Hindawi
Scientific Programming
Volume 2022, Article ID 5747647, 10 pages
https://doi.org/10.1155/2022/5747647

Research Article

Design of National Sports Action Feature Extraction System Based on Convolutional Neural Network

Yajun Pang [1,2]

[1]*College of Physical Education, Luoyang Institute of Science and Technology, Luoyang, Henan 471023, China*
[2]*Henan Province Engineering Research Center of Industrial Intelligent Vision, Luoyang, Henan 471023, China*

Correspondence should be addressed to Yajun Pang; yajun.pang@lit.edu.cn

Received 2 December 2021; Accepted 27 December 2021; Published 25 February 2022

Academic Editor: Baiyuan Ding

Human action recognition is one of the hotspots in computer vision research. Its purpose is to detect and recognize target actions from videos, so that computer systems can understand human actions, and thus it has great research significance. Based on the action features of famous sports, this paper proposes an action recognition scheme based on RGB-D video compression to establish action features and deep learning as a means of recognition. By establishing the connection between the bone data of the three-dimensional data type and the depth image data, the depth sequence is analyzed and expressed as a three-level structure diagram sequence, which is the overall figure sequence, partial figure sequence, and joint point figure sequence, and then passes through the two-way pool. The sorting algorithm extracts the action features in the three picture sequences and compresses and generates three types of structured images of the corresponding picture sequences, and these three types of structured images are used as the feature expression of the video. When constructing a three-level structure diagram sequence, the innovation of this paper is to splice the extracted key unit image blocks to obtain a three-level structured moving image based on the three-key unit splicing, so that the image is not only retained. In addition to time-space information, the structure information of the depth image is also strengthened, and the amount of calculation is reduced at the same time. Finally, the three types of structured images are input into the convolutional neural network, respectively, and the judgment and recognition results obtained are multiplicatively fused to obtain the final recognition rate of the action.

1. Introduction

In human-centered computer vision research (such as human detection, tracking, human posture estimation, and human motion recognition), human motion recognition is widely used; for example, video surveillance, human-machine interface, home assistance, human-machine interaction, and intelligent driving have become an important research direction in computer vision research [1–5].

According to the complexity and duration of the action, action recognition can be roughly divided into four types: gesture recognition, action recognition, interaction recognition, and group activity recognition. Specifically, gesture recognition is defined as expressing people's thoughts, opinions, and emotions through the basic movements or positions of hands, arms, human body, or head: Typically, "waves" and "nods." The posture duration is relatively short and the complexity is low. Action is defined as an activity completed by a single human body mobilizing multiple parts of the body; that is to say, an action is a combination of multiple postures, such as "walking" and "boxing." The interactive action is mainly completed by two subjects: people and things, and people and people [6–11]. This means that interaction expresses the interaction of people or characters, such as "hug" and "playing the guitar." Group activities are the most complicated type of action. It may combine three types of actions: posture, action, and interaction, involving two or more people or objects, such as "two teams play basketball" and " group meeting," shown in Figure 1.

The realization process of action recognition is generally divided into three stages: first, target detection, then feature extraction, and finally feature analysis and judgment and recognition. There are corresponding studies for each stage, and the goal is to achieve efficient action recognition. Moving target detection is not the focus of this

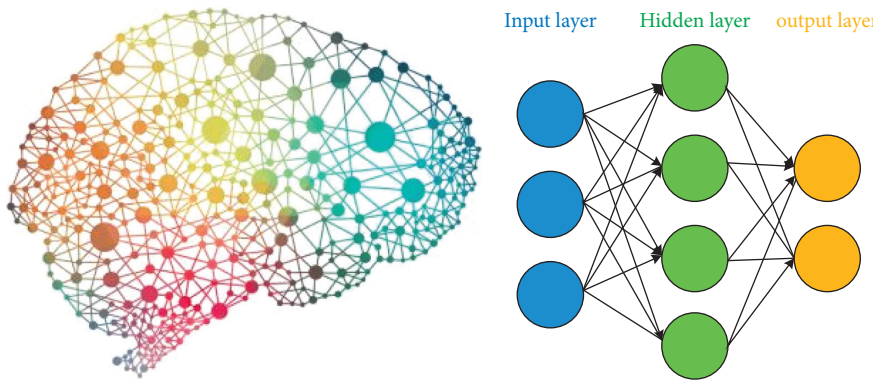

Figure 1: Neural networks.

article, so it will be not repeated here. In the feature extraction stage, the traditional method is to manually select features. However, in most cases, specific algorithm analysis is performed on the data characteristics. The generalization ability is poor, and there are often situations where the effect of different datasets is quite different. The complexity of the processing process is on different datasets. Moreover, with the advent of the era of big data, the development of datasets is gradually moving towards larger data volumes, more data categories, and wider data ranges, making it more difficult to analyze data and extract features. As it becomes more difficult to determine the features to be extracted, the step of feature extraction gradually abandons specific customization and tends to be general. People are more hopeful that the action features in the video can be extracted through nonmanual methods, which reduces the difficulty of qualitative extraction of features in the early stage and improves the efficiency of large-scale action data processing. With the deepening of research, researchers use models and algorithms to characterize features, such as image-based representation, model-based representation, time-space-based representation, and so on. In the last stage, when analyzing data features for classification, a classifier is needed for identification; that is, the extracted features are used as an abstract representation of the original input image and input into the classifier. By using the model training parameters in the classifier, a similar match with the input data is found. The main purpose of human action recognition is that, for the test data and the label data predicted by the classification model obtained through training, the higher the fit with the actual label data, the better. With the evolution of features from simple to complex, the classifier is also a process from simple to complex, from the linear binary classifier at the beginning to the logistic regression classifier, as well as a series of traditional excellent learning algorithms such as SVM and HMM. It is a typical classifier model. However, the common feature of these classifiers is that different types of features need to be matched with different classifiers, which greatly reduces the computational efficiency. Deep learning is a branch of traditional machine learning that can make up for the shortcomings of the above algorithms and is a popular algorithm among recent classification algorithms [12, 13].

