. The video information collection method is used to obtain the image sampling output result. The gray information fusion is performed on the basketball shooting action feature points to obtain the basketball shooting action feature, and the video tracking fusion formula for points is defined as follows:

$$\begin{aligned} x(k+1) &= G_i(k)x(k) + w_i(k), \quad i = 1, 2, \ldots, m, \\ u(k) &= R_i(k)x(k) + v_i(k), \quad i = 1, 2, \ldots, m. \end{aligned} \tag{9}$$

Among them, $w_i(k)$ and $v_i(k)$ are the state feature quantity and observation feature quantity extracted from the basketball shooting action feature. $G_i(k)$ and $R_i(k)$ obey the mean value of 0, and the variance is the normal distribution of x_k. According to the above analysis, the method of video sensor image tracking is used to collect basketball players' shooting motion trajectory images, and the shooting motion trajectory extraction and image information monitoring are performed according to the image collection results.

3.3. Motion Trajectory Extraction Optimization. Based on the aforementioned video sensor image tracking method for basketball player shooting motion trajectory image acquisition and edge contour feature extraction, the basketball player's shooting motion trajectory prediction and feature extraction are carried out, and the shooting motion trajectory extraction is optimized. This paper proposes an extraction method of basketball players' shooting motion trajectory based on block growth algorithm. The image fusion model of basketball player's shooting motion trajectory is established, the area linear growth method is combined to extract the image feature of basketball player's shooting motion trajectory, and the corner points are marked. The texture structure information of the basketball shooting action feature points is changed, the smooth area is distinguished, and the basketball shooting action is calculated. The amplitude modulation information of the characteristic defect image component, the component $x(t)$ of the pixel feature point of the n shooting action image of the image r_n, is defined as follows:

$$x(t) = p_t \sum_{i=1}^{n} e_i + r_n. \tag{10}$$

Among them, p_t is the pixel value of the edge contour corner point, and e_i represents the pixel value of the empirical mode decomposition of the basketball shooting action feature point. Using computer vision analysis, the basketball shooting action feature image is divided into N segmented regions, and N segmented regions perform multidimensional search iterations. Assuming that the two-dimensional feature segmentation function of the basketball player's shooting motion image satisfies $n \in N(0, \sigma_n^2)$, where σ_n^2 is the variance of the noise, combined with the regional linear growth method for the feature extraction of the basketball player's shooting motion trajectory image and the corner point labeling, the corner point distribution satisfies

$$w(i, j) = \frac{1}{x(t)} \exp\left(-\frac{d(i, j)}{h^2} \right) \tau(Z; D_X). \tag{11}$$

Among them, $\exp(-d(i, j)/h^2)$ is the least square feature quantity of the basketball player's shooting motion image, and h is the block fusion information entropy of the basketball player's shooting motion image. The fuzzy correlation fusion method is used to obtain the regional growth

function model $g(x, y) = \xi(x, y) + \psi(x, y)$ of the basketball player's shooting motion trajectory. Among them, $\xi(x, y)$ is the variance of the pixel distribution, and $\psi(x, y)$ represents the pixel intensity of the basketball player's shooting motion image. Through the regional linear growth analysis, when $\varphi_m(x, y) \in \{1, -1, 0\}$ is satisfied, the expression of the characteristic component of the regional linear growth of basketball shooting action is defined as follows:

$$p(\varphi_m(x, y)) = \begin{Bmatrix} \frac{r}{4}, & \varphi_m(x, y) = -1 \\ 1 - \frac{r}{2}, & \varphi_m(x, y) = 0 \\ \frac{4}{r}, & \varphi_m(x, y) = 1 \end{Bmatrix}. \quad (12)$$

Among them, $\varphi_m(x, y)$ is the regional linear growth characteristic of basketball shooting action, and r is the detection threshold of the basketball shooting action trajectory, $0 \le r \le 1$. The three-dimensional feature quantity of the basketball player's shooting motion image is extracted, and the three-dimensional visualization surface reconstruction method is used to reconstruct the motion track. According to the corner point distribution and edge contour distribution of the basketball player's shooting motion trajectory image, the characteristics of the basketball player's shooting motion trajectory are extracted, and the expression of the basketball shooting motion trajectory $p(x, t)$ is defined as follows:

$$p(x, t) = -\sigma \frac{\Delta u(x, t)}{p(\varphi_m(x, y))x(k)}. \quad (13)$$

Among them, $\Delta u(x, t)$ is the associated pixel point of the global threshold segmentation of the basketball player's shooting motion image, and σ is the feature quantity of the basketball player's shooting motion trajectory tracking. In summary, the optimization of the trajectory extraction of the basketball player's shooting is realized.

