# **BUPT-MCPRL at TRECVID 2018: ACtEV and INS**\*

Xinkun Cao, Zhipeng Wang, Peng Li, Xuanlu Xiang, Xinyu Li, Ya Li, Shaoqiang Xu, Liang Zhou, Zhicheng Zhao, Yanyun Zhao, Fei Su

Multimedia Communication and Pattern Recognition Lab Beijing Key Laboratory of Network System and Network Culture, Beijing Beijing University of Posts and Telecommunications, Beijing, China

{zhaozc, zyy, sufei}@bupt.edu.cn

# ABSTRACT

In this paper, we describe BUPT-MCPRL systems and evaluation results for TRECVID 2018 [15]. We join two tasks: activities in extended video and instance search.

Activities in Extended Video (ActEV): we mainly improve our method in following three runs:

- p\_baseline\_1: the baseline of our method. In this submission, we combine the prior knowledge and design different rules to extract proposals. Then we detect events by applying trajectory analysis, image classification and action classification solutions for different type of events separately.
- p\_baseline\_17: use TRN replace TSN for Open\_Trunk and Closing\_Trunk event detection.
- p\_baseline\_19: fuse the target object detection results for interaction event detection.

Instance Search (INS): We submit three runs for automatic search and one run for interactive search:

- F\_E\_BUPT\_MCPRL\_1: merge two rank list from location retrieval after person retrieval and person retrieval after location retrieval.
- F\_E\_BUPT\_MCPRL\_2: fine tune the rank score of F\_E\_BUPT\_MCPRL\_1 based on random forest classification extra score.
- F\_E\_BUPT\_MCPRL\_3: fine tune the rank score of F\_E\_BUPT\_MCPRL\_1 based on random forest classification extra score.
- I\_E\_BUPT\_MCPRL\_4: optimize rank list interactively based on F\_E\_BUPT\_MCPRL\_1.

# **1** Activities in Extended Video

In ActEV evaluation, we submit our results of all events except two events: "specialized\_talking\_phone" and "specialized\_texting\_phone". It is difficult to detect event in videos directly due to expensive computation cost and limited annotation. Considering that all events involve the person or vehicle, we firstly detect and track persons and vehicles in videos to narrow related regions. To combine the prior knowledge, we design different rules for different events to extract proposals. We divide events into different states (such as Pull and Riding) or actions (such as Opening and Closing). The states can be discriminated by poses and context information. But for actions, we consider motion patterns and action classification methods. The proposed method wins the 3<sup>th</sup> place on AD leaderboard and the 2<sup>nd</sup> place on AOD leaderboard.

# 1.1 Person and vehicle detection

Faster R-CNN [1] is the most prevalent two-stage object detection framework. We use it for person detection but with some modifications as shown in Figure 1. ResNet-50 [2] is our backbone network, and all convolution layers are shared to generate feature maps. Then, we extract the feature with ROIAlign layer and classify each ROI by extra fully-connected layers. In addition, the OHEM algorithm is used to improve feature representations. It selects comparatively difficult samples with a large loss value in the forward pass, and only these samples are valid to update weights of model

<sup>\*</sup>This work is supported by Chinese National Natural Science Foundation (61471049, 61532018), Key Laboratory of Forensic Marks, Ministry of Public Security of China.

in the backward pass. We train our model with fully-annotated videos, in which all pedestrians are labeled by bounding box. For vehicles detection, we use YOLO model [3].

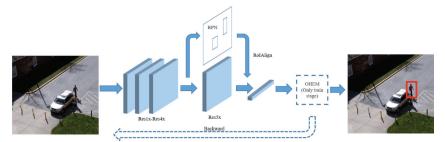


Figure 1. The architecture of our person detector.

# 1.2 Tracking

Our multiple object tracker is extended from [4]. We use it for pedestrian tracking and vehicle tracking but we replace the appearance feature with the re-identification feature. For complex variations of viewpoints, poses, lighting, occlusions and background clutter, it's hard to describe pedestrians with color histogram [4], since we use a CNN model to extract more stable features. We train our model on public re-identification datasets such as Market1501 and CUHK03. The same appearance model is adopted to track vehicles. Afterwards, we use Gaussian process regression to smooth trajectories [5].

