MSR Asia MSM at ActivityNet Challenge 2017: Trimmed Action Recognition, Temporal Action Proposals and Dense-Captioning Events in Videos

Ting Yao, Yehao Li, Zhaofan Qiu, Fuchen Long, Yingwei Pan, Dong Li, and Tao Mei
Microsoft Research, Beijing, China
{tiyao, tmei}@microsoft.com

Abstract

This notebook paper presents an overview and comparative analysis of our systems designed for the following three tasks in ActivityNet Challenge 2017: trimmed action recognition, temporal action proposals and dense-captioning events in videos.

***Trimmed Action Recognition (TAR)**: We investigate and exploit multiple spatio-temporal clues for trimmed action recognition (TAR) task, i.e., frame, short video clip and motion (optical flow) by leveraging 2D or 3D convolutional neural networks (CNNs). The mechanism of different quantization methods is studied as well. Furthermore, improved dense trajectory with fisher vector encoding over the whole trimmed video is utilized. All activities are finally classified by late fusing the predictions of one-versus-rest linear SVMs learnt on each clue.*

***Temporal Action Proposals (TAP)**: To generate temporal action proposals from videos, a three-stage workflow is particularly devised for TAP task. Given an untrimmed video, our system firstly generates an actionness curve via a snippet-level actionness classifier. The temporal actionness grouping scheme is then exploited over actionness curve to produce proposal candidates. Finally, a proposal re-ranking procedure is incorporated to select high-quality proposals via a proposal-level actionness classifier.*

***Dense-Captioning Events in Videos (DCEV)**: For DCEV task, we firstly adopt our temporal action proposal system mentioned above to localize temporal proposals of interest in video, and then generate the descriptions for each proposal. Specifically, RNNs encode a given video and its detected attributes into a fixed dimensional vector, and then decode it to the target output sentence. Moreover, we extend the attributes-based CNNs plus RNNs model with policy gradient optimization and retrieval mechanism to further boost video captioning performance.*

1. Introduction

Recognizing activities in videos is a challenging task as video is an information-intensive media with complex variations. In particular, an activity may be represented by different clues including frame, short video clip, motion (optical flow) and long video clip. In this work, we aim at investigating these multiple clues to activity classification in trimmed videos, which consist of a diverse range of human focused actions. However, most of the natural videos in the real world are untrimmed videos with complex activities and unrelated background/context information, making it hard to directly recognize activities in them. One possible solution is to quickly localize temporal chunks in untrimmed videos containing human activities of interest and then conduct activity recognition over these temporal chunks, which largely simplifies the activity recognition for untrimmed videos. Generating such temporal action chunks in untrimmed videos is known as the task of temporal action proposals, which is also exploited in this work.

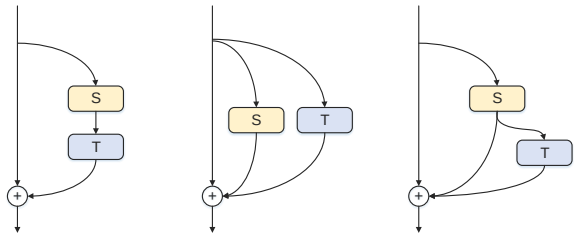
In addition to the above two tasks tailored to activity which is usually the name of action/event in videos, the task of dense-captioning events in videos is explored here which goes beyond activities by describing numerous events within untrimmed videos with multiple natural sentences.

The remaining sections are organized as follows. Section 2 presents all the features which will be adopted in our systems, while Section 3 details the feature quantization strategies. Then the descriptions and empirical evaluations of our systems for three tasks are provided in Section 4-6 respectively, followed by the conclusions in Section 7.

2. Video Representations

We extract the video representations from multiple clues including frame, short clip, motion and long clip.