3.4. Radial Neural Network Prediction Model. The process of predicting the accuracy of basketball players' shooting has a relatively high complexity, which brings a certain degree of difficulty to the establishment of mathematical models of behavior prediction. Combining this feature, we can also regard the prediction of basketball behavior as a nonlinear problem. In the prediction of solving nonlinear problems, artificial neural networks are widely used in the field of nonlinear system modeling due to their own self-adaptive and nonlinear characteristics. In this regard, this paper combines the advantages of artificial neural networks and proposes a behavior prediction method based on an improved RBF algorithm.

RBFNN is an extremely efficient feedforward neural network [26]. Compared with other neural networks, RBFNN has two unique advantages of best approximation performance and global optimal characteristics and has the basic characteristics of other neural networks such as simple structure and fast training speed. It is a typical feedforward neural network composed of three layers. The function of the input layer node is to pass the signal to the hidden layer; the hidden layer node is composed of radial basis functions; the output layer node is usually a simple linear function. In gamma, the transformation from the input layer to the hidden layer is nonlinear. The role of the hidden layer is to perform a nonlinear transformation on the input vector, and the transformation from the hidden layer to the output layer is linear, that is, the output of the network is a hidden node, the linear weighted sum of the output. The topology of the specific single-output RBF network is shown in Figure 1.

The commonly used hidden layer radial basis functions have the following forms:

(i) Multiquadric function:

$$\phi(r) = \left(1 + (\varepsilon r)^2\right)^{1/2}, \quad \text{among them } \varepsilon > 0,\ r \in R. \quad (14)$$

(ii) Inverse multiquadric function:

$$\phi(r) = \frac{1}{\left(1 + (\varepsilon r)^2\right)^{1/2}}, \quad \text{among them } \varepsilon > 0,\ r \in R. \quad (15)$$

(iii) Gauss function:

$$\phi(r) = \exp\left(-\frac{r^2}{2\sigma^2}\right), \quad \text{among them } \sigma > 0,\ r \in R. \quad (16)$$

In formulas (14)–(16), ε is a constant, and r is the expansion constant of the RBF.

The hidden layer in the RBFNN in this paper uses the Gaussian function as the radial basis function, and the specific form is defined as follows:

$$\phi_k(x) = \exp\left(\frac{-\|x - \mu_k\|^2}{\sigma_k^2}\right), \quad k = 1, 2, \ldots, K. \quad (17)$$

Among them, x represents the l-dimensional input vector; μ_k is the center of the k -th radial basis function, a vector with the same dimension as x, and σ_k is the width of the radial basis function of the k -th hidden layer neuron. $\|x - \mu_k\|$ is the Euclidean norm of the vector $x - \mu_k$, which usually represents the radial distance between x and μ_k. $\phi_k(x)$ has a unique maximum value at μ_k, and as $\|x - \mu_k\|$ increases, it quickly decays to 0. For a given input $x \in R'$, only a small part of the input close to the center μ_k is activated, that is, the radial basis function is a function with local induction characteristics. Suppose the number of hidden layer neurons of the RBFNN is K, then the output form of the network can be defined as follows:

$$\hat{y} = f(x) = w_0 + \sum_{k=1}^{K} w_k \exp\left(-\|x - \mu_k\|^2 / \sigma_k^2\right). \quad (18)$$

Among them, w_0 represents the bias, which is the connection weight between the hidden layer and the output layer.

It can be seen from the structure of the neural network that the construction and training of one is to determine the

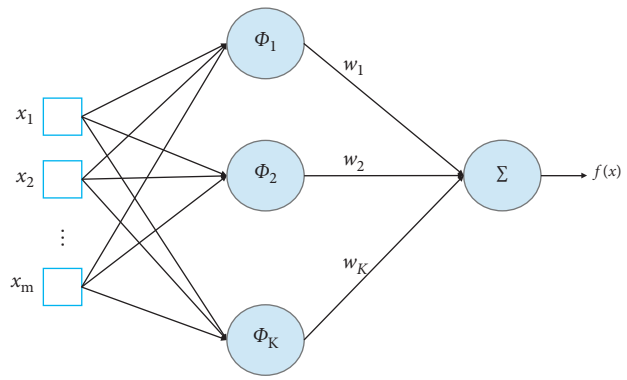

Figure 1: RBF network topology diagram.

number of hidden layer neurons K, the center μ of the radial basis function of each hidden layer neuron, the width σ, and the connection weight of the hidden layer to the output layer. Value w is the learning process of these parameters to complete the required mapping from the input to the output. For the RBFNN, the performance of its network mainly depends on the selection strategy of hidden layer neurons and the update process of the connection weights of the hidden layer to the output layer in the network.

