T-STAM: An End-to-End Action Recognition Model Based on Two-Stream Spatio-Temporal Attention Mechanism Postprint

Authors: Shi Xiangbin, Li Yiying, Liu Fang, Dai Qin

Date: 2020-09-28T00:00:00+00:00

Abstract

To address the issues of ignoring interdependencies among feature channels and the presence of substantial redundant spatiotemporal information in features when employing the two-stream approach for video action recognition, we propose T-STAM, an end-to-end action recognition model based on a two-stream spatiotemporal attention mechanism that achieves full utilization of key spatiotemporal information in videos. First, a channel attention mechanism is introduced into the two-stream backbone network to calibrate channel information by modeling dependencies among feature channels, thereby enhancing feature representation capability. Second, we propose a CNN-based temporal attention model that learns attention scores for each frame using minimal parameters, focusing on frames with significant motion magnitude. Simultaneously, a multi-spatial attention model is proposed to calculate attention scores for each spatial location within every frame from multiple perspectives, extracting multiple motion-salient regions. Subsequently, spatiotemporal features are fused to further enhance video feature representation. Finally, the fused features are fed into a classification network, where the outputs of the two streams are integrated with different weights to obtain action recognition results. Experimental results on the HMDB51 and UCF101 datasets demonstrate that T-STAM can effectively recognize actions in videos.

Full Text

Preamble

Vol. 38 No. 3
Application Research of Computers
Accepted Paper

T-STAM: An End-to-End Action Recognition Model Based on Two-Stream Network with Spatio-Temporal Attention Mechanism

Shi Xiangbin[1],[2], Li Yiying[1]†, Liu Fang[2], Dai Qin[3]
(1. College of Information, Liaoning University, Shenyang 110036, China;
2. College of Computer Science, Shenyang Aerospace University, Shenyang 110136, China;
3. College of Information, Shenyang Institute of Engineering, Shenyang 110136, China)

Abstract: Existing two-stream methods for video action recognition often overlook inter-channel relationships among features and suffer from redundant spatio-temporal information. To address these issues, we propose an end-to-end action recognition model called T-STAM (Two-Stream Spatio-Temporal Attention Model) that fully utilizes key spatio-temporal information in videos. First, we introduce a channel attention mechanism into the two-stream backbone network to recalibrate channel information by modeling dependencies between feature channels, thereby enhancing feature representation capability. Second, we propose a CNN-based temporal attention model that learns attention scores for each frame with minimal parameters, focusing on frames with significant motion amplitude. Simultaneously, we introduce a multi-spatial attention model that calculates attention scores for each spatial location from multiple perspectives to extract multiple motion-salient regions. We then fuse spatio-temporal features to further enhance video representation. Finally, the fused features are fed into a classification network, and the outputs from both streams are combined with different weights to obtain the final action recognition result. Experimental results on the HMDB51 and UCF101 datasets demonstrate that T-STAM can effectively recognize actions in videos.

Keywords: action recognition; two-stream; channel information; spatio-temporal attention; motion saliency areas

0 Introduction

Action recognition has widespread applications in video surveillance, smart homes, video retrieval, human-computer interaction, and various other domains. Videos are characterized by complex environments, large viewpoint variations, and significant changes in human motion range, which introduce substantial redundant information in both temporal and spatial dimensions during feature representation. Therefore, effectively utilizing information from key regions on frames with significant motion—such as objects and body parts involved in human-object interactions—is crucial for action recognition.

Video-based action recognition methods can be categorized into two classes: traditional methods and deep learning-based methods. Traditional methods have achieved some progress but rely heavily on hand-crafted features, resulting

in limited generalization capability. Deep learning methods can automatically learn video features for classification. Among them, two-stream approaches effectively combine spatio-temporal information and demonstrate relatively superior performance. Simonyan et al. first proposed the two-stream model, feeding single RGB images and multi-frame optical flow fields into spatial and temporal streams respectively, followed by feature fusion and classification. Wang et al. introduced temporal segment networks using sparse sampling and video-level supervision to further improve accuracy. However, two-stream methods cannot effectively leverage key spatio-temporal information and ignore differences in information represented by different channels during feature extraction. To obtain salient region information, several studies employ object detection or pose estimation to extract multiple key regions or body parts before feeding them into networks for action recognition. However, pre-processing videos with object detection or pose estimation increases computational cost, and the quality of detection/estimation results affects recognition performance.