Frame. To extract frame-level representations from video, we uniformly sample 25 frames for each video/proposal, and then use pre-trained 2D CNNs as



(a) P3D-A (b) P3D-B (c) P3D-C

Figure 1. Three Pseudo-3D blocks.

frame-level feature extractors. We choose the most popular 2D CNNs in image classification—ResNet [4].

Short Clip. In addition to frame, we take the inspiration from the most popular 3D CNN architecture C3D [18] and devise a novel Pseudo-3D Residual Net (P3D ResNet) architecture [15] to learn spatio-temporal video clip representation in deep networks. Particularly, we develop variants of bottleneck building blocks to combine 2D spatial and 1D temporal convolutions, as shown in Figure 1. The whole P3D ResNet is then constructed by integrating Pseudo-3D blocks into a residual learning framework at different placements. Our P3D ResNet model is pre-trained on Sports-1M dataset [5]. We fix 16 frames as the length of short clip, and sample rate is set to 25 per video.

Motion. To model the change of consecutive frames, we apply another CNNs to optical flow "image," which can extract motion features between consecutive frames. When extracting motion features, we follow the setting of [21], which fed 32 optical flow images, consisting of two-direction optical flow from 16 consecutive frames, into ResNet/P3D ResNet network in each iteration. The sample rate is also set to 25 per video.

Long Clip. For long/trimmed clip, we choose the state-of-the-art hand-crafted features—improved dense trajectory (iDT) [20] on each trimmed clip. Specifically, trajectory feature, histogram of oriented gradients (HOG), histogram of flow (HOF), and motion boundary histogram (MBH) are computed for each trajectory obtained by tracking points in video clips. Furthermore, Fisher vector encoding is used to quantize the features and create high dimensional representations for each clip.

3. Feature Quantization

In this section, we describe two quantization methods to generate video-level representations from frame-level or clip-level features.

Average Pooling. Average pooling is the most common method to extract video-level features from consecutive frames, short clips and long clips. For a set of frame-level or clip-level features $F = \{f_1, f_2, ..., f_N\}$, the video-level representations are produced by simply averaging all the features in the set:

$$R_{pooling} = \frac{1}{N} \sum_{i:f_i \in F} f_i \quad , \tag{1}$$

where $R_{pooling}$ denotes the final representations.

Deep Quantization. Moreover, we present our recently proposed network-based quantization method called Deep Quantization (DQ) [14]. A generative neural network with parameters θ is trained on the top of feature extraction network. Then, following the fisher kernel method, the video-level representations are defined as

$$\begin{aligned} &L_{Generative}(\theta) = \sum_{f \in TrainingSet} -log\, p(f, \theta) \\ &\widehat{\theta} = \arg\max_{\theta} L_{Generative}(\theta) \\ &R_{DQ} = normalize(\sum_{i:f_i \in F} \frac{\partial(-log\, p(f_i, \widehat{\theta}))}{\partial \widehat{\theta}}) \end{aligned} \quad , \tag{2}$$

where $p(f, \theta)$ is the generative network output. After optimizing parameters θ, the gradient calculating and accumulating can be processed end-to-end during backpropagation, no extra storage is required. To further improve the ability of representations, we propose a semi-supervised optimizing function as:

$$\begin{aligned} &L(\theta) = L_{Generative}(\theta) + \lambda L_{Classification}(\theta) \\ &\widehat{\theta} = \arg\max_{\theta} L(\theta) \\ &R_{DQ} = normalize(\sum_{i:f_i \in F} \frac{\partial(-log\, p(f_i, \widehat{\theta}))}{\partial \widehat{\theta}}) \end{aligned} \quad . \tag{3}$$

Readers can refer to [14] for more technical details of our deep quantization network.