Deep learning is an important branch of traditional machine learning. With the development of hardware equipment and big data, deep learning has gradually become one of the hot points in the research of computer vision research. Deep learning builds a hierarchical learning and training model and establishes a progressive learning mechanism between input and output data. With each step forward, the extracted feature dimension rises by one level, so that the final trained model can extract the original. The high-dimensional features of the image are conducive to the final recognition and judgment. Because deep learning has the characteristics of autonomous learning and does not require artificial design of related algorithms, it is a more efficient and generalized feature extraction method. So far, deep learning has good experimental results in the fields of target detection, target recognition, image processing, and so on. One of the more prominent advantages of deep learning is that its processing speed for big data is much faster than traditional hand-craft features. With the rapid development of the Internet, deep learning is more in line with the characteristics of the times and can play a role from massive amounts of data. Information makes the future development of deep learning very impressive [14–17].

In recent years, the development of the field of deep learning has provided new methods for the judgment and recognition of human actions in the later stages. Deep learning algorithms can extract higher-level action features and give better classification results. The system that applies deep learning for action recognition has achieved a high recognition rate. The good classification effect of deep learning has been recognized, and it has been practically applied in driverless cars, medical image recognition, and image retrieval. Convolutional neural networks are a common way to learn high-level features in deep learning. G.W. Taylor et al. proposed an action recognition framework based on convolutional neural networks, which directly process continuous image sequences and use convolutional neural networks to learn color texture maps. S. Ji et al. used a three-dimensional convolutional neural network to extract human action information that contains both time and space features. A. Karpathy et al. conducted a comparative study on a variety of convolutional neural networks in the time domain and provided an empirical evaluation of convolutional neural networks in high-

resolution video classification. At the same time, the article also pointed out that the use of convolutional neural networks to process each frame in the video sequence alone has almost the same result as processing a series of frames at the same time; that is, the convolutional neural network for comparative research in the article learns the characteristics of time and space [18–22]. The above is not well integrated. In order to better mine the spatial and temporal features in the action sequence, K. Simonyan et al. used convolutional neural networks to process spatial data streams and temporal data streams separately and finally used specific methods to perform learning results on different features [23–28]; see Figure 2.

In order to solve the above problems, this paper proposes a novel action recognition framework, which compresses the action sequence based on depth information, effectively compresses a video sequence into several pictures, and finally uses the convolutional neural network to improve the picture and learning and classification capabilities to complete the learning and classification of action sequences. This paper applies sorting pooling to the depth image sequence to obtain three levels of dynamic images, compress the depth video sequence, and then use three convolutional neural networks to judge and recognize the three types of images, respectively, and finally merge the results to obtain best effect. In the research, we introduced AlexNet, which has a good effect on human action recognition in deep learning and improved the accuracy of action recognition. Compared with other algorithms, the recognition effect and robustness of this method are far stronger than other methods, and the trained network can allow anyone to control the UAV by gestures according to regulations, which has good universality [29–33].

2. Convolutional Neural Network

One of the most common algorithms for image recognition is a model built based on a convolutional neural network (CNN), which can be regarded as a forward feedback neural network that includes convolution operations. After layers of convolution and pooling operations, the model can gradually extract the features of the target image, and the proposed features have translation invariance. The biggest feature of convolutional neural network is weight sharing and local perception, so when training, it needs to learn a huge number of feature parameters. What needs to be explained here is that each extraction of feature values requires multiple core convolution kernels. Different cores correspond to different features, as shown in Figure 3. At the same time, its corresponding two-dimensional spatial characteristics help it in computer vision tasks and demonstrate a good characterization ability on the problem. Therefore, the classification model based on CNN mainly includes the following modules.

2.1. Convolutional Layer. After inputting the current data features in the convolutional layer, by setting the appropriate number of convolutions, the size of the convolution kernel, and the convolution method, the feature expression of the input data is completed. After completing this operation, you can multiply the data and the convolution kernel to get each feature point on the corresponding feature map. Traverse all the data according to the convolution step length, and realize the convolution between the convolution kernel parameters and the input data in the receptive field. The process can be expressed as follows:

$$\begin{aligned}\mathbf{Z}^{l+1}(i,j) &= \left[\mathbf{Z}^{l} \otimes \mathbf{w}^{l+1}\right](i,j) + b \\ &= \sum_{k=1}^{K_l} \sum_{x=1}^{f} \sum_{y=1}^{f} \left[\mathbf{Z}_k^{l}(s_0 i + x, s_0 j + y)\mathbf{w}_k^{l+1}(x,y)\right] + b, \\ (i,j) \in \{0,1,\ldots,L_{l+1}\}, L_{l+1} &= \frac{(L_l + 2p - f)}{s_0} + 1,\end{aligned} \tag{1}$$

where b represents the corresponding deviation, $Z^{l+1}|_{,}Z^{l}$ are, respectively, used to represent the output and input values of the neural network. L_{l+1} represents the feature size of the output image, $Z(i,j)$ represents the feature value, and K represents the number of convolution kernels, that is, the number of channels; f, s_0, p represent the number of convolution operations involved in this operation, convolution kernel size.

When the convolution step size and the size of the convolution kernel are both 1, it is called unit convolution, and the convolutional layer composed of these unit convolutions is called the net in the net. In the case of unchanged input features, unit convolution can perform feature fusion on multiple channels, which helps to reduce the number of corresponding model parameters, thereby reducing a certain amount of calculation.

