VireoJD-MM @ TRECVid 2019: Activities in Extended Video (ActEV)

Zhijian Hou[†], Ying-Wei Pan[⋆], Ting Yao[⋆], Chong-Wah Ngo[†]

[†] *Video Retrieval Group (VIREO), City University of Hong Kong*
http://vireo.cs.cityu.edu.hk

[⋆] *JD AI Research*

Abstract

In this paper, we describe the system developed for Activities in Extended Video(ActEV) task at TRECVid 2019 [1] and the achieved results.

Activities in Extended Video(ActEV): The goal of Activities in Extended Video is to spatially and temporally localize the action instances in a surveillance setting. We have participated in previous ActEV prize challenge. Since the only difference between the two challenges is evaluation metric, we maintain previous pipeline [2] for this challenge. The pipeline has three stages: object detection, tubelet generation and temporal action localization. This time we extend the system for two aspects separately: better object detection and advanced two-stream action classification. We submit 2 runs, which are summarised below.

- VireoJD-MM_Pipeline1: This run achieves Partial AUDC=0.6012 using advanced two-stream action classification. It has been recognized in many papers [3, 4] that two-stream structure increases action recognition performance. In our prize challenge model, we only use RGB frames as input. For the submission this time, we extend the action classification stage into an advanced two-stream action classification module.

- VireoJD-MM_SecondarySystem: This run achieves Partial AUDC=0.6936 using better object detection model. The CMU team released the groundtruth of object bounding box provided by Kitware as well as their object detection and tracking code[1] based on VIRAT dataset. They build a system to detect and track small objects in outdoor scenes for surveillance videos. For the submission this time, we replace our object detection and tracking code with their code and keep the remaining stages of tubelet generation and temporal action localization.

1 Activities in Extended Video(ActEV)

1.1 Framework

We adopt a three-stage system to automatically detect and temporally localize all instances of given activities in the video. The system is composed of object detection [5], tubelet generation and temporal activity localization [6] stages. We first apply object detection algorithm to localize associated objects related to activities in videos, and then utilize object tracking method to link the detected objects into a

[1] https://github.com/JunweiLiang/Object_Detection_Tracking

long-term tubelet. Finally, we employ sliding window method to generate temporal activity proposals in each tubelet and Pesudo-P3D algorithm to classify every proposals into corresponding categories. Figure 1 illustrates the framework of our proposed system.