4. Trimmed Action Recognition

4.1. System

Our trimmed action recognition framework is shown in Figure 2 (a). In general, the trimmed action recognition process is composed of three stages, i.e., multi-stream feature extraction, feature quantization and prediction generation. For deep feature extraction, we follow the multi-stream approaches in [8, 13], which represented input video by a hierarchical structure including individual frame, short clip and consecutive frame. In addition to deep features, one most complementary hand-crafted feature, i.e., iDT, is exploited to further enrich the video representations. After extraction of raw features, different quantization and pooling methods are utilized on different features to produce global representations of each trimmed video. Finally, a linear SVM is trained on each kind of video representations and the predictions from multiple SVMs are combined by linearly fusion. When training SVM, we fix $C = 100$.

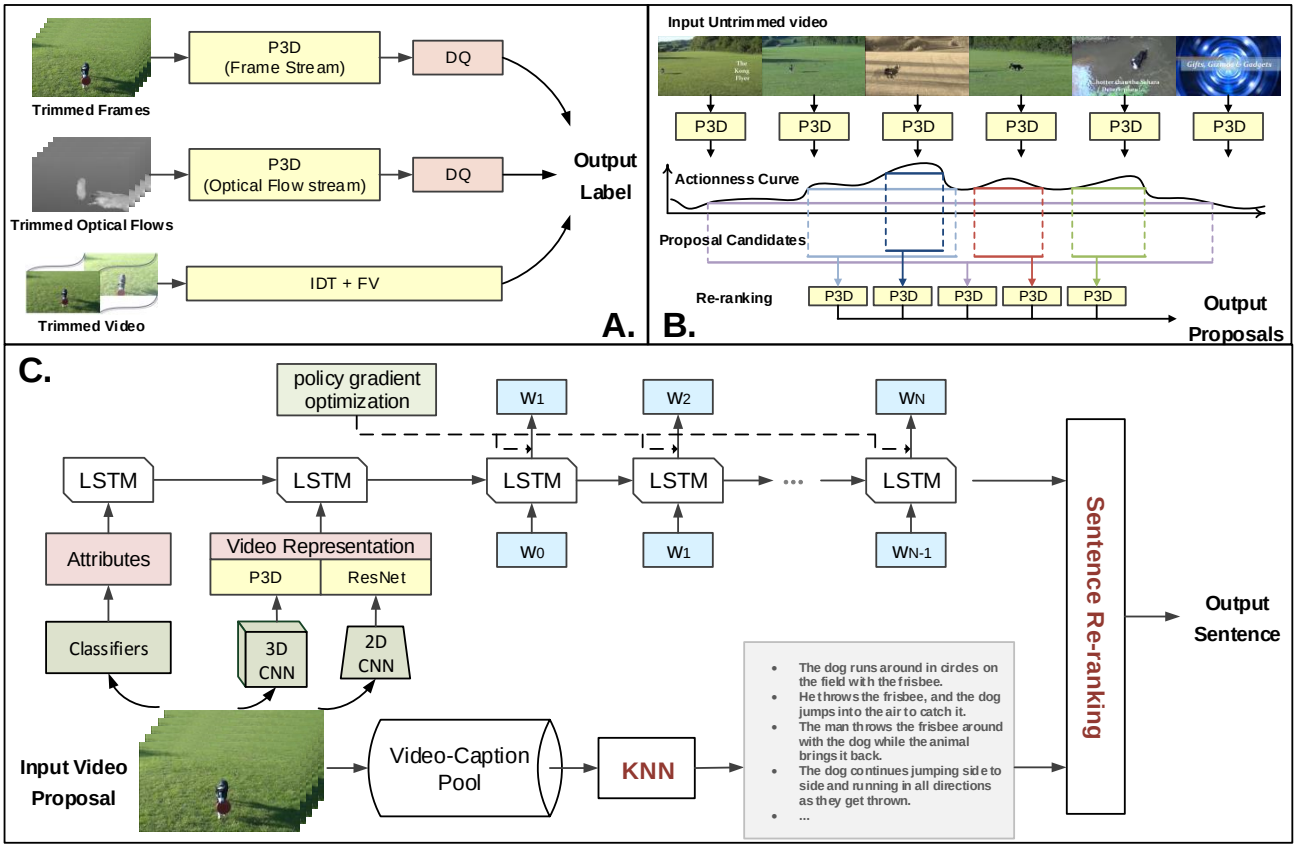


Figure 2. Frameworks of our proposed (a) trimmed action recognition system, (b) temporal action proposals system and (c) dense-captioning system.

