

Research Article

Analysis of Basketball Technical Movements Based on Human-Computer Interaction with Deep Learning

Xu-Hong Meng, Hong-Ying Shi , and Wei-Hong Shang

Basic Teaching Department, Hebei Vocational University of Industry and Technology, Shijiazhuang 050091, China

Correspondence should be addressed to Hong-Ying Shi; 1101962@sjzc.edu.cn

Received 16 February 2022; Revised 9 March 2022; Accepted 17 March 2022; Published 14 April 2022

Academic Editor: Qiangyi Li

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

With the continuous development of computer technology, analysis techniques based on various types of sports data sets are also evolving. One typical representative is image-based motion recognition technology, which enables video action recognition with a certain degree of feasibility. In basketball technical action videos, technical action has obvious characteristics. The athletes in the footage in sports videos are relatively fixed, and the scenes are relatively homogeneous, so technical action analysis of basketball technical action videos has certain advantages. However, there are many challenges in basketball technical action recognition, mainly including the fact that basketball techniques are numerous and complex. To address the above issues, this research proposes a 3D convolutional neural network framework that two different resolution image inputs are performed on the basketball technical action dataset. The experimental results show that the algorithmic process designed in this study is effective for action recognition on the basketball technical action dataset.

1. Introduction

In recent years, with the awakening of the population's awareness of physical fitness, various sports have been promoted and popularized. In China, basketball is one of the most popular sports for young people between the ages of 20 and 35. The equipment is simple, and a single court can accommodate 10–20 people. In addition, basketball is an all-round exercise for the human body. Smaller players can use their speed to move through the crowd, while bigger players can use their physical strengths to play on the back. Thus, no matter what physical condition an athlete is in, he or she can find the right position on the court. At the same time, basketball can be not only a one-on-one game but also a competitive team sport, where the sense of achievement of a beautifully organized attacking score is not dissimilar to that of a great dummy pass.

With the upgrade of mobile devices and the rise of the video industry, more and more athletes are focusing on using video to analyze their skills during basketball training [1, 2]. At the same time, recognition and classification of

image data and even video data have become possible with the increase in GPU computing power and the proposal of excellent models in the field of deep learning [3–5]. As an example, models based on convolutional neural networks have an important place in deep learning [6–9]. In addition, there are many other deep learning algorithms that are constantly being developed, such as BP neural network algorithm [10, 11], back-gradient algorithm [12], generative adversarial network [13, 14], transformer [15, 16], and life cycle assessment [17]. The convolutional neural network was first applied in the field of images, and after achieving excellent performance in this field, researchers have successively proposed algorithmic models that apply it to video recognition, such as large-scale video classification [18], multimemory convolutional neural network [19], and tube convolutional neural network [20]. If this technology can be used to recognize technical basketball moves as they appear in the video, this would be an important application of deep learning in sport. Such practical applications hold great promise for basketball training. For example, it could be used to inform professional players, analysts, or basketball

coaches about technical movements, and it could also assist referees in judging games on the court. Deep learning techniques such as wearing sensors on human joints to extract skeletal point location information for movement recognition have significant limitations, whereas it is easier and more efficient to recognize human movements in the dynamic video [21–23]. Therefore, a large number of researchers have investigated action recognition tasks through deep learning [24–26], by improving on the classical structures of convolutional neural networks to obtain frame-specific features and interimage timing information in videos.

Therefore, it is relevant to carry out work on the recognition of basketball technical movements through the use of deep learning methods. For all types of coaches and professional athletes, this technology will help them to better analyze the video in order to provide more effective coaching and training for their players.

2. Target Detection Method Based on Convolutional Neural Networks

The traditional target detection algorithm has roughly flowed as shown in Figure 1. The main process is to first intercept the image by means of a sliding window. The shape and size of the interception window vary due to the size of the target, which results in a large number of redundant windows [27]. Next, the information in the window is extracted for features. There are a number of methods for feature extraction, including methods based on shape and color according to rules specified for different tasks. This step of feature extraction is critical as algorithm performance is often causally related to a large extent to the quality of the extracted features. After feature extraction, the target within the window requires to be classified and determined. Commonly used classifiers include SVM and AdaBoost, and the final detection result is generated.