Attention mechanism-based action recognition methods can automatically learn key information from videos. Hu et al. designed a channel attention network to model channel-wise features and highlight important channels. Sharma et al. proposed a spatial attention model to emphasize salient regions in each frame. Du et al. used RNN-based temporal attention models to assign weights to different frames, enabling effective utilization of key frames. Yang et al. designed spatio-temporal attention models using bidirectional LSTM. However, existing approaches have several limitations: (a) RNN/LSTM-based temporal attention models contain numerous parameters and have fixed sequential structures that must process video frames in chronological order, resulting in low recognition efficiency. (b) When extracting spatial salient information, using only one spatial attention model to extract multiple motion regions often yields inaccurate region localization.

To address these problems, we propose an end-to-end action recognition model based on two-stream network with spatio-temporal attention mechanism (T-STAM). The contributions of T-STAM are as follows: (a) We integrate channel attention into the two-stream backbone network to recalibrate channel information while preserving two-stream features, thereby enhancing feature representation capability. (b) We propose a CNN-based temporal attention model to focus on temporally discriminative frames. Compared with RNN-based approaches, our model uses CNN to compute attention scores for each frame along the temporal dimension, resulting in fewer parameters and lower computational cost. Additionally, CNN enables parallel computation across multiple frames, improving overall efficiency. (c) We propose a multi-spatial attention model that employs multiple models to learn spatial location weights from different perspectives, obtaining multiple discriminative motion regions (e.g., human-object interactions, moving body parts) while reducing background interference. Spatio-temporal features are fused to further enhance video representation. (d) We conduct experimental validation on UCF101 and HMDB51 datasets, demonstrating that T-STAM is an efficient, end-to-end action recognition model.

1 Two-Stream Spatio-Temporal Attention Action Recognition Model

A video can be viewed as comprising spatial and temporal components. Spatially, RGB images contain scene and object appearance information, while temporally, optical flow images capture object motion information. Therefore, we design our model based on an appearance stream using RGB images and a motion stream using optical flow images. We propose T-STAM to strengthen feature representation, enabling discrimination between different channel features and focusing attention on multiple motion-salient regions within discriminative frames for action recognition. The overall architecture of T-STAM is shown in [Figure 1: see original paper]. To obtain appropriate input segments, T-STAM performs sparse sampling on videos: each video is divided into N equal intervals, with one frame randomly sampled from each interval. The RGB and optical flow images of sampled frames are then fed into the two-stream network.

T-STAM builds upon appearance and motion streams, with each stream containing three modules: SE-BN-Inception, spatio-temporal attention, and classification. The SE-BN-Inception module distinguishes differences in features represented by different channels, extracting expressive video features holistically. The appearance stream outputs F_{rgb} and the temporal stream outputs F_{flow} after this module. The spatio-temporal attention module further enhances video representation by highlighting discriminative frames and multiple motion-salient regions through temporal and multi-spatial attention models. The classification module consists of an FC layer and softmax function. Spatio-temporal features S_t and S_s from both streams are fed into their respective classification modules to obtain appearance stream output x_{rgb} and motion stream output x_{flow}. Final recognition results are obtained by fusing both stream outputs with different weights.

2 SE-BN-Inception Module

Convolutional networks produce multi-channel feature vectors when extracting video frame features, where each channel describes the frame from a specific aspect and channels vary in importance. However, previous deep learning methods ignore these differences, resulting in weak feature representation. Channel attention mechanisms can learn the importance of each feature channel, enhancing useful channels while suppressing less discriminative ones. Therefore, we introduce SE-Net (Squeeze-and-Excitation Networks) into the two-stream backbone BN-Inception to create the SE-BN-Inception module for recalibrating channel information and enhancing feature expressiveness.

SE-Net is shown in Figure 2: see original paper. The implementation proceeds as

follows: First, input features undergo global average pooling along the channel dimension to compress features. Two fully connected layers then model channel dependencies: the first FC layer reduces channel dimension to 1/16 of the original to reduce computation, followed by ReLU activation for non-linearity; the second FC layer restores the original dimension. A sigmoid function obtains normalized weights, which are then reweighted onto each channel' s features through feature recalibration.

