# Human Action Recognition Using Deep Learning Methods

Zeqi Yu and Wei Qi Yan

Auckland University of Technology, Auckland 1010 New Zealand

***Abstract*—The goal of human action recognition is to identify and understand the actions of people in videos and export corresponding tags. In addition to spatial correlation existing in 2D images, actions in a video also own the attributes in temporal domain. Due to the complexity of human actions, e.g., the changes of perspectives, background noises, and others will affect the recognition. In order to solve these thorny problems, three algorithms are designed and implemented in this paper. Based on convolutional neural networks (CNN), Two-Stream CNN, CNN+LSTM, and 3D CNN are harnessed to identify human actions in videos. Each algorithm is explicated and analyzed on details. HMDB-51 dataset is applied to test these algorithms and gain the best results. Experimental results showcase that the three methods have effectively identified human actions given a video, the best algorithm thus is selected.**

***Keywords—behavior recognition, convolutional neural network, deep learning, LSTM, 3D CNN, Two-Stream CNN (****key words****)***

## I. INTRODUCTION

Human action recognition from videos is based on the analysis of a sequence of video frames by using computers, so as to automatically find human actions without manual operations [1]. In the era of the Internet with mobile phones, people's daily lives have been surrounded by access control such as building gates, traffic sensors, security cameras, and many others. The ubiquitous cameras enable everyone's actions in public to be monitored, identification of human actions in surveillance videos has tremendous significance in the field of cybersecurity [17,18,19]. In addition, analysis and understanding of human actions in digital videos encapsulate multiple interesting research topics such as object detection [20,21], semantic segmentation, motion analysis, etc. Hence, human action analysis has a broad spectrum of applications including intelligent surveillance, intelligent care, etc.

Traditional methods in machine learning for human action recognition extracted visual features primarily based on human observations [2]. It is subject to a vast amount of human experience and background knowledge. Most of these algorithms only performed well on the exact dataset for a specific experiment. At present, there are a huge number of digital video footages available on the Internet, like YouTube, it is impossible to satisfy the demand to annotate all videos with tags and extract the features only based on our human labor.

Fortunately, the surge of deep learning methods in recent years has provided a solution. Deep learning algorithms generate feature maps based on artificial neural networks [3]. Deep neural networks have remarkable achievements in the field of computer vision, natural language processing, robotics. However, as deep learning is still at its early stage. Meanwhile, human motion is relatively complicated, the relevant motion analysis is affected by various determinants such as chaotic background, various lighting conditions, unstable image acquisition, and insufficient pattern classes, etc. [4]. Deep learning thus has a large room for developing human action recognition. Additionally, deep learning has extremely vital value to implement self-learning and transfer learning.

CNN is a core net in deep learning. Human action recognition from digital images or video frames is to identify a story in one shot. The objective using CNNs is to group a sequence of motion pictures into a class of human actions by using the end-to-end way. In this paper, we optimize three end-to-end algorithms which are all closely related to CNNs. The three deep learning algorithms include Two-Stream CNN, CNN+LSTM, and 3D CNN which are specifically appropriate for spatiotemporal data analysis like human actions in video footages. Assisted by a given public dataset, we compare the three algorithms, our aim is to find the best one for human action recognition.

In this paper, our related work is addressed in Section II, our method is explicated in Section III, our results are demonstrated in Section IV, our conclusion and future work are depicted in Section V.

## II. RELATED WORK

In recent years, a great deal of breakthroughs have been attained in the field of machine vision and deep learning. A plenty of methods for human action recognition based on deep learning have been explored and exploited [28,29]. Compared with machine learning methods for human behaviour recognition, deep learn approaches do not require a specific type of human experience and knowledge. Instead, human actions in a video are identified directly in the end-to-end way [3]. According to feature extraction methods, the approaches are grouped into two categories, i.e., human action recognition based on skeletons, human action recognition based on feature maps. Amongst the deep learning methods, spatiotemporal networks and Two-Stream networks are the salient ones [5]. In these methods, CNN and RNN are most popular [30,31].