The convolutional layer has two characteristics of local perception and weight sharing at the same time. The neurons in the front and back layers are connected in pairs. For example, in a 1000*1000 image, the input layer has 106 nodes. Therefore, it needs to learn a huge number of feature parameters during the training process. The feature value is extracted when the convolution operation slides in the previous layer of the convolution kernel, and the weight of the core parameter needs to be kept unchanged during each operation, and it is only at the same time. Establish a connection with a part of the neurons in the previous layer. It can be seen that the corresponding local perception not only is more in line with human's cognitive characteristics of things from part to the whole, but also greatly reduces the number of characteristic parameters. The process of applying the learning information in a certain local area to other places is called weight sharing. In simple terms, the whole image is filtered on the image, which is also conducive to greatly reducing the parameters.

Each feature extraction needs to use multiple convolution kernels, the features obtained by different convolution kernel extraction are different, and each convolution kernel will extract a certain aspect of the feature. At the same time, its two-dimensional spatial characteristics help it have good characterization capabilities for computer vision tasks.

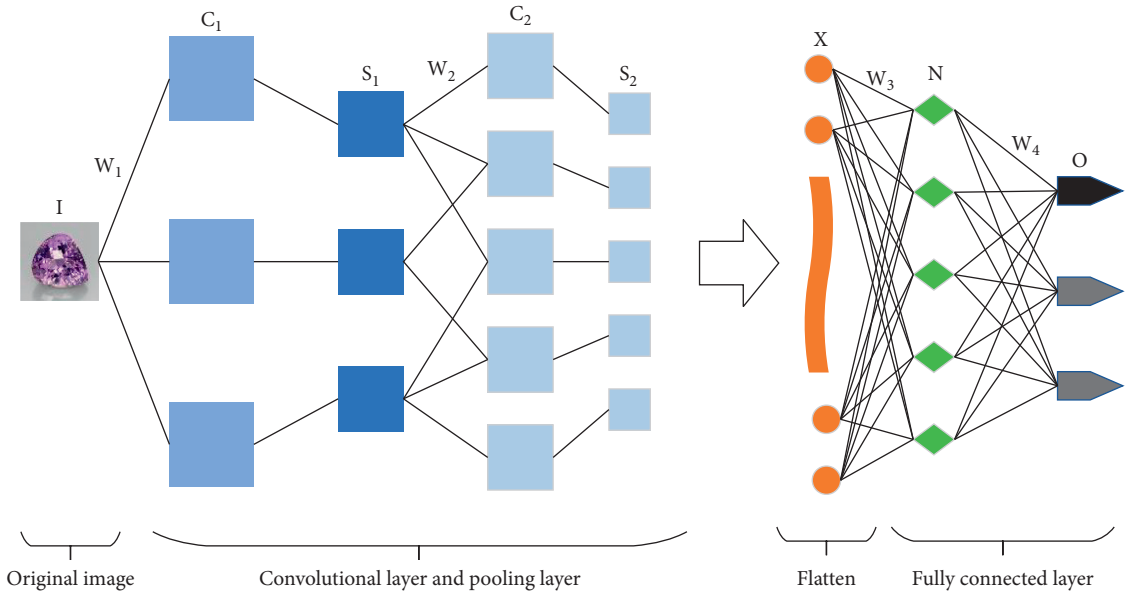

Figure 2: Convolutional neural network structure.

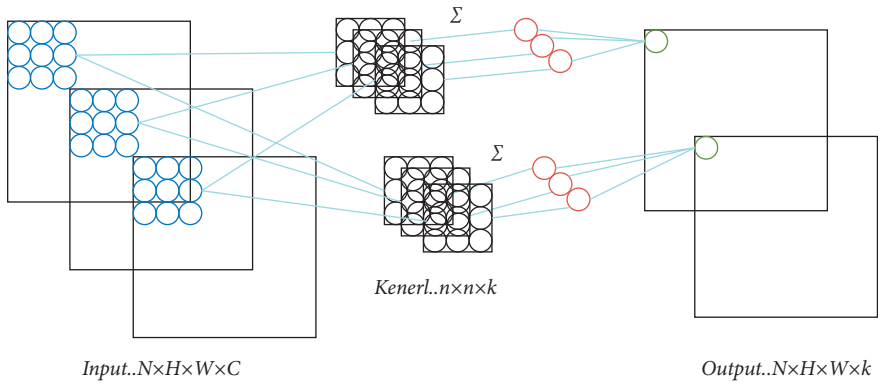

Figure 3: Process of convolutional neural network structure.

2.2. Pooling Layer. After the data input of the previous layer is completed, it is passed to the pooling layer. In this layer, the data filtering of the previous layer is mainly completed. The standard for filtering is the data extraction standard of the convolutional layer. This process can be understood, in order to imitate the human visual system to complete the dimensionality reduction of the data and finally obtain the image representation features of higher-level features. Pooling helps to reduce the redundancy of information, and at the same time it can improve the invariance of the model scale, thereby helping to avoid the adverse effects of the model due to overfitting. In general, the pooling layer can be further divided into mean pooling and maximum pooling. The advantage of the latter is that it can learn the edge and texture structure of the image. The advantage of the former is that it can effectively reduce the deviation of the estimated mean value and improve the anti-interference ability of the established model. The formula of the pooling operation can be expressed as

$$\mathbf{A}_k^l(i,j) = \left[\sum_{x=1}^{f}\sum_{y=1}^{f}\left[\mathbf{A}_k^l(i,j)(s_0 i + x, s_0 j + y)^p\right]\right]^{\frac{1}{p}}. \quad (2)$$

In the above formula, s_0 represents the pooling step size, when $p = 1$ is the average pooling; when p tends to infinity, it is the maximum pooling.