The SE-BN-Inception module structure is shown in Figure 2: see original paper. BN-Inception contains 9 inception operations, with SE-Net added after each inception. Since FC layer outputs are insensitive to spatial location while convolutional outputs retain spatial structure to some extent, we preserve BN-Inception up to its final convolutional layer.

---

3 Spatio-Temporal Attention Module

The spatio-temporal attention module comprises a CNN-based temporal attention model, a multi-spatial attention model, and spatio-temporal feature fusion. The temporal and multi-spatial attention models focus on key frames and multiple motion-salient regions from temporal and spatial dimensions respectively, while feature fusion effectively combines these key spatio-temporal cues to further enhance video representation and improve action recognition accuracy.

3.1 CNN-Based Temporal Attention Model

Actions are continuous processes where different video frames contribute differently to recognition. Frames containing rich information with obvious motion changes should be prioritized. Temporal attention models assign greater attention to key frames. However, previous temporal attention models are RNN-based, featuring numerous parameters, complex structures, and inability to parallelize across time.

To address this, we propose a CNN-based temporal attention model that generates attention scores for each frame using CNNs. These scores determine each frame' s importance for action recognition, enabling selective focus on key frames and temporal feature enhancement. Our design has fewer parameters, a simpler structure, and can compute attention scores for all frames in parallel, fully leveraging GPU hardware. The CNN-based temporal attention model is shown in [Figure 3: see original paper].

Features after the SE-BE-Inception module are represented as $X \in \mathbb{R}^{N\times C\times W\times H}$, where N denotes the number of selected frames, C represents feature dimension, and $W \times H$ is the feature map grid size. For the i-th frame' s feature vector x_i, we first apply a fully connected layer for linear mapping to obtain $\hat{x}_i$ using shared parameters across frames, as shown in Equation (1):

$$\hat{x}_i = w_1 x_i + b_1, \quad i = 1, 2, \ldots, N$$

where w_1 and b_1 are learnable parameters. The mapped features for the entire video are $\hat{X} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_N)$. A convolutional layer with kernel size 1×1 transforms the video feature dimension to 256: $\hat{X} \in \mathbb{R}^{N \times 256}$. Softmax along the temporal dimension yields temporal attention scores α_i for each frame, computed as in Equation (2):

$$\alpha_i = \frac{\exp(\text{conv}(\hat{x}_i))}{\sum_{i=1}^{N} \exp(\text{conv}(\hat{x}_i))}$$

where conv represents the convolution operation. α_i indicates the i-th frame' s contribution to action recognition. After obtaining α_i, we multiply it with feature $\hat{x}_i$ to get the i-th frame's temporal feature. Summing all frames'temporal features yields the entire video' s temporal feature f_t, as in Equation (3):

$$f_t = \sum_{i=1}^{N} \alpha_i \hat{x}_i$$

3.2 Multi-Spatial Attention Model

Videos consist of sequential images where each frame spatially comprises motion-salient regions and other areas. For action recognition videos, motion-salient regions typically correspond to moving body parts and object locations. For example, in the "drinking" action, features from the arm, head region, and cup are sufficient for accurate recognition. Therefore, we should focus on these motion-salient regions in each frame. Previous methods using object detection or pose estimation are labor-intensive and complex. Spatial attention mechanisms can solve these issues, but existing approaches use only one spatial attention model to extract different salient region information, often yielding inaccurate localization.

To accurately extract different region information interacting with actions, we propose a multi-spatial attention model, detailed in [Figure 4: see original paper]. Rather than spatially decomposing input images based on feature map grid size, this model extracts frame spatial information from multiple perspectives, computing attention scores for each spatial location to identify different motion-salient regions. This approach reduces interference from irrelevant background information and mitigates issues caused by human pose variations, further enhancing spatial feature representation. The number of spatial attention models determines the quantity of learned motion-salient regions, with the optimal value determined experimentally.

We employ multiple spatial attention models to extract motion-salient regions. Each model primarily consists of two convolutional layers and a softmax function.

For the j-th spatial attention model, feature X first passes through a 1×1 convolutional layer with tanh activation to reduce dimension to $C/2$, decreasing computational cost. It then undergoes a second convolutional layer to obtain feature s_{ij}, implemented as in Equation (4). Batch Normalization (BN) is added after each convolutional layer to address covariate shift and stabilize training, with BN implementation shown in Equation (5):

$$s_{ij} = w_2^j * (\tanh(w_1^j * X + b_1^j)) + b_2^j$$

where $w_1^j, w_2^j, b_1^j, b_2^j$ are learnable network parameters. The second convolutional layer uses 5×5 kernels with stride 1. j indexes the spatial attention model number.