A multimodal learning approach was proposed for the recognition and classification of human actions [6,29]. In 2017, 3D convolutional neural network (3D CNN) and two-way long short-term memory network (ConvLSTM) were trained based on multimodal and spatiotemporal data to fulfil human action recognition by using support vector machine (SVM) [7,8]. A deep dynamic neural network (DDNN) was designed to implement action recognition from input data under multimodal framework, which extracts spatiotemporal features from RGB and RGB-D images [9]. A scene-flow dynamic model was deployed to generate visual features from RGB and depth images, which were imported for training by using CNN networks [10,32]. A 3D deep convolutional neural network was taken into account to learn high-level features from the original images, fix the position and angle of bone joint information. The two features were fused by using SVM for human action classification [11]. In 2018, CNN and RNN were integrated together to cope with the spatiotemporal

978-1-7281-8579-8/20/$31.00 ©2020 IEEE

information of human actions and achieved promising results [12].

## III. Our Methods

The ultimate goal of this paper is to implement human action recognition from the given videos. We segment the video footages and imported the video frames as the input data. Three deep learning methods are applied to generate feature maps for human action recognition. Throughout network training, we recognize human actions and finally export the class tags.

### A. CNN+LSTM Model

CNNs are a class of feedforward neural networks, which are principally comprised of input layer, convolutional layer, pooling layer, full connection layer, and output layer [5]. The convolutional layer of a CNN encompasses one or more feature planes. Each feature plane is related to numerous neurons in a region, the neurons in the same plane share the same weights. The shared weights consist of network parametric set, the better weights are gained in the process of model training [13]. By extracting local features and synthesizing them at a higher level, CNNs not only yield global features but also lessen a number of neuron nodes. At this point, the number of neurons is still very large, by setting the weight for each neuron equally, the number of network parameters will be greatly diminished. On the first convolution layer, the output is $y_m$, then the output after $k$ times of convolution operations is

$$y_k^m = \delta(\sum_{y_i^{n-1} \in M_k} y_i^{m-1} * W_{ik}^m + b_k^m) \quad (1)$$

where $\delta(\cdot)$ is an activation function, $M_k$ is based on a layer of feature collection, $W_{ik}^m$ refers to convolution kernels, $*$ means a convolution, $b_k^m$ stands for offset.

In CNNs, pooling layer follows the convolutional layer to reduce dimensionality and accelerate the convergence of network training. The other is to remove redundant features so as to prevent overfitting. Each neuron in the full connection layer is linked with all neurons in the proceed layer. Throughout the full connections, all local features are integrated together to form the overall features. Each neuron in the full connection layer operates with an activation function, which is transferred to the output layer.

In RNNs, the memory units have not ability to measure the value of information. It is impossible to distinguish the importance of state information, which results in the useless information being stored in memory. However, the truly valuable information is squeezed. Each unit of LSTM network contains memory unit, input gate, forget gate, and output gate.

Behavioral videos contain not only spatial data but also temporal information. Using CNN, the temporal information of a given video cannot be fully used. The output of LSTM is determined by using combined actions of current input and historical output. Temporal information is applied to represent a sequence of video frames. The structure for CNN+LSTM model is shown in Fig. 1.

In CNN, a video is decomposed into single frames so as to form a large image dataset. This set is imported as the input of single-channel CNN+LSTM for pretraining. The training results are stored and the sequence of features are generated. The dataset is then imported into LSTM network as input data. The sequence of video frames is used to train LSTM network. After the training, the parameters of CNN are exported as spatial features for human action recognition.

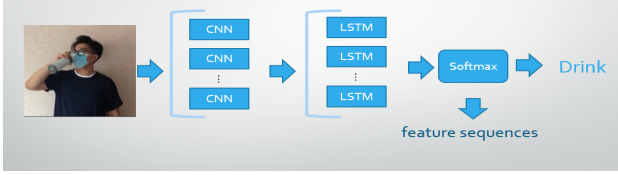


Fig. 1. The structure of CNN+LSTM algorithm

In each video sequence, every 8 video frames are treated as a group of input. The spatial feature is imported into the LSTM to learn the temporal relationship of the frame sequence so as to fix the network parameters. In the test, every 8 video frames extracted from a video equally are taken as the input data of CNN+LSTM model. After spatial feature extraction and temporal feature selection, the output tags of LSTM are thought as the final classification result.

### B. Two-Stream CNN

Two-Stream CNN employs RGB (color map) and optical flow to construct a CNN network. The core idea is to use two CNNs to tackle RGB values of video frames and dense optical flow of adjacent frames, respectively [14]. Two-Stream CNN models were trained separately. The results of the two deep learning models are fused. Multimode fusion and information complementation of visual data are implemented. The network architecture of Two-Stream CNN is shown as Fig.2.

