A Survey of Vision-Based Human Action Recognition Algorithms: Postprint

Authors: Chen Yuping, Weigen Qiu

Date: 2018-07-09T00:00:00+00:00

Abstract

Human action recognition finds extensive applications and represents a hot research topic in the field of artificial intelligence. Conducting a comprehensive survey and summary of human action recognition algorithms holds significant reference value. Centered on action recognition, this work also encompasses datasets, action segmentation, and related content. The introduction section primarily describes the fundamental pipeline of human action recognition; the dataset section summarizes commonly used datasets in this domain; the action segmentation methods section reviews the current development status and commonly employed techniques of temporal segmentation; the traditional methods section elaborates on classical approaches; and the deep learning methods section compiles the latest and most popular deep learning methodologies. By incorporating action segmentation and combining it with action recognition, continuous human action recognition can be achieved, rendering action recognition applicable to real-world scenarios rather than merely recognizing artificially edited individual videos, which is of great significance for practical applications.

Full Text

Preamble

Survey of Human Action Recognition Algorithms Based on Vision

Chen Yuping, Qiu Weigen
(School of Computers, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: Human action recognition has extensive applications and represents a hot research topic in artificial intelligence. Summarizing human action recognition algorithms holds significant reference value. This paper focuses on action recognition while also covering datasets and motion segmentation. The introduction describes the basic pipeline of human action recognition. The dataset

section summarizes commonly used datasets for human action recognition. The motion segmentation section reviews the development status and common methods of temporal segmentation. Traditional methods explain classic algorithms for human action recognition, while deep learning methods summarize the latest and most popular deep learning approaches for action recognition. The introduction of motion segmentation, combined with action recognition, enables continuous human action recognition, making it applicable to real-world scenarios rather than merely recognizing manually edited individual videos. This has great practical significance.

Keywords: human action recognition; dataset; motion segmentation; deep learning; two-stream network

0 Introduction

Human action recognition primarily analyzes human behavior from captured video and finds widespread application in video surveillance, medical rehabilitation, fitness assessment, human-computer interaction, and other domains, making it a hot research topic in computer vision. From an implementation perspective, human action recognition can be categorized into sensor-based and vision-based approaches, as well as combinations thereof. Sensor-based recognition requires wearing corresponding sensors, which lacks flexibility, involves complex operation, offers limited scalability, and fails to guarantee effective user experience, thus restricting its use to specific domains. Vision-based recognition can be further divided into single-image and video-based approaches. Single-image recognition cannot effectively capture temporal continuity information about actions, often leading to misclassification, whereas video-based recognition can effectively extract both spatial and temporal information from videos, significantly improving accuracy. Due to its strong scalability and high flexibility, video-based action analysis has received extensive research and application.

The general processing pipeline for human action recognition can be divided into three steps: feature extraction, feature processing, and learning algorithm output. First, features are extracted from raw video, processed to form a feature descriptor, and finally classified through a learning algorithm. For some learning algorithms with fixed input dimensions but variable feature descriptors, aggregation methods must be employed to ensure fixed input dimensions. This paper incorporates motion segmentation to achieve continuous human action recognition, with the flowchart shown in Figure 1 [Figure 1: see original paper].

This paper primarily introduces four aspects: datasets, motion segmentation, traditional methods, and deep learning methods. We first present commonly used human action recognition databases, including both 2D and 3D datasets. Next, we discuss motion segmentation, which is essential for continuous action recognition and offers greater advantages over traditional single-action recognition. Traditional methods cover commonly used approaches for action recognition, while deep learning methods focus on recent techniques based on deep

learning.

1 Datasets

Numerous human action recognition datasets have emerged to facilitate experiments, which can be divided into 2D and 3D datasets. 2D datasets are typically collected using ordinary cameras, while 3D datasets use special cameras like Kinect that capture depth information. Since 3D datasets contain depth information, they are more informative.

1.1 2D Datasets

2D datasets emerged earlier and have evolved toward increasing complexity, with richer action categories, more diverse scenes, and larger sample sizes per action, posing more stringent challenges for algorithms. Table 1 lists commonly used 2D datasets.

KTH [2] is one of the earliest human action datasets, containing relatively little data: only 6 action categories with 2,391 videos from 25 subjects. Although simple, this dataset played a milestone role in human action recognition. **Weizmann** [3,4] contains 10 action categories with 93 videos from 9 subjects. **IXMAS** [5] includes 13 actions with 180 videos, a relatively small dataset but containing 5 viewpoints, providing reliable data for multi-view research. These three datasets are early, commonly used benchmarks with small data volumes, simple scenes, and no complex backgrounds, and are rarely used today.

Hollywood [6] consists of video clips from Hollywood movies, containing 8 action categories with 663 videos from 32 movies. **Hollywood 2** [7] extends this, containing 12 action categories with 3,669 videos from 69 movies. Since these two datasets approximate real-world scenarios with complex backgrounds, lighting variations, and self-occlusion, they present certain challenges.