Liu et al. proposed a region convolutional neural network (R-CNN) algorithm based on deep learning, which used convolutional neural networks to recognize the images [28]. This research focuses on introducing R-CNN, fast R-CNN [29], and faster R-CNN [30].

2.1. R-CNN. Within the field of deep learning-related target detection, R-CNN is the first algorithm to apply convolutional neural networks to target detection. In contrast to traditional target detection algorithms, R-CNN first obtains a preselected box where the target may be located and performs feature extraction within the preselected box. This process is achieved through a selective search algorithm, and thus, window redundancy can be largely avoided, resulting in a definite improvement in the detection speed and accuracy of the network. As shown in Figure 2, the R-CNN acquires preselected frames from the input image and then transforms the image to make the preselected frames consistent in size for subsequent processing. Finally, it is fed into the network model, and the feature vector is output, which is used by the classifier to determine whether it belongs to a certain category.

Compared to sliding window target detection algorithms, R-CNN algorithms are faster, but there is still a large gap in real-time detection results. The main reason for this is that the feature extraction process involves a lot of redundant computations, as each candidate region of the image is convolved. In addition, image compression and image cropping operations are performed on each candidate region, and some of the image feature information is lost to a certain extent.

2.2. Fast R-CNN. Compared to the traditional loss function, fast R-CNN proposes a multitask loss function that combines the loss of classification and the loss of border regression and incorporates it into the CNN as a way to correct the position information of candidate frames.

The loss function includes classification loss and regression loss, which are shown in the following formula:

$$L(p, u, t^u, v) = L_c(p, u) + \lambda[u \geq 1] \times L_r(t^u, v), \quad (1)$$

where L_c refers to classification loss.

$$L_c(p, u) = -\log(p_u), \quad (2)$$

where L_r indicates regression loss.

$$L_r(t^u, v) = \sum_{i \in \{x, y, w, h\}} \text{smooth}_{L_1}(t_i^u - v_i),$$

$$\text{smooth}_{L_1}(t_i^u - v_i) = \begin{cases} 0.5 \times (t_i^u - v_i)^2, & |(t_i^u - v_i)| < 1, \\ |(t_i^u - v_i)| - 0.5, & \text{other.} \end{cases} \quad (3)$$

2.3. Faster R-CNN. The performance of fast R-CNN target detection algorithm is much improved compared to the R-CNN algorithm, but the bottleneck of fast R-CNN is the step of the selective search algorithm to generate the detection frame of the image. Faster R-CNN generates candidate frames by doing a region proposal network (RPN). At this point, the faster R-CNN is considered to be an end-to-end detection algorithm.

As shown in Figure 3, faster R-CNN uses RPN to extract prediction frames on the feature map, map the prediction frames onto the feature map, and feed them to the end-to-end detection algorithm. The prediction frames are mapped onto the feature map and fed into the Fast R-CNN for classification and regression.

3. Video Action Recognition Based on 3D-CNN

In basketball training, when performing human action recognition, the input data are a sequence of video frames. Therefore, it is necessary to consider not only the representation of the action in space but also the sequence between the atomic actions of the action in the video frame sequence. The traditional 2D convolutional networks cannot handle the sequential order of atomic actions when dealing with action recognition in video. The ability to determine an action from just one frame in a real-world scenario is very limited. Features need to be extracted from a continuous



FIGURE 1: Traditional target detection algorithm.

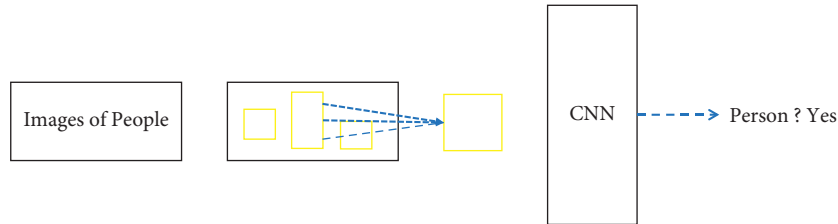


FIGURE 2: Framework of R-CNN.