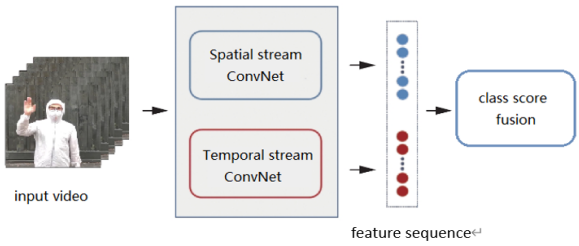


Fig. 2. The structure of the Two-Stream CNN

In the training process of CNNs, the backpropagation (BP) algorithm is accommodated to carry out parameter updating at each layer of the neural network, which includes two passes: Forward pass and back pass.

In the CNN, convolutional layer, pooling layer, and full connection layer all contain corresponding weighting parameters. Let current layer be $k$, the output of the current hidden layer is shown in eq. (2) and eq.(3):

$$u^k = W^k x^{k-1} + b^k \quad (2)$$

$$x^k = f(u^k) \quad (3)$$

where $W^k$ and $b^k$ represent the weight and bias of neurons at layer $k$, $k$=1,2,…, $n$. $x^k$ is the output of the current layer and the input of the next hidden layer, $u^k$ represents the output of neuron at layer $k$, $f(\cdot)$ is the activation function such as sigmoid function, rectified linear unit (ReLU) function, etc.

Throughout calculating, the prediction results including high-level semantics are obtained and best matched with the ground truth. A training dataset with ground truth has $C$ categories and $N$ samples. For each individual sample $n$, the training error is expressed by using eq.(4).

$$E^n = \frac{1}{2}\sum_{l=1}^{c}(t_l^n - y_l^n)^2, \quad (4)$$

where $t$ is the ground truth of the sample corresponding to the sample $n$ in class $c$ , $y$ is the network estimation corresponding to the sample $n$ in class $c$ . The total error is the sum of the error of each sample, which is calculated as

$$E = \frac{1}{2}\sum_{n=1}^{N}\sum_{l=1}^{c}(t_l^n - y_l^n)^2. \quad (5)$$

In the backpropagation, the total error $E$ takes the derivative of each node of each hidden layer. The partial derivative will be calculated by using gradient descent method for parameter updating at each layer.

The backpropagation algorithm calculates the partial derivative layer by layer and transmits as well as reduces the errors layer by layer. It updates the parameters of each layer along the direction of negative gradient. The partial derivative of $E$ with respect to $b$ is shown as

$$\frac{\partial E}{\partial b} = \frac{\partial E}{\partial u}\frac{\partial u}{\partial b} = \delta. \quad (6)$$

The weights are updated by using

$$\Delta W^K = -\eta\frac{\partial E}{\partial W^k}, \quad (7)$$

where $\eta$ represents learning rate which is a parameter to control the convergent speed of the updating iterations. By setting an appropriate learning rate, the network keeps training and updating weights, till the loss function reaches a local minimum, the model is optimal.

Through the training, the CNN network weights reach the local minimum with respect to the loss function. In the network testing, only the calculations of forward pass are accomplished. The test dataset is imported into the network with the fixed parameters after training. The classification results are compared with the ground truth. The average error rate of all test samples is calculated to measure the classification performance of the network.

### *C. 3D CNN*

In 2D CNN, convolution is applied to 2D images, the features are calculated only from spatial domain. By using video data, we expect to capture motion information encoded in consecutive frames. Therefore, it is proposed to carry out 3D convolution in the CNN so as to reflect the spatiotemporal characteristics. 3D convolution is a stack of several consecutive frames so as to form a cube. Then, 3D convolution kernel is applied to the cube. By using this designed structure, the feature map in convolutional layer is connected to multiple adjacent frames in the previous layer so as to capture motion information [15,16]. The position of a feature map is obtained through local perception of the same position of three consecutive frames in the convolution layer [36, 37].

It is important that 3D convolution kernel only extracts feature maps from one cube. Because the kernels are the same throughout the process of convolution, we use multiple convolution kernels to extract multiple features. Based on the 3D convolution, various architectures can be deployed and designed. Thus, 3D CNN architecture which has been developed for human action recognition is implemented as shown in Fig.3.