The UCF series includes multiple datasets that have attracted widespread attention due to their diversity and challenge. **UCF Sports** [8] contains data from BBC and ESPN television channels, including 10 action categories with 150 high-resolution videos approaching natural scenes. **UCF YouTube** [9], now called UCF11, sources data from YouTube, containing 11 action categories with 1,600 videos grouped by common characteristics. **UCF50** [10] extends UCF YouTube to 50 action categories with 6,676 videos, presenting considerable challenges. **UCF101** [11] further extends UCF50, containing 101 action categories with 13,320 videos. Due to its inclusion of many low-quality videos with varying lighting conditions, this dataset is extremely challenging.

HMDB51 [12] primarily sources data from movies and public resources, containing 51 action categories with 6,849 videos. With its diverse sources, complex scenes, and varying lighting conditions, it is currently one of the most challenging datasets. **Sports-1M** [13] from YouTube, released by Google in 2014, is a

large-scale dataset containing 289 action categories with 1,133,158 videos (1,000-3,000 videos per category). Its massive scale of over one million videos represents an insurmountable advantage over previous datasets, with diverse scenes and categories making it highly challenging.

1.2 3D Datasets

Due to self-occlusion issues in human actions, 2D data cannot adequately address these problems, whereas 3D data provides additional information that compensates for occlusion, making recognition relatively easier but increasing dataset complexity and processing difficulty. However, with advances in computer hardware, 3D data collection and processing have become more accessible, exemplified by Microsoft Kinect. Table 2 lists commonly used 3D datasets.

CMU Motion Capture (Mocap) [14] is a 3D dataset released by Carnegie Mellon University, captured using 8 infrared cameras and providing 41 human joint points. It contains 6 major action categories and 23 subcategories with 2,605 sequences, each major category containing one or more action types, enabling construction of complete 3D human action models.

With the emergence of Microsoft Kinect, many databases have been built using it. **MSR Action 3D** [15] uses Kinect to provide 20 joint skeleton data and depth maps, containing 20 action categories with 567 sequences. The videos have clean backgrounds, though noise still presents challenges, making it widely used. **MSR Daily Activity 3D** [16] also uses Kinect to capture daily life activities, containing 10 action categories with 320 sequences in real-world backgrounds, making it more challenging. **UCF Kinect** [17] similarly uses Kinect but employs OpenNI rather than Microsoft's SDK to evaluate skeleton sequences, with 15 skeleton joints per sequence, containing 16 action categories with 1,280 sequences.

N-UCLA Multiview Action3D [18] uses three Kinect cameras, thus containing three viewpoints, with 10 action categories and 1,493 sequences. Each action is captured from different viewpoints, presenting certain challenges. **UTD-MHAD** [19] uses both Kinect and IMU, providing 20 skeleton joints with 17 action categories and 861 sequences. **NTU RGB+D** [20] uses second-generation Kinect, providing 25 skeleton points with 60 action categories and 56,880 sequences.

2 Motion Segmentation

Motion segmentation here refers to segmenting continuous actions from videos, i.e., temporal segmentation. If a video contains actions like walking, running, and jumping, the algorithm should accurately identify each action' s boundaries and segment them from the original video. Since current action recognition is performed on pre-segmented datasets while real-world captured data is unsegmented, motion segmentation is crucial for achieving continuous human action recognition.

2.1 PCA-Based Methods

Barbič et al. [21] proposed three methods: PCA, PPCA, and GMM. The PCA method is based on the idea that the intrinsic dimensionality of a motion sequence containing a single action should be smaller than that containing multiple actions. By computing discrete errors d_i, when d_i rises sharply beyond a fixed threshold, it is considered a transition point. PPCA improves upon PCA based on the assumption that "action sequences follow a Gaussian distribution, and two different actions will have significant differences." It uses a sliding window mechanism (both forward and backward) to find transition points by computing the Mahalanobis distance of the sliding window; when reaching a maximum, it is considered a transition point. GMM is based on the assumption that "each action in a sequence follows different Gaussian distributions." It projects onto a hyperplane using PCA and employs Expectation Maximization (EM) to estimate Gaussian model parameters for segmentation.

PCA-based methods rely on certain assumptions and have limitations. The GMM method also requires prior knowledge of the number of action categories in each video, which is often unknown. However, these methods have low hardware requirements, are relatively simple to implement, and can be easily applied to scenarios that meet their assumptions.

2.2 Clustering-Based Methods

Zhou et al. [22] proposed ACA (Aligned Cluster Analysis), which extends standard k-means clustering in two ways: a) clusters contain variable numbers of features, and b) Dynamic Time Warping (DTW) kernels achieve temporal invariance. They use DTAK (dynamic time alignment kernel) to measure two time series since DTW is not a properly defined metric and fails to satisfy triangle constraints. Zhou et al. [23] proposed HACA, an improvement over ACA that provides natural embedding for clustering and visualizing time series data with hierarchical decomposition across multiple temporal scales. Temporal clustering is formulated as energy minimization, and minimizing HACA is an NP problem, for which they propose an efficient coordinate descent method via dynamic programming.

Xia et al. [24] proposed a method based on SSC (Sparse Subspace Clustering). SSC performs subspace clustering, then uses triangle constraints to prevent similar frames in different time periods from being assigned to the same cluster, ensuring temporal continuity. Correntropy suppresses non-Gaussian noise in subspace clustering. Finally, the absolute average of all coefficient matrices is used for the final segmentation similarity matrix to reconstruct relationships between different joints without ignoring joint relationships by treating the entire sequence as a whole.