The learning algorithm of the RBFNN can be divided into two categories according to the time: offline learning, also called batch learning, and online learning, also called sequential learning.

(1) Offline Learning. The offline training method trains the RBF network with a fixed structure. The center is extracted from the distribution pattern of the training sample space. After the center point is selected, the structure of the network is determined. When the data mode changes, the network cannot make corresponding changes. Although the network characteristics can be adjusted as much as possible by adjusting the weights, this method is not fundamental, and the adjustment range is limited. Especially when there is a sample that has not been learned, the network will lose its recognition ability. Therefore, the offline method is not strong in fitting time-varying systems.

(2) Online Learning. The offline learning mode is adopted, and the network parameters are adjusted only after complete learning is performed with all training samples, that is, an epoch. The learning process is one epoch after one epoch, until the network parameters stabilize and the average output error on the entire training set converges to a certain minimum, the learning can be ended. When using online learning algorithms, after each training sample enters the network and is calculated, the network parameters are adjusted. Specifically, suppose an epoch contains a training sample $((X(1), Y(1)), \ldots, (X(N), Y(N)))$. When the first sample $(X(1), Y(1))$ enters the network, the network output is calculated. Then, the learning algorithm is run to update the network parameters. This process is repeated until the learning of the first training sample is completed. The online RBFNN learning algorithm is actually a variable-structure training method. The so-called "variable structure" means that, in the training process, the current RBFNN is allocated or added hidden layer neurons according to the input samples in real time, and the network topology is dynamically constructed. At the same time, in the process of correcting the weight, the position of the "center" is further corrected. Adopting the concise "central selection" principle and effective weight training algorithm can make it have the characteristics of real time and rapidity.

At present, many scholars have conducted research on online learning algorithms. Typical algorithms are as follows:

3.4.1. RAN (Resource Allocating Network) Learning Algorithm. Resource allocating network (RAN) reflects the complexity of the original function to be simulated by adjusting the number of hidden layer units [27]. In Platt's RAN learning algorithm, the "novelty" of the training sample is used as the standard for introducing hidden layer neurons, and then the network parameter LMS (least mean square) is updated through the algorithm. When a data point is far away from the existing basis function center and there is a large error between the network output and the actual output, the data point is considered "novel." If the input sample does not meet the requirements of novelty, no hidden layer neurons are added, but the LMS algorithm is started to adjust the existing network parameters including center, width, and weight.

3.4.2. RANEKF (RAN Extended Kalman Filter) Learning Algorithm. The RANEKF algorithm is an improvement on the basis of the RAN learning algorithm. Its hidden layer neuron introduction strategy is the same as that of the RAN learning algorithm. The difference is that the adjustment of network parameters uses the extended Kalman filter (EKF) instead of the LMS method [28]. The extended Kalman filter method has a faster convergence rate than the LMS method but requires more computer resources. However, with the development of computer hardware technology, the EKF method has more advantages in the case of a small problem.

3.4.3. MRAN (Minimal RAN) Learning Algorithm. The MRAN algorithm not only combines the hidden layer node growth criterion of the RAN learning algorithm but also introduces a hidden layer neuron deletion strategy in order to obtain an ideal minimum neural network structure. Cheng proposed a method to delete hidden nodes in the batch learning algorithm in 1994 [29]. In this method, each epoch must check the weight of each hidden layer node, and those hidden layer nodes with a weight value less than a certain threshold will be deleted.

Inspired by Cheng's method, L. Ying Wei et al. proposed another hidden node deletion strategy for the MARN algorithm and called this RBFNN with the addition of the deletion strategy and the RMS sliding window MRAN (minimal resource allocating network). The first difference between MRAN's deletion strategy and Cheng's method is

that the MRAN is for sequential learning algorithms instead of batch learning algorithms. In addition, the MRAN deletion strategy not only considers the weight of the hidden node [30, 31] but also considers the output of the hidden node.