2.3. Fully Connected Layer. The fully connected layer is composed of several neurons "holding hands" with each other, and each neuron in the latter part is interconnected with the neuron in the former part. This is equivalent to the aforementioned forward feedback network, as the end layer of the convolution model. It is the integration of the features provided by the previous convolution and pooling operations. The feature map is expanded into a one-dimensional column vector and used as the input corresponding to the fully connected layer. It is easy to realize the nonlinearity of the input feature by using the activation function and finally

extract feature vectors with more expressive ability and then realize feature classification.

All in all, the structure corresponding to a typical convolutional neural network used to process image tasks can usually be expressed as follows.

Because convolutional neural networks have certain advantages in processing images, they are generally regarded as a commonly used method in the field of image recognition. In the traditional network structure, there are many classic structures, such as AlexNet network, GoogleNet network, VGGNet network, and ResNet network.

2.3.1. Gradient Descent. The standard gradient descent can be described as

$$\theta = \theta - \eta \cdot \nabla_\theta J(\theta). \tag{3}$$

The standard gradient descent refers to the replacement of the secondary parameters of the gradient of the overall calculation example. This standard gradient descent method has certain drawbacks such as relatively slow calculation speed and certain application limitations.

2.3.2. Stochastic Gradient Descent (SGD). Different from calculating the gradient after calculating the loss of all samples in the standard gradient descent case, SGD calculates the gradient once for each sample and updates the parameters. It can be described as

$$\theta = \theta - \eta \cdot \nabla_\theta J\left(\theta; x^{(i)}; y^{(i)}\right). \tag{4}$$

2.3.3. Minibatch Gradient Descent (MBGD). This method is a relative compromise between batch stochastic gradient descent and gradient descent. Among them, the main idea of the minibatch gradient descent method is as follows: based on a dataset of n training samples, update the corresponding parameters in real time and select a minibatch data sample of size m ($m < \mathrm{n}$) to calculate its corresponding. The gradient of the formula is as follows:

$$\theta = \theta - \eta \cdot \nabla_\theta J\left(\theta; x^{(i;i+n)}; y^{(i;i+n)}\right). \tag{5}$$

AdaGrad is a method of adaptive learning rate, which implements a high learning rate for low-frequency parameters and a low learning rate for high-frequency parameters. This feature makes it more suitable for processing sparse data. Standard parameters are a very small constant, generally 10e-8, which represents the global learning rate, and usually, the final gradient accumulation variable r.

$$\begin{aligned} g &\leftarrow \nabla_\theta J(\theta), \\ r &\leftarrow r + g^2, \\ \Delta\theta &\leftarrow \frac{\delta}{\sqrt{r+\epsilon}} \cdot g. \end{aligned} \tag{6}$$

3. Action Feature Model Algorithm Based on Convolutional Neural Network

In order to simplify the processing of video sequences and make the computer's judgment and recognition of the types of human actions more efficient, a method of compressing the depth of human actions is proposed, shown in Figure 4. This method considers the spatial and temporal characteristics of the video at the same time, which can save the video action characteristics and reduce the information redundancy. The algorithm includes the following steps: remove the background interference in the depth video, leaving only the shape of the human body. After removing the background interference from a k-frame depth video sequence, it is expressed as

$$\langle d_1, d_2, ..., d_t, d_k \rangle, \tag{7}$$

where dt represents the average value of the depth features of all frames as of t time. The deep depth image is shown in Figure 5.

Rank pooling processing for depth videos: At each time t, define a score value, and the score value must meet the conditions; the more the current frame number, the greater the score value.

Perform rank pooling on deep video sequences to extract features. The process of rank pooling is to find an optimal solution that satisfies the following objective function:

$$\begin{aligned} &\arg\min \frac{1}{2}\|\omega\|^2 + \lambda \sum_{i>j} \xi_{ij}, \\ &s.t. \omega^T\left(d_i - d_j\right) \geq 1 - \xi_{ij}, \xi_{ij} \geq 0. \end{aligned} \tag{8}$$

In the depth image, each joint point is used as the center point for extension, and the depth image is cropped using a frame of size q∗p to obtain the image block of the joint point. In order to make the extracted image blocks have the same scale feature, we find the maximum range of motion for the same unit in a video frame sequence and define it as mask. For each unit, use 0 to fill the unit to the mask size. In this way, the relative scale between the units can be maintained, which is very necessary for the preservation of spatial information, as shown in Figure 6.

Convolutional neural network has an excellent effect on image and speech processing. By using a single neuron to respond to the coverage and surrounding pixels, convolutional neural networks have unique advantages for large-scale image processing. The structural design of the weight sharing of the convolutional neural network is more like the principle of the human neural network. Through the pooling layer, the weight sharing fully linked layer design greatly reduces the parameters of the entire neural network. The design uses multiple convolutional templates for feature extraction, which is far better than the existing machine learning methods in terms of image classification and recognition. Deep learning through convolutional neural networks can automatically extract useful features from images and use these features to

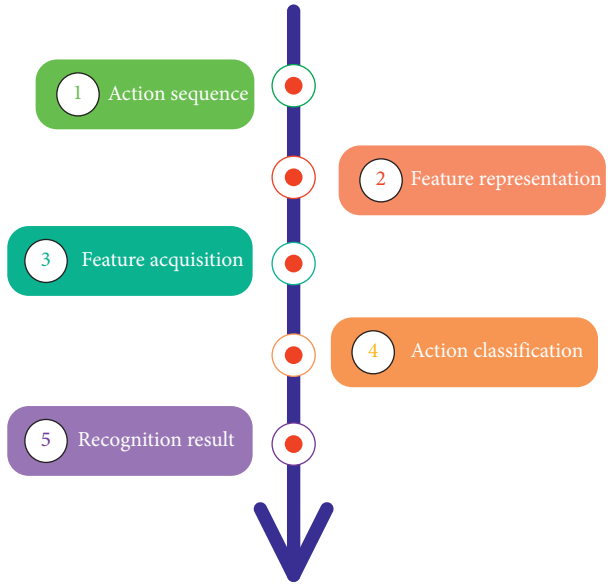

Figure 4: Action recognition framework flowchart.

