Retraction

Retracted: Deep Learning-Based Posture Recognition for Motion-Assisted Evaluation

Mobile Information Systems

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope

(2) Discrepancies in the description of the research reported

(3) Discrepancies between the availability of data and the research described

(4) Inappropriate citations

(5) Incoherent, meaningless and/or irrelevant content included in the article

(6) Peer-review manipulation

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] Y. Yuan and D. Zheng, "Deep Learning-Based Posture Recognition for Motion-Assisted Evaluation," *Mobile Information Systems*, vol. 2022, Article ID 7581079, 9 pages, 2022.

Research Article

Deep Learning-Based Posture Recognition for Motion-Assisted Evaluation

Yi Yuan [ID] and DongXia Zheng

Hunan University of Science and Technology, Xiangtan 411201, China

Correspondence should be addressed to Yi Yuan; 1170061@hnust.edu.cn

With the development of computer vision technology, human action pose recognition has gradually become a popular research direction, but there are still some problems in the application research based on pose recognition in sports action assisted evaluation. In this paper, the human motion pose recognition technology based on deep learning is introduced into this field to realize the intelligence of sports-assisted training. Firstly, we analyze the advantages and limitations of the state-of-the-art human motion pose recognition algorithms in computer vision in specific fields. On this basis, a human motion space recognition method based on periscope neural network is proposed. Firstly, the classical radar signal processing method is used to preprocess the echo signal of human spatial position and generate the frequency image in the process of human spatial position. Then, the periscope neural network (CNN) is constructed, and the time-frequency image is used as the input data of CNN to train the network parameters. Finally, the method is tested by using the open dataset in the network. The experimental results show that the designed CNN can accurately identify four different types of physical motion, and the accuracy coefficient is at least 97%.

1. Introduction

The recognition of human motion pose has become a concern and is widely used in computer vision [1]. Video-based pose recognition usually means inputting video data and extracting and analyzing video features through various image processing and recognition methods [2]. In order to achieve the purpose of human action recognition in video, it has a wide range of applications [3]. The key of video-based posture state recognition is to extract appropriate video features and analyze and recognize these features reasonably and accurately [4]. Physical motion recognition technology is widely used in various fields. Applying natural person and natural person recognition technology to the purpose of sports recognition can accurately identify sports, compare them with existing sports, and identify and correct irregular sports [5].

Artificial intelligence technology began to emerge in the 1990s, and machine vision technology after 20 years of development has been widely used in video surveillance, virtual imaging, film and television production, and other industries [6]. In particular, the technology of character modeling to generate two-dimensional animation is one of the current hot spots of scientific and technological research [7, 8]. With the application of machine learning in the field of image processing gradually mature, the combination of deep learning and computer 2D animation imaging technology has become possible, [9] proposed the use of Toronto University's general model - wireframe model modeling, modeling method is more efficient, simple image extraction, but the data noise is too large, affecting the accuracy of the action is affected by the excessive data noise [10]. Other domestic research units, still in the academic exploration and research stage, the proposed algorithm in the application of the hardware often has high requirements for computing power [11]. From the domestic and international research, the key to the recognition of character pose and 2D animation generation lies in the pose extraction of each action of the character itself, image compression and subsequent refinement of the convolutional neural networks in

the processing of medical images show a strong advantage [12. In this paper, we focus on the effective combination of deep neural networks and pose recognition and propose an improved convolutional neural network architecture to achieve real-time character pose output in complex scenes of multiplayer motion.

2. Related Work

The detection and recognition technology of human motion gesture can be applied not only in smart home but also in military field, which will greatly promote the development of intelligent weapons, so it has important application prospects. At present, the main method of subject space recognition is the recognition of visible light based on vision and microwave. Radar microwave-based human motion gesture recognition is not affected by light, can protect user privacy, and can penetrate certain obstacles for recognition [13]. Therefore, radar microwave-based human motion gesture recognition technology has an irreplaceable position in the fields of smart home, remote control, and intelligent weapons.

The key to radar microwave-based human motion pose recognition is to extract and identify the micro-Doppler features of the echoes. In the literature, micro-Doppler features are extracted from human action postures for recognition and classification by traditional algorithms such as support vector machine (SVM), orthogonal matching tracking (OMP), and dynamic time regularization (DTW). Although the above traditional algorithms can achieve high accuracy, they are limited to traditional supervised learning, which requires human extraction of features from micro-Doppler information, and the extracted features are difficult to migrate for application due to the limitation of the recognition object, while deep learning algorithms can overcome this limitation. In the literature [14, 15], deep learning algorithms such as CNNs and dual-stream fusion neural networks (TS-FNNs) were used to extract and recognize features from R-D (range-Doppler) maps of gestures generated by FM continuous wave radar, and the accuracy rate was significantly improved compared with traditional algorithms. This shows that the deep learning algorithm can bring a great improvement to the accuracy of radar gesture recognition. However, deep learning algorithms require a large amount of data and are prone to overfitting and error transfer for small datasets, resulting in poor recognition results [16].