3.5. Shooting Prediction Based on the CIRAN Online RBFNN. The MRAN algorithm is developed on the basis of the RAN learning algorithm and the RANEKF learning algorithm. The learning process of the MRAN algorithm involves the introduction of new hidden layer neurons, the adjustment of network parameters, and the deletion of hidden layer neurons. The learning algorithm is defined as follows:

(i) For each input, calculate

$$\phi_k(x) = \exp\left(\frac{-\|x_i - \mu_k\|^2}{\sigma_k^2}\right), \quad k = 1, 2, \ldots, K,$$

$$f(x_i) = w_0 + \sum_{k=1}^{K} w_k \phi_k(x_i),$$

$$d_i = \min_{1 \le k \le K} \|x_i - \mu_k\|, \tag{19}$$

$$\varepsilon_i = \max\{\gamma^i \varepsilon_{\max}, \varepsilon_{\min}\},$$

$$e_{rms}^i = \sqrt{\sum_{j=i-B+1}^{i} \frac{\|e_j\|^2}{B}}.$$

Among them, d_i represents the Euclidean distance from the center of the hidden layer closest to x_i, $\varepsilon_{\max}$ represents the maximum distance between input data, $\varepsilon_{\min}$ represents the minimum distance between input data, and $0 < \gamma < 1$ is an attenuation coefficient. As the input data increase, ε_i decreases at an exponential rate until $\varepsilon_{\min}$. B is the width of the RMS sliding window (generally 40–50, empirical setting is required), and e_{rms}^i is the root mean square (RMS) of the input error after the window is added when the i sample enters the RBFNN.

(ii) If three conditions $|e_i| > \lambda$, $d_i > \varepsilon$, and $e_{rms}^i > \lambda_i$ (where λ is the desired approximation accuracy and λ_i is the threshold set in advance) are met at the same time, then a new hidden layer neuron is added to the network, and $K = K + 1$; then, the hidden layer neuron parameters are the following three formulas:

$$\begin{aligned} w_{K+1} &= e_i, \\ \mu_{K+1} &= x_i, \\ \sigma_{K+1} &= \kappa d_i. \end{aligned} \tag{20}$$

Among them, κ is the overlap factor, which determines the response width of hidden layer neurons. If the conditions are not satisfied, then the EKF is used to update the network parameters.

(iii) Update the network parameters according to the following formula:

$$\begin{aligned} u_i &= u_{i-1} + K_i e_i \\ K_{i_{(z \times \text{node}_y)}} &= P_{i-1} A_i \left[R_i + A_i^T P_{i-1} A_i\right]^{-1}, \\ P_i &= \left[I_{z \times z} - K_i A_i^T\right] P_{i-1} + I_{z \times z}. \end{aligned} \tag{21}$$

Among them, $u_i = [w_0^i, w_1^i, \mu_1^i, \sigma_1^i, \ldots, w_k^i, \mu_K^i, \sigma_K^i]^T$ represents the parameter state after the i sample enters the network, and $K_{i_{(z \times node_y)}}$ is the Kalman gain matrix:

$$K_{i_{(z \times \text{node}_y)}} = P_{i-1} A_i \left[R_i + A_i^T P_{i-1} A_i\right]^{-1}. \tag{22}$$

Among them, $z = \text{node}_y + K_i \times (j + \text{node}_y + 1)$ is the number of network parameters, node_y is the number of output nodes, and node_y is the variance matrix of the measurement noise. $A_i = \nabla_v f(x_i)$ is the gradient matrix of $f(x)$ with respect to the parameter vector v_i, and P_i is the error covariance matrix everywhere. Use the following formula to update:

$$P_i = \left[I_{z \times z} - K_i A_i^T\right] P_{i-1} + ss_0 I_{z \times z}. \tag{23}$$

Among them, $I_{z \times z}$ is the identity matrix, and ss_0 is a scalar, representing a random step length, used to determine the size of a random walk in the gradient direction. When a new hidden layer node is introduced into the network, the dimension of P_i is increased, and new rows and columns need to be added to P_{i-1}:

$$P_i = \begin{pmatrix} P_{i-1} & 0 \\ 0 & p_0 I_{z1 \times z1} \end{pmatrix}. \tag{24}$$

Among them, p_0 is a parameter value initially estimated, here is the covariance of sample data x_i and y_i, and z_1 is the number of new parameters added due to the introduction of new hidden layer nodes, $z_1 = l + \text{node}_y + 1$.

(iv) Calculate the output vectors $(\rho_{k1}^i, \ldots, \rho_{kj}^i, \ldots, \rho_{k\text{node}_y}^i)$ and $\|\rho_{j,\max}^i\|$ of all hidden nodes, which represent the maximum absolute value of all hidden nodes to the j output unit when sample i is input. Calculate the normalized output vector of each hidden node:

$$r_{kj}^i = \frac{\left\|\rho_{k,j}^i\right\|}{\left\|\rho_{j,\max}^i\right\|}, \quad k = 1, \ldots, K. \tag{25}$$

During the continuous input of N_w samples, if $r_{kj}^i < \lambda_2$ is established, then the k hidden node can be deleted, and the dimension of P_i can be reduced accordingly to facilitate the adjustment of the EKF parameters in the next step.