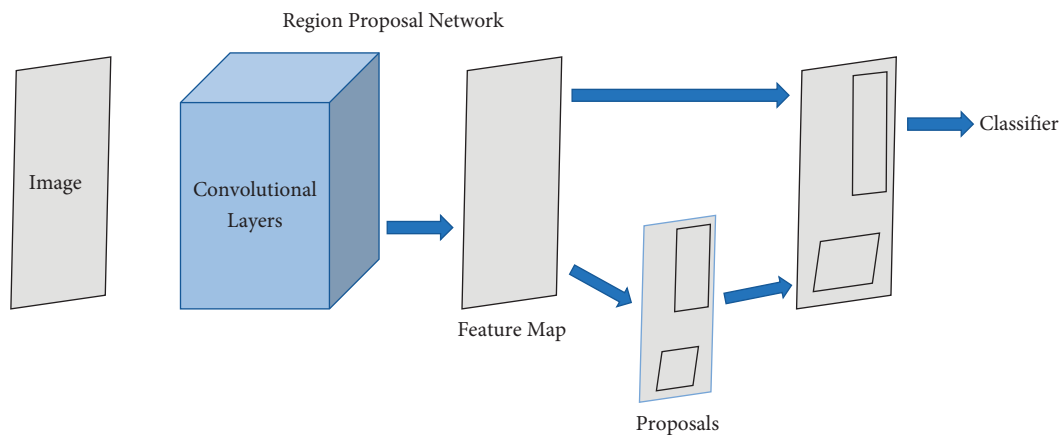


FIGURE 3: Framework of faster R-CNN.

video frame in order to classify actions more accurately. This problem can be solved in 3D convolutional networks.

3.1. Framework for Dual-Resolution 3D-CNN. The 3D-CNN (3D convolutional neural networks) has the ability to process continuous information between motion images by extending the traditional CNN to include a temporal dimension on 2D so that the video data contain both temporal and spatial dimensions. 2D convolutional operations are compared to 3D convolutional operations as shown in Figures 4 and 5. Figure 5 captures the motion information in three consecutive frames by computing them in a temporal dimension of 3.

In the same way as 2D convolutional kernels, 3D convolutional kernels extract certain types of features from stacked video frames using a weight-sharing approach. To get a better feature representation, more types of features can be extracted by increasing the variety of convolutional kernels or adjusting the convolutional kernel weights compared to the same convolutional kernel in the whole video cube. This is because the deeper the network is, the more data are available in the feature map, and more types of abstract features need to be generated from the combination of shallow features.

The most straightforward way to speed up network training is to reduce the number of convolutional layers and fully connected layer parameters, but this reduces the performance of the network. In order to ensure the performance of the training without changing the size of the network, the high-resolution data can be downsampled to low-resolution data, and then, the training experiment can be conducted. The experimental results show that although the network is faster, the low-resolution input data still lose some feature information, and certain details of the high resolution have an impact on the recognition rate.

As shown in Figure 6, the aim is to process successive technical action frames at two different resolution inputs and then fuse the features to further improve the recognition capability. A video clip with a resolution of 112×112 and 16 consecutive frames is used as input to the network. The upper stream is the original frame image after video pre-processing, and the lower stream is the cropped image after SSD target detection (circled in red). The lower stream is the cropped image after SSD target detection (circled in red), with a resolution size of 64×64 .

A picture is composed of pixel points based on a mixture of the three primary colors of RGB in different proportions. When the original picture data are fed directly into the network, the amount of input data will be larger, increasing

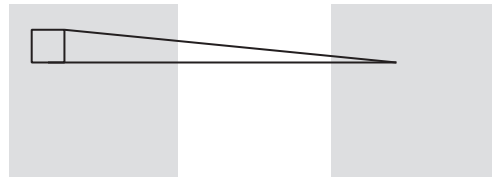


FIGURE 4: 2D convolution operation.