4.2. Experiment Results

Table 1 shows the performances of all the components in our trimmed action recognition system. Overall, our Deep Quantization on P3D ResNet achieves the highest top1 accuracy (72.66%) and top5 accuracy (90.74%) of single component. For the final submission, we train the SVMs using training and validation sets. All the components are linearly fused using the weights tuned on validation set.

5. Temporal Action Proposals

5.1. System

Figure 2 (b) shows the framework of temporal action proposals, which is mainly composed of three stages:

Actionness curve generation. We treat every 16 continuous frames as one snippet and the stride size is 8 frames. Then, similar to video highlight detector in [22], a binary actionness classifier is trained over snippets to distinguish whether the snippets contain human activities. Accordingly, an actionness curve can be generated by accumulating all the actionness probabilities of snippets via snippet-level actionness classifier.

Temporal actionness grouping. Given an actionness curve, the classic watershed algorithm [17] is utilized to produce a set of "basins" corresponding to the temporal region with high actionness probability. Then, the temporal actionness grouping scheme [25] is leveraged to connect small basins, resulting in proposal candidates. Finally, the highly overlapped proposal candidates are filtered out via Non-maximal suppression.

Proposal re-ranking. To select the action proposals with high actionness probabilities, we additionally train the proposal-level actionness classifier to measure the actionness probability of each proposal candidate and then re-rank all the proposal candidates. In our experiments, only the top 100 proposals are finally outputted.

5.2. Experiment Results

Table 2 shows the results of actionness classifiers trained with different 2D/3D architectures (i.e., ResNet [4] and P3D ResNet [15]). Each 2D/3D architecture is pre-trained on different sources (e.g., ImageNet [2], Sports1M [5] and Kinetics [6]). For all the single stream runs w or w/o re-ranking scheme, the setting based on P3D ResNet pre-trained on Kinetics achieves the highest AUC. Moreover, by

Table 1. Comparison of different components in our framework on Kinetics validation set for trimmed action recognition task.

Stream	Feature	Layer	Quantization	Top1	Top5
Frame	ResNet	pool5	Ave	70.70%	89.75%
	ResNet	res5c	DQ	71.50%	90.20%
Short Clip	P3D ResNet	pool5	Ave	71.24%	90.01%
	P3D ResNet	res5c	DQ	**72.66%**	**90.74%**
Long Clip	iDT+FV	-	-	45.09%	69.73%
Motion	ResNet	pool5	Ave	59.84%	82.54%
	ResNet	res5c	DQ	61.03%	83.51%
	P3D ResNet	pool5	Ave	61.92%	84.19%
	P3D ResNet	res5c	DQ	63.24%	85.53%

Table 2. Area Under the average recall vs. average number of proposals per video Curve (AUC) of different 2D/3D architectures and pre-trained sources on ActivityNet validation set for temporal action proposals task.

Network	Pre-trained	Re-ranking	AUC
ResNet	ImageNet		56.96%
ResNet	Kinetics		59.75%
P3D ResNet	Sports1M		58.79%
P3D ResNet	Kinetics		59.90%
ResNet	ImageNet	$\surd$	59.03%
ResNet	Kinetics	$\surd$	60.13%
P3D ResNet	Sports1M	$\surd$	60.76%
P3D ResNet	Kinetics	$\surd$	61.13%
Fusion all			**63.12%**

additionally incorporating the re-ranking scheme, our system is consistently improved under different deep architectures. For the final submission, we fusion all the proposals from the eight streams with different settings and then select the top 100 proposals based on their weighted actionness probabilities. The linear fusion weights are tuned on validation set.