The main problems of the aforementioned MRAN learning algorithm are as follows: firstly, due to the use of the extended Kalman filter to adjust the network parameters, the parameters must be updated in each iteration, which leads to the process of updating the parameters with the hidden layer neurons. The scale of the matrix is very large, which increases the computational complexity of the RBFNN structure, causes the algorithm to calculate too much burden, consumes a lot of computer resources, and limits the real-time application of the MRAN algorithm; then, initializing the algorithm, there are too many parameters, and improper selection of initialization parameters will greatly reduce the performance of the algorithm. Sometimes, an exhaustive method has to be used for multiple trials. This will consume a lot of time and cause the algorithm promotion performance to drop significantly.

In this regard, this paper proposes a CIRAN (convergent improved RAN) improved algorithm, which is mainly reflected in the following:

(i) In order to reduce the initialization parameters of the algorithm, the idea of the GAP-RBP algorithm is absorbed, only the parameters of the hidden layer neuron closest to the current input data are updated, and the definition and estimation formula for measuring the importance of the hidden layer neuron are introduced, which reduces the algorithm number of initialization parameters and improves the generalization performance of the algorithm to a certain extent. The importance of hidden layer neurons is defined as follows:

$$E_{\text{imp}}(k) = \|w_k\|_q \left(\int_X \exp\left(-\frac{q\|x-\mu_k\|^2}{\sigma_k^2} \right) p(x)\,\mathrm{d}x \right)^{1/q}. \tag{26}$$

Among them, $\mu_k = (\mu_{k,1}, \ldots, \mu_{k,j})^T \in R^l$ and $\sigma_k (k = 1, \ldots, K)$ are the center and width of the radial basis function, and l is the dimension of the input vector.If the importance of a hidden layer neuron k is less than the realization of the given learning accuracy λ, then the hidden layer neuron k is considered no longer important and is deleted from the network; otherwise, the hidden layer neuron k should be retained in the network.

(ii) The dynamic adjustment method of the coincidence degree threshold is introduced into the algorithm so that the CIRAN need not set the values of the parameters $\varepsilon_{\max}$, $\varepsilon_{\min}$, and γ in the MRAN. Parameter γ can be dynamically obtained during the execution of the algorithm. Its updated formula is

$$\varepsilon_i = \max\left(0,\ (1-\tau)\left|\frac{f(x_i)}{e_i}\right| \right). \tag{27}$$

Among them, τ is the expected accuracy of the single-point output, $f(x_i)$ is the current actual output, and e_i is the current output error.

(iii) A new adaptive adjustment method for the width of the radial basis function of hidden layer neurons is

$$\left\{ \begin{array}{l} K = \dfrac{e_i}{e_i\sqrt{-\ln\lambda/e_i}} \\ \\ \sigma_{K+1} = \dfrac{e_i d_i}{e_i\sqrt{-\ln\varepsilon/e_i}} \end{array} \right\}. \tag{28}$$

So, the CIRAN learning algorithm can be defined as follows.

Input: given estimated error λ and single-point output expected accuracy τ. For the input sample (x_i, y_i), $x_i \in R^l$,

(i) Calculate the network output:

$$f(x_i) = w_0 + \sum_{k=1}^{K} w_k \exp\left(-\|x_i - \mu_k\|^2/\sigma_k^2 \right). \tag{29}$$

Among them, K is the number of hidden layer nodes.

(ii) Calculate the following quantities in the novelty criterion:

$$\begin{aligned} \varepsilon_i &= \max\left(0,\ 1-(1-\tau)\left|\frac{f(x_i)}{e_i}\right| \right), \\ e_i &= y_i - f(x_i), \\ d_i &= \|x_i - \mu_{ir}\|. \end{aligned} \tag{30}$$

Among them, μ_{ir} is the center of the hidden layer neuron closest to x_i in the sense of the Euclidean norm.

(iii) Apply the novelty criterion to judge whether to add hidden layer neurons.If $d_i > \varepsilon_i$ and $\|e_i\|\left(\int_X \exp\left(-(q\|x-x_i\|^2)/(\kappa^2\|x_i-\mu_{ir}\|^2)\right)p(x)\,\mathrm{d}x\right)^{1/q} > \lambda$, then a new hidden layer neuron $K+1$ is added to the network, and the corresponding parameters are set as

$$\begin{aligned} w_{K+1} &= e_i, \\ \mu_{K+1} &= x_i, \\ \sigma_{K+1} &= \kappa d_i. \end{aligned} \tag{31}$$

Otherwise, use the EKF method to update the node parameters of the hidden layer neuron closest to the current input in the network: w_{ir}, μ_{ir}, and σ_{ir}, and check whether the hidden layer neuron meets the output conditions; if $E_{imp}(ir) = \|w_{ir}\|\left(\int_X \exp\left(-\|x-\mu_{ir}\|^2/\sigma_{ir}^2\right)p(x)\,\mathrm{d}x\right)^{1/q} < \lambda$, delete the ir hidden layer neuron, and correspondingly, reduce the dimensionality of the EKF. When using too many hidden layer neuron nodes, the model will tend to overtrain the data, the generalization ability will be poor, and the classification effect will become less and less obvious. Therefore, the CIRAN learning algorithm introduces a hidden layer

Table 1: Shooting accuracy analysis.