This paper proposes a CNN-based microwave recognition method for human action posture. CNN can automatically extract the depth features of action echoes without human extraction, and the model has strong generalization ability [17]. Compared with the traditional BP (back-propagation) neural network, CNN uses convolutional kernels for local connectivity and weight sharing, which reduces the number of parameters and improves the learning efficiency of the network and can better solve the overfitting and error transmission problems caused by small datasets [18]. In this paper, LFMCW radar is used to acquire the human action posture echo signals, generate the time-

frequency maps of human action postures, and recognize the radar echo images of four types of human action postures: walking, sitting, standing, and falling, by CNN [19, 20]. The final recognition accuracy for walking, sitting, standing, and falling movements reaches over 97%.

3. Methodology

The algorithm is based on a bottom-up human pose recognition algorithm, which is the first to identify the key points of human movement in a complex environment with multiple people and then form a skeletal map of human movement after a reasonable linkage of key points. When using convolutional neural network to process the basic image, only one convolution is needed to complete the analysis. Firstly, according to the coordinates of human joints, joint levels, and types, the feature map of human directed links is established, which facilitates the digital processing of images and then completes the convolution operation, as shown in Figure 1.

In the human feature map represented in Figure 1, the coordinates, levels, and types of key joint points are identified in the form of feature vectors, and for the feature point (x, y), the corresponding feature vector is

$$\{P_c, T_{\text{to}}\}, \tag{1}$$

where P_c represents the probability value of the type to which the feature point (x, y) and its corresponding joints belong and T_{to} represents the offset value of the coordinates of the parent node of the feature point from the coordinates of the feature point itself, which is the value of the feature vector.

3.1. Acquisition of Differential Beat Signal. Figure 2 shows the time-frequency relationship between the LFMCW radar transmit signal, the echo signal, and the differential beat signal.

In Figure 2, f_0 is the starting frequency of the signal, τ_{max} is the maximum time delay, T_r is the period of the signal, B is the bandwidth of the signal, the effective time of the signal is T, i.e., $T = T_r - \tau_{\text{max}}$, and the effective bandwidth of the signal is usually smaller than B.

Considering the multiperiod LFMCW radar echo signal, to simplify the analysis, ignoring the initial phase, the sawtooth LFMCW radar signal in the. The complex form of the emitted signal in the $(k + 1)$ sweep period is

$$s_{t,k+1}(t) = A_0 \exp\left\{ j\left[2\pi\left(f_0(t - kT_r) + \left(\frac{\mu(t - kT_r)^2}{2} \right) \right) \right] \right\},$$

$$t \in \left[\left(k - \frac{1}{2} \right)T_r, \left(k + \frac{1}{2} \right)T_r \right], \tag{2}$$

where A_0 is the random amplitude of the transmit signal at $t = kT_r$, f_0 is the instantaneous frequency of the transmit signal at $t = kT_r$, and $\mu = B_r/T_r$ is the FM slope (B_r is the FM bandwidth and T_r is the sweep period). At time $t = 0$, assuming that a point target has an initial distance of R_0 with respect to the radar and approaches the radar with radial
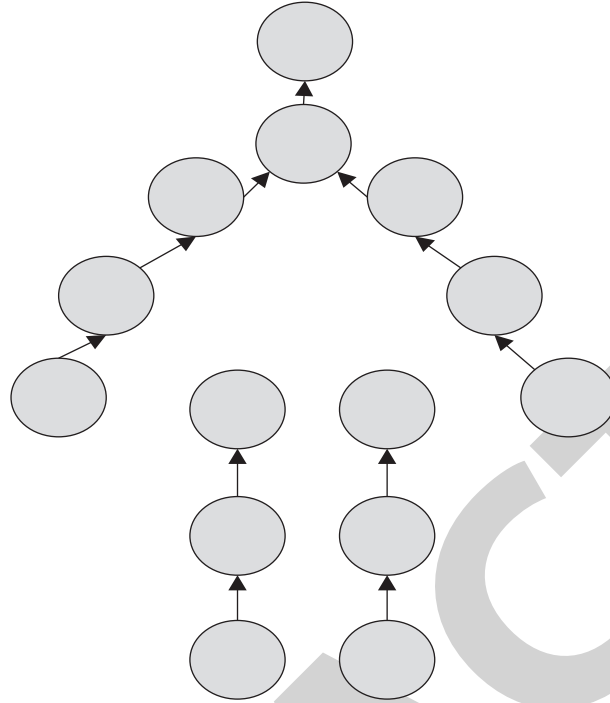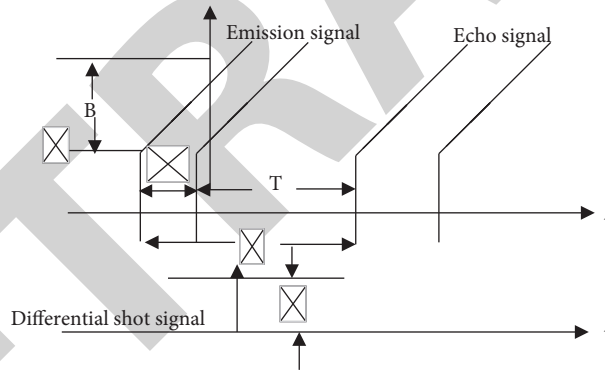
Figure 1: Diagram of human directed link.