## **1.3 Event detection**

The proposed event detection framework is illustrated in Figure 2. Based on detection and tracking results, proposals for different events are extracted by using hand-crafted rules. Besides, we also take differences of events into account and adopt different solutions to classify them. We detect the vehicle turning event by trajectory analysis. As for the other events, we divide them into key pose class and action class, and use image classification method and action classification method separately.

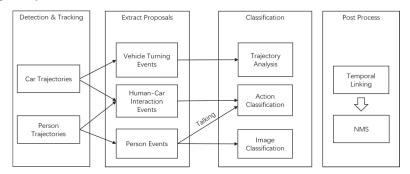


Figure 2. Event detection framework.

### **1.3.1 Vehicle turning**

Vehicle turning events consist of left turn, right turn and u-turn event. We detect the events based on variation rules of trajectory direction. All vehicle trajectories are considered as proposals. Given a vehicle track S, we sample N points  $\{p_0, p_1, p_2, ..., p_{N-1}\}$  with an equal interval. The direction of line between points  $p_{i-1}$  and  $p_i$  is

$$\theta_i = tan^{-1}[(y_i - y_{i-1})/(x_i - x_{i-1})] , \qquad (1)$$

where  $(x_i, y_i)$  is the coordinate of  $p_i$ , i=1,2,...,N-1. The variation of  $\theta_i$  is

$$\Delta \theta_i = \theta_i - \theta_{i-1} \quad , \tag{2}$$

If there are continuous K points that satisfy

$$\Delta \theta_k > T_1 \quad , \tag{3}$$

and  $K > T_K$ , then one turning event between point 1 and point K is detected.  $T_1$  denotes the threshold of direction change we set, k=1,2,...,K in (3), and  $T_K$  is another threshold. Finally, we calculate

$$\Delta \theta = |\theta_K - \theta_1|. \tag{4}$$

The rule that we discriminate vehicle turning event is

$$\begin{cases} u - \text{turn,} & \text{if } \Delta \theta > T_2 \\ \text{left turn,} & \text{if } \Delta \theta \le T_2 \text{ and } \theta_K - \theta_1 > 0 \\ \text{right turn,} & \text{if } \Delta \theta \le T_2 \text{ and } \theta_K - \theta_1 < 0 \end{cases}$$

Generally, u-turn events have a larger direction change than other turning events. For the events occurring on the boundary of scenes, we draw some critical regions and detect these events when vehicles pass through them.

#### 1.3.2 Human-vehicle interaction events

We regard Opening, Closing, Entering, Exiting, Loading, Unloading, Open Trunk and Closing Trunk as human-vehicle interaction events. Since above events involve the interaction between person and vehicle, we extract the proposals in which the vehicle and the person are close to each other. Specifically, we firstly enlarge the width and height of the vehicle detection results, and then extract proposals where the IoU between person and vehicle bounding boxes is greater than 0.5. Furthermore, we consider that is an Exiting or Entering proposal when a person trajectory appearing or disappearing surrounding a vehicle. We treat human-vehicle interaction events as the action class and use action classification method to recognize them. To locate the event, we split all proposal to small temporal segments and use action classification model to predict the score for each segment. We first use a binary-classification TSN [6] model to filter out most negative segments. Then we train different action classification models to capture the motion feature and temporal information. We employ them for different events to classify each segment, i.e. CDC [7] for Opening and Closing, TSN for Loading and Unloading, TRN [8] for Open\_Trunk and Closing Trunk, and 3D-ResNet [9] for Entering and Exiting, and remove these segments if the confidence is less than 0.5. Finally, we temporally link the segments to get event detection result and use Non-Maximum Suppression (NMS) to remove duplicate results.