Group no.	0–100	101–200	201–300	301–400	401–500	501–600	601–700	701–800	801–900	901–1000
Correct analysis	99	98	95	96	100	99	97	92	100	99
Unrecognized	1	2	5	4	0	1	3	8	0	1
Prediction success rate (%)	99	98	95	96	100	99	97	92	100	99

neuron deletion strategy. Its network structure is simpler than that of the traditional one. The number of hidden layer neurons is less, and the running time required is less, which can better meet the real-time requirements.

3.6. Shooting Accuracy Prediction. Using the CIRAN online RBFNN-based shooting prediction proposed in this paper, from equation (1), the basketball should be hollow when entering the net or the basketball should be within a certain speed when it hits the inner basket, except for the conditions already obtained, when the ball is in the air, the ball cannot touch the basket. The basketball and the basket have custom sizes. Set the diameter of the basketball to d and the diameter of the basket to D, ignoring the air resistance. L is the shooting distance, h is the shooting height, and θ is the shooting angle; then, the shooting speed is defined as follows:

$$v = \sqrt{\frac{L^2 g^2}{2\cos^2\theta(L\tan\theta - h)}}. \tag{32}$$

Combining the distributed trajectory of ball shooting (13), if the following formula holds, then the shooting accuracy predicts success.

$$\lim_{x \longrightarrow v} f(x_i) \mp p(x,t)\omega_j \sim E_{\text{imp}}(ir). \tag{33}$$

4. Experiments

In order to verify the effectiveness of the shooting prediction based on the CIRAN online RBFNN proposed in this paper, the shooting prediction and running time are performed on the sample number of basketball players' shooting motion images.

4.1. Model Evaluation. In the basketball player shooting motion assistance training system, choose 1000 groups of images as the test set and random variables $w_0 = 0.2$, $w_k = 5$, $\mu = 3$, and $\text{TT} = \sigma$, divided into 10 groups; the shooting prediction success rate is shown in Table 1.

As seen from Table 1, the number of unrecognized experiments in each group is relatively small, and the accuracy of the correctly analyzed shots is relatively large, reaching more than 95%. The effectiveness of the shot prediction based on the CIRAN online RBFNN can be seen.

4.2. Prediction Ratio. In order to determine the weight of equation (33), let ratio $= \omega_1/\omega_2$, and determine the weights ω_1 and ω_2 by comparing the relationship between ratio and

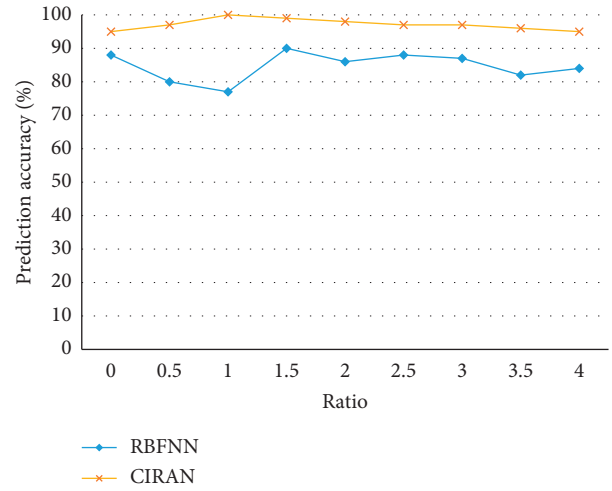

Figure 2: Prediction accuracy of the RBFNN and CIRAN.