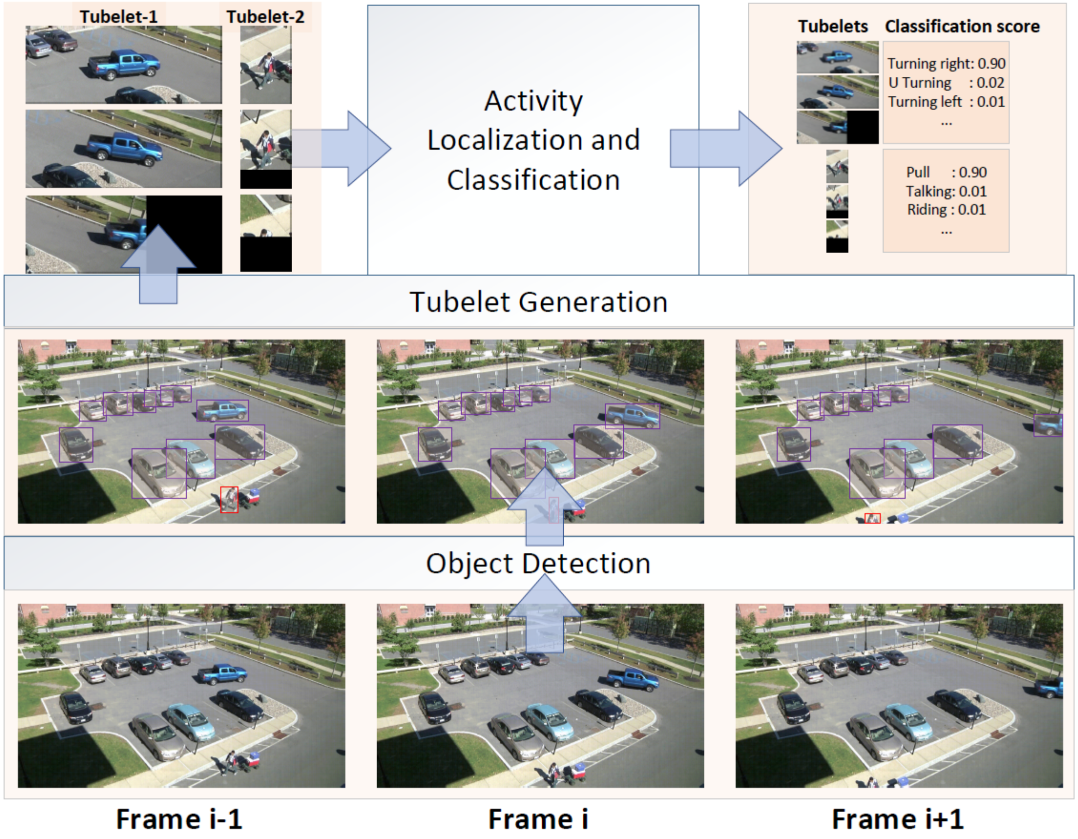

Figure 1: Framework of our three-stage system for Activity Detection in Extended Videos.

1.2 VireoJD-MM_Pipeline1

In this run, we focus on the action classification part. For the stage of object detection and tubelet generation, readers can refer to our ActivityNet paper [2]. After tubelet jittering, we get temporal action proposals with various durations. The next thing is to classify each temporal proposals as corresponding activities or background. This time we use both RGB frames and optical flow as input following two-stream inflated 3D ConvNets [7]. As we did before, we exploit Pseudo-P3D framework to process 64 image frames to get spatial stream result. Additionally, we also employ P3D to process optical flow to get temporal stream result. Finally, we average those two results and obtain final classification score. We expect that advanced two-stream action classification can improve the performance of activity classification, thus leading to overall performance gain.

1.3 VireoJD-MM_SecondarySystem

In this run, we concentrate on the object detection process. This year the ActEV organizers only provide the annotation of bounding box during the period of action. In price challenge, we conclude that the finetuned detection model on the partial annotation actually degrades detection performance. Luckily, CMU team releases full annotation of object bounding box which they got from Kitware previously and also their detection algorithm[1] finetuned on this full annotation. We start to learn their code and expect it can shed light on proper approaches to achive detection performance improvement for VIRAT dataset, then we integrate their code with our code, hoping to witness overall performance boost.

Unfortunately, it is difficult to compare detection performance among different methods since there is no general consensus about evaluation details for VIRAT dataset. The first issue is lack of official train/evaluation split for training and evaluation. The split of COCO dataset is about 115k train frames and 5k validation frames. The ratio of train frames to valiation frames is about 23. However, the total frame number of train videos and evaluation videos are 267,139 and 201,953 respectively. Problems, like improper ratio of train frames to valiation frames and long evaluation time, will occur if we simply use frames of train videos to train and frames of evaluation videos to evaluate. What's more, we should also lay down the evaluation metric, more specifically whether we should adopt the COCO evaluation metric or PASVAL VOC evaluation metric. Additionally, we should define what kind of object categories with which proposed models train and evaluate as well.

CMU team indeed solve above three issues. As for dataset split, they first filter out the frames of train videos with no object annotation, leaving 241,517 frames out of initial 267,139 frames. Then they carefully select 5 train videos (10,737 frames) out of the whole 64 train videos as the validation set of object detecion stage and the remaining 59 train videos (230,780 frames) are used as the training set of object detecion stage. No original evaluation video is involved. In regard to evaluation metric, they declare that the ground truth bounding boxes are not accurate/tight and opt to adopt PASVAL VOC evaluation metric. With regard to object categories, detailed information about the category names for different settings can be obtained from table 1. They apply comprehensive categories to train proposed models and the models then evaluate on five dominant categories involved in activities: vehicle, person, prop, push_pulled_object, bike.