The feature s_{ij} after two convolutional layers is fed into softmax to compute probability scores α_{jk}^i for each spatial region in frame i, as in Equation (6):

$$\alpha_{jk}^i = \frac{\exp(s_{ijk})}{\sum_{k=1}^{W \times H} \exp(s_{ijk})}$$

We multiply α_{jk}^i with each mapped feature for element-wise multiplication to obtain weighted spatial features. With l spatial attention models, each frame extracts l spatial features. Summing the j-th spatial features across all selected frames yields the entire video' s j-th spatial feature f_{s_j}, as in Equation (7):

$$f_{s_j} = \sum_{i=1}^{N} \sum_{k=1}^{W \times H} \alpha_{jk}^i \hat{x}_i$$

3.3 Spatio-Temporal Feature Fusion

Spatio-temporal feature fusion combines extracted temporal and spatial features to determine human action categories. The fused spatio-temporal features represent changes in motion-salient regions (body parts, interactive objects, etc.) of key frames, further enhancing feature expressiveness for more accurate action recognition. For instance, in the "golf swinging" action, frames with obvious swinging motion receive more attention through the temporal attention model, while spatial attention models extract key regions like arms, golf clubs, and balls. Combining these features enables focused attention on multiple motion-salient regions in key frames for better action recognition.

Spatio-temporal feature fusion is illustrated in [Figure 5: see original paper]. After obtaining l spatial features f_{s_j} and temporal feature f_t, we first map each spatial feature onto the temporal feature by adding spatial features f_{s_j} to temporal feature f_t, yielding l features F_j. These features are then concatenated to obtain the video' s spatio-temporal feature F, as shown in Equations (8) and (9):

$$F_j = f_t + f_{s_j}$$

$$F = \text{concatenate}(F_1, F_2, \ldots, F_l)$$

where concatenate denotes the concatenation operation.

---

4 Experiments

4.1 Datasets and Evaluation Metrics

We evaluate our method on two publicly available video action recognition datasets: UCF101 and HMDB51.

UCF101 contains 13,320 videos across 101 action categories. The dataset exhibits strong diversity in motion acquisition, including camera motion, object appearance changes, pose variations, and background changes. Action categories are grouped into five types: human-object interaction, body motion, human-human interaction, playing musical instruments, and sports. The dataset presents challenges such as large intra-class variation and small inter-class differences. HMDB51 contains 6,676 videos across 51 action categories, primarily sourced from movies, YouTube, and Google videos, many of which are low-quality. Consequently, action recognition on these datasets is challenging. For both datasets, we adopt the official split: each dataset is divided into three splits, with 70% of videos for training and 30% for testing per split.

We use Top-1 recognition accuracy (hereinafter referred to as recognition accuracy) as the evaluation metric. The reported accuracy for each dataset is the weighted average across its three splits.

4.2 Experimental Setup

Experiments are conducted on GPU-enabled PyTorch. We use BN-Inception as the backbone, an upgraded version of GoogLeNet that balances accuracy and efficiency. The network is initialized with pre-trained ImageNet parameters. To align optical flow data with RGB data, we use the tool provided by Wang et al. to extract optical flow via the TV-L1 algorithm, quantizing the flow data to [0,255] through linear transformation.

Training: Input frames are resized to 240$\times$320, $then randomly cropped from fixed corners with horizontal flip$ $batch SGD with batch size 32, weight decay 0.0005, and momentum 0.9. The appearance stream learning rate starts a$ at epochs 30 and 60, for a total of 80 epochs. The motion stream learning rate starts at 0.001, decreasing by 10$\times$ at epochs 190 and 300, for a total of 340 epochs.

Testing: We select 25 frames per sample using average sampling. For each frame, data augmentation via cropping and flipping generates 10 test samples. The final classification result is obtained by averaging the output class probabilities across these 10 samples.

4.3 Experimental Analysis

This section presents comparative experiments analyzing video segment numbers, spatial attention model numbers, two-stream fusion weights, and the effectiveness of channel attention. Finally, we compare our method with state-of-the-art approaches.