3D CNN architecture in this paper includes a hidden layer, 3 convolutional layers, 2 lower sampling layers, and a full connection layer. Each 3D convolution kernel convolutes the cube with 8 consecutive frames, the patch size of each frame is $60 \times 40$.

On the first layer, we applied a fixed kernel to tackle the original frame in our experiment, generate multiple channels of information, and deal with multiple channels separately. Finally, we combine the information of all channels to get the final description. This layer actually encodes our prior knowledge of the features, which is better than random initialization.

In this paper, from each frame, we extract the information of five channels, namely, the gradients in $x$ and $y$ directions, the optical flow in $x$ and $y$ directions as shown in Fig. 3, where the first three are calculated from per frame, the horizontal and vertical optical flows need two consecutive frames to be determined. There are 33 feature maps in total.

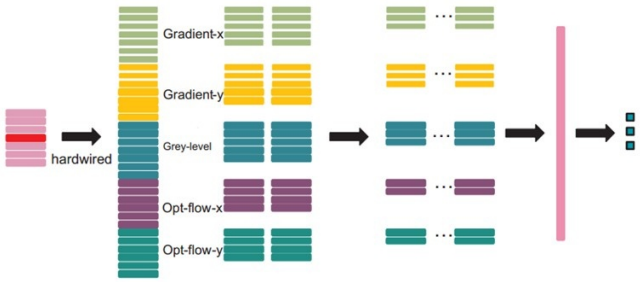


Fig. 3. The information extracted from each frame

A 3D convolution kernel ($7 \times 7$ in spatial domain, 3 in temporal domain) is applied to convolute each of the five channels, respectively. In order to increase the number of feature maps, our experiment utilizes two different convolution kernels at each position. Thus, in the two feature maps at the $C_2$ layer, each group contains 23 feature maps. The patch size of each frame is $54 \times 34$.

The next sampling layer is $S_3$ layer. In max pooling, the feature maps of $C_2$ layer are sampled under $2 \times 2$ window. Therefore, we get the same number of feature maps with lower spatial resolution in the output.

$C_4$ is a 3D convolution kernel with $7 \times 6 \times 3$ in 5 channels. In order to increase the number of feature maps, three different convolution kernels are employed for each location. This gives us six sets of feature maps, each has 13 feature maps. The patch size of each frame is $21 \times 12$.

At this stage, the number of frames in temporal domain is already very small. In this experiment, we convolute only within the spatial domain. The convolution kernel is $7 \times 4$, and the output feature maps are reduced to $1 \times 1$. $C_6$ layer contains 128 feature maps, each of which is fully connected with all 78 ($13 \times 6$) feature maps in $S_5$ layer, then each feature map goes to $1 \times 1$.

After multilayer convolution and downsampling, the input image with its successive 7 frames is converted into a 128-dimensional feature vector. This eigenvector captures the motion of the input frame. The number of nodes in the output layer is as same as that of human actions, each node is fully connected to the 128 nodes in $C_6$ as shown in Fig. 4. Hereinafter, the CNN model utilizes a linear classifier SVM to classify the 128D feature vectors to implement human action recognition.

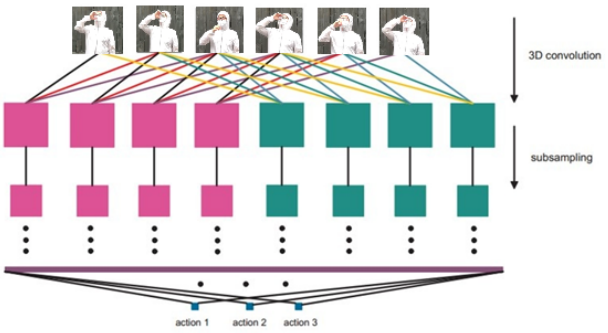


Fig. 4. Human action recognition using 3D CNN net

## IV. Results and Analysis

In our experiment, HMDB-51 dataset was selected as the training set. HMDB-51 has a total of 51 categories and 6,766 short videos. The human action videos with one person were chosen as the test data. The selected actions are shown in Fig.5. There are four actions to be classified in this paper, namely, clapping, waving, hugging, and drinking.