Clustering-based methods generally outperform PCA-based methods overall, with relatively higher time complexity but low hardware requirements, making them ideal for applications when considering cost and demand.

2.3 Deep Learning-Based Methods

Lea et al. [25] proposed TS-CNN, introducing spatiotemporal CNN (ST-CNN) for low-level encoded visual information and a semi-Markov model for capturing high-level temporal information. ST-CNN' s spatial component is a VGG variant for encoding fine-grained tasks of object states, positions, and inter-object relationships. The segmentation component uses semi-Markov and Conditional Random Fields (CRF) to jointly segment and classify actions.

Lea et al. [26] proposed TCN (Temporal Convolutional Networks), shown in Figure 2 [Figure 2: see original paper]. TCN includes ED-TCN and Dilated TCN. ED-TCN introduces encoder-decoder networks, while Dilated TCN is adapted from WaveNet. Both share common characteristics: a) hierarchical execution, meaning each time step updates synchronously rather than frame-by-frame; b) convolutions computed across time; c) predictions at each frame are functions of fixed-length time periods called receptive fields. ED-TCN outperforms Dilated TCN.

These methods are currently research hotspots, generally using CNN (or autoencoder) combined with other machine learning methods, achieving better results than other approaches. However, they require high hardware configurations and depend on large amounts of data, making implementation difficult, though model miniaturization is a viable option.

2.4 Other Methods

Devanne et al. [27] studied motion trajectory shapes in Riemannian shape space and achieved action segmentation via dynamic naive Bayes classifiers. Vögele et al. [28] used neighborhood graph methods to segment action sequences. Borzeshi et al. [29] used extended HMM (HMM-MIO) models for joint action segmentation and classification. Liu et al. [30] proposed TS-WMCS, using curvature of time series warping for segmentation.

3 Traditional Methods

Traditional methods primarily involve manual feature extraction, building models to represent human actions, and recognizing actions on these models. They can be categorized into holistic and local representation methods based on representation approach.

3.1 Holistic Representation Methods

Holistic methods represent human actions as a whole for analysis. Bobick [31] proposed classic MEI (Motion Energy Images) and MHI (Motion History Images), encoding motion-related information through a single image. MHI templates show how motion images move, with each pixel being a function of the motion' s temporal history at that point (higher intensity corresponds to more recent motion). Thus, MEI and MHI templates contain useful contextual video

information. Blank et al. [3] proposed a volumetric extension of MEI templates, representing actions through 3D shapes induced by silhouettes in spacetime, as shown in Figure 3 [Figure 3: see original paper]. Weinland et al. [33] suggested representing MHI templates through spatiotemporal volumes and demonstrated that 3D volumetric extensions increase robustness to viewpoint changes.

Yilmaz et al. [34] determined actions based on different characteristics of spatiotemporal volumes (STV). STV is built by stacking object contours along the time axis. Changes in STV' s direction, velocity, and shape characterize underlying actions. Action descriptions are sets of attributes (e.g., Gaussian curvature) extracted from STV surfaces, showing robustness to viewpoint changes.

Holistic methods have limitations. Dollar et al. [35] showed that holistic methods are too rigid to effectively capture viewpoint, occlusion, and other variations. Matikainen et al. [36] argued that contour-based representations cannot capture details within contours. Consequently, local representation methods and deep features are now favored.

3.2 Local Representation Methods

Local representation methods use local regions in videos to describe human motion. Laptev et al.' s [37] spatiotemporal interest points (STIPs) laid the foundation for local representation methods in action recognition, which follow three processes: interest point detection, local descriptor extraction, and local descriptor aggregation.

Interest Point Detection: Extracting spatiotemporal interest points requires building STIP detectors, which can be constructed in various ways. Laptev et al. [38] extended the Harris corner detector [39] to 3D-Harris, which identifies points with large spatial variation and non-constant motion, requiring temporal significance beyond rich spatial structure. To prune irrelevant interest points triggered by camera shake, Liu et al. [9] suggested using statistical properties of detected interest points. Many other methods and improvements exist, but the Harris corner detector remains the most classic.

Local Descriptor Extraction: After extracting spatiotemporal interest points, they must be processed to form local descriptors representing human actions. Kläser et al. [40] proposed using histograms of oriented gradients as motion descriptors, inspired by Histogram of Oriented Gradients (HoG) [41] but spanning spatiotemporal domains, hence named HoG3D. Laptev et al. [6] used Histograms of Optical Flow (HoF) in local regions as spatiotemporal descriptors. A more robust extension of HoF is Motion Boundary Histograms (MBH) introduced by Dalal et al. [43]. Local Binary Patterns (LBP) are intensity-based 2D descriptors successfully used in various vision problems including face recognition and texture analysis [44], computed by quantifying a pixel' s neighborhood relative to its intensity. Zhao et al. [45] introduced various extensions of 2D LBP to spatiotemporal domains: Volume LBP (VLBP), where local volumes are encoded by histograms of binary patterns. LBP has several

variants [46,47]. Sanin et al. [49] described image regions through second-order statistics.