	Object Category Names
Comprehensive categories for VIRAT datset	Other, Person, Prop, Push_Pulled_Object, Vehicle, Bike, Door, Dumpster, Parking_Meter, Tree, Animal, Construction_Vehicle, Trees, Skateboard and Construction_Barrier
Categories involved in activities	Person, Prop, Push_Pulled_Object, Vehicle , Bike, Door and Construction_Vehicle
Adopted COCO categories of last challenge	Person, Car, Truck and Bicycle

Table 1: Object category names under three settings. The whole annotation provided by Kitware have fifteen object categories. The annotation given this year only contains seven object categories involved in activities. Last row show adopted COCO category names in last challenge since we only exploit pretrained models on COCO.

Following CMU evaluation metric, we compare the performance of finetuned detection model using full annotations. The method of last challenge without pretraining adopts COCO object category, which is incomparable with the setting. We have tested two tries: reproduction of their work and finetuned algorithm of last challenge. But the results of both are worse than that of CMU released finetuned model. We reckon the inferiority comes from our inferior training skills and the fact that CMU team adds dilated CNN [8] in backbone besides the Faster-RCNN and FPN module under Mask-RCNN pipeline.

Their code integrates both object detection and object tracking. And the object tracking part borrows one multi-object tracking open-sourced code, named simple online and realtime tracking with a deep association metric(Deep SORT) [9]. For simplicity, we replace both object and tracking part with their code. For the stage of temporal action localization and classification, readers can refer to our ActivityNet paper [2].

1.4 Performance comparison with price challenge work

We compare above two improved models with previous price challenge work in this section. Since the primary evaluation metric changes this time, we will use former primary evaluation metric (mean-w_p_miss@0.15rfa) for comparison. The less the value of mean-w_p_miss@0.15rfa is, the better the final performance is.

Through advanced two-stream action classification, mean-w_p_miss@0.15rfa decreases from 0.7682 to 0.7285. We can see that some improvement indeed happens. We reckon it is becauese the result of action classification have direct influence on whether predicted action instance can match corresponding groundtruth instance, thus straight affecting the evaluation metric as well. It convinces us that we should pay more attention to the stage of temporal action localization.

As for the second run based on better object detection, to our surprise, mean-w_p_miss@0.15rfa increases from 0.7682 to 0.77857. We guess that there are several reasons. First, unfortunately, even adopted with full annotation and the performance improves, the detection performance under surveillance videos is still not satisfying. CMU team only use the detection boxes belonging to person and vehicle to form the tubelet later. Other categories, such as prop, bike and so on, are not considered due to unsatisfactory performance. They also observe that newly proposed ideas on COCO dataset don't really work for this surveillance setting[1]. Second, the detection performance is an interim evaluation, which has indirect effect on final evaluatin metric. Third, new tracking method may lag the performance. Fourth, ActEV organizers have slightly changed the computation of this evaluation metric.

We believe the performance of current model can be improved if equiped with advanced tracking method. We can observe some person and vehicle trajectories spilt into several segments using current tracking method. So that one groundtruth long-term tubelet will be predicted as several small-duration tubelets. When we further apply temporal action localization to those small-duration tubelets, the duration of predicted action instances will be relatively short. This will reduce the chance that those predicted action instances match original long-duration groundtruth action instances, thus affecting final evaluation metric.

1.5 Conclusion and future work

We experimented two-stream network and more rigorous object detection model in this challenge. Compared to previous submission, only two-stream network improves performance. In the future, we will study multi-target tracking and focus on temporal action localization and classification.

References

[1] G. Awad, A. Butt, K. Curtis, Y. Lee, J. Fiscus, A. Godil, A. Delgado, A. F. Smeaton, Y. Graham, W. Kraaij, and G. Quénot, "Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval," in *Proceedings of TRECVID 2019*. NIST, USA, 2019.

[2] F. Long, Q. Cai, Z. Qiu, Z. Hou, Y. Pan, T. Yao, and C.-W. Ngo, "VireoJD-MM at Activity Detection in Extended Videos," *arXiv preprint arXiv:1906.08547*, 2019.

[3] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," in *Advances in neural information processing systems*, 2014, pp. 568–576.

[4] C. Feichtenhofer, A. Pinz, and A. Zisserman, "Convolutional two-stream network fusion for video action recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1933–1941.

[5] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[6] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5533–5541.

[7] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[8] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *arXiv preprint arXiv:1511.07122*, 2015.

[9] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *2017 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2017, pp. 3645–3649.