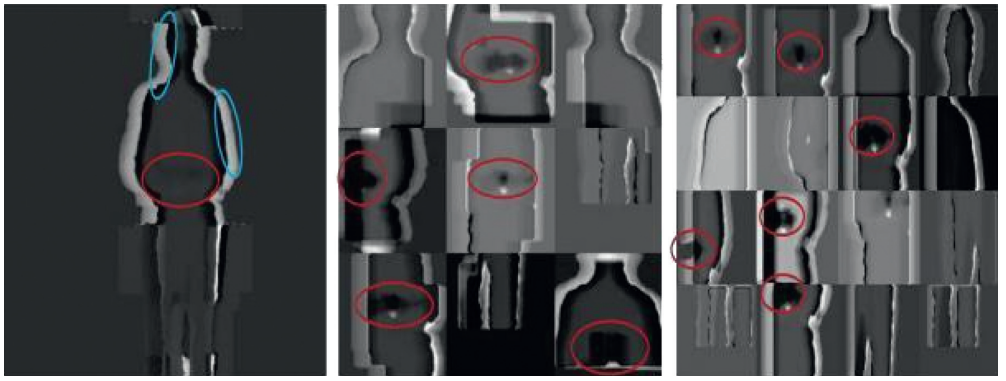

Figure 5: Deep depth image.

classify images. The level of perfection of learning features exceeds many existing methods of manually specifying features.

In order to achieve a good classification effect, we adopted the AlexNet network structure, which has achieved remarkable results on the ImageNet dataset, as the neural network we use to classify gestures. The network structure has 5 convolutional layers and 3 full link layers.

There is the following relationship between every two convolutional layers of the network structure, as shown in Figure 7.

Among the three fully connected layers in AlexNet, each fully connected layer contains 4096 neurons. Such a network maximizes the multiclass logistic regression objective; that is, it maximizes the average log probability of the correct label in the training sample under the prediction distribution, thereby making the classification more accurate. In order to make the convolutional neural network get good results faster, the model trained on the ImageNet of the AlexNet network is used in this article to initialize the network parameters.

3.1. Hardware Platform. The workstation for training in the experiment is equipped with an Intel E5-2300 CPU and 16 GB DDR3 memory using GPUNvidia TitanX to accelerate the training process of the neural network. The predicted result is plotted in Figure 8.

3.2. Software Platform. The deep learning platform used in the experiment is Caffe (Convolutional Architecture for Fast Feature Embedding), which is a general framework for deep learning algorithms. The framework uses many libraries that can perform fast calculations, models for fast data storage, and function templates that can be directly called, allowing developers to quickly implement the network structure envisioned. The architecture abstracts many common operations of convolutional neural networks, and they are implemented with CPU and GPU, respectively. The entire calculation can be seamlessly switched between CPU and GPU. Caffe allows users to implement convolutional neural networks only by specifying the network structure in the configuration file.

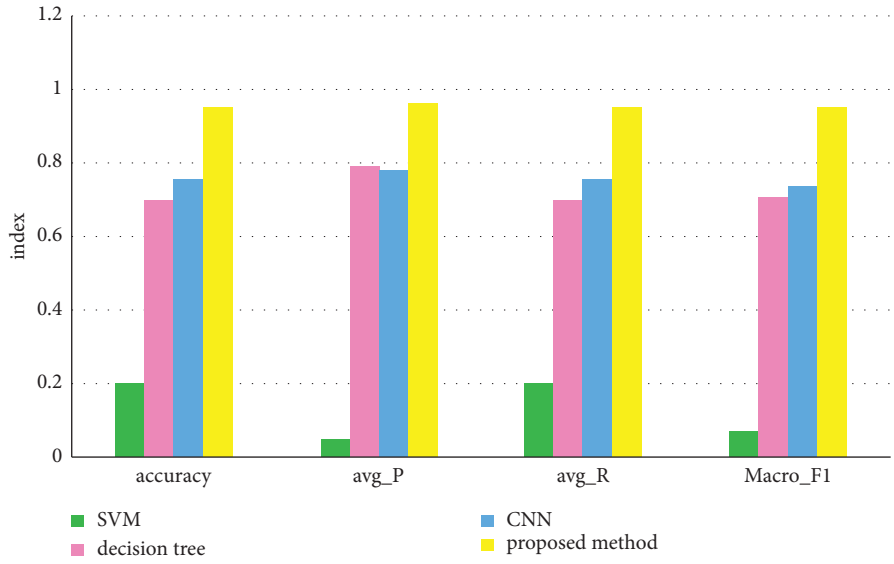

Figure 6: Statistical table of evaluation indexes for different algorithms.

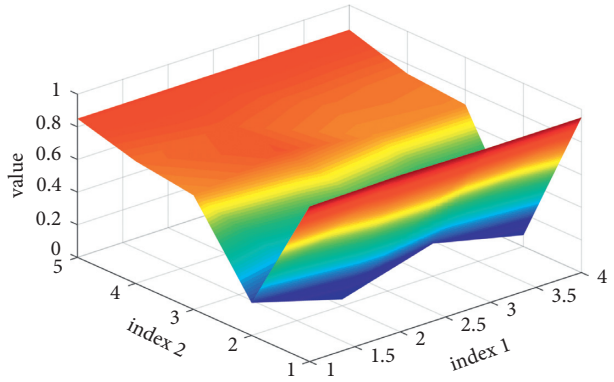

Figure 7: Value vs. index.