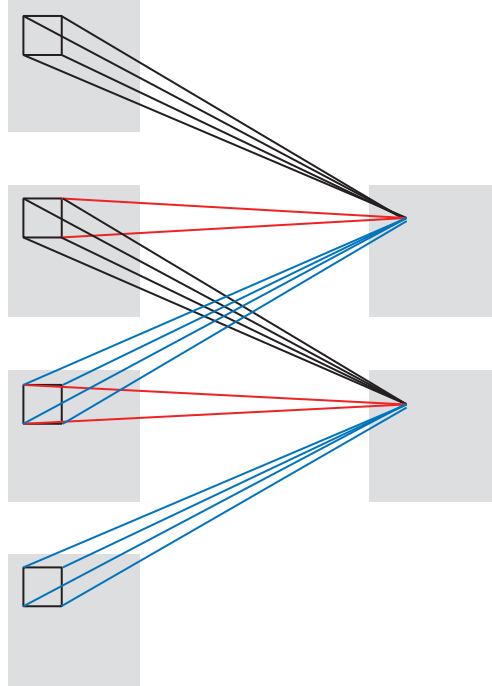


FIGURE 5: 3D convolution operation.

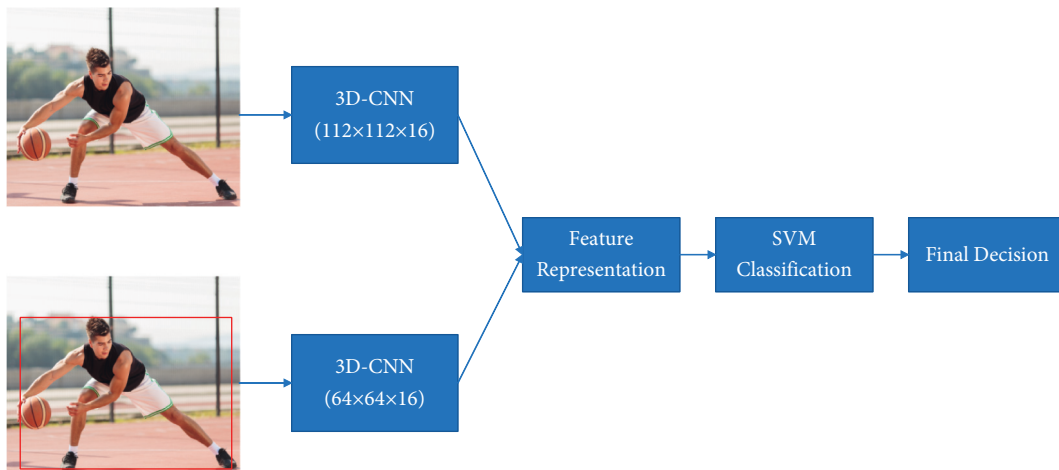


FIGURE 6: Framework of dual-resolution 3D-CNN.

the training time. However, in the process of human action recognition, the color information utilization of the picture is relatively low, and grey scaling of the original image can be considered. This is because even in a greyscale image, each

pixel of the image still has a certain range of variation and can still retain the vast majority of the information such as gradients in the original image. Using the network to process greyscale images can reduce the amount of computation to a

certain extent, thus achieving more efficient use of memory. In basketball technical action recognition, differences in the background of the court and changes in lighting and the color of the person's clothing can have an impact on the recognition. By grey scaling the images, the effects of these distracting elements can be ignored to a certain extent, so grey scaling of technical action images is feasible.

3.2. Initialization Weighting Strategy. In order to improve the training efficiency, this research proposes a method to initialize the weights by combining temporal information with the spatial convolution of image frames. In the 3D convolution layer, the 3D convolution weight parameters are initialized by using ImageNet's 2D weight parameters.

3.2.1. Mean Initialization. Since continuous image frames of technical actions have similar backgrounds, similar image information can be extracted through the following equation:

$$W_t^{3D} = \frac{W^{2D}}{T}. \quad (4)$$

3.2.2. Initialization by Scaling. According to equation (5), by introducing some diversity, any combination of constants can be used.