Figure 2: Time-frequency relationship of different signals.

velocity v (with the velocity away from the radar as positive and the velocity close to the radar as negative), the echo signal $s_{r,k+1}(t)$ of the moving target in the effective time period of the $(k + 1)$ sweep period is expressed as

$$
s_{r,k+1}(t) = \beta_r A_0 \exp\left\{ j2\pi\left[f_0\left(t - kT_r - \tau_k(t)\right) + \left(\frac{\mu\left(t - kT_r - \tau_k(t)\right)^2}{2}\right)\right]\right\},
$$

$$
t \in \left[\left(k - \frac{1}{2}\right)T_r + \tau_{\max}, \left(k + \frac{1}{2}\right)T_r\right],
$$

(3)

where β_r is the attenuation constant, which reflects the influence of the environment on the electromagnetic wave and the ability of the target to scatter the electromagnetic wave; $\tau_k(t)$ is the instantaneous delay of the target echo in the $(k + 1)$ period; and $\tau_k(t) = 2(R_0 - vt)/c$,

in which c is the speed of light. By mixing the transmitting signal $s_{t,k+1}(t)$ and the target echo signal $s_{r,k+1}(t)$ in the effective band in the $(k + 1)$ period, the resulting differential beat signal $s_{b,k+1}(t)$ can be expressed as

$$s_{b,k+1}(t) = \left(\frac{\beta_r A_0^2}{2}\right)\exp\left\{j\left[2\pi\left(\left(\frac{-\mu\tau_k^2(t)}{2}\right) + f_0\tau_k(t) + \mu\tau_k(t)(t - kT_r)\right)\right]\right\},$$

$$t \in \left[\left(\left(k - \frac{1}{2}\right)T_r + \tau_{max}, \left(k + \frac{1}{2}\right)T_r\right].\right.$$

(4)

Let $t' = t - kT_r$; then, substitute t for t', bring $\tau_k(t + kT_r) = 2(R_0 - vt - vkT_r)/c$ into (4), and ignore c^2, t^2, and we get

$$s_{b,k+1}(t) \approx \left(\frac{\beta_r A_0^2}{2}\right)\exp\left\{j\left[2\pi\left(\left(\frac{(2\mu R_0 - 2v f_0 - 2\mu vkT_r)t}{c}\right) + \left(\frac{2f_0 R_0 - 2v f_0 kT_r}{v}\right)\right)\right]\right\},$$

$$t \in \left[-\frac{T_r}{2} + \tau_{max}, \frac{T_r}{2}\right].$$

(5)

Fourier transform (FT) of (5) on the $(k+1)$ interval $[-T/2, T/2]$ yields the spectrum of the differential beat signal.

$$S_{b,k+1}(f) \approx \int_{-T/2}^{T/2}\left(\frac{\beta_r A_0^2}{2}\right)\exp\left\{j\left[2\pi\left(\left(\frac{(2\mu R_0 - 2v f_0 - 2\mu vkT_r)t}{c}\right) + \frac{(2f_0 R_0 - 2v f_0 kT_r)}{c}\right)\right]\right\}\exp(-j2\pi ft)dt$$

$$\approx \left(\frac{\beta_r A_0^2}{2}\right)\exp\left[\frac{j2\pi f_0(2R_0 - 2vkT_r)}{c}\right] \cdot TSa\left(\pi T\left(\left(\frac{f - 2\mu R_0}{c}\right) + \left(\frac{2v f_0}{c}\right) + \frac{2\mu vkT_r}{c}\right)\right).$$

(6)

There is background clutter in the differential signal spectrum, which needs to be processed by MTI. The background clutter is mainly fixed target echo and slow moving clutter. In this paper, the high-pass Butterworth filter is chosen as the MTI filter to suppress the clutter.

3.2. STFT Transformation to Generate Echo Time-Frequency Map.
When $f = (2(R_0 - vkT_r)\mu/c) - (2v f_0/c)$, equation (6), $S_{b,k+1}(f)$, obtains the maximum value, i.e.,

$$S_{b,k+1}(f) = \left(\frac{\beta_r A_0^2 T}{2}\right)\exp\left[\frac{j2\pi f_0(2R_0 - 2vkT_r)}{c}\right]. \quad (7)$$

It can be seen that the frequency points corresponding to the peak of the single-period signal spectrum contain both distance and velocity information. It is necessary to perform time-frequency analysis on the spectral components of all repeated-period signals within the same frequency point by STFT, so as to obtain the Doppler shift information of the differential beat signal and convert it into two-dimensional information and then convert it into a time-frequency map [21–23].

$$STFT[x(t)] = \int_{-\infty}^{+\infty}[x(t)h(t - u)]\exp(-j2\pi ft)dt, \quad (8)$$

where $x(t)$ is the spectral component of all repeated periodic signals at the same frequency point, $h(t)$ is the Hanning window, and u is the window function shift distance.

To facilitate computer processing, the signal is discretized, and the discrete form of (8) is

$$STFT[x(n)] = \sum_{n=-\infty}^{\infty}[x(n)h(n - mk)\exp(-j2\pi nF)], \quad (9)$$

where $x(n)$ is the discrete spectral component of all repetitive periodic signals within the same frequency point, $h(n)$ is the Hanning window, m is the single move step of the window function, k is the number of move steps, and F is the digital frequency.