Spatiotemporal interest points may not be located at identical spatial positions within cuboids' temporal extents, so features extracted from cuboids may not necessarily describe the interest points themselves. Trajectories are features correctly tracked over time, as shown in Figure 4 [Figure 4: see original paper]. Trajectory-based local feature extraction was proposed by Messing et al. [50] and Matikainen et al. [36], both using trajectory velocity as local features. Jiang et al. [51] and Wang et al. [52] improved earlier trajectories using camera motion correction.

Regarding the choice between sparse and dense interest points, Wang et al. [53] conducted detailed comparisons, generally showing dense approaches outperform sparse ones but with higher temporal and spatial complexity.

Local Descriptor Aggregation: After extracting local features from videos, learning algorithms like SVM are typically used for training. However, most such algorithms only accept fixed-size vector inputs, requiring mechanisms to aggregate local features into fixed-size descriptors. Three main mechanisms exist. The first uses Bag-of-Visual-Words (BoV), where local descriptors' distribution over a "visual vocabulary" or "codebook" serves as the descriptor. Related work includes Dollar et al. [35,40,6,53], though Fisher Vector encoding (FV) [54-56] has recently become preferred. A simplified FV version is Vector of Locally Aggregated Descriptors (VLAD) [57], successfully applied in Jain et al. [58-61]. The second uses spatiotemporal dictionary learning and sparse coding for aggregation, represented by Zhu et al. [62-65]. The third aggregates through temporal consistency by incorporating temporal information into video descriptors, with research focusing on Hidden Markov Models (HMM) [66] and Conditional Random Fields (CRF) [67], represented by Hongeng et al. [68-73]. Many such methods exist, with choices depending on specific problems. Currently, the second and third methods are more commonly used, with the third generally achieving the best overall results.

4 Deep Learning Methods

Compared to traditional methods, deep learning approaches do not require manual feature extraction, preserving more valuable information from videos and generally achieving superior results. Deep learning for human action recognition must utilize both spatial and temporal video information, which is the focus of research.

4.1 Spatiotemporal Networks

Spatiotemporal networks focus on extracting temporal information from videos, typically using CNNs for spatial features and other methods like LSTM for temporal features. Time and space information follow a series architecture similar

to electrical circuits, which was popular in early methods and generally outperforms traditional approaches.

Li et al. [74] proposed a method based on LSTM and CNN, extracting multiple artificially defined features input into 3 LSTM networks and 7 CNN networks, then fusing these 10 networks. They proposed three fusion methods: max fusion, average fusion, and element-wise multiplication fusion, with element-wise multiplication performing best.

Karpathy et al. [13] proposed late fusion, early fusion, and slow fusion in convolutional networks to enable continuous multi-frame input, as shown in Figure 5 [Figure 5: see original paper], allowing temporal information capture. A CNN then processes this information, with similar work by Chen et al. [61-63]. Donahue et al. [76] proposed LRCN, which first extracts spatial information via CNN, then uses an LSTM network to extract temporal information for classification. Sun et al. [77] also proposed LSTM-based methods for capturing temporal information in video sequences.

Ji et al. [78] proposed 3D CNN methods. 3D CNN adds a temporal dimension to 2D CNN, enabling simultaneous learning of spatial and temporal information from input videos, outperforming 2D CNN methods. Wang et al. [79] combined 3D CNN with LSTM, simultaneously performing saliency detection on raw videos to effectively reduce network parameters and training difficulty. They also pre-trained 3D CNN on Sports-1M, achieving 84.0% accuracy on UCF-101. Since 3D CNN can only stack a fixed number of input frames, it cannot capture entire video temporal information like LSTM and has higher complexity, though its performance surpasses pure CNN-LSTM combinations, demonstrating that temporal convolution effectively captures video temporal information despite fixed frame counts. Combining 3D CNN with LSTM is also a good approach.

4.2 Two-Stream Networks

Two-stream networks process temporal and spatial information in parallel, similar to parallel circuits in electrical engineering, with two networks operating independently before fusion. Simonyan et al. [80] first proposed the creative two-stream network in 2014, as shown in Figure 6 [Figure 6: see original paper], using two identical CNNs: one network inputs video frames for spatial information, while the other inputs optical flow information for temporal information, with final fusion via averaging or SVM classification, the latter performing best. Many improvements have since been proposed.

Feichtenhofer et al. [81] improved fusion strategies by fusing intermediate layers rather than only at the end, as shown in Figure 6, achieving better results than the original two-stream network while significantly reducing parameters. Wang et al. [82] used improved trajectories (iDT) instead of optical flow for temporal information while keeping the spatial network unchanged, pooling local ConvNet responses on trajectory-centered spatiotemporal tubes. The resulting

descriptors are called TDD, and Fisher Vectors aggregate local TDDs across the entire video into global hypervectors, with linear SVM as the classifier for action recognition.