4.3.1 Analysis of Action Recognition Performance with Different Video Segment Numbers Using TSN' s sparse sampling method, we analyze the impact of different video segment numbers on recognition performance. Experiments are conducted on the first split of HMDB51, sampling 3, 4, 5, and 6 segments per video. Results on the appearance stream are shown in [Figure 6: see original paper]. Recognition accuracy gradually increases with more segments, peaking at 6 segments, as the network learns from more samples. However, the improvement rate slows beyond 5 segments, and limited GPU memory prevents testing more segments. Therefore, we use 6 segments per video in subsequent experiments.

4.3.2 Analysis of Action Recognition Performance with Different Spatial Attention Model Numbers Our multi-spatial attention model extracts multiple motion-salient regions. As the number of models increases, more regions are extracted. We analyze this effect on the first split of HMDB51, with results shown in [Figure 7: see original paper]. Accuracy improves as the number increases to 4, peaking at 4 models, then declines at 5 models due to limited GPU memory for larger numbers. Thus, we adopt 4 spatial attention models in subsequent experiments.

4.3.3 Analysis of Action Recognition Performance with Different Two-Stream Fusion Weights We analyze the impact of different fusion weights between appearance and motion streams, as shown in . Using only the motion stream achieves higher accuracy than using only the appearance stream, and two-stream fusion outperforms single streams. The best results are obtained with a 1/4 RGB stream and 3/4 optical flow stream weighting. Therefore, we use a 1:3 fusion ratio for subsequent experiments.

4.3.4 Analysis of Action Recognition Performance with Channel Attention Network To validate the effectiveness of channel attention, we compare TSN integrated with SE-Net against standard TSN on both datasets, using identical experimental parameters. Results in show that SE-Net integration improves accuracy by 0.2% on UCF101 and 1.3% on HMDB51, demonstrating that

channel attention highlights discriminative channel information and enhances feature expressiveness.

4.3.5 Comparison with State-of-the-Art Methods 1) Comparison with Attention-Based Action Recognition Methods: We compare T-STAM (without SE-Net) against other attention-based methods in . Our method achieves higher accuracy: (a) Compared with RNN-based Temporal Attention, T-STAM (without SE-Net) improves HMDB51 accuracy by 6.3%, as Temporal Attention only extracts key frames while our method captures both key frames and spatial motion-salient regions, proving that spatio-temporal fusion enhances recognition. (b) T-STAM outperforms RSTAN and ISTPAN (both using BN-Inception backbones), showing our simpler yet more effective spatio-temporal attention model. (c) T-STAM surpasses Attention Cluster, Bi-LSTM Attention, and R-STAN, despite those methods using stronger ResNet backbones, proving our model compensates for BN-Inception' s limitations by accurately extracting key spatio-temporal information. (d) Adding SE-Net further boosts T-STAM' s accuracy on both datasets, showing channel attention recalibration improves performance.

2) Comparison with Recent Classical Action Recognition Methods: We compare T-STAM with classical methods in . Results show: (a) T-STAM outperforms traditional IDT, demonstrating effective key spatio-temporal information extraction and simplified end-to-end computation. (b) Compared with Two-Stream Fusion and TSN, T-STAM improves UCF101 accuracy by 3.2% and 0.8%, and HMDB51 accuracy by 6.5% and 2.5%, proving that our spatio-temporal attention model effectively extracts more motion features from key frames. (c) T-STAM also outperforms TDD, C3D, ST-ResNet, ST-Pyramid, ARTNet, and TSM, demonstrating that T-STAM' s channel recalibration and spatio-temporal attention model comprehensively mine key video information to obtain enhanced features for robust action description.

---

5 Conclusion

To address two-stream methods' limitations in ignoring inter-channel information differences and their inability to distinguish redundant frames and background information—resulting in weak overall feature expressiveness and low recognition rates—we propose an end-to-end action recognition model based on two-stream spatio-temporal attention mechanisms. We first integrate channel attention into the two-stream architecture to recalibrate channel information through channel-wise feature modeling, enhancing video feature expressiveness. We then design CNN-based temporal and multi-spatial attention models to focus on multiple motion-salient regions within discriminative frames, further strengthening video representation. Experiments on UCF101 and HMDB51 demonstrate superior accuracy compared to recent advanced methods, proving our model effectively distinguishes channel features and concentrates atten-

tion on key spatio-temporal information for more accurate action recognition. However, our motion and appearance streams currently share identical network structures, whereas human understanding of motion and appearance involves distinct processes that warrant different architectures. Future work will explore different network structures for each stream and investigate integration with other deep learning models to further improve accuracy.