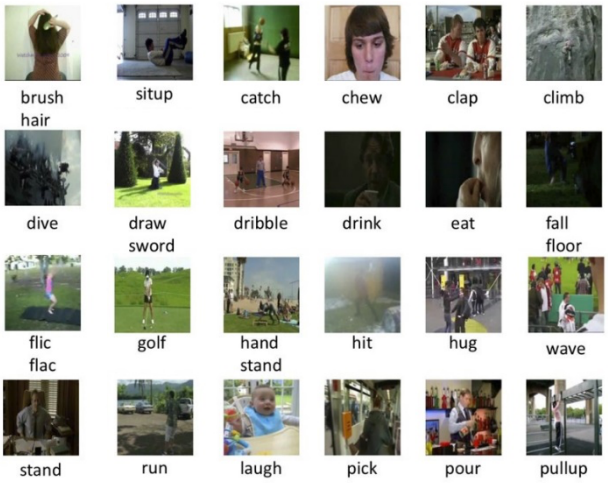


Fig. 5. Examplar actions in HMDB-51 dataset

### A. Our Results

In this section, we present the recognition results of the three algorithms. Table 1 shows the accuracies for human action recognition with 850 iterations by using these three models.

TABLE I. Recognition Accuracy of The Three Models

| | Results | |
|---|---|---|
| | **Methods** | **Accuracies** |
| Recognition accuracy | CNN+LSTM | 89.74% |
| | Two-Stream CNN | 82.37% |
| | 3D CNN | 86.54% |

The experimental results of the three models are shown in Table I. CNN+LSTM model has the highest accuracy rate 89.74%. 3D CNN model was at the second position up to 86.54%. Finally, the Two-Stream CNN model has 82.37%. There are more than 85% in the two out of three methods.

Figure 6 shows the experimental accuracies of the three models. Throughout 850 iterations, the final accuracy of each algorithm is obtained, the polylines indicate the trends with the growth of iterations. The results show the best performance is from the CNN+LSTM model. During the iterations, accuracy rates are converged to stable levels.

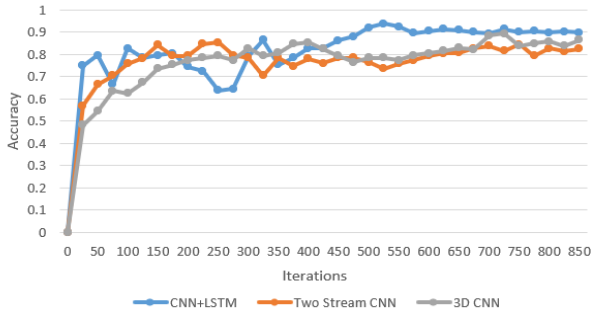


Fig. 6. The accuracy rates of three methods

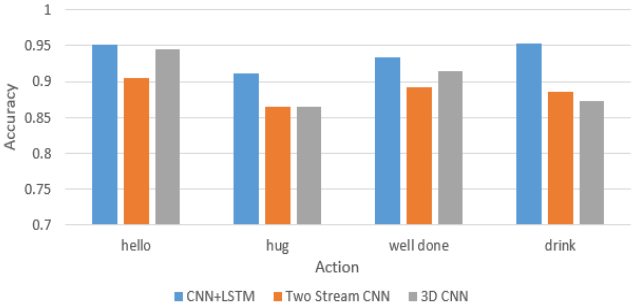


Fig. 7. The accuracy of the three models for a single action recognition

Figure 7 shows the accuracy rates of the three models for identifying each action. Each action has a recognition rate more than 90%. Regarding CNN+LSTM, the accuracy rates for action Hello and action Drink are up to 95%. Pertaining to the Two-Stream CNN model, the best action to be identified is the action Hello (Waving). Other three are all under 90%, but over 85%. With regard to 3D CNN model, the accuracy rates are all over 85%, action Hello is the highest one. But the recognition accuracy of action Well-Done (Clapping) is very high, up to 92%.

Fig. 8. The results of human action recognition by using CNN+LSTM

The results from CNN+LSTM model are shown in Fig. 8. The results unveil that human actions have been accurately identified. However, the experiments are not working well to identify the movements under dim lighting conditions when

the video is almost invisible even with our human eyes. However, if we wear reflective clothes, the situation is completely altered, the recognition results are shown in Fig.9. The results indicate that our actions are able to be recognized even in complete dark environment. In addition, we experimented with multiple shooting angles. The results show that the actions on the videos with front and side views are able to be accurately identified.


