Hindawi
Mobile Information Systems
Volume 2022, Article ID 4652946, 12 pages
https://doi.org/10.1155/2022/4652946

Research Article

Research on Human Action Feature Detection and Recognition Algorithm Based on Deep Learning

Zhipan Wu[1] and Huaying Du[2]

[1]*School of Computer Science and Engineering, Huizhou University, Huizhou 516007, Guangdong, China*
[2]*School of Information Technology City College of Huizhou, Huizhou 516025, Guangdong, China*

Correspondence should be addressed to Huaying Du; duhuaying@tm.hzc.edu.cn

Received 13 July 2022; Revised 5 August 2022; Accepted 20 August 2022; Published 29 August 2022

Academic Editor: Yajuan Tang

With the improvement of computer computing power and storage capacity, the emergence of massive data makes the methods based on human action feature detection and recognition unable to meet people's needs due to poor generalization ability. Based on the detection and recognition of human action features based on deep learning algorithms, a suitable neural network can be constructed to identify locked human action features from surveillance video and analyze whether it is a specific behavior. In this paper, a deep learning algorithm is proposed to optimize the detection of human action features, and a multiview reobservation fusion action recognition model of 3D pose is designed. Several factors affecting the recognition of human action features are analyzed, and a detailed summary is made from the detection environment. Experiments show that adding one or two layers of feature attention enhancement to the multiview observation fusion network can improve the accuracy by 1% to 3%. In this way, the model can integrate action features from multiple observation angles to judge actions and learn to find observation angles suitable for action recognition, thereby improving the performance of action recognition.

1. Introduction

In recent years, the application of deep learning in the field of computer vision and pattern recognition has been in-depth, especially the research on human action recognition, which has become one of the important technologies in deep learning application research. It has huge development potential in intelligent monitoring, video understanding, human-computer interaction, medical diagnosis, and so on.

This paper conducts in-depth research on computer vision technology in deep learning and uses deep learning technology to solve the problems of intelligence and automation of monitoring systems. According to the three-dimensional recognition of human body, a multiview fusion model based on attention mechanism is proposed, and a set of deep learning monitoring system based on deep learning is developed. The system can detect, identify, track and identify people in the monitored scene, and analyze the obtained information, so as to play the role of supervision, early warning, and prevention. Therefore, the research on deep learning monitoring system has important application value and significance.

In order to study the problem of 3D human action recognition, this paper proposes a multiview reobservation fusion model based on attention mechanism. In the fusion process, the attention mechanism is used to evaluate whether the observation perspective is helpful to the action recognition according to the action sequence information. In this way, the model can learn to find suitable viewpoints for action recognition among multiple viewpoints based on action information.

2. Related Work

Human action recognition is widely used in intelligent applications, and key human detection is a hot topic in the field of action recognition. In order to improve the identification rate of key points in the human body, Thrall et al. used an artificial intelligence program to extract "radioactive" information from images that cannot be discerned by visual inspection, thereby increasing the diagnostic and prognostic value derived from image datasets [1]. Mamoshina et al. outlined next-generation artificial intelligence and blockchain technology and proposed innovative solutions that can be used to accelerate biomedical research, enable patients to control and monetize personal data with new tools, and incentives for continuous health monitoring [2]. Jiang et al. proposed a novel unified framework that jointly exploits feature relations and class relations to improve classification performance. The proposed Regularized DNN (rDNN) is more suitable for video semantic modeling by endowing the DNN with the ability to better utilize features and class relations [3]. Cheng et al. conducted a comprehensive survey of recent advances in network acceleration, compression, and accelerator design from both algorithmic and hardware perspectives, while the computational complexity and resource consumption of these networks are increasing. In terms of hardware implementation of deep neural networks, a number of accelerators based on Field Programmable Gate Array (FPGA) or Application Specific Integrated Circuit (ASIC) have been proposed in recent years [4]. Dong and Wang studied the robust exponential stability problem of uncertain discrete-time stochastic neural networks with time-varying delays based on output feedback control by choosing an enhanced Lyapunov-Krasovskii functional, which establishes a sufficient condition for asymptotically stable delay correlation in the mean square for a class of discrete-time stochastic neural networks with time-varying delays [5]. Lucas et al. reviewed deep learning techniques for solving such inverse problems in imaging. Specifically popular neural network architectures for imaging tasks, they provided some insights into how these deep learning tools can solve the inverse problem [6]. Yin et al. proposed BinaryRelax, a simple two-stage algorithm for training deep neural networks with quantized weights, and aimed to test BinaryRelax on benchmark CIFAR and ImageNet color image datasets to demonstrate the superiority of relaxed quantization methods and higher accuracy than state-of-the-art training methods [7]. Sarabu and Santra used two CNNs that use pretrained ImageNet models to extract spatiotemporal features, combining the results of the two CNNs in the first step as input to CLSTM to obtain an overall classification score [8]. Liu et al. made full use of video and deep skeleton data and propose a dual-stream network (SV-GCN) based on RGB-D action recognition, which can be said to be a dual-stream architecture that processes two different data. To provide the model with richer skeleton point features, they replaces the traditional random sampling layer with an atrous convolutional layer, which can better utilize the depth features, and finally fuses the two-stream information to achieve action recognition [9]. These studies are instructive to a certain extent, but in some cases, the demonstrations are insufficient or inaccurate and can be further improved.

3. Action Recognition Method Based on Deep Learning

Deep learning is a learning method using distributed features, which can automatically learn from low-level features to high-level features, eliminating the dependence on manual feature extraction methods. And the powerful feature extraction ability of deep learning makes it have huge advantages in classification and recognition. Traditional human action recognition aims to extract design features and classify them according to the features. Human target detection and recognition based on deep learning algorithm constructs a suitable neural network, which can identify a movement of a locked person from surveillance video, and analyze whether it is a specific behavior [10–12]. The traditional human action recognition flowchart is shown in Figure 1. Deep learning performs feature extraction at the position with the greatest saliency by defining a saliency function, and obtains local feature descriptors.

Traditional human action recognition is mainly based on RGB images or videos, and the effect is not satisfactory due to factors such as scale, illumination changes, and background noise. The use of deep learning is popular in the field of computer vision. At present, the recognition effect of this human action recognition method through deep learning is far superior to the traditional recognition method and gradually replaces the traditional human body recognition [13–15]. The basic framework of the human action recognition framework based on deep learning is shown in Figure 2.

3.1. Mask RCNN Algorithm. The Mask RCNN algorithm framework can be widely used in many vision tasks such as target detection, target instance segmentation, target key point detection and so on by fine-tuning the network structure [16]. Since it inherits the advantages of both the classic segmentation network FCN and the classic target detection network Faster RCNN, Mask RCNN can simultaneously ensure high classification accuracy, detection accuracy, and instance segmentation accuracy. Although Mask RCNN is more complex than Faster RCNN, the speed of the two is comparable. Therefore, the above advantages make Mask RCNN extremely outstanding in multiple vision tasks [17].

The schematic diagram of the overall structure of Mask RCNN is shown in Figure 3. The first half of the network structure is similar to the network structure of Faster RCNN. First, the input image is extracted as a feature map through the convolution layer, and then the feature map is input to the RPN network, so that the RPN generates the region of interest corresponding to the feature map. Then, the feature maps of the region of interest are uniformly scaled through the RoIAlign layer, and finally, the obtained uniform-size feature maps are input to multiple branch networks. Two of the fully connected layer branches are used to obtain the classification score results and the regression parameters of the target frame respectively, and another branch generates a mask image of the same size as the input original image through the image segmentation network FCN, that is, the semantic segmentation result of the image classifies each pixel of the image [18].