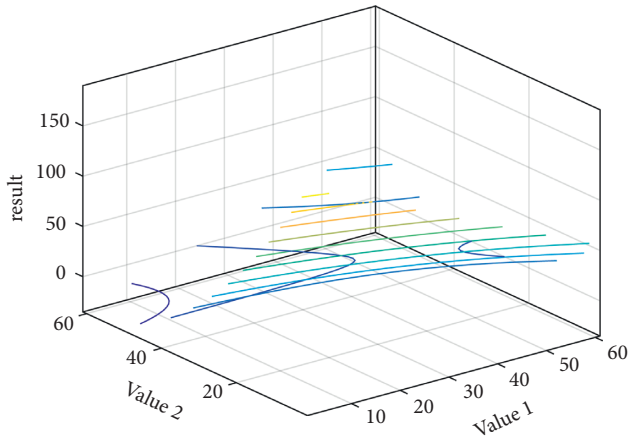

Figure 8: Predicted result.

The parameters of the network set in the experiment are as follows: The network uses 256 training images (batch size) for one iteration. According to the size of the training dataset, a total of about 90 cycles (epoch) are trained. In these 90 cycles, the initial learning rate is set to 0.001 (base learning rate), which drops to 0.0001 after 60 cycles. For the rest of the training parameters, we refer to the method proposed by A. Krizhevsky et al.

In the verification of the algorithm in this article, a total of five human motion datasets are used. They are MSR Activity 3D dataset, G3D dataset, MSR Daily Activity dataset, SYSU 3D HOI dataset, and UTD-MHAD dataset. These datasets cover most types of actions, including single actions, in-game actions, daily action actions, human and object interaction actions, and fine-grained actions. Therefore, these datasets can show that the algorithm proposed in this paper is universal.

3.2.1. MSR Action 3D Dataset. MSR Action 3D dataset contains 20 simple actions performed by 10 people facing the camera, and each person performs each action 2 to 3 times. The experiment adopts the method of cross-validation. The human movement data labeled 1,3,5,7, and 9 are trained, and the movement data labeled 2,4,6,8, and 10 are tested.

3.2.2. G3D Dataset. G3D is a game action type dataset, which is a series of action data shots for scenes in the game, including 20 game actions displayed by 10 people. The experimental verification method is to keep the first five labeled people as the training data and the last five labeled people as the test data.

3.2.3. MSR Daily Activity 3D Dataset. The MSR Daily Activity 3D dataset contains 16 actions displayed by 10 people, and each person performs each action twice, once in a standing position and once in a sitting position. Most of this dataset contains interactive actions of characters. The experiment adopts a cross-validation method, and people's movement data with labels of 1, 3, 5, 7, 9, 11, 13, and 15 are used for training, and those with labels of 2, 4, 6, 8, 10, 12, 14, and 16 action data are tested.

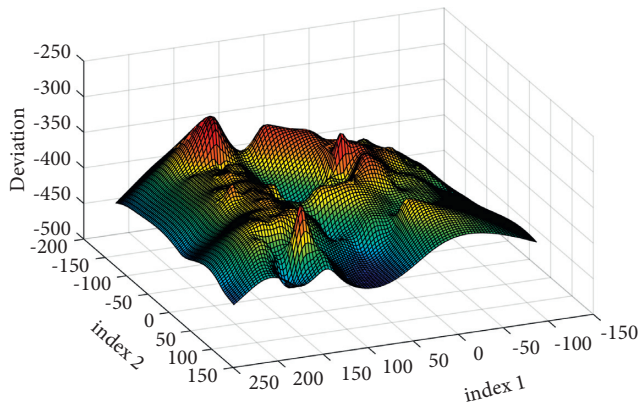

Figure 9: Deviation.

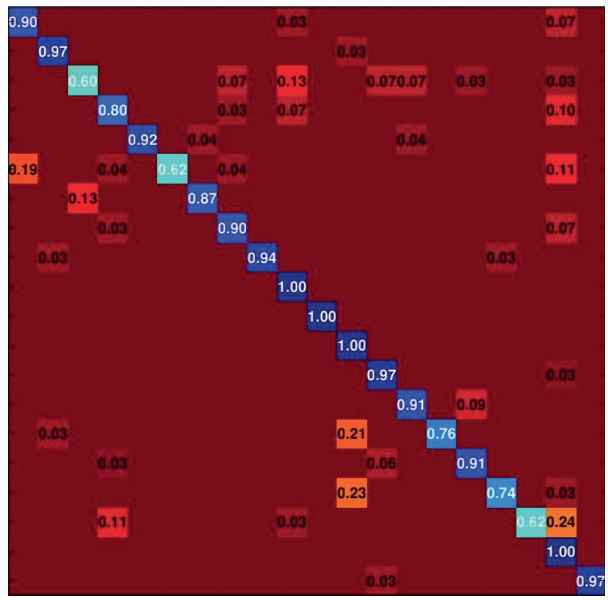

Figure 10: Matrix of MSR Action 3D.

3.2.4. SYSU 3D HOI Dataset. SYSU 3D HOI dataset was designed and collected by Sun Yat-Sen University, and it is aimed at the action types of character interaction. This dataset consists of 40 people showing 12 actions. For each action, each person interacts with the following six items: phone, chair, school bag, wallet, mop, and broom. The experiment adopts the method of cross-validation, the action data with odd numbers of people is trained, and the action data with even numbers is tested.