Wang et al. [83] proposed TSN (Temporal Segment Networks) based on two-stream networks with segmental and sparse sampling ideas, as shown in Figure 7 [Figure 7: see original paper]. This reduces complexity while enabling fusion across multiple segments to capture more contextual information. The temporal stream uses warped optical flow fields instead of original optical flow to eliminate camera motion effects. Additional techniques like cross-modality pre-training, regularization, and data augmentation further improve the network. Analysis shows that whether the temporal network input uses optical flow, trajectories, or their improved variants has little impact on final results; the decisive factors are network structure and final fusion method.

Chen et al. [84] incorporated semi-coupled concepts into two-stream networks for extremely low-resolution action recognition, proposing additive fusion, concatenation fusion, and convolutional fusion, with convolutional fusion performing best. Wang et al. [85] used 3D CNN instead of 2D CNN. To support arbitrary video sizes and lengths, they used STPP (Spatial Temporal Pyramid Pooling) instead of ordinary pooling in the final convolutional layer to ensure consistent output feature dimensions. Each network incorporates LSTM or CNN-E to learn temporal information, with fusion via element-wise max, element-wise sum, or concatenation, as shown in Figure 8 [Figure 8: see original paper].

Gammulle et al. [86] proposed a two-stream LSTM network using ImageNet-pretrained VGG16 for initial feature extraction followed by LSTM, with four fusion strategies, two-stream LSTM performing best. Zhao et al. [87] used 3D CNN for the spatial network and RNN with bidirectional GRU for the temporal network, inputting human skeleton sequences and achieving good results on NTU RGB+D.

In summary, two-stream networks are currently the most popular framework, not only due to their effectiveness but also for providing excellent ideas for parallel architectures in action recognition. TSN currently achieves state-of-the-art results on UCF101, demonstrating the power of two-stream networks. However, deep learning' s high hardware requirements and extreme dependence on massive data pose new challenges for practical applications.

4.3 Other Networks

Beyond spatiotemporal networks, other excellent architectures exist, particularly unsupervised methods. Annotating video data is expensive, making unsupervised techniques significantly advantageous. Yan et al. [88] introduced Dynencoder, a deep autoencoder that captures video dynamics. Dynencoder has proven successful for synthesizing dynamic textures and can be viewed as a compact representation of video spatiotemporal information. Thus, reconstruction error from Dynencoder can be used for classification.

Srivastava et al. [89] proposed LSTM autoencoder models (Figure 9 [Figure 9: see original paper]) consisting of encoder and decoder LSTMs. The encoder LSTM accepts a sequence as input and learns a compact representation containing both appearance and dynamic temporal information. The decoder LSTM receives this representation to reconstruct the input sequence.

Currently, unsupervised and weakly supervised methods are widely favored due to their application value without requiring manual labels or only needing few labels, representing a promising future research direction. However, their performance has not yet matched supervised methods.

Time coherence is a weakly supervised approach. If models are fed ordered sequences as positive samples and unordered sequences as negative samples, temporal correlations can be learned through deep models. This concept has been used by Goroshin et al. [90] and Wang et al. [91] to learn features from unlabeled videos. Misra et al. [92] studied how to use time coherence to train deep models for action recognition and pose estimation (Figure 10 [Figure 10: see original paper]). Siamese networks [93-95] are trained with tuples to determine whether given sequences are consistent. Another related study is Wang et al.' s [96] action recognition, performed in two stages as shown in Figure 11 [Figure 11: see original paper]. For video frame sets, prerequisite set X_p (Equation (1)) and effect set X_e (Equation (2)) are input to the same network, with actions labeled by transformations mapping high-level descriptors from X_p to those from X_e.

5 Conclusion

This paper systematically reviews datasets and methods in human action recognition, covering both traditional approaches and recent popular deep learning methods. Deep learning has become the mainstream trend, evolving from simple to complex models, from supervised to weakly supervised and unsupervised methods. Introducing motion segmentation enables continuous action recognition, though current segmentation accuracy remains low and far from application requirements, indicating a long road ahead. Integrating motion segmentation with action recognition is also a future development trend.

References

[1] Zhu Honglei, Zhu Changsheng, Xu Zhigang. Research advances on human activity recognition datasets [J/OL]. ACTA AUTOMATICA SINICA: 1-27. (2018-04-30). https://doi.org/10.16383/j.aas.2018.c170043.

[2] Schuldt C, Laptev I, Caputo B. Recognizing human actions: a local SVM approach [C]// Proc of the 17th International Conference on Pattern Recognition. 2004: 32-36.

[3] Blank M, Gorelick L, Shechtman E, et al. Actions as space-time shapes [C]// Proc of the 10th IEEE International Conference on Computer Vision. 2005:

1395-1402.

[4] Gorelick L, Blank M, Shechtman E, et al. Actions as space-time shapes [J]. IEEE transactions on pattern analysis and machine intelligence, 2007, 29(12): 2247-2253.

[5] Weinland D, Ronfard R, Boyer E. Free viewpoint action recognition using motion history volumes [J]. Computer vision and image understanding, 2006, 104(2-3): 249-257.

[6] Laptev I, Marszalek M, Schmid C, et al. Learning realistic human actions from movies [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2008: 1-8.

[7] Marszalek M, Laptev I, Schmid C. Actions in context [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2009: 2929-2936.

[8] Rodriguez M D, Ahmed J, Shah M. Action mach: a spatio-temporal maximum average correlation height filter for action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2008: 1-8.