6. Dense-Captioning Events in Videos

6.1. System

The main goal of dense-captioning events in videos is jointly localizes temporal proposals of interest in videos and then generate the descriptions for each proposal/video clip. Hence we firstly leverage the temporal action proposal system described above in Section 5 to localize temporal proposals of events in videos (50 proposals for each video). Then, given each temporal proposal (i.e., video segment describing one event), our dense-captioning system runs two different video captioning modules in parallel—the generative module for generating caption via the LSTM-based sequence learning model, and the retrieval module which can directly copy sentences from other visually similar video segments through KNN. Finally, a sentence re-ranking module is exploited to rank and select the final most consensus caption from the two parallel video captioning modules by considering the lexical similarity among all the sentence candidates. The overall architecture of our dense-captioning system is shown in Figure 2 (c).

Generative module with LSTM. Taking inspiration from the recent successes of probabilistic sequence models leveraged in image/video captioning [9, 10, 11, 19, 23], we follow our previous state-of-the-art image captioning model [24] and formulate the generative video captioning module in an end-to-end fashion based on LSTM which encodes the given video segment and its detected attributes/categories into a fixed dimensional vector and then decodes it to the target output sentence. Specifically, the third design LSTM-A_3 in [24] which firstly encodes attribute representations into LSTM and then transforms video representations into LSTM at the second time step is adopted as the basic architecture. Here, we uniform sample 2 frames/clips per second for each video segment and each word in the sentence is represented as "one-hot" vector (binary index vector in a vocabulary). For the input video representations, we take the output of 2048-way $pool5$ layer from the ResNet [4] pre-trained on Kinetics dataset [6] and 2048-way $pool5$ layer from P3D ResNet [15] pre-trained on Sports-1M video dataset [5] as frame/clip representation respectively, and then concatenate the features from ResNet and P3D ResNet as the input video representation. For representation of attributes/categories, we treat all the 200 categories on Activitynet dataset [1] as the high-level semantic attributes and train the attribute detectors with our previous video classification system [12], resulting in the final 200-way vector of probabilities. The dimension of the input and hidden layers in LSTM are both set to 1,024.

Furthermore, different from the common training strategy with maximum likelihood estimation (MLE) in LSTM-A_3, we employ the policy gradient optimization method with reinforcement learning [16] to boost the video captioning performances specific to both CIDEr-D and METEOR metrics. Moreover, it should be noted that we additionally incorporate context information from other neighboring events into this generative module like [7].

Table 3. Performance of our proposed dense-captioning models on ActivityNet captions validation set, where B@N, M, R and C are short for BLEU@N, METEOR, ROUGE-L and CIDEr-D scores. All values are reported as percentage (%).

Model	B@1	B@2	B@3	B@4	M	R	C
LSTM-A_3	**17.50**	9.62	**5.54**	**3.38**	7.71	13.27	**16.08**
LSTM-A_3 + policy gradient	17.49	**9.73**	5.38	3.07	8.47	14.28	13.82
LSTM-A_3 + policy gradient + retrieval	17.27	9.70	5.39	3.13	**8.73**	**14.29**	14.75

Retrieval module with KNN. Another direction of image/video captioning is search-based approaches which "generate" sentence for an image/video by directly copying sentences from other visually similar images/videos. Although the approaches in this dimension cannot produce novel descriptions, it indeed can achieve human-level descriptions as all sentences are from existing human-generated sentences. Hence we design the retrieval module in this dimension to leverage the "crowdsourcing" human intelligence for producing diverse sentences from other angles. In particular, we utilize KNN to find the visually similar video segments based on the extracted video representations. The captions associated with the top similar video segments are regarded as sentence candidates in retrieval module. In the experiment, we mainly choose the top 300 nearest neighbors for generating sentence candidates.