Mask RCNN introduces the RoIAlign structure to replace the RoI pooling layer of the Faster RCNN part. The

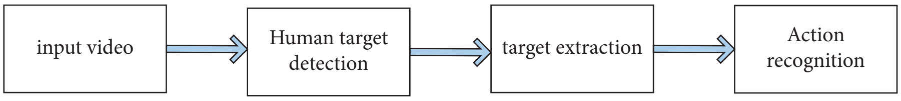


Figure 1: Traditional human action recognition flowchart.

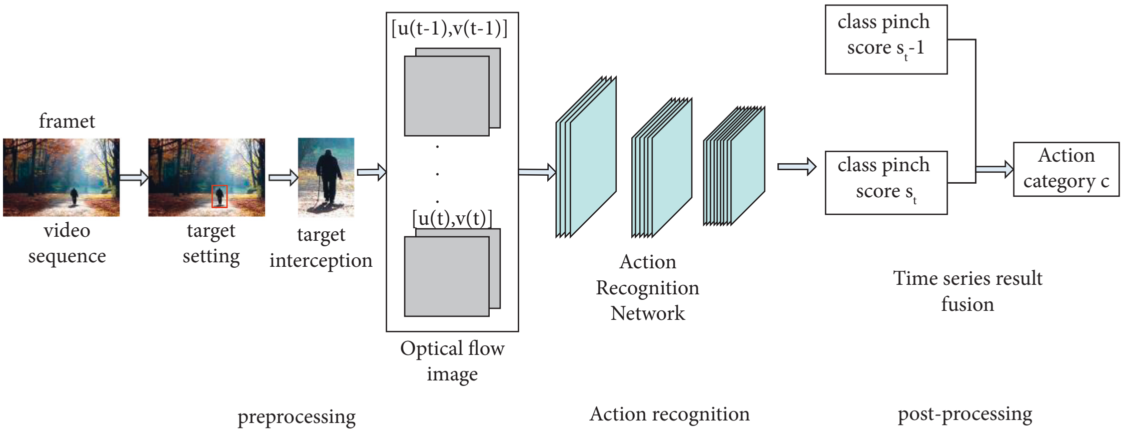


Figure 2: Action recognition framework.

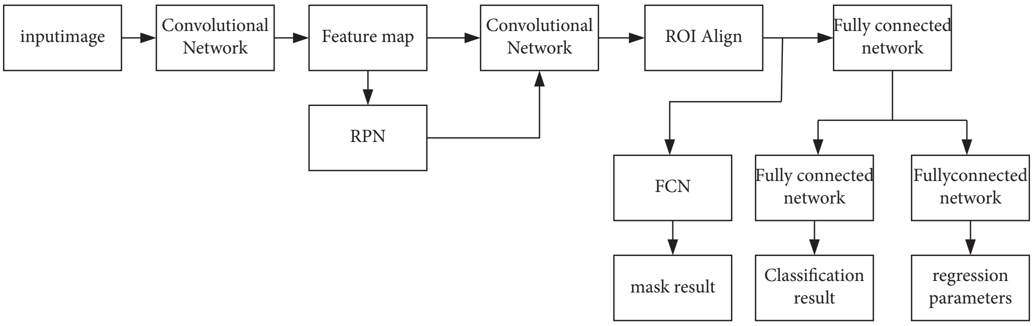


Figure 3: Schematic diagram of Mask RCNN framework.

main purpose of this layer is also to extract the feature map corresponding to the region of interest and unify the feature maps of the region of interest of different sizes to the same size, so as to facilitate the input of the subsequent network [19].

3.2. Fully Convolutional Neural Networks. The network introduces a deconvolution structure to generate a mask map that is the same size as the input feature map, thereby generating a mask value classification for each pixel, and implementing instance segmentation for each pixel [20]. Through the convolution and pooling operations of the input data, the full-volume neural network starts from the shallow features, extracts the deep features layer by layer, and directly inputs the original image data, avoiding the complex preprocessing of the image. Deconvolution can actually be understood as the reverse operation of ordinary convolution operations, and its schematic diagram is shown in Figure 4. It first fills the feature map with 0 between each feature point and then performs the convolution calculation of the convolution kernel to obtain a feature map with a larger size after upsampling [21, 22].

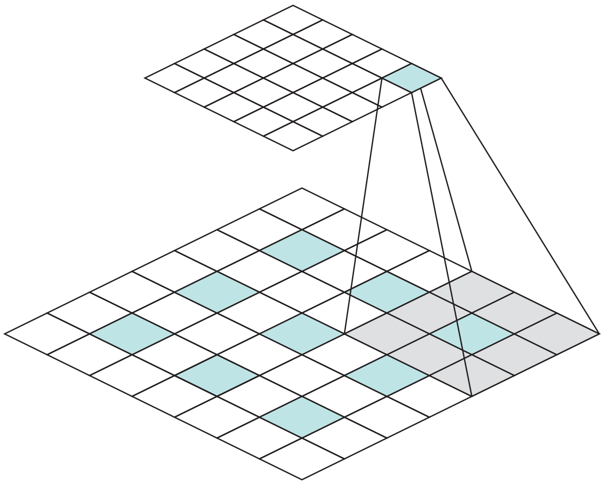


Figure 4: Schematic diagram of deconvolution.

The feature extraction process of the convolutional local connection in Figure 4 can be regarded as a process of convolution of each region in the image by multiple convolution kernels. After the input image has undergone multiple convolutions, the resolution of the feature map becomes smaller and smaller. Then, the size of the low-resolution feature map needs to be restored to the same size as the input feature map in order to generate a mask feature map to classify and score each pixel [23]. Full-volume neural networks are fault-tolerant and adaptive, and robust to specific poses, lighting, displacements, scaling, or more types of image distortions. FCN deconvolutes the feature map finally output by the original convolutional network by 32 times and can directly obtain the feature map of the same size as the original image, but such a result can only reflect the deep overall features, without using some intermediate features. The detailed local features extracted by the layer, so FCN introduces a skip-level structure.

By training the parameters of the FCN network, the network is finally able to generate a mask of the same size as the input image, that is, each pixel has a corresponding classification score, and the output feature map has layers in depth. That is, there is a category, and the score of each pixel in each layer represents the confidence score that the pixel belongs to the current layer category. First, the classification probability is obtained by performing Softmax on the score of each layer and then the target position or key point position belonging to the category of the current layer is found through a certain threshold limit.

3.3. Human Key Point Detection Based on Improved Mask RCNN. In this experiment, a new method for initializing the size of the anchor candidate frame is introduced. The size of the character annotation frame in the dataset is classified by the clustering method, and then the average value of each type of cluster is calculated, so that the anchor point candidate frame size that is more suitable for the human key point detection task is calculated through the dataset itself [24–26]. The general idea of the K-means algorithm is to first select a sample from the training set as the center of the cluster and evaluate the distance between all the sample points and the center of the cluster. For each sample point, it is divided into the cluster to which the cluster center with the smallest distance value belongs. After such a clustering, the cluster center is recalculated and updated for each cluster, and so on until the model reaches a satisfactory convergence accuracy. The algorithm flow is as follows:

(1) Set the initial value for each cluster center according to the set number of clusters. Generally, a group of samples is randomly selected as the initial cluster center $\mu_1, \mu_2, , \mu_n$.