3.2.5. UTD-MHAD Dataset. UTD-MHAD dataset contains 27 actions displayed by 8 people, 8 people contain 4 men and 4 women, and each person performs each action 4 times. The experiment adopts the method of cross-validation. The human movement data labeled 1, 3, 5, and 7 are trained, and the movement data labeled 2, 4, 6, and 8 are tested. The deviation is plotted in Figures 9 and 10.

4. Conclusions

This paper analyzes the framework structure of action recognition and designs a new framework from two directions: feature extraction and feature classification. In terms of feature extraction, in order to retain more information about the original video, this article starts from the time-space structure and uses the hierarchical sorting and pooling dynamic depth map DDI to obtain three types of dynamic depth maps, namely, the whole body DDI, part DDI, and joint DDI; in the feature classification stage, this

article uses the AlexNet convolutional neural network provided by the deep learning platform Caffe to train the model; multiplicative fusion of the final result can get a better action recognition rate.

4.1. Bidirectional Pooling Sorting Algorithm. The pooling algorithm is based on the premise of a forward and backward time assumption. It extracts and sorts video features and compresses the sorting information on a picture, which is verified on RGB color images. It is a video compression method with excellent performance. Two-way pooling reverses this assumption of forward and backward time and trains them separately. This is quite effective for actions that are sensitive to time information and can obtain more comprehensive information. On the other hand, more data can make the model training more adequate.

4.2. Hierarchical Image Segmentation Combination. This paper proposes a simple and effective way of extracting video spatial information. The human body is divided according to joint points and combined according to the whole body, parts, and joint points. In the three types of pictures obtained in this way, the whole body DDI can provide body contour information, and the partial and joint DDI can provide detailed information, which plays a complementary role and is of great significance for the next step of recognition. And this way of segmentation is easy to understand, and it is very convenient for other researchers to reverify.

4.3. The Challenge of Human Motion Recognition. The difference between within and between classes is the same action, and the performance of different people may vary greatly. For example, because of the differences between individuals, people with different simple actions of running have different speeds and step lengths. A robust action recognition method should have good generalization; the environment here can be divided into the complex background where the action is executed and the camera environment. The environment in which the action occurs is an important differentiating factor. In a complex and chaotic background, it is difficult to accurately track and locate feature points of interest. Moreover, important parts of the human body are likely to be occluded by objects and other people. At the same time, the lighting conditions will further affect the appearance of the person's contours, which will greatly interfere with the recognition of human movements. The problem of occlusion can be alleviated by using multiple cameras to observe movement from different perspectives, but this will face the problem of synchronization and cannot achieve real time. In addition, a moving camera will increase the difficulty of positioning and tracking the human body and will also cause the ability to change the scale of the human body, which is not affected by changes between classes. At the same time, it should be able to distinguish the differences between different categories. However, as the types of actions increase, there is a certain overlap between actions, which makes the recognition more difficult.

Generally, research assumes that actions are easily segmented in the time dimension. Although this assumption reduces the segmentation burden in the recognition task, another segmentation process must be added in advance, which makes the real-time performance of the entire recognition worse. The speed of people performing actions varies greatly, and it is difficult to determine the starting point of the action, which has the greatest impact when extracting features from the video to represent the action. Therefore, a robust human action recognition method should be invariant to the execution speed of the action.

Data Availability

The dataset can be accessed upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported by Science and Technology Projects of Henan Science and Technology Department, Assessment of action in sports video based on multi-feature fusion, 182102310041, Science and Technology Projects of Henan Science and Technology Department, Assessment of Taichi Action Based on Vision Transformer, 222102320016, and Sanmenxia Vocational and Technical College Project, Assessment of action in low-quality large-displacement sports videos, SZYGCCRC-2020-005.

References

[1] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas, NV, USA, 2016.

[2] M. Yun, J. Zhao, J. Zhao, X. Weng, and X. Yang, "Impact of in-vehicle navigation information on lane-change behavior in urban expressway diverge segments," *Accident Analysis & Prevention*, vol. 106, no. 1, pp. 53–66, 2017.

[3] S. Kumar Dwivedi, R. Amin, V. Satyanarayana, and R. Chaudhry, "Blockchain-based secured event-information sharing protocol in internet of vehicles for smart cities," *Computers & Electrical Engineering*, vol. 86, no. 1, pp. 1–9, 2020.

[4] Z. Khan and S. Amin, "Bottleneck model with heterogeneous information," *Transportation Research Part B: Methodological*, vol. 112, no. 1, pp. 157–190, 2018.

[5] J. M. Cairney, K. Rajan, D. Haley et al., "Mining information from atom probe data," *Ultramicroscopy*, vol. 159, no. 1, pp. 324–337, 2020.

[6] J. Yu and P. Lu, "Learning traffic signal phase and timing information from low-sampling rate taxi GPS trajectories," *Knowledge-Based Systems*, vol. 110, no. 1, pp. 275–292, 2016.

[7] K. P. Wijayaratna, V. V Dixit, L. Denant-Boemont, and S. Travis Waller, "An experimental study of the Online Information Paradox: does en-route information improve road network performance?" *PLoS ONE*, vol. 12, no. 9, pp. 184–191, 2017.

[8] Z. Wang, H. Ren, Q. Shen, W. Sui, and X. Zhang, "Seismic performance evaluation of a steel tubular bridge pier in a five-span continuous girder bridge system," *Structures*, vol. 31, no. 1, pp. 909–920, 2021.

[9] S. Nakayama and J. Takayama, "Traffic network equilibrium model for uncertain demands," in *Proceedings of the 82nd Transportation Research Board Annual Meeting*, Washington, DC, USA, 2003.