3.3. Recognition Using CNN.
The time-frequency map is used as the input data and the network parameters are trained. Due to the small dataset, a CNN with fewer layers is constructed to reduce overfitting and error transmission, as shown in Figure 3 and Tables 1 and 2.

Two convolutional layers ($C1$, $C2$) with 5×5 convolutional kernel size and 16 and 32 convolutional kernels respectively, both in steps of 1; two pooling layers ($P1$, $P2$) with 3×3 and 2×2 pooling window matrices respectively, in
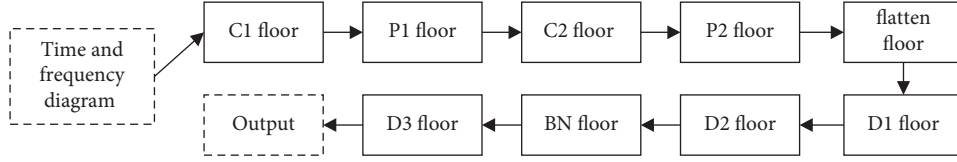
Figure 3: Schematic diagram of convolutional neural network.

Table 1: Convolutional and pooling layer network parameters.

Network layer number	Network layer	Convolution kernel/pooling window matrix dimension	Number of convolution kernels	Step	Network layer graph dimension
C1	Convolutional layer 1	5×5	16	1	$16 \times 220 \times 220$
P1	Pooling layer 1	3×3		3	$16 \times 73 \times 73$
C2	Convolutional layer 2	5×5	32	1	$32 \times 69 \times 69$
P2	Pooling layer 2	2×2		2	$32 \times 34 \times 34$

Table 2: Full connectivity layer network parameters.

Network layer number	Network layer	Weight matrix dimension	Output vector dimension
D1	Fully connected layer 1	36992×64	1×64
D2	Fully connected layer 2	64×32	1×32
D3	Fully connected layer 3	32×4	1×4

steps of 3 and 2; three fully connected layers ($D1$, $D2$ and $D3$) with $36\,992 \times 64$, 64×32 and 32×4 weight matrix dimensions respectively). The activation functions of $D1$ and $D2$ are Relu1, except for the activation function of the fully connected layer D3, which is softmax.

The convolutional layers ($C1$ and $C2$) use multiple convolutional kernels to extract depth features from the image. Let the original image be $X = (x_{i_1 j_1})_{\alpha \times \alpha}$, the convolution kernel be $W = (w_{i_2 j_2})_{\beta \times \beta}$, the convolution kernel dimension be d, and the convolution kernel move step be e. X, W perform the convolution operation, and the output is $Y = (y_{i_3 j_3})_{\gamma \times \gamma}$, and then the activation function Relu returns the negative value, i.e.,

$$\text{Relu}(x) = \begin{cases} 0, & x < 0, \\ x, & x \geq 0. \end{cases} \tag{10}$$

In the pooling layers ($P1$ and $P2$), the pool window matrix is used to extract the local maximum value of reservoir output, sample the matrix of each channel, output the dimension set in the pool window matrix, and move the pool window matrix. Let a be the input vector with dimension 1×36992; W_1, W_2, W_3 be the weight matrices of fully connected layers 1, 2, and 3 with dimensions 36992×4, 64×32, and 32×4, respectively; h_1, h_2, h_2' be the output vectors of D1, D2, and BN (batch normalization) layers with dimensions 1×64, 1×32, and 1×32, respectively; and out be the predicted value of the network with dimension 1×4, respectively. b_1, b_2, b_3 is the bias of fully connected layers $D1$,

$D2$, and $D3$, respectively. Let the output vector h_2 of $D2$ layer be $(x_1, x_2, \ldots, x_N)$; then, the BN layer can be expressed as

$$\text{BN}(x_1, x_2, \ldots, x_N) = \left(\left(\frac{(x_1 - \mu_h)\sigma_2}{\sigma_1 + \mu_h'} \right), \left(\frac{(x_2 - \mu_h)\sigma_2}{\sigma_1} \right) \right.$$
$$\left. + \mu_h', \ldots, \frac{(x_N - \mu_h)\sigma_2}{\sigma_1 + \mu_h'} \right). \tag{11}$$

Let the input vector of softmax be $(y_1, y_2, \ldots, y_M)$; then,

$$\text{Softmax}(y_1, y_2, \ldots, y_M) = \left[\sum_{i=1}^{M} \exp(y_i) \right]^{-1} (\exp(y_1), \tag{12}$$
$$\exp(y_2), \ldots, \exp(y_M)).$$

The network model is

$$\left. \begin{array}{l} h_1 = \text{Relu}(aW_1 + b_1) \\ h_2 = \text{Relu}(h_1 W_2 + b_2) \\ h_2' = \text{BN}(h_2) \\ \text{out} = \text{Softmax}(h_2' W_3 + b_3) \end{array} \right\}. \tag{13}$$