Sentence re-ranking. Given the sentence candidates generated by generative and retrieval modules for input video segment, we need to re-rank all the sentence candidates and select the best one as the final output result. Inspired by [3], we treat the consensus sentence which has the highest average lexical similarity to the other candidates as the best one. Specifically, we linearly fuse two kinds of sentence similarities (i.e., CIDEr-D and METEOR) as the lexical similarity between two sentence candidates.

6.2. Experiment Results

Table 3 shows the performances of our proposed dense-captioning models. Here we compare three variants derived from our proposed dense-captioning framework. In particular, by additionally incorporating the policy gradient optimization scheme into the basic LSTM-A_3 architecture, we can clearly observe the performance boost in METEOR. Moreover, our dense-captioning model (LSTM-A_3 + policy gradient + retrieval) is further improved by injecting the sentence candidates from retrieval module in METEOR.

7. Conclusion

In ActivityNet Challenge 2017, we mainly focused on multiple visual features, different strategies of feature quantization and video captioning from different dimensions. Our future works include more in-depth studies of how fusion weights of different clues could be determined to boost the action recognition/temporal action proposals performance and how to generate open-vocabulary sentences for events in videos.

References

[1] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.

[2] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[3] J. Devlin, S. Gupta, R. Girshick, M. Mitchell, and C. L. Zitnick. Exploring nearest neighbor approaches for image captioning. *arXiv preprint arXiv:1505.04467*, 2015.

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[5] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[6] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[7] R. Krishna, K. Hata, F. Ren, L. Fei-Fei, and J. C. Niebles. Dense-captioning events in videos. *arXiv preprint arXiv:1705.00754*, 2017.

[8] Q. Li, Z. Qiu, T. Yao, T. Mei, Y. Rui, and J. Luo. Action recognition by learning deep multi-granular spatio-temporal video representation. In *ICMR*, 2016.

[9] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, 2016.

[10] Y. Pan, Z. Qiu, T. Yao, H. Li, and T. Mei. Seeing bot. In *SIGIR*, 2017.

[11] Y. Pan, T. Yao, H. Li, and T. Mei. Video captioning with transferred semantic attributes. In *CVPR*, 2017.

[12] Z. Qiu, D. Li, C. Gan, T. Yao, T. Mei, and Y. Rui. Msr asia msm at activitynet challenge 2016. In *CVPR workshop*, 2016.

[13] Z. Qiu, Q. Li, T. Yao, T. Mei, and Y. Rui. Msr asia msm at thumos challenge 2015. In *THUMOS'15 Action Recognition Challenge*, 2015.

[14] Z. Qiu, T. Yao, and T. Mei. Deep quantization: Encoding convolutional activations with deep generative model. In *CVPR*, 2017.

[15] Z. Qiu, T. Yao, and T. Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*, 2017.

[16] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel. Self-critical sequence training for image captioning. *arXiv preprint arXiv:1612.00563*, 2016.

[17] J. B. Roerdink and A. Meijster. The watershed transform: Definitions, algorithms and parallelization strategies. *Fundamenta informaticae*, 2000.

[18] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.

[19] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015.

[20] H. Wang and C. Schmid. Action recognition with improved trajectories. In *ICCV*, 2013.

[21] L. Wang, Y. Xiong, Z. Wang, and Y. Qiao. Towards good practices for very deep two-stream convnets. *arXiv preprint arXiv:1507.02159*, 2015.

[22] T. Yao, T. Mei, and Y. Rui. Highlight detection with pairwise deep ranking for first-person video summarization. In *CVPR*, 2016.

[23] T. Yao, Y. Pan, Y. Li, and T. Mei. Incorporating copying mechanism in image captioning for learning novel objects. In *CVPR*, 2017.

[24] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei. Boosting image captioning with attributes. In *ICCV*, 2017.

[25] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, D. Lin, and X. Tang. Temporal action detection with structured segment networks. *arXiv preprint arXiv:1704.06228*, 2017.