## 1.3.3 Person events

Person events include Talking, Interacts, Transport\_HeavyVehiclery, Pull, Activity\_Vehiclering and Riding event. They all belong to the key pose class except the Talking event, as it is hard to judge if two persons are talking or just passing by each other with appearance feature. Therefore, we adopt action classification method for Talking event detection. Considering persons usually gather together and do not move widely when talking, we first calculate the velocity of each frame in all person trajectories, and find out the still points where the velocity is lower than a constant threshold v. Then, we gather those still points with high IoU overlap and get the proposals. Finally, we train a 3D-Resnet model and use it to predict the score of each proposal.

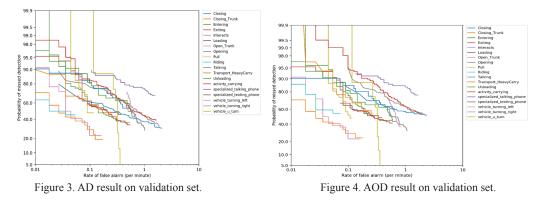
For the remaining 5 events, we treat them as the key pose class action. All person trajectories are extracted as proposals for each event, but the ones with low velocity are removed for Riding event. In order to balance the positive and negative samples for Transport\_HeavyVehiclery, Pull and Activity\_Vehiclering events, we train a 4-class (no-action or one of the three actions) ResNet to

recognize each frame of all proposal trajectories and truncate the trajectory according to the score Besides. considering that target objects to refine proposals. are helpful for Transport HeavyVehiclery, Pull and Activity Vehiclering events, we use motion foreground detection methods based on inter-frame differences to expand person bounding boxes to contain both person and target object. Then, we train a ResNet classifier for each event to predict the confidence of each frame of all proposals. After that, we keep the frame of which the confidence is greater than 0.5. For Interacts event, the target objects also provide important context information. However, they usually do not move along with the person, which makes foreground detection methods not work. Therefore, we train FPN-Faster-RCNN [10] to detect target objects, which are interacted with person in the scene. At last, we link the frames to temporal sequence and use Non-Maximum Suppression (NMS) to remove duplicate results.

### 1.4 Results and conclusion

We evaluate our method on the validation set and results of AD and AOD, and the results are shown in Figure 3 and Figure 4. In addition, we achieve similar metrics on test dataset and win the 3<sup>th</sup> place on the AD leaderboard and the 2<sup>nd</sup> place on the AOD leaderboard. We can see that our method has good performance on the human-vehicle interaction events, but it does not work well on some events with small movement, such as Interacts and Talking.

In this year ActEV evaluation, we attempt some solution based on action classification and have achieved good results. In the future, we will explore the action detection scheme and utilize the temporal and spatial relation between person and other objects.



# **2** Instance Search

This year, we propose a similar search framework for both automatic and interactive search tasks. Video key frames with a sample rate of 2 fps are extracted for retrieval. To retrieve specific persons in specific locations, we consider which methods and features are more appropriate for locations or person retrieval respectively. The results are summarized in Table 1. More details will be given in the following sections.

Table 1. Results for each run	
Run ID	mAP
F_E_BUPT_MCPRL_1	37.0
F_E_BUPT_MCPRL_2	38.0
F_E_BUPT_MCPRL_3	37.8
I_E_BUPT_MCPRL_4	44.7

#### 2.1 Location retrieval

For locations retrieval, we use two independent methods to extract features. The first one extracts local and global features to describe the image. In our experiment, we use Hessian-Affine detector with RootSIFT descriptor, MSER detector with RootSIFT descriptor and CNN features extracted from conv5 of AlexNet as local features. Then we adopt Bag-of-Words (BoW) model on local features to represent the images. As for global features, we adopt spatial pyramid pooling [11] and Gram matrix weighting [12] for convolutional layers of ResNeXt. For the second method, we fine-tune publicly available VGG-16 model, GoogleNet model and ResNet-152 models, which have all been trained on Places365 dataset, to fit with the task. We then extract the features of the fully connected layer for location retrieval. Subsequently, feature fusion scheme is followed to improve the initial retrieval performance.