[9] Liu Jingen, Luo Jiebo, Shah M. Recognizing realistic actions from videos "in the wild" [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2009: 1996-2003.

[10] Reddy K K, Shah M. Recognizing 50 human action categories of Web videos [J]. Machine Vision and Applications, 2013, 24(5): 971-981.

[11] Soomro K, Zamir A R, Shah M. UCF101: a dataset of 101 human action classes from videos in the wild [J]. Computer Science, 2012, 12(1): 1-7.

[12] Kuehne H, Jhuang H, Garrote E, et al. HMDB: a large video database for human motion recognition [C]// Proc of IEEE International Conference on Computer Vision. 2011: 2556-2563.

[13] Karpathy A, Toderici G, Shetty S, et al. Large-scale video classification with convolutional neural networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2014: 1725-1732.

[14] CMU graphics lab motion capture database [DB/OL]. (2016-09-27). http://mocap.cs.cmu.edu.

[15] Li Wanqing, Zhang Zhengyou, Liu Zicheng. Action recognition based on a bag of 3D points [C]// Proc of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. Washington DC: IEEE Computer Society, 2010: 9-14.

[16] Wang Jiang, Liu Zicheng, Wu Ying, et al. Mining actionlet ensemble for action recognition with depth cameras [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2012: 1290-1297.

[17] Ellis C, Masood S Z, Tappen M F, et al. Exploring the trade-off between accuracy and observational latency in action recognition [J]. International Journal

of Computer Vision, 2013, 101(3): 420-436.

[18] Wang Jiang, Nie Xiaohan, Xia Yin, et al. Cross-view action modeling, learning and recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2014: 2649-2656.

[19] Chen Chen, Jafari R, Kehtarnavaz N. Utd-mhad: a multimodal dataset for human action recognition utilizing a depth camera and a wearable inertial sensor [C]// Proc of IEEE International Conference on Image Processing. 2015: 168-172.

[20] Shahroudy A, Liu Jun, Ng T T, et al. NTU RGB+D: a large scale dataset for 3D human activity analysis [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2016: 1010-1019.

[21] Barbič J, Safonova A, Pan J Y, et al. Segmenting motion capture data into distinct behaviors [C]// Proc of Graphics Interface. Ontario: Canadian Human-Computer Communications Society, 2004: 185-194.

[22] Zhou Feng, De la Torre F, Hodgins J K. Aligned cluster analysis for temporal segmentation of human motion [C]// Proc of the 8th IEEE International Conference on Automatic Face & Gesture Recognition. 2008: 1-7.

[23] Zhou Feng, De la Torre F, Hodgins J K. Hierarchical aligned cluster analysis for temporal clustering of human motion [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2013, 35(3): 582-596.

[24] Xia Guiyu, Sun Huaijiang, Feng Lei, et al. Human motion segmentation via robust kernel sparse subspace clustering [J]. IEEE Trans on Image Processing, 2018, 27(1): 135-150.

[25] Lea C, Reiter A, Vidal R, et al. Segmental spatiotemporal cnns for fine-grained action segmentation [C]// Proc of European Conference on Computer Vision. Cham: Springer, 2016: 36-52.

[26] Lea C, Flynn M D, Vidal R, et al. Temporal convolutional networks for action segmentation and detection [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2017: 156-165.

[27] Devanne M, Berretti S, Pala P, et al. Motion segment decomposition of RGB-D sequences for human behavior understanding [J]. Pattern Recognition, 2017, 61(1): 222-233.

[28] Vögele A, Krüger B, Klein R. Efficient unsupervised temporal segmentation of human motion [C]// Proc of ACM SIGGRAPH/Eurographics Symposium on Computer Animation. Copenhagen: Eurographics Association, 2014: 167-176.

[29] Borzeshi E Z, Concha O P, Da Xu R Y, et al. Joint Action Segmentation and Classification by an Extended Hidden Markov Model [J]. IEEE Signal Processing Letters, 2013, 20(12): 1207-1210.

[30] Liu Shenglan, Feng Lin, Liu Yang, et al. Manifold warp segmentation of human action [J]. IEEE Trans on Neural Networks and Learning Systems, 2018, 29(5): 1414-1426.

[31] Bobick A F, Davis J W. The recognition of human movement using temporal templates [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2001, 23(3): 257-267.

[33] Weinland D, Ronfard R, Boyer E. Free viewpoint action recognition using motion history volumes [J]. Computer Vision and Image Understanding, 2006, 104(2-3): 249-257.

[34] Yilmaz A, Shah M. Actions sketch: a novel action representation [C]// Proc of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005: 984-989.

[35] Dollar P, Rabaud V, Cottrell G, et al. Behavior recognition via sparse spatio-temporal features [C]// Proc of Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. 2006: 65-72.

[36] Matikainen P, Hebert M, Sukthankar R. Trajectons: action recognition through the motion analysis of tracked features [C]// Proc of IEEE International Conference on Computer Vision. 2009: 514-521.

[37] Laptev I. On space-time interest points [J]. International Journal of Computer Vision, 2005, 64(2-3): 10-123.