(2) Update the labels of the clusters to which all samples belong, and the cluster label of the first sample is y_i. In this paper, two feature components $x_i = (x_i^1, x_i^2)$ are set for the position of the labeling target frame. By calculating the Euclidean distance between the feature vectors composed of these two values, the similarity in size and size of the two position annotation boxes can be better represented. Calculated as follows:

Table 1: Box selection size of points based on K-means clustering algorithm.

Candidate frame number	Candidate box size
Candidate box 1	45 × 20
Candidate box 2	22 × 55
Candidate box 3	31 * 76
Candidate box 4	59 * 145
Candidate box 5	82 * 200

$$y_i = \underset{\mathrm{j}}{\operatorname{argmin}}\, \operatorname{dist}\left(\mathrm{x}_i, \mu_j\right) j \in [1, c],$$
$$\operatorname{dist}\left(x_i, \mu_j\right) = \sqrt{\sum_{k=1}^{2} \left|x_i^k - \mu_j^k\right|^2}. \tag{1}$$

(3) Update the center point of each cluster, the formula is as follows:

$$\mu_{\mathrm{j}} = \frac{\sum_{k \in S_{\mathrm{j}}} x_k}{K_j}, j = 1, 2, \ldots . \tag{2}$$

In the formula, K_j represents the number of the j-th sample points assigned in this iteration process, and S_j represents the set of samples that are assigned to the j-th sample point.

(4) Repeat steps 1 to 3 until the classification ability of the cluster center makes the model reach the convergence accuracy.

Through the above K-means clustering training process, the size and height-width ratio of the annotation frame in the human key point detection dataset are used as the two dimensions of the feature vector as the classification criteria of the cluster, and the classification result is obtained [27]. For each cluster in the classification result, the average size of the annotation frame in the same cluster is obtained, so as to obtain a set of anchor candidate frame sizes that are representative of the size of the person in the human key point dataset. In this paper, a clustering algorithm is performed on the character annotation boxes provided by the MSCOCO dataset. Considering that the characters are rich in actions, resulting in a large gap in the aspect ratio of the characters, the number of clusters is increased to 13, and the average size of the annotation box in each cluster is calculated, and finally, the anchor point candidate frame size is obtained as shown in Table 1.

For the multiperson target scene, the confidence is appropriately lowered, and a variety of calculation methods are introduced to adjust the confidence penalty. Finally, it analyzes which penalty function calculation method can better improve the recognition performance of the algorithm in dense human scenes and reduce the sensitivity of the model to the confidence threshold through experiments [28].

Three kinds of common weight calculation methods are selected as the object of discussion, namely linear function, Gaussian function, and exponential function [29–31]. These

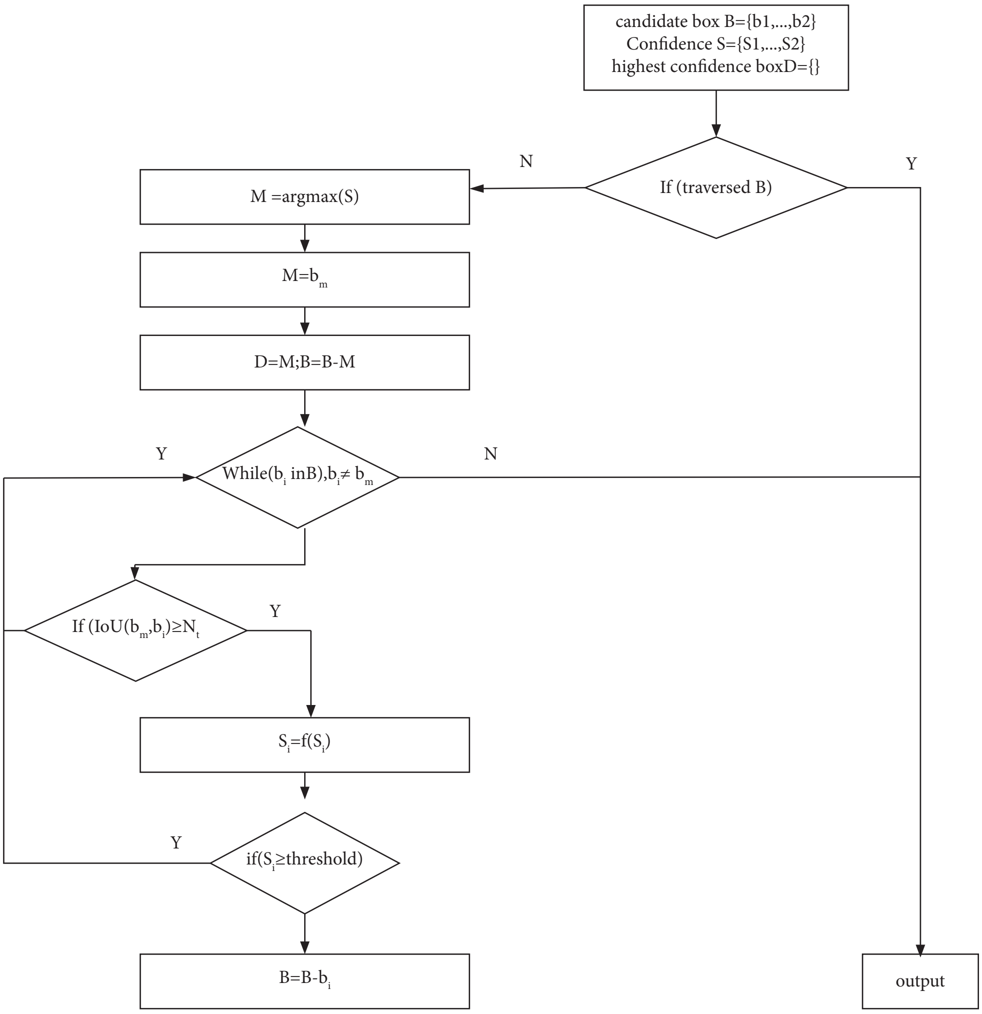


Figure 5: Flowchart of the improved nonmaximum suppression algorithm.

three types of functions can calculate a penalty value. When a candidate box whose intersection with the marked box is greater than the threshold appears in the process of nonmaximum suppression, its confidence will be penalized, so as to further judge whether to retain the candidate box. The program flowchart of the improved nonmaximum suppression is shown in Figure 5, where *f* represents the type of penalty function used, and N_t is the threshold of the intersection and union ratio. *Threshold* represents the confidence threshold, which is used to judge whether the confidence after punishment meets the conditions that need to be eliminated.

The penalty function is defined below according to three types of weighting methods. The first is a linear penalty function, whose penalty function is defined as follows:

$$s_i = \begin{cases} s_i, & \mathrm{IOU}(b_m, b_i) < N_t, \\ a\bullet s_i\left(1 - \mathrm{IOU}(b_m, b_i)\right), & \mathrm{IOU}(b_m, b_i) \geq N_t, \end{cases} \tag{3}$$

where a is a coefficient weight, ranging from 0 to 1, to control the strength of the penalty. S_i represents the confidence score of the candidate frame currently being judged, b_m represents the position coordinate of the candidate frame with the highest confidence score discharged in this round of iteration, and b_i represents the position coordinate of the candidate frame currently being judged. When the intersection ratio of the two is greater than the threshold, it is necessary to impose a certain penalty on the confidence of the candidate frame. If the values of a and N_t are different, the degree of penalty will also be different. The parameter sensitivity will be analyzed in subsequent experiments.