In this paper, the network parameters are updated by the gradient descent method, and the loss function cross entropy (cross entropy) is
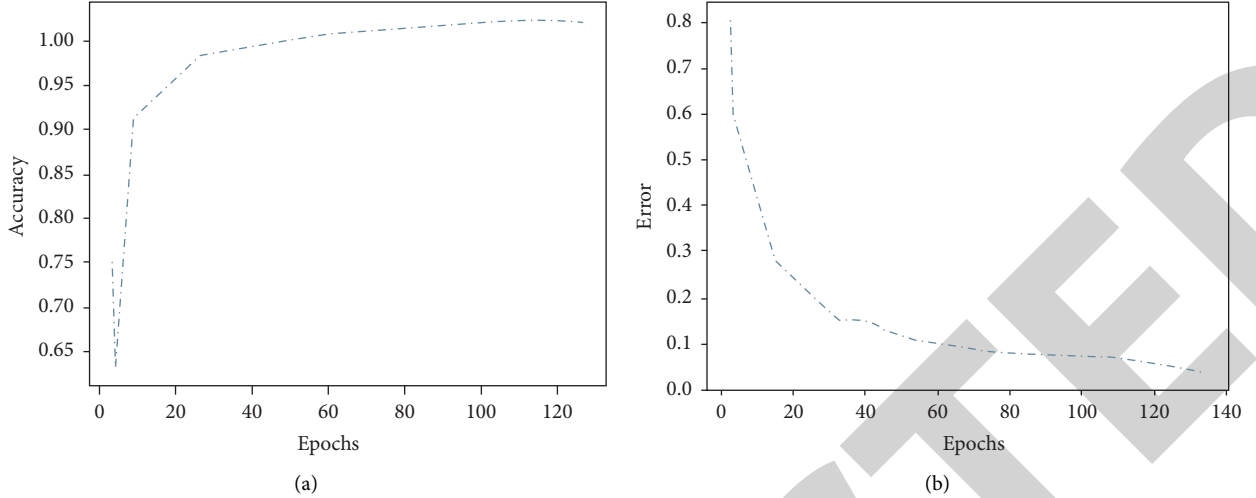
FIGURE 4: Variation of accuracy and error with the number of iteration rounds in the training set. (a) Accuracy. (b) Error.

$$\rho_{\text{loss}} = -\sum_{i=1}^{M} O_i \ln \widehat{O}_i. \tag{14}$$

The process of updating the parameters can be expressed as

$$
\begin{aligned}
g_\theta &= \nabla_\xi \rho_{\text{loss}}, \\
\lambda_\theta &= \beta_1 \lambda_{\theta-1} + (1 - \beta_1) g_\theta, \\
\rho_\theta &= \beta_2 \rho_{\theta-1} + (1 - \beta_2) g_\theta^2, \\
\widehat{\lambda}_\theta &= \left(\frac{\lambda_\theta}{1 - \beta_1^\theta} \right), \\
\widehat{\rho}_\theta &= \left(\frac{\rho_\theta}{1 - \beta_2^\theta} \right), \\
\xi_\theta &= \left(\frac{\xi_{\theta-1} - \eta \widehat{\lambda}_\theta}{\widehat{\rho}_\theta^{1/2} + \varepsilon} \right).
\end{aligned}
\tag{15}
$$

4. Experiments and Analysis

The above method is experimentally validated using a publicly available dataset on the Web [24–27]. The dataset is obtained from the LFMCW radar, which detects four types of human gestures: walking, sitting, standing, and falling. The experiments were conducted in an indoor environment with 106 participants to obtain the motion data, and each motion was repeated 2-3 times. STFT used a Hanning window with a length of 0.2 s and an overlap time of 0.19 s [28, 29].

4.1. CNN Generalization Performance. In order to avoid the phenomenon of slow convergence due to too small learning rate and oscillation of accuracy when the parameters

TABLE 3: Accuracy of 4-class image classification in the test set.

| Human motion image type	Classification accuracy (%)
Walking	92
Sitting	99
Standing up	100
Falling	100

converge to near the optimal point due to too large learning rate, this paper adopts a segmented decay strategy of learning rate, i.e., $\eta = 5 \times 10^{-4}$ when iterating within 20 rounds, $\eta = 1 \times 10^{-4}$ from 20 to 30 rounds, $\eta = 5 \times 10^{-5}$ from 30 to 40 rounds, and $\eta = 1 \times 10^{-5}$ above 40 rounds.

The accuracy and error of the training set with the number of iteration rounds are shown in Figure 4, and the accuracy of the test set with 4 classes of image classification is shown in Table 3.

From Figure 4, the accuracy of the training set has reached more than 90% within 5 iterations, indicating that the network parameters have converged to a smaller range, but the curve oscillation amplitude is more obvious due to the large learning rate, and when the learning rate decreases after 40 iterations, the curve oscillation amplitude decreases significantly due to the reduction of the learning rate, and the accuracy reaches more than 99% after 150 iterations, and the average error is 0.0114. Due to the slight overfitting, the accuracy of the test set is always slightly smaller than that of the training set, and after 150 iterations, the accuracy is 97.208%, with an average error of 0.1106.

4.2. Effect of Network Parameters on the Recognition Effect of CNN. The accuracy and error of the training set with the number of iterations are shown in Figures 5(a)–5(c).