[38] Harris C. A combined corner and edge detector [C]// Proc of Alvey Vision Conference. Manchester: Alvey Vision Club, 1988: 147-151.

[40] Klaser A, Marszałek M, Schmid C. A spatio-temporal descriptor based on 3d-gradients [C]// Proc of the 19th British Machine Vision Conference. Leeds: BMVC Press, 2008: 275: 1-10.

[41] Dalal N, Triggs B. Histograms of oriented gradients for human detection [C]// Proc of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2005: 886-893.

[43] Dalal N, Triggs B, Schmid C. Human detection using oriented histograms of flow and appearance [C]// Proc of the European Conference on Computer Vision. Berlin: Springer, 2006: 428-441.

[44] Ojala T, Pietikäinen M, Mäenpää T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2000, 24(7): 971-987.

[45] Zhao Guoying, Pietikainen M. Dynamic texture recognition using local binary patterns with an application to facial expressions [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2007, 29(6): 915-928.

[46] Kellokumpu V, Zhao Guoying, Pietikainen M, et al. Human activity recognition using a dynamic texture based method [C]// Proc of the British Machine Vision Conference. Leeds: BMVC Press, 2008: 88. 1-88. 10.

[47] Norouznezhad E, Harandi M T, Bigdeli A, et al. Directional space-time oriented gradients for 3d visual pattern analysis [C]// Proc of the European Conference on Computer Vision. Berlin: Springer, 2012: 736-749.

[49] Sanin A, Sanderson C, Harandi M T, et al. Spatio-temporal covariance descriptors for action and gesture recognition [C]// Proc of IEEE Workshop on Applications of Computer Vision. 2013: 103-110.

[50] Jiang YuGang, Dai Qi, Xue Xiangyang, et al. Trajectory-based modeling of human actions with motion reference points [C]// Proc of the European Conference on Computer Vision. Florence: Springer-Verlag, 2012: 425-438.

[51] Wang Heng, Schmid C. Action recognition with improved trajectories [C]// Proc of IEEE International Conference on Computer Vision. 2014: 3551-3558.

[52] Wang Heng, Ullah M M, Klaser A, et al. Evaluation of local spatio-temporal features for action recognition [C]// Proc of the British Machine Vision Conference. London: BMVC Press, 2009: 124. 1-124. 11.

[53] Kovashka A, Grauman K. Learning a hierarchy of discriminative space-time neighborhood features for human action recognition [C]// Proc of Computer Vision and Pattern Recognition. 2010: 2046-2053.

[54] Dan O, Verbeek J, Schmid C. Action and event recognition with fisher vectors on a compact feature set [C]// Proc of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2013: 1817-1824.

[55] Peng Xiaojiang, Zou Changqing, Qiao Yu, et al. Action recognition with stacked fisher vectors [C]// Proc of the European Conference on Computer Vision. Cham: Springer, 2014: 581-595.

[56] Wang Heng, Kläser A, Schmid C, et al. Action recognition by dense trajectories [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2011: 3169-3176.

[57] Arandjelovic R, Zisserman A. All about VLAD [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2013: 1578-1585.

[58] Jain M, Jegou H, Bouthemy P. Better exploiting motion for better action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2013: 2555-2562.

[59] Xing Dong, Wang Xianzhong, Lu Hongtao. Action recognition using hybrid feature descriptor and VLAD video encoding [C]// Proc of IEEE Conference on Computer Vision. Singapore: Springer, 2014: 99-112.

[60] Kantorov V, Laptev I. Efficient feature extraction, encoding, and classification for action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2014: 2593-2600.

[61] Chen Sun, Nevatia R. Large-scale Web video event classification by use of Fisher Vectors [C]// Proc of IEEE Workshop on Applications of Computer Vision. Washington DC: IEEE Computer Society, 2013: 15-22.

[62] Zhu Yan, Zhao Xu, Fu Yuncai, et al. Sparse coding on local spatial-temporal volumes for human action recognition [C]// Proc of the Asian Conference on Computer Vision. Queenstown. Springer, 2011: 660-671.

[63] Guha T, Ward R K. Learning sparse representations for human action recognition [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2012, 34(8): 1576-1588.

[64] Somasundaram G, Cherian A, Morellas V, et al. Action recognition using global spatio-temporal features derived from sparse representations [J]. Computer Vision & Image Understanding, 2014, 123(7): 1-13.

[65] Sadanand S, Corso J J. Action bank: a high-level representation of activity in video [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2012: 1234-1241.

[66] Rabiner L R. A tutorial on hidden Markov models and selected applications in speech recognition [M]// Readings in Speech Recognition. 1990: 267-296.

[67] Lafferty J D, Mccallum A, Pereira F C N. Conditional random fields: probabilistic models for segmenting and labeling sequence data [C]// Proc of the 8th International Conference on Machine Learning. San Francisco: Morgan Kaufmann, 2001: 282-289.

[68] Hongeng S, Nevatia R. Large-Scale Event detection using semi-hidden Markov models [C]// Proc of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2003: 1455.

[69] Koller D, Tang K, Li F F. Learning latent temporal structure for complex event detection [C]// Proc of Computer Vision and Pattern Recognition. 2012: 1250-1257.