The second is the Gaussian penalty function, which modifies the Gaussian function to wait until the penalty function:

$$S_i = S_i e^{-\mathrm{IOU}(b_m, b_i)^2/\sigma}. \tag{4}$$

Among them, σ represents the penalty for the confidence, S_i represents the confidence score of the candidate frame currently being judged, b_m represents the position coordinate of the candidate frame with the highest confidence score discharged in this round of iteration, and b_i represents the position coordinates of the candidate frame currently being judged.

The third is the exponential penalty function. Compared with the linear function, the transition of the penalty function at the threshold point is smoother. Compared with the Gaussian function, it can maintain a higher degree of confidence in the stage where the intersection and union are low. The calculation formula is as follows:

$$S_i = \begin{cases} S_i, & \mathrm{IOU}\left(b_m, b_i\right) < N_{\mathrm{t}}, \\ S_i e^{\left(N_{\mathrm{t}} - IOU\left(b_m, b_i\right)\right)}, & \mathrm{IOU}\left(b_m, b_i\right) \geq N_{\mathrm{t.}} \end{cases} \quad (5)$$

The above three penalty functions are brought into the whole process of nonmaximum suppression. Each time the intersection ratio is judged, the above three types of penalty functions are introduced to calculate the confidence after penalty. If the confidence is still greater than the set confidence threshold, the candidate frame is retained as the region of interest, otherwise the candidate frame is rejected [32].

4. Multiview Reobservation Fusion Action Recognition Model Based on 3D Pose

In the problem of human action recognition, most of the current video-based methods directly process the entire video image, and let the network classify actions from the entire image in a data-driven way [33]. These methods rarely focus on the role of action elements, and it is difficult to explain whether the model learns human actions or more information about the appearance of the environment, especially when the current datasets are small. Compared with other deep learning-based algorithm architectures, the multipose human action recognition architecture mainly has the advantages of high recognition accuracy, low complexity, and strong complementarity of action information between multiple modalities.

Human action recognition is an important technology in the field of video understanding and analysis [34]. In recent years, with the rapid development of video-based position estimation algorithms, bone-based methods for human action recognition have become popular. The framework of action recognition based on skeleton data is mainly divided into three steps: skeleton data acquisition, action feature extraction, and action classification. All of these methods see the human skeleton as a graph, with joint points as graph nodes and bone connections as edges on the graph. Graph-based methods provide a good characterization of the skeletal structure. In the current graph-based method, nodes are convolved and pooled. However, most of the movements do not require the participation of all the joint points throughout the body, and a few of them play a central role. By introducing an attention model, the model can focus more on a small number of important joint points.

In reality, there is a high diversity of movements, and many actions will have different forms in different situations, such as eating when people may be sitting or standing [35]. This diversity of actions poses great difficulty for the generalization of the model. The observation perspective has an important influence in action recognition. On the one hand, human movements can be observed from many perspectives, and the observation data from different perspectives will vary greatly, which causes the diversity of action data. The model needs to consider the data changes brought about by the change of the observation perspective to ensure the identification effect. On the other hand, for many actions, it is difficult to identify from some angles and particularly difficult to identify from other angles. In this case, it will be helpful to identify actions from easy to identify.

Multipose feature fusion is crucial for recognition performance because it can enhance the representational ability of features and reduce the redundancy of features. In the fusion process, the model also adopts the attention mechanism to evaluate all the observation angles according to the action sequence information and give the corresponding weights. The observation perspective will achieve higher weight in the fusion process. In this way, the model can integrate the action information of multiple observation angles to judge the action and learn to find the observation angle suitable for action recognition, so as to improve the performance of action recognition.

4.1. LSTM Neural Network. RNN realizes the accumulation of historical information through the design of cyclic structure, which is suitable for processing sequence data, and has been widely used in sequence problems such as action recognition in recent years [36]. The output h_t of the current step of the RNN is determined by the current input x_t and the output h_{t-1} of the previous step of the RNN (that is, the historical information accumulated in the RNN). In this way, RNN realizes the utilization of historical information of sequence data.

$$h_t = \varphi\left(W_x \bullet x_t + W_h \bullet h_{t-1} + b\right). \quad (6)$$

In the above formula, W and b are the parameters to be learned by the RNN unit. In theory, RNN can handle sequence information of arbitrary length and dependencies between sequences in this way. However, there are many practical problems that hinder the realization of the idealization, such as gradient disappearance during training, and gradient explosion. Long Short-Term Memory (LSTM) is an improved variant of RNN for handling such long-term sequence problems. Unlike the simple perceptron-like structure in the original RNN unit, the LSTM unit contains multiple gates to control the flow of information in the LSTM. These gate units also build a temporally linear connection, which alleviates the problems of vanishing and exploding gradients to a certain extent. The data computation in the LSTM cell looks like this:

$$\begin{aligned} \mathrm{f}_t &= \sigma\left(W_{xf*g} X_t + W_{hf*g} H_{t-1} + b_f\right), \\ \mathrm{i}_{\mathrm{t}} &= \sigma\left(W_i \bullet \left[h_{t-1}, x_t\right] + b_i,\right. \\ \mathrm{o}_{\mathrm{t}} &= \sigma\left(W_O \bullet \left[h_{t-1}, x_t\right] + b_O,\right. \\ \mathrm{u}_{\mathrm{t}} &= \tan h\left(W_c \bullet \left[h_{t-1}, x_t\right] + b_c,\right. \\ \mathrm{C}_{\mathrm{t}} &= \mathrm{f}_{\mathrm{t}} * C_{t-1} + i_t * u_t, \\ \mathrm{h}_{\mathrm{t}} &= \mathrm{o}_{\mathrm{t}} * \tan h\left(C_t\right). \end{aligned} \quad (7)$$

The cell state C refers to the state information in the LSTM cell. The forget gate, input gate, and output gate are calculated according to the input x_t of the current step and the unit state C_{t-1} of the previous step to control the flow of information inside the LSTM unit. Forget gate f_t controls how much the cell state of the previous step (that is, the historical information accumulated by the cell) C_{t-1} should be retained; input gate i_t controls how much current input x_t should be input into the cell calculation; output gate o_t controls the conversion calculation from cell state to output.

The proposed action recognition model is also based on the LSTM unit. LSTM can better handle long-term and short-term time series memory by adding control gates. It has excellent performance in time series problems and is the most widely used RNN variant.