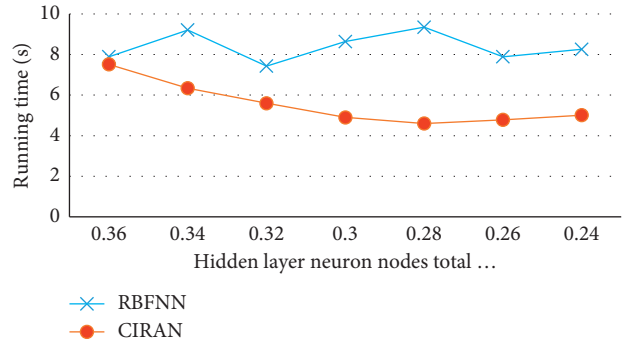

Figure 3: Running time of the RBFNN and CIRAN.

prediction accuracy under different ratios. As shown in Figure 2, when the ratio of the two is 0.98, CIRAN-based online RBFNN has a much better prediction accuracy than the traditional offline RBFNN. The shooting prediction accuracy of the method proposed in this paper is above 90%, while for the RBFNN method, it is only above 70%. The shooting accuracy prediction of the basketball player in this method is more accurate than that of the traditional method.

Through analysis, the accuracy of both the traditional RBFNN and the CIRAN-based online neural network for the prediction of shooting accuracy is above 70%. The prediction accuracy of the online neural network for shooting is higher than that of the traditional one. This is mainly because the online update function of the learning algorithm can better adjust the corresponding structure with the development of the game and has a better generalization ability.

4.3. Running Time Analysis. The CIRAN learning algorithm introduces a hidden layer neuron deletion strategy, which requires less running time and can better satisfy shooting accuracy prediction analysis. It can be seen from Figure 3 that when the ratio of hidden layer neuron nodes to the total number of nodes is 0.28, the running time is the shortest. In general, the CIRAN takes less time to predict the shooting accuracy than the RBFNN.

5. Conclusions

In today's highly information-based society, sports training is also quietly undergoing changes. In the arena of various sports, it is no longer purely a competition of athletes' sports skills and physical and psychological qualities but a comprehensive competition of the development of science and technology and the cohesion of people between countries. Introducing artificial intelligence and neural network into basketball shooting prediction can greatly improve the success rate of prediction. This paper proposes a shooting prediction based on the CIRAN online RBFNN and improves the RBFNN learning algorithm. The results show that the method proposed in this paper has a high accuracy rate in predicting the accuracy of basketball players' shooting and less running time. In the future, we will further analyze the behavior prediction of team members through the prediction of shooting accuracy to improve team cooperation; through the prediction of shooting, the trajectory of basketball players' shooting is analyzed, so as to improve the accuracy of basketball players' shooting.

Data Availability

The 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.

References

[1] Q. Huang, W. Gao, H. Yao et al., "Event tactic analysis based on broadcast sports video," *IEEE Transactions on Multimedia*, vol. 11, no. 1, pp. 49–67, 2009.

[2] B. A. Jasani, M. Wu, S.-K. Lam, and P. K. Meher, "Threshold-guided design and optimization for Harris corner detector architecture," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 12, pp. 3516–3526, 2018.

[3] S.-W. Wang, H.-W. Chen, Volleyball players arm trajectory simulation research on optimization of recognition," *Computer Simulation*, vol. 34, no. 2, pp. 270–273, 2017.

[4] B. Yeong-Hyeon, D. Kim, J. Lee, and K. C. Kwak, "Body and hand–object ROI-based behavior recognition using deep learning," *Sensors*, vol. 21, no. 5, pp. 1838–1860, 2021.

[5] S. Alexandros and P. Ronald, "Learn to cycle: time-consistent feature discovery for action recognition," *Pattern Recognition Letters*, vol. 141, pp. 1–7, 2021.

[6] Xiaohong Dong, "Shooting motion mathematical model of the optimal solution," *Technology Wind*, no. 28, pp. 43-44, 2018.

[7] J. Landlinger, T. Stöggl, S. Lindinger, H. Wagner, and E. Müller, "Differences in ball speed and accuracy of tennis groundstrokes between elite and high-performance players," *European Journal of Sport Science*, vol. 12, no. 4, pp. 301–308, 2012.

[8] H. Zhang, Z. Liu, H. Zhao, and G. Cheng, "Recognizing human activities by key frame in video sequences," *Journal of Software*, vol. 5, no. 8, pp. 818–825, 2010.

[9] W. Wang and D. Zhou, "A multi-level approach to highly efficient recognition of Chinese spam short messages," *Frontiers of Computer Science*, vol. 12, no. 1, pp. 135–145, 2018.

[10] H. Wu, J. Weng, X. Chen, and W. Lu, "Feedback weight convolutional neural network for gait recognition," *Journal of Visual Communication and Image Representation*, vol. 55, no. 8, pp. 424–432, 2018.

[11] M. R. Malgireddy, J. J. Corso, S. Setlur, V. Govindaraju, and D. Mandalapu, "Framework for hand gesture recognition and spotting using sub-gesture modeling," in *Proceedings of the ICPR 2010, 20th International Conference*, pp. 3780–3783, Istanbul, Turkey, August 2010.