Fig. 9. The recognition results of wearing reflective clothes

### B. *Analysis and Discussions*

The advantage of the Two-Stream CNN model is that two convolutional neural networks are able to acquire pretty rich features of human actions. In the model, the diversity of behavioural description is increased and more determinants are obtained. But CNN is limited by its own attributes, the Two-Stream CNN is apt. This structure ignores the temporal relationship of the video frames. It is difficult to cope with video samples with complicated spatiotemporal relationships and abrupt changes. In addition to optimize the performance of the CNN, we should consider modelling the temporal information of video frames such as adding more temporal convolution operations.

3D CNN increases the dimensionality of data input even if the training samples are not raised. Therefore, 3D CNN needs more samples of human actions to train the network well. Under the same training condition, the recognition accuracy is not excellent. In addition, 3D CNN only copes with adjacent frames if short-term motion information of the actions is available. This method is still unable to model the full video sequence. In time series analysis, 3D convolutions are offered to enrich 2D convolutions, a basic network is employed to extract spatial features and short-time motion features for complicated action recognition.

CNN+LSTM model makes up the CNN for recognizing human actions. Human action recognition benefits from visual information of digital videos and the temporal relationship between video adjacent frames. Although the architecture of CNN+LSTM is clear, what information the structure needs is still ambiguous. By effectively integrating multimodal actions, we are able to learn from each other and gain much discriminative action descriptors.

Overall, all three methods accurately identified human actions. More accurate identification is attained by adjusting the parameters of these nets. In our experiments, though CNN+LSTM has the best performance, the Two-Stream CNN and 3D CNN have shown better outcomes and will be continuously to be improved.

## V. Conclusion and Future Work

Feature extraction is the most critical step in human action recognition, which relies on domain knowledge and human experience that cannot meet the demands of data growth. Therefore, we take deep learning methods as our start point in this paper. The mainstream method in deep learning is based on CNNs. Therefore, three recognition algorithms, i.e., the Two-Streams CNN, CNN+LSTM, and 3D CNN are chiefly taken into account in this paper. Throughout feature selection, the algorithms successfully recognized human actions from a given video, they are distinct in dealing with time series problems. Our experiments show that LSTM better deals with this temporal coherence. Therefore, LSTM+CNN recognizes human actions from the videos more effectively.

Videos have temporal information compared to 2D images, which is complicated and diverse. The annotation of video data is time-consuming, laborious, and expensive. Therefore, deep learning models for video-based classification are relatively slow compared with static images. The performance of deep learning models in human action classification, interaction recognition, and motion detection has been verified. However, in real complicated scenarios such as intelligent video surveillance [40], there are too many problems in scene understanding [41], interactive recognition [42], and spatiotemporal actions located from multimodal data. Our previous work was focusing on human gait [43-48] and behavior recognition [49-52], we will devote to resolve these problems in the near future.

## References

[1] M. Murakami, J. K. Tan, H. Kim and S. Ishikawa, Human motion recognition using directional motion history images, Proceedings of SICE Annual Conference 2010, Taipei, 2010, pp. 1512-1514.

[2] L. S. Xu, M. Q. Meng and K. Q. Wang, Pulse image recognition using fuzzy neural network, 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Lyon, 2007, pp. 3148-3151.

[3] B. Zhang, C. Quan and F. Ren, Study on CNN in the recognition of emotion in audio and images, IEEE/ACIS International Conference on Computer and Information Science (ICIS), Okayama, 2016, pp. 1-5.

[4] J. Xu, L. Tu, Z. Zhang, L. Zhang and C. Zhou, The region partition of quality and coating for tongue image based on color image segmentation method, IEEE International Symposium on IT in Medicine and Education, Xiamen, 2008, pp. 817-821.

[5] S. Deep and X. Zheng, Leveraging CNN and transfer learning for vision-based human activity recognition, International Telecommunication Networks and Applications Conference, Auckland, New Zealand, 2019, pp.1-4.

[6] S. Khan, H. Rahmani, S. Shah, M. Bennamoun, G. Medioni, S. Dickinson, A Guide to Convolutional Neural Networks for Computer Vision, Morgan & Claypool, 2018.

[7] L. Jing, Y. Ye, X. Yang and Y. Tian, 3D convolutional neural network with multi-model framework for action recognition, IEEE International Conference on Image Processing (ICIP), Beijing, 2017, pp. 1837-1841, doi: 10.1109/ICIP.2017.8296599.