4.2. Attention Mechanism is used to Enhance Pose Information and Feature Representation. For many human actions, usually only a few limb joints play a decisive role, and the states of other joints or limbs are irrelevant to the action. For example, no matter what the posture of an object is, as long as he puts something to eat in his mouth, it constitutes the action of "eating." Based on this idea, this chapter introduces an attention mechanism to learn to assign different attention weights to different skeleton joint points according to action information to assist in action recognition. Since in the action recognition task, all the joint points may have an influence on the determination of the action, this chapter adopts the attention method of the soft attention mechanism. At step t of the LSTM network time series, skeleton data $x_t = \lfloor x_t^1, x_t^2, \ldots x_t^n \rfloor$ (x_t^{i} refer to the three-dimensional coordinates of the i-th joint point, n is the number of joint points in the skeleton model) is input into the network for processing for action recognition. During data input, the attention mechanism is used to assign an additional weight to the joint points in the skeleton, indicating the importance of the joint points under the current action sequence information. Attention weights are calculated based on input and historical information:

$$A_{\mathrm{ja}} = U_j\Big(\tanh\left(W_{jh}\bullet \mathrm{h}_{t-1} + W_{jx}\bullet \mathrm{x}_t + b_{j1}\right)\Big) + b_{j1}. \tag{8}$$

In the computation of attention weights, the state h_{t-1} of the previous step of the LSTM represents the sequence history information accumulated in the LSTM unit. According to the action history information h_{t-1} and the current skeleton data input x_t, the importance of the current step skeleton joints is predicted W_{ja}. W_{ja} is the attention weight of the joint point, corresponding to the J joint points in the skeleton model. This weight is used to revise the skeleton data, emphasizing the joint points according to their importance:

$$\tilde{x}_t = \left[x_t^1, x_t^2, \ldots x_t^n,\right]\bullet A_{ja}. \tag{9}$$

The corrected input contains the importance $\tilde{x}_t$ of the joint points. Through this enhancement of the attention mechanism, the movements of the more important joints are amplified, while the movements of the less important joints are suppressed. In this way, the sequence information or interaction information of important joint points is emphasized, which is helpful for the identification of related actions.

In a multilayer LSTM network, the output of the previous layer of LSTM is used as the input of the next layer of LSTM, and the attention LSTM enhances the attention mechanism of the features passed between the LSTM layers. The attention mechanism of general features is similar to the implementation of the attention mechanism on the skeleton joints. For the l-th LSTM layer in the network:

$$A_{\mathrm{t}} = U_t\bullet\Big(\tanh\left(W_{l,h}\bullet h_{\mathrm{l},t-1} + W_{l,x}\bullet h_{l-1,t} + b_{l1}\right)\Big) + b_{l2}. \tag{10}$$

Here, $h_{\mathrm{l}-1,t}$ is the output of the $l-1$-th layer LSTM, which is the input of the l-th layer LSTM. $h_{\mathrm{l},t-1}$ is the previous state of the lth layer LSTM. Similarly, attention weights are calculated based on feature input $h_{\mathrm{l}-1,t}$ and historical information $h_{\mathrm{l},t-1}$. Then, according to the attention weights, the features are modified:

$$\widetilde{h_{\mathrm{l}-1,t}} = h_{\mathrm{l}-1,t}\bullet A_{\mathrm{t}}. \tag{11}$$

4.3. Multiview Fusion Model Based on Attention Mechanism. The multiview fusion action recognition model of the attention mechanism first reobserves the input action sequence from multiple perspectives, then uses the deep network to process the observation data separately, and finally fuses the processing results of all observations to determine the final action category. The observation angle of action has a great influence on 3D action recognition, which is reflected in two aspects. First, action data can be collected from different observation perspectives, which increases the diversity of action data. Therefore, these increased sample diversity needs to be considered in the model or training to obtain a good algorithm generalization effect. Second, in general, many actions are hard to discern from some perspectives and easy to discern from others. In this way, finding observations from a suitable perspective will be of great help to improve the performance of action recognition.

Figure 6 illustrates the multiview reobservation fusion model. The model first performs N three-dimensional transformations on the skeleton data input x_t to simulate the observation of the skeleton data from N perspectives. The learned temporal features are extracted from the graph through a fully convolutional neural network, and the sequence of human skeleton key points can be represented by a series of undirected graphs. The obtained N observations are separately processed by the main LSTM network. In practice, observations under different viewing angles are obtained by 3D rotation of the skeleton s. Multiple new observations are obtained by performing N different rotations on the original skeleton s.

$$s_{vn} = \mathrm{Rotate}_n(s). \tag{12}$$

Skeleton sequence-based action recognition tasks have a huge dependence on timing information, so skeleton key points are encoded into multiple two-dimensional pseudoimages, which are then input into convolutional neural networks to learn timing features. When rotating the

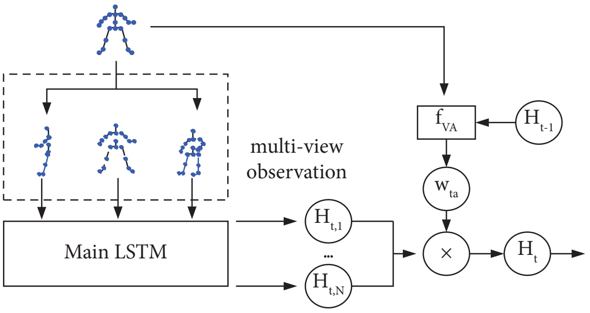


Figure 6: Multiview reobservation fusion based on attention and mechanism.

skeleton, first reposition the skeleton with its hip-center as the origin to reduce the joint drift caused by the rotation. Then, it rotates the skeleton in a 3D coordinate system. In the three-dimensional coordinate system, for a joint point $s^i = [x^i, y^i, z^i]^T$ in the skeleton, the result obtained by its rotation can be expressed as:

$$\widetilde{s}^i = R_x R_y R_z s^i. \tag{13}$$

Among them, $R_x R_y R_z$represent the rotation around the X, Y, and Z axes of the coordinate system, respectively. For example, a rotation around the Z axis can be represented by the following matrix:

$$R_y(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{14}$$

In the matrix, θ represents the rotation angle around the Z axis. In reality, most of the perspective transformations are caused by changes in the horizontal direction of the observation perspective, which can be roughly represented by the rotation of the skeleton around the Z-axis in a three-dimensional coordinate system. Therefore, in order to reduce the amount of computation, in the model implementation process, only the horizontal rotation around the Z axis is now considered, as shown in Figure 7.

As discussed earlier, among all observational perspectives, some may be helpful for action recognition, and some may be detrimental to action recognition. The multiframe images or video segments input in the 3D full-volume network pass through the network and the output is a 3D feature map. The timing information of the video is effectively preserved, and the features in the timing information are accurately extracted. Therefore, the attention mechanism is used to evaluate all observation perspectives, and the perspectives that are helpful for action recognition are given higher weights, while the perspectives that are unfavorable for action recognition are given lower weights. During fusion, the results of all views are fused according to the attention weights. In this way, the model is able to learn to pick viewpoints that are beneficial for action recognition. The attention weight for the observation perspective is also generated based on the previous output h_t of the model and the current input x_t:

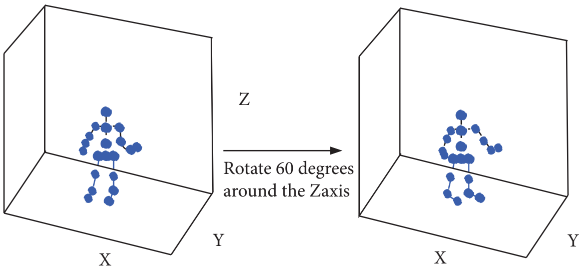


Figure 7: Skeleton model rotated 60 degrees in 3D space.

$$A_t = \mathrm{Softmax}\left(U\left(\tanh\left(W_{tx} * X + W_{th} * H_{t-1} + b_{t1}\right) + b_{t2}\right).\right. \tag{15}$$