From Figure 5, we can get the following. ①When the activation function is changed, after 150 iterations, the accuracy of the training set is 98.6%, and the average error is 0.1668; compared with that before the parameters are changed, the oscillation amplitude of the training set is
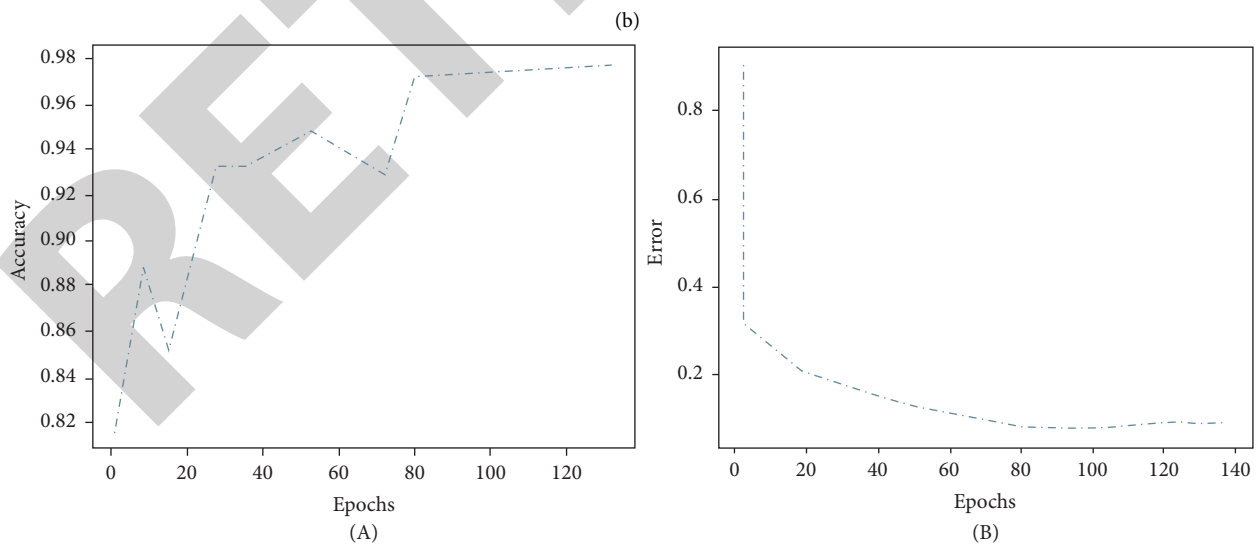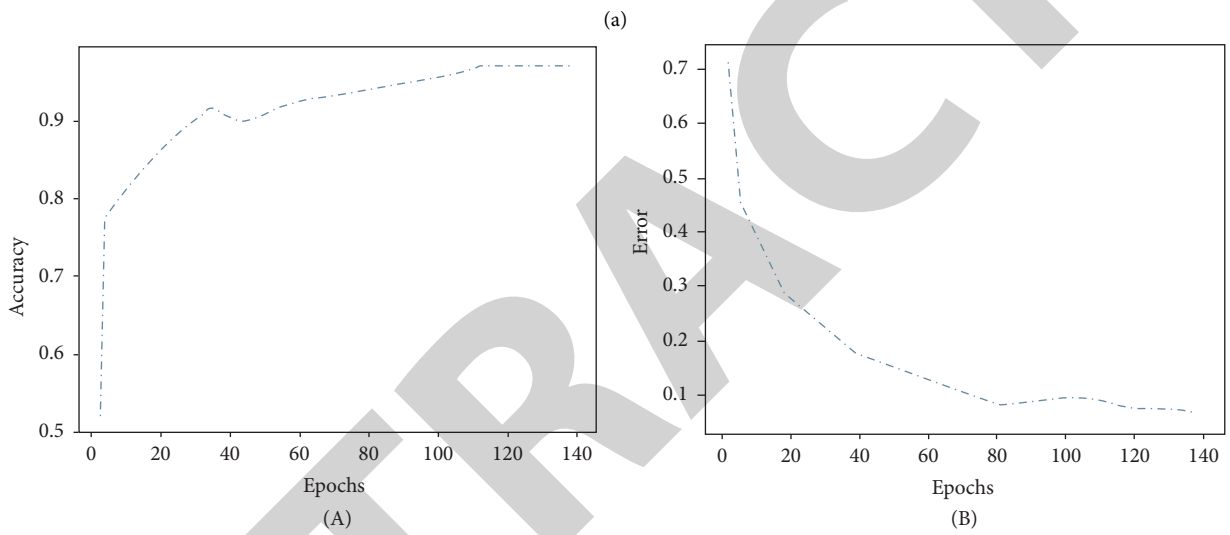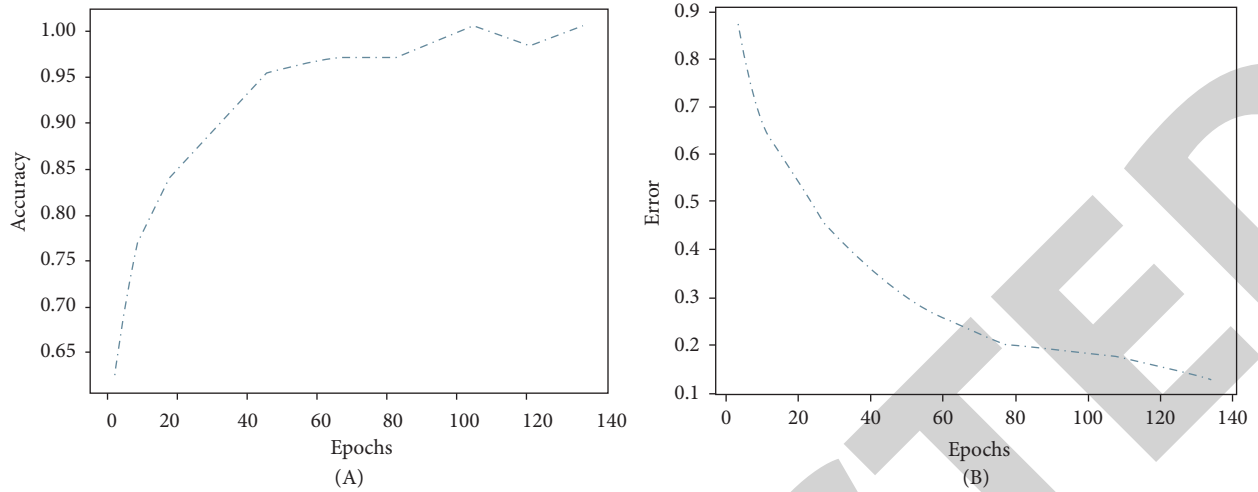
FIGURE 5: (a) Training set accuracy and error when changing the activation function. (A) Accuracy of training set. (B) Error of training set. (b) Training set accuracy and error when changing the optimizer. (A) Accuracy of training set. (B) Error of training set. (c) Training set accuracy and error when changing the learning rate. (A) Accuracy of training set. (B) Error of training set.