[8] C. Li, S. Sun, X. Min, W. Lin, B. Nie and X. Zhang, End-to-end learning of deep convolutional neural network for 3D human action recognition, IEEE International Conference on Multimedia & Expo Workshops (ICMEW), Hong Kong, 2017, pp. 609-612.

[9] D. Wu et al., "Deep dynamic neural networks for multimodal gesture segmentation and recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(8) 1583-1597, 2016.

[10] Y. Wang, W. Zhou, Q. Zhang, X. Zhu and H. Li, Weighted multi-region convolutional neural network for action recognition with low-latency online prediction, IEEE International Conference on Multimedia & Expo Workshops (ICMEW), San Diego, CA, 2018, pp. 1-6.

[11] J. Li, Parallel two-class 3D-CNN classifiers for video classification, International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), 2017, pp. 7-11.

[12] M. A. Russo, A. Filonenko and K. Jo, Sports classification in sequential frames using CNN and RNN, International Conference on Information

and Communication Technology Robotics (ICT-ROBOT), 2018, pp. 1-3.

[13] W. Ye, J. Cheng, F. Yang and Y. Xu, "Two-stream convolutional network for improving activity recognition using convolutional long short-term memory networks, " IEEE Access, vol. 7, pp. 67772-67780, 2019.

[14] W. Dai, Y. Chen, C. Huang, M. Gao and X. Zhang, Two-stream convolution neural network with video-stream for action recognition, International Joint Conference on Neural Networks (IJCNN), Hungary, 2019, pp. 1-8.

[15] S. Park and D. Kim, Study on 3D action recognition based on deep neural network, International Conference on Electronics, Information, and Communication (ICEIC), New Zealand, 2019, pp. 1-3.

[16] L. Liu, F. Hu and J. Zhao, Action recognition based on features fusion and 3D convolutional neural networks, International Symposium on Computational Intelligence and Design (ISCID), 2016, pp. 178-181.

[17] D. Schonfeld, "MotionSearch: Context-Based Video Retrieval and Activity Recognition in Video Surveillance, " IEEE International Conference on Advanced Video and Signal Based Surveillance, Genova, 2009, pp. 194-194.

[18] A. H. Meghdadi and P. Irani, "Interactive Exploration of Surveillance Video through Action Shot Summarization and Trajectory Visualization," in IEEE Transactions on Visualization and Computer Graphics, 19(12) 2119-2128, 2013.

[19] J. D. Prange, Detecting, recognizing and understanding video events in surveillance video, IEEE Conference on Advanced Video and Signal Based Surveillance, 2003, pp. 4.

[20] E. Liu, Research on video smoke recognition based on dynamic image segmentation and detection technology, International Conference on Intelligent Computation Technology and Automation (ICICTA), 2019, pp. 240-243.

[21] H. Wei, Z. Ou and J. Zhang, Fingerprint identification based on ridge lines and graph matching, World Congress on Intelligent Control and Automation, 2006, pp. 9965-9968.

[22] G. Xu, H. Jiang and Q. Chen, Research of comparison of intelligent extraction methods based on high spatial resolution image, International Conference on Intelligent Computing and Integrated Systems, 2010, pp. 433-436.

[23] R. Zhao, R. Zhang Rui, Z. Li, Study of the algorithms for image matching in intelligent transportation system, International Conference on Industrial Informatics - Computing Technology, Intelligent Technology, Industrial Information Integration, 2015, pp. 14-17.

[24] A. Sevik, P. Erdogmus and E. Yalein, Font and Turkish letter recognition in images with deep learning, International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT), 2018, pp. 61-64.

[25] S. Zhang, X. Pan, Y. Cui, X. Zhao and L. Liu, "Learning affective video features for facial expression recognition via Hybrid Deep Learning, " in IEEE Access, vol. 7, pp. 32297-32304, 2019.

[26] Y. Wang, T. Bao, C. Ding and M. Zhu, Face recognition in real-world surveillance videos with deep learning method, International Conference on Image, Vision and Computing (ICIVC), 2017, pp. 239-243.

[27] Z. Li, Y. Tie and L. Qi, Face recognition in real-world Internet videos based on deep learning, International Symposium on Next Generation Electronics (ISNE), 2019, pp. 1-3.

[28] S. G. Pleshkova, A. B. Bekyarski and Z. T. Zahariev, Based on artificial intelligence and deep learning hand gesture recognition for interaction with mobile robots, National Conference with International Participation (ELECTRONICA), 2019, pp. 1-4,.