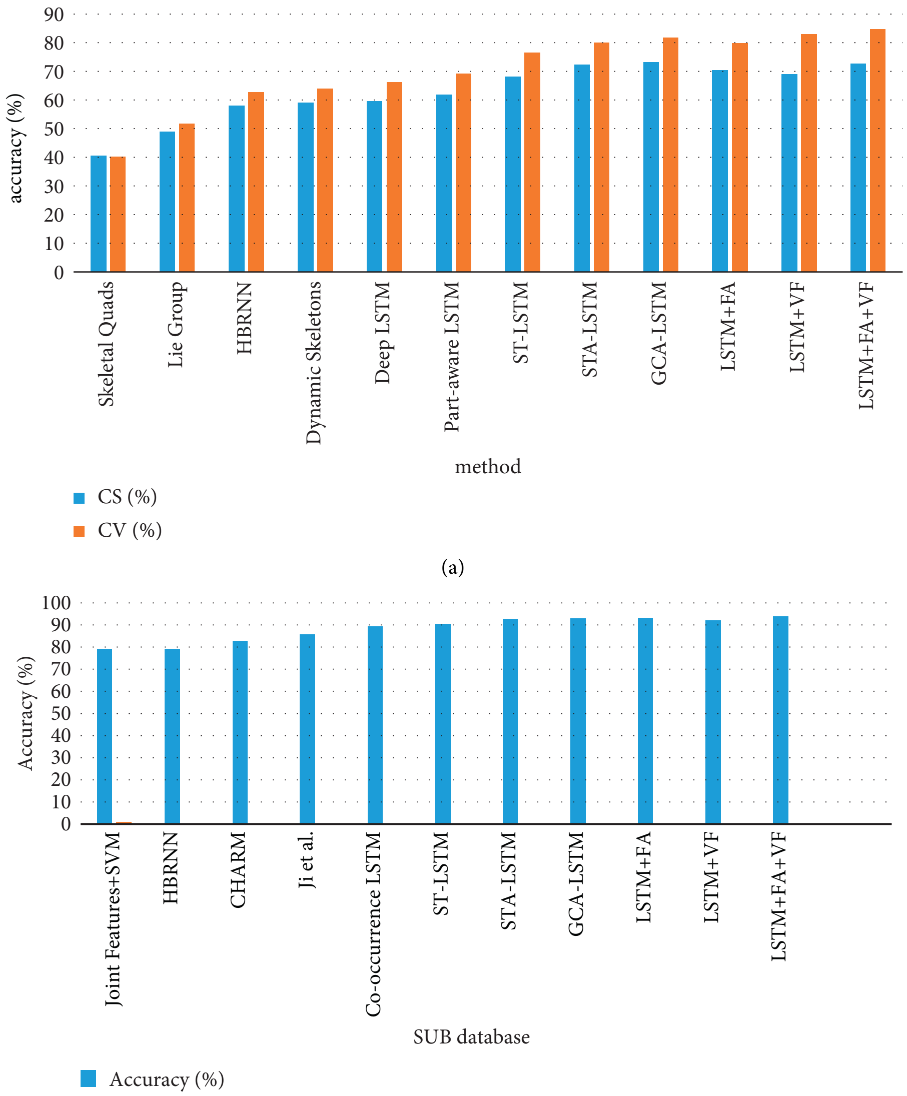
The processing results of observation data from different perspectives are weighted and summed according to the attention weight to obtain the final output result:

$$\mathrm{h}_t = \sum_{i=1}^{N} A_{t,i} \bullet h_{t,i}. \tag{16}$$

In the above formula, $[h_{t,1}, h_{t,2}, \ldots, h_{t,i}]$ is the result of the main LSTM processing of observation data from different perspectives, and the weighted summation of them obtains the output of the model fused with multiple observation perspectives. h_t is deal after Softmax processing; the final category recognition result is obtained. The model can detect, identify, track, and identify people in the monitored scene, and analyze the obtained information, so as to play the role of supervision, early warning, and prevention.

4.4. Experiment and Result. The model is mainly experimentally verified on the following two current mainstream and challenging 3D action recognition databases: NTU RGB + D is currently the largest 3D action recognition database, which includes a total of 45 action categories and 35,760 records. The 45 types of movements include daily activities, health-related movements, two-person interaction movements, and more. Actions are demonstrated by 38-bit objects and captured simultaneously from 3 different viewpoints. The scale of data in this database is large, and there are multiple perspective changes. In addition, there are many actions with high similarity. It is currently the most challenging database.

LSTM + FA in the table represents a three-layer Attention LSTM combined with a multilayer feature attention mechanism. LSTM + VF represents a multiview integrated network that uses the basic three-layer LSTM network as the main network. LSTM + FA + VF represents the final model combining multilayer feature attention mechanism and multiview observation fusion architecture. In the fusion process, the attention mechanism is used to evaluate different observation perspectives, and the perspective suitable for the recognition of the action will get a higher fusion weight. As can be seen from the comparison in the table, for the cross-view evaluation method, the effect of this method



(a)

(b)

Figure 8: Comparison results with other state-of-the-art methods on NTU RGB + D and SBU databases.

is the current state-of-the-art method, which is 3% higher than the GCA-LSTM method. For the cross-object evaluation method, the effect is similar to that of GCA-LSTM, with a difference of only 0.6% as shown in Figure 8.