[10] H. Shao, W. H. K. Lam, and M. L. Tam, "A reliability-based stochastic traffic assignment model for network with multiple user classes under uncertainty in demand," *Networks and Spatial Economics*, vol. 6, no. 3, pp. 173–204, 2019.

[11] A. Chen, J. Kim, S. Lee, and Y. Kim, "Stochastic multi-objective models for network design problem," *Expert Systems with Applications*, vol. 37, no. 2, pp. 1608–1619, 2020.

[12] H. Wang, W. H. K. Lam, X. Zhang, and H. Shao, "Sustainable transportation network design with stochastic demands and chance constraints," *International Journal of Sustainable Transportation*, vol. 9, no. 2, pp. 126–144, 2015.

[13] S.-M. Hosseininasab and S.-N. Shetab-Boushehri, "Integration of selecting and scheduling urban road construction projects as a time-dependent discrete network design problem," *European Journal of Operational Research*, vol. 246, no. 3, pp. 762–771, 2015.

[14] S.-M. Hosseininasab, S.-N. Shetab-Boushehri, S. R. Hejazi, and H. Karimi, "A multi-objective integrated model for selecting, scheduling, and budgeting road construction projects," *European Journal of Operational Research*, vol. 271, no. 1, pp. 262–277, 2018.

[15] M. Johnson, M. Schuster, Q. Le et al., "Google's multilingual neural machine translation system: enabling zero-shot translation," *Transactions of the Association for Computational Linguistics*, vol. 5, no. 1, pp. 339–351, 2017.

[16] M. D. Moreno, "Translation quality gained through the implementation of the iso en 17100:2015 and the usage of the blockchain," *Babel*, vol. 1, no. 2, pp. 1–9, 2020.

[17] X. Wang, X. Yu, L. Guo, F. Liu, and L. Xu, "Student performance prediction with short-term sequential campus behaviors," *Information*.vol. 11, no. 4, p. 101, 2020.

[18] Q. Guo, Z. Zhu, Q. Lu, D. Zhang, and W. Wu, "A dynamic emotional session generation model based on Seq2Seq and a dictionary-based attention mechanism," *Applied Sciences*, vol. 10, no. 6, pp. 1–10, 2020.

[19] H. Ren, Xi Mao, W. Ma, J. Wang, and L. Wang, "An English-Chinese machine translation and evaluation method for geographical names," *ISPRS International Journal of Geo-Information*, vol. 9, no. 3, pp. 193–201, 2020.

[20] J. Arús-Pous, T. Blaschke, S. Ulander, J.-L. Reymond, H. Chen, and O. Engkvist, "Exploring the GDB-13 chemical space using deep generative models," *Journal of Cheminformatics*, vol. 11, no. 1, pp. 20–29, 2019.

[21] T. Moon, T. In Ahn, and E. S. Jung, "Long short-term memory for a model-free estimation of macronutrient ion concentrations of root-zone in closed-loop soilless cultures," *Plant Methods*, vol. 15, no. 1, pp. 1–12, 2019.

[22] N. Pourdamghani and K. Knight, "Neighbors helping the poor: improving low-resource machine translation using related languages," *Machine Translation*, vol. 33, no. 3, pp. 239–258, 2019.

[23] L. Bote-Curiel, S. Muñoz-Romero, A. Gerrero-Curieses, and J. L. Rojo-Álvarez, "Deep learning and big data in healthcare: a double review for critical beginners," *Applied Sciences*, vol. 9, no. 11, pp. 1–11, 2019.

[24] J. Zhang and T. Matsumoto, "Corpus augmentation for neural machine translation with Chinese-Japanese parallel corpora," *Applied Sciences*, vol. 9, no. 10, pp. 1–12, 2019.

[25] Y. Chen, Y. Ma, X. Mao, and Q. Li, "Multi-task learning for abstractive and extractive summarization," *Data Science and Engineering*, vol. 4, no. 1, pp. 14–23, 2019.

[26] P. Zhou and Z. Jiang, "Self-organizing map neural network (SOM) downscaling method to simulate daily precipitation in the Yangtze and Huaihe River Basin," *Climatic and Environmental Research*, vol. 21, no. 5, pp. 512–524, 2016.

[27] X. Xiao, "Analysis on the employment psychological problems and adjustment of retired athletes in the process of career transformation," *Modern Vocational Education*, vol. 5, no. 12, pp. 216-217, 2018.

[28] S. Sahoo and M. K. Jha, "Pattern recognition in lithology classification: modeling using neural networks, self-organizing maps and genetic algorithms," *Hydrogeology Journal*, vol. 25, no. 2, pp. 311–330, 2016.

[29] Y. Zhou and B. Yang, "Sports video athlete detection using convolutional neural network," *Journal of Natural Science of Xiangtan University*, vol. 39, no. 1, pp. 95–98, 2017.

[30] J. Pang, "Research on the evaluation model of sports training adaptation based on self-organizing neural network," *Journal of Nanjing Institute of Physical Education*, vol. 16, no. 1, pp. 74–77, 2017.

[31] G. Querzola, C. Lovati, C. Mariani, and L. Pantoni, "A semi-quantitative sport-specific assessment of recurrent traumatic brain injury: the TraQ questionnaire and its application in American football," *Neurological Sciences*, vol. 40, no. 9, pp. 1909–1915, 2019.

[32] J. Wang, X. Luo, and H. Yan, "Correlation analysis between injuries and functional movement screening for athletes of the National Shooting Team," *Journal of Capital Institute of Physical Education*, vol. 5, no. 4, pp. 352–355, 2016.

[33] G. Ma, "Research on the design of juvenile football players' sports injury prediction model," *Automation Technology and Application*, vol. 277, no. 7, pp. 141–144, 2018.