References

[1] Li Ruifeng, Wang Liangliang, Wang Ke. A survey of human body action recognition [J]. Pattern Recognition and Artificial Intelligence, 2014, 27(1): 35-48.

[2] Wang Heng, Schmid C. Action recognition with improved trajectories [C]// Proc of IEEE International Conference on Computer Vision. 2013.

[3] Zhang Jie, Wu Jianzhang, Tang Jiali, et al. Human action recognition method based on spatio-temporal image segmentation and interactive area detection [J]. Application Research of Computers, 2017, 34(1): 302-305.

[4] Simonyan K, Zisserman A. Two-stream convolutional networks for action recognition in videos [J]. Advances in Neural Information Processing Systems, 2014, 1(4): 568-576.

[5] 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. Piscataway, NJ: IEEE Press, 2016: 1933-1941.

[6] Yue-Hei Ng J, Hausknecht M, Vijayanarasimhan S, et al. Beyond short snippets: Deep networks for video classification [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2015: 4694-4703.

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

[8] Wang Yifan, Song Jie, Wang Limin, et al. Two-Stream SR-CNNs for Action Recognition in Videos [C]// BMVC. 2016.

[9] Tu Zhigang, Xie Wei, Qin Qianqing, et al. Multi-stream CNN: Learning representations based on human-related regions for action recognition [J]. Pattern Recognition, 2018, 79: 32-43.

[10] Chéron G, Laptev I, Schmid C. P-cnn: Pose-based cnn features for action recognition [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2015: 3218-3226.

[11] Hu Jie, Li Shen, Gang Sun. Squeeze-and-excitation networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. 2018.

[12] Sharma S, Kiros R, Salakhutdinov R. Action recognition using visual attention [C]// Proc of International Conference on Machine Learning. 2015: 48-57.

[13] Liu Zhikang, Tian Ye, Wang Zilei. Improving human action recognition by temporal attention [C]// Proc of IEEE International Conference on Image Processing. 2017: 870-874.

[14] Du Wenbin, Wang Yali, Yu Qiao. Recurrent spatial-temporal attention network for action recognition in videos [J]. IEEE Transactions on Image Processing, 2017, 27(3): 1347-1360.

[15] Du Yang, Yuan Chunfeng, Li Bing, et al. Interaction-aware spatio-temporal pyramid attention networks for action classification [C]// Proc of European Conference on Computer Vision. Cham: Springer, 2018.

[16] Long Xiang, Gan Chuang, De Melo G, et al. Attention clusters: Purely attention based local feature integration for video classification [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2018: 7834-7843.

[17] Yang Haodong, Zhang Jun, Li Shuohao, et al. Bi-direction hierarchical LSTM with spatial-temporal attention for action recognition [J]. Journal of Intelligent & Fuzzy Systems, 2019, 36(1): 775-786.

[18] Liu Quanle, Che Xiangjiu, Mei Bie. R-STAN: Residual spatial-temporal attention network for action recognition [J]. IEEE Access, 2019, 7: 18073-18082.

[19] Yan Shiyang, Smith J S, Lu Wenjin, et al. CHAM: Action recognition using convolutional hierarchical attention model [C]// Proc of IEEE International Conference on Image Processing. 2017: 3958-3962.

[20] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift [J]. Computer Science, 2015.

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

[22] Tran D, Bourdev L, Fergus R, et al. Learning spatiotemporal features with 3D convolutional networks [C]// Proc of IEEE International Conference on Computer Vision. Washington DC: IEEE Computer Society, 2015: 4489-4497.

[23] Feichtenhofer C, Pinz A, Wildes R. Spatiotemporal residual networks for video action recognition [C]// Advances in Neural Information Processing Systems. 2016: 3468-3476.

[24] Wang Yunbo, Long Mingsheng, Wang Jianmin, et al. Spatiotemporal pyramid network for video action recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2017.

[25] Wang Limin, Li Wei, Li Wen, et al. Appearance-and-relation networks for video classification [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2018: 1430-1439.

[26] Lin Ji, Gan Chuang, Han Song. TSM: Temporal shift module for efficient video understanding [C]// Proc of IEEE International Conference on Computer Vision. 2019: 7083-7093.

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