[12] B.-Y. Su, J. Jiang, Q.-F. Tang, and M. Sheng, "Human dynamic action recognition based on functional data analysis," *Acta Automatica Sinica*, vol. 43, no. 5, pp. 866–876, 2017.

[13] J. Zhong, H. W. Liu, and C. L. Lin, "Human action recognition based on hybrid features," *Applied Mechanics and Materials*, vol. 2594, no. 748, pp. 1188–1191, 2013.

[14] E. Ramasso, C. Panagiotakis, D. Pellerin, and M. Rombaut, "Human action recognition in videos based on the Transferable Belief Model," *Pattern Analysis and Applications*, vol. 11, no. 1, pp. 1–19, 2008.

[15] J. Kittler, B. Christmas, S. W. Lee, and M. C. Roh, "Gesture spotting for low-resolution sports video annotation, Pattern Recognition," *The Journal of the Pattern Recognition Society*, vol. 41, no. 3, pp. 1124–1137, 2008.

[16] O. Camps, J. Min, and R. Kasturi, "Extraction and temporal segmentation of multiple motion trajectories in human motion," *Image and Vision Computing*, vol. 26, no. 12, pp. 1621–1635, 2008.

[17] X. F. Tong, L. Y. Duan, and C. S. Xu, "Local motion analysis and its application in video based swimming style recognition," in *Proceedings of the 18th International Conference on Pattern Recognition*, pp. 32–50, Hong Kong, China, August 2006.

[18] X. Song, "Research on basketball shooting trajectory automatic capture method based on background difference method," *Automation and Instrumentation*, no. 7, pp. 42–45, 2020.

[19] G. Liu, "3D vision based correction method for basketball shooting angle," *Modern Electronics Technique*, vol. 40, no. 5, pp. 45–48, 2017.

[20] Y. Zhang, "Research on visual analysis based standardization judgment method for basketball shooting action," *Modern Electronics Technique*, vol. 40, no. 3, pp. 47–50, 2017.

[21] D. Ma, Y. Chen, H. Chen, and L. Yun, "Augmented reality based basketball shoot trajectory analysis and simulation," *Computer Applications and Software*, vol. 31, no. 5, pp. 53–56, 2014.

[22] H.-W. Xu, W.-G. Wan, B. Cui, J.-C. Lin, and K.-Y. Zhang, "Simulation of basketball shooting in virtual reality," *Journal of Applied Sciences*, vol. 27, no. 4, pp. 414–418, 2009.

[23] D. Zhang and Y. Chen, "Realization of basketball training simulation system," *Journal of Graphics*, vol. 36, no. 5, pp. 789–794, 2015.

[24] Y.-H. Chai, "Application of digital image processing technology to analysis of factors influencing the basketball shot rate," *Journal of Hefei University of Technology(Natural Science)*, vol. 30, no. 12, pp. 1607–1609, 2007.

[25] B. Liu and W. Liu, "The lifting factorization of 2D 4-channel nonseparable wavelet transforms," *Information Sciences*, vol. 456, pp. 113–130, 2018.

[26] X.-J. Wei, N.-Z. Li, X.-Z. Zhou, J. Ding, and W.-C. Ding, "Optimization of RBFNN based on dynamic multiple subpopulation collaboration quantum-behaved particle swarm optimization algorithm," *Journal of Lanzhou Jiaotong University*, vol. 33, no. 3, pp. 98–103, 2014.

[27] Z.-J. Liu, Z.-C. Zhu, H. Deng, and X. Liu, "Power allocation algorithm with respect to system capacity maximization based on RAN architecture of wireless access network," *Application Research of Computers*, vol. 30, no. 5, pp. 1486–1488, 2013.

[28] Y. Hou and Y. Wang, "Research on attitude solution of four-rotor based on improved EKF," *Application of Electronic Technique*, vol. 43, no. 10, pp. 83–85, 2017.

[29] Y. H. Cheng and C. S. Lin, "A learning algorithm for radial basis function networks: with capability of adding and pruning neurons," in *Proceedings of the 1994 IEEE Internationl Conference on Neural Networks*, pp. 797–801, Orlando, FL, USA, June 1994.

[30] B. Li and X.-P. Lai, "An improved GGAP-RBF algorithm and its application to function approximation," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 20, no. 2, pp. 230–235, 2007.

[31] Q. Bao and W. Song, "GAP-RBF neural network learning algorithm based ON population partitioning optimisation," *Computer Applications and Software*, vol. 33, no. 11, pp. 215–220, 2016.