TABLE 4: Test results after changing parameters.

Change parameter type	Accuracy (%)	Average error
Activation function	94.588	0.3576
Optimizer	97.859	0.1101
Learning rate	98.552	0.0697

significantly larger, the overfitting is aggravated, and the generalization ability of the model is reduced. ②When the optimizer is changed, after 150 iterations, the accuracy of the training set is 99.8%, and the average error is 0.0162; compared with that before the parameters are changed, the oscillation amplitude of the training set is basically the same, the overfitting is reduced, and the model generalization ability is basically the same. When the learning rate is changed, after 150 iterations, the accuracy of the training set is 99.6%, and the average error is 0.0219. Compared with that before the parameters are changed, the oscillation amplitude of the training set is slightly increased, the overfitting is slightly reduced, and the generalization ability of the model is improved. The test results are shown in Table 4.

Therefore, when individual network parameters are changed, the generalization ability of the network model will be affected to some extent, but the accuracy of the test set always remains above 94% (see Table 4), which indicates that the network model has certain robustness and can better extract and recognize the micro-Doppler features of some simple human action postures.

5. Conclusion

A CNN-based human posture action recognition method is proposed for motion action judgment. The method obtains the time-frequency map of human action gestures by two-dimensional Fourier transform and then uses CNN to extract micro-Doppler features from the radar time-frequency map for classification. Compared with the traditional BP (backpropagation) neural network, it improves the learning efficiency of the network and better solves the problems of overfitting and mistransmission caused by small datasets. The robustness and superiority of the method are evaluated from various aspects, and the experiments are perfect and effective. Specifically, high recognition accuracy was achieved in the classification of four human action poses, namely, walking, sitting, standing, and falling, and the final recognition accuracy reached more than 97%, which achieved the expected goal.

Data Availability

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

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] J. I. Bisson, R. Deursen, B. Hannigan et al., "Randomized controlled trial of multi-modular motion-assisted memory desensitization and reconsolidation (3MDR) for male military veterans with treatment-resistant post-traumatic stress disorder," *Acta Psychiatrica Scandinavica*, vol. 142, no. 2, pp. 141–151, 2020.

[2] M. J. van Gelderen, M. J. Nijdam, J. F. Haagen, and E. Vermetten, "Interactive motion-assisted exposure therapy for veterans with treatment-resistant posttraumatic stress disorder: a randomized controlled trial," *Psychotherapy and Psychosomatics*, vol. 89, no. 4, pp. 215–227, 2020.

[3] S. Zhang and V. Callaghan, "Real-time human posture recognition using an adaptive hybrid classifier," *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 2, pp. 489–499, 2021.

[4] L. Chen and S. Li, "Human motion target posture detection algorithm using semi-supervised learning in internet of things," *IEEE Access*, vol. 9, Article ID 90529, 2021.

[5] G. Zhang and L. Zhong, "Research on volleyball action standardization based on 3D dynamic model," *Alexandria Engineering Journal*, vol. 60, no. 4, pp. 4131–4138, 2021.

[6] A. Nadeem, A. Jalal, and K. Kim, "Automatic human posture estimation for sport activity recognition with robust body parts detection and entropy Markov model," *Multimedia Tools and Applications*, vol. 80, no. 14, pp. 21465–21498, 2021.