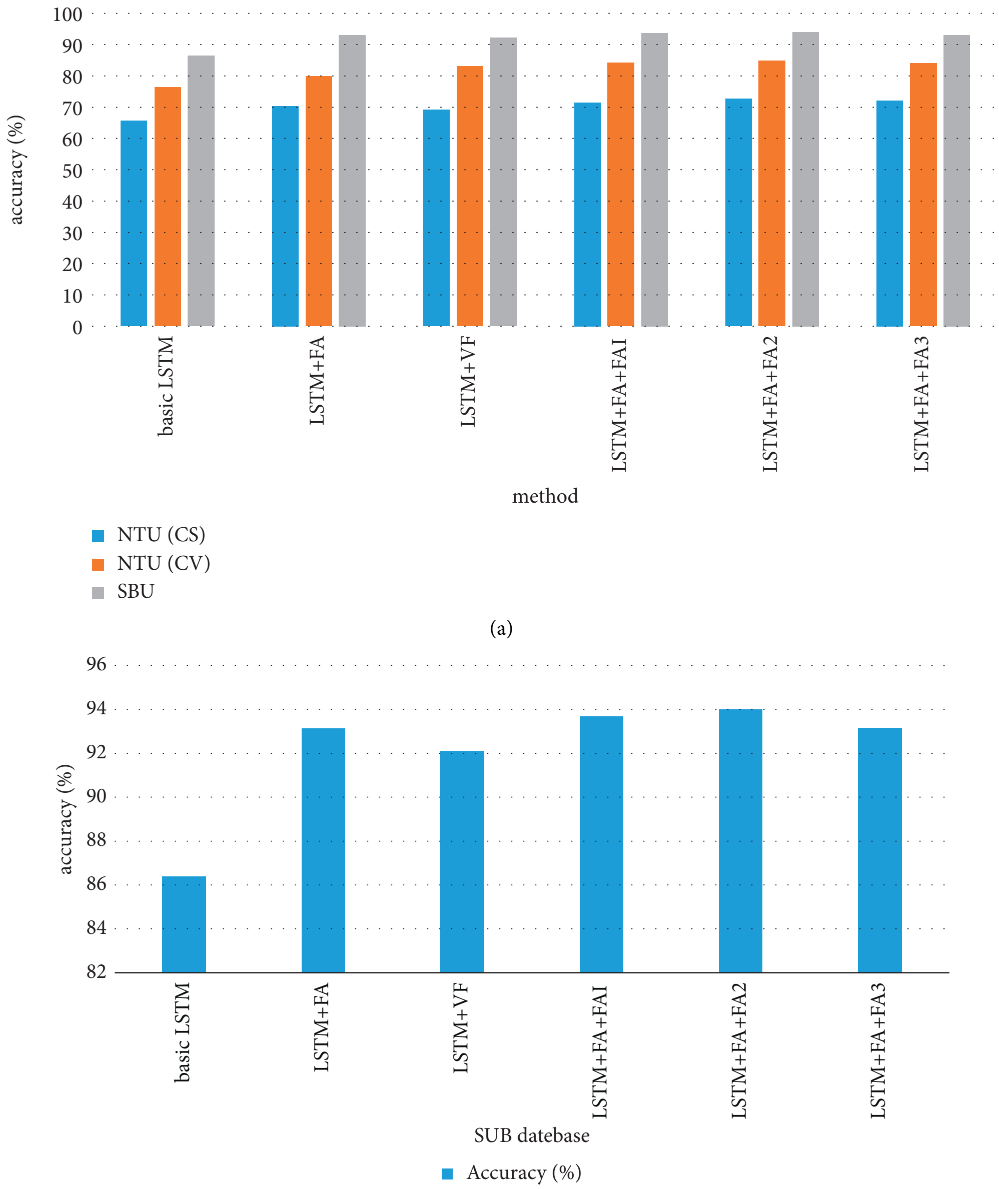
Figure 8 shows that processing the features in LSTM with attention improves the performance of action recognition. Moreover, processing with multilayer attention mechanisms in the LSTM network can further improve the performance. However, when stacking the three-layer attention mechanism, the performance improvement is not obvious, and there is a large drop in accuracy on the smaller database such as SBU. Through analysis, it is believed that this is because the multilayer attention operation greatly increases the complexity of the model, thereby increasing the risk of overfitting in model training, which is reflected in the sharp drop in the accuracy of SBU. The results showed that the application of multilayer feature attention mechanism in LSTM can effectively improve the performance of action

Table 2: Experimental results of multiview fusion method on NTU RGB + D database.

Method	CS (%)	CV (%)
Basic LSTM	65.67	76.36
LSTM + VF (ave)	66.27	81.5
LSTM + VF (tanh)	67.15	80.83
LSTM + VF (softmax)	69.12	83.08

Table 3: Experimental results of multiview fusion method on SBU database.

Method	Accuracy (%)
Basic LSTM	86.4
LSTM + VF (ave)	90.25
LSTM + VF (tanh)	90.64
LSTM + VF (softmax)	92.12



(a)

(b)

Figure 9: Comparison results with other state-of-the-art methods on NTU RGB + D and SBU databases.

recognition, but the superposition of too many layers of attention mechanism will also bring the risk of model overfitting, so it needs to be handled with caution in use.

Tables 2 and 3 indicate the experimental results of different observation fusion methods. In the experiment, three fusion strategies, average, attention weight, and attention weight were used to fuse the processing results of multiple observations to obtain the final result. For the NTU RGB + D database, three observation angles [0°, ±60°] were used in the experiment. For the SBU Kinect Interaction database, three observation angles [0°, ±90°] are used.

The experimental results indicate that multiview observation fusion is of great help to improve the performance of action recognition. Even simply averaging the processing results of multiple time observations can significantly improve the recognition accuracy. The application of the attention mechanism in the fusion process has brought further significant improvements. Experimental comparisons also show that generating weights by Softmax performs better in fusion tasks than simply generating weights with tanh activation. Overall, this experiment shows that the multiview observation fusion method proposed in this chapter is effective in 3D action recognition. It is worth mentioning that the NTU RGB + D database contains data collected from multiple perspectives, while all the data in the SBU Kinect Interaction database are collected from the same perspective. In experiments, the multiview observation fusion method achieves good results on both databases. This shows that fusing observations from different perspectives is a general 3D action recognition improvement method. The final model fuses the multilayer feature attention mechanism in LSTM and the multiview observation fusion method to form an end-to-end network model. Experiments on the integration of methods are carried out on two databases, NTU RGB + D and SBU Kinect Interaction.

The relevant experimental results on the two databases of NTU RGB + D and SBU Kinect Interaction are shown in Figure 9. Experiments show that adding 1 or 2 layers of feature attention mechanism enhancement to the multiview observation fusion network can increase the accuracy by 1% to 3%. However, when the three-layer feature attention mechanism is added to enhance, the accuracy rate will drop. In conclusion, the combination of multiview observation fusion and feature attention mechanism method allows boosting the behavior of the model further in action recognition problems. According to the experimental results and in the final model, two layers of feature focus operations are used.

5. Discussion

The development of deep learning has solved many common problems that highly rely on feature design in computer vision. Deep networks automatically learn feature extraction under data drive and can easily obtain more powerful features compared with the traditional tedious and manually designed features. Although deep learning-based methods have achieved far more results than traditional methods in action recognition, the current theoretical research of deep learning is imperfect, and its internal information is more like a black box.

Deep learning method is used to identify human movements. In action recognition, most deep learning methods directly input images, and through data-driven methods, human action recognition is regarded as a simple classification task. Through the whole volume neural network about human action recognition, put forward based on 3D attitude human action recognition situation, through the attention mechanism of multiple observation fusion action recognition model in different environments. By identifying the human skeleton data effective information enhancement and perspective transformation, the recognition method improvement integrated into an end-to-end action recognition network.

6. Conclusion

Validated by demonstrating the rationality and effectiveness of both CS and CV algorithms on NTU RGB + D and SBU datasets. First, the accuracy of the new attention mechanism module and the new mechanism are compared, and the importance of the feedback mechanism in the recognition of human action features in multiview reobservation fusion of 3D poses is verified. Finally, the model is validated on the NTU-RGB + D Cross-Object (CS) and Cross-Perspective (CV) datasets, respectively, proving that the network model is more accurate in performance and recognition than other mainstream human action features. The rate has been improved algorithm.

Data Availability

The data that support the findings of this study can be obtained from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Acknowledgments

This work was supported by the Independent Innovation Capability Improvement Program of Huizhou University (no. HZU202054).

References

[1] J. H. Thrall, X. Li, Q. Li et al., "Artificial intelligence and machine learning in radiology: opportunities, challenges, pitfalls, and criteria for success," *Journal of the American College of Radiology*, vol. 15, no. 3, pp. 504–508, 2018.

[2] P. Mamoshina, L. Ojomoko, Y. Yanovich et al., "Converging blockchain and next-generation artificial intelligence technologies to decentralize and accelerate biomedical research and healthcare," *Oncotarget*, vol. 9, no. 5, pp. 5665–5690, 2018.

[3] Y. G. Jiang, Z. Wu, J. Wang, X. Xue, and S. F. Chang, "Exploiting feature and class relationships in video categorization with regularized deep neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 2, pp. 352–364, 2018.