[70] Sun C, Nevatia R. ACTIVE: activity concept transitions in video event classification [C]// Proc of IEEE International Conference on Computer Vision. 2013: 913-920.

[71] Quattoni A, Wang S, Morency L P, et al. Hidden conditional random fields [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2007, 29(10): 1848-1852.

[72] Wang Yang, Mori G. Hidden part models for human action recognition: probabilistic versus max margin [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2011, 33(7): 1310-1323.

[73] Song Yale, Morency L P, Davis R. Action recognition by hierarchical sequence summarization [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2013: 3562-3573.

[74] Li Chuankun, Wang Pichao, Wang Shuang, et al. Skeleton-based action recognition using LSTM and CNN [C]// Proc of IEEE International Conference on Multimedia & Expo. 2017: 585-590.

[76] Donahue J, Hendricks L A, Guadarrama S, et al. Long-term recurrent convolutional networks for visual recognition and description [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2015: 2625-2634.

[77] Sun Lin, Jia Kui, Yeung D Y, et al. Human action recognition using factorized spatio-temporal convolutional networks [C]// Proc of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2015: 4597-4605.

[78] Ji Shuiwang, Yang Ming, Yu Kai. 3D convolutional neural networks for human action recognition [J]. IEEE Trans on Pattern Analysis & Machine Intelligence, 2013, 35(1): 221-231.

[79] Wang Xuanhan, Gao Lianli, Song Jingkuan, et al. Beyond frame-level cnn: Saliency-aware 3D CNN with lstm for video action recognition [J]. IEEE Signal Processing Letters, 2017, 24(4): 510-514.

[80] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos [C]// Advances in Neural Information Processing Systems. Montreal: NIPS Press, 2014: 568-576.

[81] Feichtenhofer C, Pinz A, Zisserman A. Convolutional Two-Stream Network Fusion for Video Action Recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2016: 1933-1941.

[82] Wang Limin, Qiao Yu, Tang Xiaoou. Action recognition with trajectory-pooled deep-convolutional descriptors [C]// Proc of the IEEE Conference on Computer Vision and Pattern Recognition. 2015: 4305-4314.

[83] Wang Limin, Xiong Yuanjun, Wang Zhe, et al. Temporal segment networks: Towards good practices for deep action recognition [C]// Proc of the European Conference on Computer Vision. Cham: Springer, 2016: 20-36.

[84] Chen Jiawei, Wu J, Konrad J, et al. Semi-coupled two-stream fusion convnets for action recognition at extremely low resolutions [C]// Proc of IEEE Winter Conference on Applications of Computer Vision. 2017: 246-253.

[85] Wang Xuanhan, Gao Lianli, Wang Peng, et al. Two-stream 3-D convNet fusion for action recognition in videos with arbitrary size and length [J]. IEEE Trans on Multimedia, 2018, 20(3): 634-644.

[86] Gammulle H, Denman S, Sridharan S, et al. Two stream LSTM: a deep fusion framework for human action recognition [C]// Proc of IEEE Winter Con-

ference on Applications of Computer Vision. 2017: 177-186.

[87] Zhao Rui, Ali H, van der Smagt P. Two-stream RNN//CNN for action recognition in 3D videos [J]. Intelligent Robots and Systems, 2017, 10(1): 426-433.

[88] Yan Xing, Chang Hong, Shan Shiguang, et al. Modeling video dynamics with deep dynencoder [C]// Proc of the European Conference on Computer Vision. Cham: Springer, 2014: 215-230.

[89] Srivastava N, Mansimov E, Salakhudinov R. Unsupervised learning of video representations using lstms [C]// Proc of International Conference on Machine Learning. 2015: 843-852.

[90] Goroshin R, Bruna J, Tompson J, et al. Unsupervised learning of spatiotemporally coherent metrics [C]// Proc of IEEE International Conference on Computer Vision. 2015: 4086-4093.

[91] Wang Xiaolong, Gupta A. Unsupervised learning of visual representations using videos [C]// Proc of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2015: 2794-2802.

[92] Misra I, Zitnick C L, Hebert M. Unsupervised learning using sequential verification for action recognition [J]. arXiv: Computer Vision and Pattern Recognition, 2016.

[93] Chopra S, Hadsell R, Lecun Y. Learning a Similarity Metric Discriminatively, with Application to Face Verification [C]// Proc of IEEE Computer Society Conference on Computer Vision and Pattern Recognition. Washington DC: IEEE Computer Society, 2005: 539-546.

[94] Jiwen Lu, Junlin Hu, YapPeng Tan. Nonlinear metric learning for visual tracking [J]. IEEE Trans on Circuits and Systems for Video Technology, 2016, 26(11): 2056-2068.

[95] Varior R R, Shuai B, Lu J, et al. A siamese long short-term memory architecture for human re-identification [C]// Proc of the European Conference on Computer Vision. Cham: Springer, 2016: 135-153.

[96] Wang Xiaolong, Farhadi A, Gupta A. Actions~transformations [C]// Proc of IEEE International Conference on Computer Vision and Pattern Recognition. 2016: 2658-2667.

Note: Figure translations are in progress. See original paper for figures.