$$W_t^{3D} = a_t \times W^{2D}. \quad (5)$$

3.3. Feature Fusion. A 3D-CNN network with different resolutions after the original frames and cropped frames, respectively, will result in a corresponding weight file. Using the weights as parameters for the model, the technical action frames are predicted separately to obtain a 1024-dimensional feature vector for the corresponding frame sequence. After extracting the features, the features are fused, which is done for subsequent classification and recognition of the fused features. There are many ways to fuse the features, but in this paper, we choose the form of equation (6). As long as the dimensions are the same, the dual-stream models can be fused at the corresponding positions, even if there are minor differences that can be solved by filling in the differences by means of translation. Therefore, this research will fuse the features of the dual resolution streams by summing the results of the two feature maps. Let the final feature representation for the input continuous technical action frame be Y . At the same spatial locations i, j and feature channel d , the equation for Y can be expressed as follows:

$$Y = X_{i,j,d}^a + X_{i,j,d}^b. \quad (6)$$

As there are only six categories for classifying basketball technical movements and a total of 1800 video frame sequences, this sample size is suitable for application to a support vector machine (SVM). SVM is a statistical learning method based on the principles of structural risk minimization and VC dimensionality. The main logic of the SVM

algorithm is to train with labeled training samples and output a hyperplane that is optimized and separates the majority of high-dimensional data while minimizing the overall risk of the structure. The distinction between two-dimensional point data is simple: there are an infinite number of straight lines between the two categories that allow the training data to be classified correctly. In general, straight lines that are too close to a sample of a class are not optimal for classification because the bounds of such classification are more sensitive to the noise signal of the data, resulting in poor generalization. For linearly divisible data, the SVM obtains the optimal bounds by minimizing the distance between the bounds and all the training samples. When the training sample data are linearly indistinguishable, the SVM converts the nonlinear low-dimensional data into linearly divisible in the higher dimensional space by mapping it to some extent by means of a kernel function. This operation then converts the feature vectors in the input space into separable vectors and finally obtains the best performing classification hypersurface in the higher dimensional space by linearly separable methods. This is illustrated in Figure 7.

3.4. Comparison of Experimental Results. The number of frames has an impact on the recognition performance when the scale of the input video frames is the same. If the number of frames is too small, for example, if only one frame is input, it degrades to an action recognition based on a single frame, which is based on one frame only without temporal information and does not perform well for continuous action recognition. If the number of frames is too high, it is necessary to consider whether the number of frames per video is sufficient and to consider using data augmentation to generate more training data when there are not enough data. Also, the higher the number of frames, the higher the training volume of the network model, and the longer the time required. Table 1 shows the comparison of the recognition rate of the dual-stream 3D-CNN model for different frame counts of the dataset input in this study. From the experimental results, it can be seen that the algorithm achieves the highest recognition efficiency of 96.9% when 15 consecutive video image frames are input.

Taking 15 consecutive video image frames as an example, in order to verify the performance of dual-resolution 3D-CNN and single-resolution 3D-CNN for technical action recognition on basketball technical action video sets, this section compares the performance of the original frame 3D-CNN recognition algorithm and the cropped frame 3D-CNN recognition algorithm on basketball technical action video sets. From the experimental results, it can be concluded that the performance of the multiresolution 3D-CNN on basketball technical action recognition has been improved compared to the single-resolution approach of original frames and cropped frames, especially for jump shooting action and turning action with a recognition rate of over 95%. The results from the confusion matrix show that the experiments have some misclassification. For example, the turnaround and Sam Gord move, which may be due to

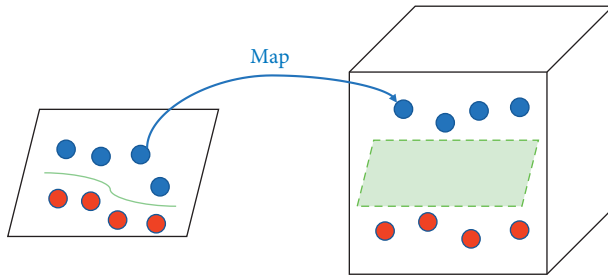


FIGURE 7: SVM maps features from linearly nonpartitionable to linearly partitionable.

TABLE 1: Comparison of experimental results.

Frame	Accuracy (%)
6	83.6
11	87.3
12	89.1
15	96.9

some similarities in that both mislead the defender’s defensive line before breaking through from the other side. There is also some confusion between the turn and the Dunbay move, probably due to the fact that there are few frames of combined ball movement between both technical moves, so there is some misclassification.