[4] J. Cheng, P. S. Wang, G. Li, Q. Hu, and H. Lu, "Recent advances in efficient computation of deep convolutional neural networks," *Frontiers of Information Technology&Electronic Engineering*, vol. 19, no. 1, pp. 64–77, 2018.

[5] Y. Dong and H. Wang, "Robust output feedback stabilization for uncertain discrete-time stochastic neural networks with time-varying delay," *Neural Processing Letters*, vol. 51, no. 1, pp. 83–103, 2020.

[6] A. Lucas, M. Iliadis, R. Molina, and A. K. Katsaggelos, "Using deep neural networks for inverse problems in imaging: beyond analytical methods," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 20–36, 2018.

[7] P. Yin, S. Zhang, J. Lyu, S. Osher, Y. Qi, and J. Xin, "BinaryRelax: a relaxation approach for training deep neural networks with quantized weights," *SIAM Journal on Imaging Sciences*, vol. 11, no. 4, pp. 2205–2223, 2018.

[8] A. Sarabu and A. K. Santra, "Human action recognition in videos using convolution long short-term memory network with spatio-temporal networks," *Emerging Science Journal*, vol. 5, no. 1, pp. 25–33, 2021.

[9] Y. Liu, R. Ma, H. Li, C. Wang, and Y. Tao, "RGB-D human action recognition of deep feature enhancement and fusion using two-stream ConvNet," *Journal of Sensors*, vol. 2021, no. 1, Article ID 8864870, 10 pages, 2021.

[10] M. Taddeo and L. Floridi, "Regulate artificial intelligence to avert cyber Ar ms race," *Nature*, vol. 556, no. 7701, pp. 296–298, 2018.

[11] S. Wan, L. Qi, X. Xu, C. Tong, and Z. Gu, "Deep learning models for real-time human activity recognition with

smartphones," *Mobile Networks and Applications*, vol. 25, no. 2, pp. 743–755, 2019.

[12] H. Wei and N. Kehtarnavaz, "Semi-supervised Faster Rcnn-Based Person Detection and Load Classification for Far Field Video Surveillance," *Computer Vision*, vol. 1, no. 3, pp. 756–767, 2019.

[13] V. Dignum, "Ethics in artificial intelligence: introduction to the special issue," *Ethics and Information Technology*, vol. 20, no. 1, pp. 1–3, 2018.

[14] S. Ding, S. Qu, Y. Xi, and S. Wan, "Stimulus-driven and concept-driven analysis for image caption generation," *Neurocomputing*, vol. 398, pp. 520–530, 2020.

[15] S. Zhou, M. Ke, and P. Luo, "Multi-camera transfer GAN for person re-identification," *Journal of Visual Communication and Image Representation*, vol. 59, pp. 393–400, 2019.

[16] X. M. Zhang and Q. L. Han, "State estimation for static neural networks with time-varying delays based on an improved reciprocally convex inequality," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 4, pp. 1376–1381, 2018.

[17] Y. Guo, "Globally robust stability analysis for stochastic cohen–grossberg neural networks with impulse control and time-varying delays," *Ukrainian Mathematical Journal*, vol. 69, no. 8, pp. 1220–1233, 2018.

[18] C. Sánchez-Sánchez and D. Izzo, "Real-time optimal control via Deep Neural Networks: study on landing problems," *Journal of Guidance, Control, and Dynamics*, vol. 41, no. 5, pp. 1122–1135, 2018.

[19] N. Liu, J. Han, T. Liu, and X. Li, "Learning to predict eye fixations via multiresolution convolutional neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 2, pp. 392–404, 2018.

[20] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "Model compression and acceleration for deep neural networks: the principles, progress, and challenges," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 126–136, 2018.

[21] H. Wei and N. Kehtarnavaz, "Determining Number of Speakers from Single Microphone Speech Signals by Multi-Label Convolutional Neural Network," in *Proceedings of the IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*, Washington, DC, USA, October 2018.

[22] Z. Lv, S. Zhang, and W. Xiu, "Solving the security problem of intelligent transportation system with deep learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4281–4290, 2021.

[23] W. Wei, Y. Sheng, J. Wang, and I. D. S. Hast, "Learning hierarchical spatial-temporal features using deep neural networks to improve intrusion detection," *IEEE Access*, vol. 6, no. 99, pp. 1792–1806, 2018.

[24] P. Cao, W. Xia, M. Ye, J. Zhang, and J. Zhou, "Radar-ID: human identification based on radar micro-Doppler signatures using deep convolutional neural networks," *IET Radar, Sonar & Navigation*, vol. 12, no. 7, pp. 729–734, 2018.

[25] E. Nabiel Al-Khanak, S. Peck-Lee, S. Ur-Rehman-Khan et al., "A heuristics-based cost model for scientific workflow scheduling in cloud," *Computers, Materials & Continua*, vol. 67, no. 3, pp. 3265–3282, 2021.

[26] S. Sengan, P. Vidya-Sagar, O. Ibrahim Khalaf, and R. Dhanapal, "The optimization of reconfigured real-time datasets for improving classification performance of machine learning algorithms," *Mathematics in Engineering, Science and Aerospace (MESA)*, vol. 12, 1 page, 2021.

[27] O. Öztimur Karadağ, "An adversarial framework for open-set human action recognition using skeleton data," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 29, no. 2, pp. 717–729, 2021.

[28] M. Attique-Khan, M. Alhaisoni, A. Armghan et al., "Video analytics framework for human action recognition," *Computers, Materials & Continua*, vol. 68, no. 3, pp. 3841–3859, 2021.

[29] L. Xia and Z. Li, "A new method of abnormal behavior detection using LSTM network with temporal attention mechanism," *The Journal of Supercomputing*, vol. 77, no. 4, pp. 3223–3241, 2021.

[30] S. Zhang, W. Tan, Q. Wang, and N. Wang, "A new method of online extreme learning machine based on hybrid kernel function," *Neural Computing & Applications*, vol. 31, no. 9, pp. 4629–4638, 2019.

[31] C. H. Chen, S. Fangying, F. J. Hwang, and L. Wu, "A probability density function generator based on neural networks," *Physica A: Statistical Mechanics and Its Applications*, vol. 541, 2020.

[32] J. Dai, H. Song, and G. Sheng, "Prediction method for power transformer running state based on LSTM network," *Gaodianya Jishu/High Voltage Engineering*, vol. 44, no. 4, pp. 1099–1106, 2018.

[33] H. Zhao, S. Sun, and B. Jin, "Sequential fault diagnosis based on LSTM neural network," *IEEE Access*, vol. 6, no. 99, pp. 12929–12939, 2018.

[34] S. Gai, X. Zeng, and T. Yuan, "Parking volume forecast of railway station garages based on passenger behaviour analysis using the LSTM network," *Journal of Advanced Transportation*, vol. 2021, no. 722, Article ID 6688609, 14 pages, 2021.

[35] J. Venskus, P. Treigys, and J. Markevičiūtė, "Unsupervised marine vessel trajectory prediction using LSTM network and wild bootstrapping techniques," *Nonlinear Analysis Modelling and Control*, vol. 26, no. 4, pp. 718–737, 2021.

[36] D. Gupta, V. Kumar, I. Ayus, M. Vasudevan, and N. Natarajan, "Short-term prediction of wind power density using convolutional LSTM network," *FME Transactions*, vol. 49, no. 3, pp. 653–663, 2021.