[29] P. Gao, D. Zhao and X. Chen, "Multi-dimensional data modelling of video image action recognition and motion capture in deep learning framework," IET Image Processing, 14(7) 1257-1264, 2020.

[30] Y. Liu, P. Wang and H. Wang, Target tracking algorithm based on deep learning and multi-video monitoring, International Conference on Systems and Informatics (ICSAI), 2018, pp. 440-444.

[31] A. Baisware, B. Sayankar and S. Hood, Review on recent advances in human action recognition in video data, International Conference on Emerging Trends in Engineering and Technology - Signal and Information Processing (ICETET-SIP-19), 2019, pp. 1-5.

[32] S. Deep and X. Zheng, Leveraging CNN and transfer learning for vision-based human activity recognition, International Telecommunication Networks and Applications Conference (ITNAC), 2019, pp. 1-4.

[33] Murugan V. Madras, Vijaykumar V. R. Anna, Nidhila A. Madras, A deep learning R-CNN approach for vehicle recognition in traffic surveillance system, International Conference on Communication and Signal Processing (ICCSP), 2019, pp. 0157-0160.

[34] S. Kamada and T. Ichimura, A video recognition method by using adaptive structural learning of long short-term memory based deep belief network, International Workshop on Computational Intelligence and Applications (IWCIA), 2019, pp. 21-26.

[35] A. S. Keçeli, A. Kaya and A. B. Can, Action recognition with skeletal volume and deep learning, Signal Processing and Communications Applications Conference (SIU), 2017, pp. 1-4.

[36] S. A. Rahman and D. A. Adjeroh, Estimating biological age from physical activity using deep learning with 3D CNN, IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2019, pp. 1100-1103.

[37] S. Park and D. Kim, Study on 3D action recognition based on deep neural network, International Conference on Electronics, Information, and Communication (ICEIC), 2019, pp. 1-3.

[38] L. Liu, F. Hu and J. Zhao, Action recognition based on features fusion and 3D convolutional neural networks, International Symposium on Computational Intelligence and Design (ISCID), 2016, pp. 178-181.

[39] J. Li, Parallel two-class 3D-CNN classifiers for video classification, International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS), 2017, pp. 7-11.

[40] W. Yan, Introduction to Intelligent Surveillance - Surveillance Data Capture, Transmission, and Analytics (3rd Edition), Springer 2019.

[41] X. Liu, M. Neuyen, W. Yan, "Vehicle-related scene understanding using deep learning," ACPR Workshops 2019, pp. 61-73.

[42] K. Zheng, W. Yan, P. Nand, "Video dynamics detection using deep neural networks," IEEE Trans. Emerg. Top. Comput. Intell. 2018, 2(3): 224-234.

[43] X. Wang, W Yan, "Non-local gait feature extraction and human identification," Multimedia Tools and Applications, 2020.

[44] X. Wang, W. Yan, "Human gait recognition based on frame-by-frame gait energy images and convolutional long short-term memory," Int. J. Neural Syst. 2020, 30(1): 1950027:1-1950027:12.

[45] X. Wang, W. Yan, "Cross-view gait recognition through ensemble learning," Neural Comput. Appl. 2020, 32(11): 7275-7287.

[46] X. Wang, J. Zhang, W. Yan, "Gait recognition using multichannel convolution neural networks," Neural Comput. Appl. 2020, 32(18): 14275-14285.

[47] X. Wang, W. Yan, "Human gait recognition based on SAHMM," IEEE/ACM Transactions on Biology and Bioinformatics, 2020.

[48] C. Liu, W. Yan, "Gait recognition using deep learning," Handbook of Research on Multimedia Cyber Security, 2020, IGI Global, pp.214-226.

[49] J. Lu, W. Yan, M. Nguyen, Human behaviour recognition using deep learning. IEEE AVSS 2018, pp. 1-6.

[50] J. Lu, M. Nguyen, W. Yan, Deep learning methods for human behavior recognition, IEEE IVCNZ, 2020.

[51] J. Lu, M. Nguyen, W. Yan, Comparative evaluations of human behaviour recognition using deep learning, Handbook of Research on Multimedia Cyber Security, 2020, IGI Global, pp. 176-189.

[52] J. Lu, J. Shen, W. Yan, "An empirical study for human behaviour analysis," International Journal of Digital Crime and Forensics, 2017, 9(3), 11-1.