4. Conclusion

This study focuses on the action recognition problem in video basketball technology and proposes a recognition architecture that combines an SSD-based target detection algorithm and a multiresolution 3D-CNN for action recognition. In addition, the study focuses on the extensive use of convolutional neural networks in the field of deep learning. The main target detection algorithms and action recognition algorithms are also presented in detail. A series of data processing processes are used to create a dataset of 1800 basketball technical action videos based on practical technical actions that occur in the professional NBA, popular highlights, and self-published basketball instruction on short video platforms, and the evolution of actions under modern basketball rules. Finally, a dual-resolution 3D-CNN architecture-based video basketball action recognition algorithm was designed to build a CNN network for action recognition. The effectiveness of the algorithm was verified by comparing the original frames and cropped frames and SVM classification after fusing features on the basketball technical action dataset.

Although some progress has been made in this study on a home-made basketball technical action dataset, there is still a certain gap from practical application and many practical problems to be solved. In action recognition based on target detection, how to better identify technical actions if the character action involves certain occlusion and interference, or if multiple near-identical sized targets appear in the same shot. In video-based basketball technical action analysis,

information can be used to include the attacker, the defender, and the interaction between the two. As there is positional information on the court such as the free-throw line and the three-point line, can this be used to indicate the position of the person and use this to infer the action. These have some promise in terms of finding new cues to achieve improved recognition. If these problems can be solved, then they can be used to improve the performance of technical basketball action recognition. Furthermore, the representation of high-level information and low-level features for technical action recognition draws on other issues previously investigated whether there are unique behavioral representations applicable to basketball actions and, in doing so, investigates better representations than the current features.

Data Availability

The labeled dataset used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Hebei Vocational University of Industry and Technology.

References

- [1] H.-T. Chen, C.-L. Chou, T.-S. Fu, S.-Y. Lee, and B.-S. P. Lin, “Recognizing tactic patterns in broadcast basketball video using player trajectory,” *Journal of Visual Communication and Image Representation*, vol. 23, no. 6, pp. 932–947, 2012.
- [2] W. Liu, C. C. Yan, J. Liu, and H. Ma, “Deep learning based basketball video analysis for intelligent arena application,” *Multimedia Tools and Applications*, vol. 76, no. 23, pp. 24983–25001, 2017.
- [3] L. Dematté and D. Prandi, “GPU computing for systems biology,” *Briefings in Bioinformatics*, vol. 11, no. 3, pp. 323–333, 2010.
- [4] J. R. Cheng and M. Gen, “Accelerating genetic algorithms with GPU computing: a selective overview,” *Computers & Industrial Engineering*, vol. 128, pp. 514–525, 2019.
- [5] P. Messmer, P. J. Mullooney, and B. E. Granger, “GPULib: GPU computing in high-level languages,” *Computing in Science & Engineering*, vol. 10, no. 5, pp. 70–73, 2008.
- [6] B. Zhong, X. Xing, P. Love, X. Wang, and H. Luo, “Convolutional neural network: deep learning-based classification of building quality problems,” *Advanced Engineering Informatics*, vol. 40, pp. 46–57, 2019.
- [7] S. Indolia, A. K. Goswami, S. P. Mishra, and P. Asopa, “Conceptual understanding of convolutional neural network-A deep learning approach,” *Procedia Computer Science*, vol. 132, pp. 679–688, 2018.
- [8] K. Yasaka, H. Akai, A. Kunitatsu, S. Kiryu, and O. Abe, “Deep learning with convolutional neural network in radiology,” *Japanese Journal of Radiology*, vol. 36, no. 4, pp. 257–272, 2018.