[7] M. A. R. Ahad, M. Ahmed, A. Das AntarDas Antar, Y. Makihara, and Y. Yagi, "Action recognition using kinematics posture feature on 3D skeleton joint locations," *Pattern Recognition Letters*, vol. 145, pp. 216–224, 2021.

[8] L. Zhao and W. Chen, "Detection and recognition of human body posture in motion based on sensor technology," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 15, no. 5, pp. 766–770, 2020.

[9] A. Huang and J. Wang, "Wearable device in college track and field training application and motion image sensor recognition," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–14, 2021.

[10] V. Igelmo, A. Syberfeldt, D. Högberg, F. Rivera, and E. Luque, "Aiding observational ergonomic evaluation methods using MOCAP systems supported by AI-based posture recognition," in *Proceedings of the 2020 6th International Digital Human Modeling Symposium*, vol. 11, pp. 419–429, Skövde, Sweden, September 2020.

[11] E. Rocha-Ibarra, M. I. Oros-Flores, D. L. Almanza-Ojeda et al., "Kinect validation of ergonomics in human pick and place activities through lateral automatic posture detection," *IEEE Access*, vol. 9, Article ID 109067, 2021.

[12] R. Ali, M. Afzal, M. Hussain et al., "Multimodal hybrid reasoning methodology for personalized wellbeing services," *Computers in Biology and Medicine*, vol. 69, pp. 10–28, 2016.

[13] R. Ali, M. Afzal, M. Sadiq et al., "Knowledge-based reasoning and recommendation framework for intelligent decision making," *Expert Systems*, vol. 35, no. 2, Article ID e12242, 2018.

[14] S. Sandhya Rani, G. Apparao Naidu, and V. Usha Shree, "Kinematic joint descriptor and depth motion descriptor with convolutional neural networks for human action recognition," *Materials Today Proceedings*, vol. 37, no. 1, pp. 3164–3173, 2021.

[15] M. Lu, Y. Hu, and X. Lu, "Driver action recognition using deformable and dilated faster R-CNN with optimized region

proposals," *Applied Intelligence*, vol. 50, no. 4, pp. 1100–1111, 2020.

[16] C. Wang, J. Zhou, B. Xiao et al., "Uncertainty Estimation for Stereo Matching Based on Evidential Deep Learning," *Pattern Recognition*, vol. 124, 2021.

[17] B. Li, B. Bai, and C. Han, "Upper body motion recognition based on key frame and random forest regression," *Multimedia Tools and Applications*, vol. 79, no. 1, pp. 5197–5212, 2020.

[18] M. F. Leung and J. Wang, "Minimax and biobjective portfolio selection based on collaborative neurodynamic optimization," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 7, pp. 2825–2836, 2021.

[19] P. An, Z. Wang, and C. Zhang, "Ensemble unsupervised autoencoders and Gaussian mixture model for cyberattack detection," *Information Processing & Management*, vol. 59, no. 2, Article ID 102844, 2022.

[20] S. Nagi Alsubari, S. Deshmukh, A. Abdullah Alqarni, N. Alsharif, O. Waselallah Alsaade, and O. I Khalaf, "Data analytics for the identification of fake reviews using supervised learning," *Computers, Materials & Continua*, vol. 70, no. 2, pp. 3189–3204, 2022.

[21] S. Bharany, S. Sharma, S. Badotra et al., "Energy-efficient clustering scheme for flying ad-hoc networks using an optimized LEACH protocol," *Energies*, vol. 14, no. 19, p. 6016, 2021.

[22] M. F. Bulbul and H. Ali, "Gradient local auto-correlation features for depth human action recognition," *SN Applied Sciences*, vol. 3, no. 5, pp. 535–613, 2021.

[23] N. Nida, M. H. Yousaf, A. Irtaza, and S. A. Velastin, "Deep temporal motion descriptor (DTMD) for human action recognition," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 28, no. 3, pp. 1371–1385, 2020.

[24] A. Sabater, I. Alonso, L. Montesano, and A. C. Murillo, "Domain and view-point Agnostic hand action recognition," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7823–7830, 2021.

[25] J. Li and D. Gu, "Research on basketball players' action recognition based on interactive system and machine learning," *Journal of Intelligent and Fuzzy Systems*, vol. 40, no. 2, pp. 2029–2039, 2021.

[26] J. Li, Z. Zhou, J. Wu et al., "Decentralized on-demand energy supply for blockchain in internet of things: a microgrids approach," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 6, pp. 1395–1406, 2019.

[27] W. Duan, J. Gu, M. Wen, G. Zhang, Y. Ji, and S. Mumtaz, "Emerging Technologies for 5G-IoV Networks: Applications, Trends and Opportunities," *IEEE Network*, vol. 34, no. 5, pp. 283–289, 2020.

[28] K. Tang, A. Kumar, M. Nadeem, and I. Maaz, "CNN-based smart sleep posture recognition system," *IoT*, vol. 2, no. 1, pp. 119–139, 2021.

[29] J. Y. He, X. Wu, Z. Q. Cheng, Z. Yuan, and Y. G. Jiang, "DB-LSTM: densely-connected Bi-directional LSTM for human action recognition," *Neurocomputing*, vol. 444, pp. 319–331, 2021.