- [9] N. A. Mubin, E. Nadarajoo, H. Z. M. Shafri, and A. Hamedianfar, "Young and mature oil palm tree detection and counting using convolutional neural network deep learning method," *International Journal of Remote Sensing*, vol. 40, no. 19, pp. 7500–7515, 2019.
- [10] Y. Qian, S. Chen, J. Li et al., "A decision-making model using machine learning for improving dispatching efficiency in Chengdu Shuangliu airport," *Complexity*, vol. 2020, Article ID 6626937, 16 pages, 2020.
- [11] S. Ding, C. Su, and J. Yu, "An optimizing BP neural network algorithm based on genetic algorithm," *Artificial Intelligence Review*, vol. 36, no. 2, pp. 153–162, 2011.
- [12] L. Muñoz-González, B. Biggio, A. Demontis et al., "Towards poisoning of deep learning algorithms with back-gradient optimization," in *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 27–38, Dallas, TX, USA, 2017.
- [13] F. Zhou, S. Yang, H. Fujita, D. Chen, and C. Wen, "Deep learning fault diagnosis method based on global optimization GAN for unbalanced data," *Knowledge-Based Systems*, vol. 187, Article ID 104837, 2020.
- [14] D. Xu, C. Wei, P. Peng, Q. Xuan, and H. Guo, "GE-GAN: a novel deep learning framework for road traffic state estimation," *Transportation Research Part C: Emerging Technologies*, vol. 117, Article ID 102635, 2020.
- [15] H. Wu, G. Q. Shen, X. Lin, M. Li, and C. Z. Li, "A transformer-based deep learning model for recognizing communication-oriented entities from patents of ICT in construction," *Automation in Construction*, vol. 125, Article ID 103608, 2021.
- [16] S. Singh and A. Mahmood, "The NLP cookbook: modern recipes for transformer based deep learning architectures," *IEEE Access*, vol. 9, pp. 68675–68702, 2021.
- [17] B. Q. Cheng, K. Lu, J. C. Li, H. H. Chen, X. W. Luo, and M. Shafique, "Comprehensive assessment of embodied environmental impacts of buildings using normalized environmental impact factors," *Journal of Cleaner Production*, vol. 334, Article ID 133083, 2022.
- [18] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei, "Large-scale video classification with convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1725–1732, Columbus, Ohio, USA, June 2014.
- [19] Z. Wang, P. Yi, K. Jiang et al., "Multi-memory convolutional neural network for video super-resolution," *IEEE Transactions on Image Processing*, vol. 28, no. 5, pp. 2530–2544, 2018.
- [20] R. Hou, C. Chen, and M. Shah, "Tube convolutional neural network (T-CNN) for action detection in videos," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5822–5831, Venice, Italy, October 2017.
- [21] M. Sharifi Renani, C. A. Myers, R. Zandie, M. H. Mahoor, B. S. Davidson, and C. W. Clary, "Deep learning in gait parameter prediction for oa and tka patients wearing imu sensors," *Sensors*, vol. 20, no. 19, p. 5553, 2020.
- [22] K. Han and X. Zeng, "Deep learning-based workers safety helmet wearing detection on construction sites using multi-scale features," *IEEE Access*, vol. 10, pp. 718–729, 2021.
- [23] A. Kaku, A. Parnandi, A. Venkatesan, N. Pandit, H. Schambra, and C. Fernandez-Granda, "Towards data-driven stroke rehabilitation via wearable sensors and deep learning," in *Proceedings of the Machine Learning for Healthcare Conference*, pp. 143–171, irtual Event, Durham, August 2020.
- [24] Y. Zhu, J. K. Zhao, Y. N. Wang, and B. B. Zheng, "A review of human action recognition based on deep learning," *Acta Automatica Sinica*, vol. 42, no. 6, pp. 848–857, 2016.
- [25] Z. Huang, C. Wan, T. Probst, and L. Van Gool, "Deep learning on lie groups for skeleton-based action recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6099–6108, Honolulu, HW, USA, July 2017.
- [26] K. Charalampous and A. Gasteratos, "On-line deep learning method for action recognition," *Pattern Analysis & Applications*, vol. 19, no. 2, pp. 337–354, 2016.
- [27] X. Shi, G. Huang, X. Hao, Y. Yang, and Z. Li, "A synchronous prediction model based on multi-channel CNN with moving window for coal and electricity consumption in cement calcination process," *Sensors*, vol. 21, no. 13, p. 4284, 2021.
- [28] Z. Liu, J. Hu, L. Weng, and Y. Yang, "Rotated region based CNN for ship detection," in *Proceedings of the IEEE International Conference on Image Processing*, pp. 900–904, IEEE, Washington, DC, USA, June 2017.
- [29] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1440–1448, Santiago, Chile, December 2015.
- [30] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 28, 2015.