## 2.2 Person retrieval

With regard to person retrieval, we use two kinds of methods including face retrieval and transcript-based search.

For face retrieval, we detect face on key frames captured from the video by MTCNN [13]: face representations are extracted from bounding-box of face based on MTCNN model, and then cosine distance is employed to match faces. Considering that some faces in given queries are hard to be detected due to occlusion or blurring, we apply query expansion to re-process retrieval for top *N* retrieval results.

For transcript-based search, we locate the query character name in transcripts. As a result, the corresponding shots of transcripts will get scores in the final rank.

#### 2.3 Merge results

To combine results of location retrieval and person retrieval, we first retrieve locations based on location queries and set a location threshold to determine which frames are relative to the location query. Then we conduct person retrieval in these frames. Correspondingly, we first retrieve persons and then conduct location retrieval.

Since we can get two alternative ranks by using different order of locations retrieval and person retrieval, we then conduct a rank fusion. We cross the rank results from tow orders and take higher rank position for the same frame. We then apply some re-ranking skills based on person re-identification, transcript and random forest classification to the final scores. For random forest classification, we conduct binary classification for each frame based on multiple scores of location and person scores. We train a random forest model based on 2016 instance search topics. Afterwards, these frames, if their classification labels are 1, will be added to the final scores.

Finally, we consider the maximum frame score as the shot score and rank the video shots for evaluation.

# 2.4 Conclusion

This year, we optimize last year's retrieval system [14] from multiple perspectives. We optimize multiple convolutional neural networks and extraction schemes to get more powerful feature representation. We also improve the accuracy of person retrieval by introducing an effective face detection method. In the next year, we will explore more discriminative feature representations and more powerful re-rank skills.

# References

[1] Ren, Shaoqing, et al. "Faster r-cnn: Towards real-time object detection with region proposal networks." Advances in neural information processing systems. 2015.

[2] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

[3] Redmon, Joseph, and Ali Farhadi. "Yolov3: An incremental improvement." arXiv preprint arXiv:1804.02767 (2018).

[4] Bae S H, Yoon K J. Robust Online Multi-object Tracking Based on Tracklet Confidence and Online Discriminative Appearance Learning[C]// Computer Vision and Pattern Recognition. IEEE, 2014:1218-1225.

[5] Zhu Y, Zhou K, Wang M, et al. A comprehensive solution for detecting events in complex surveillance videos[J]. Multimedia Tools and Applications, 2018: 1-22.

[6] L. Wang, Y. Xiong, Z. Wang, Y. Qiao, D. Lin, X. Tang, and L. van Gool, "Temporal segment networks: Towards good practices for deep action recognition," Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9912 LNCS, pp. 20–36, 2016.

[7] Z. Shou, J. Chan, A. Zareian, K. Miyazawa, and S.-F. Chang, "CDC: Convolutional-De-Convolutional Networks for Precise Temporal Action Localization in Untrimmed Videos," no. 1, 2017.

[8] Zhou, Bolei, Alex Andonian, and Antonio Torralba. "Temporal relational reasoning in videos." arXiv preprint arXiv:1711.08496 (2017).

[9] K. Hara, H. Kataoka, and Y. Satoh, "Can Spatiotemporal 3D CNNs Retrace the History of 2D CNNs and ImageNet?," 2017.

[10] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature Pyramid Networks for Object Detection," Dec. 2016.

[11] He, Kaiming, et al. "Spatial pyramid pooling in deep convolutional networks for visual recognition." European Conference on Computer Vision. Springer, Cham, 2014.

[12] Wang Z, Xiang X, Zhao Z, et al. Deep Image Retrieval: Indicator and Gram Matrix Weighting for Aggregated Convolutional Features. ICME'2018, 2018: 1-6.

[13] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[14] Zhao, Zhicheng, et al. Bupt-mcprl at trecvid 2017. Proc. TRECVID. 2017.

[15] George Awad, Asad Butt, Keith Curtis et.al. TRECVID 2018: Benchmarking Video Activity Detection, Video Captioning and Matching, Video Storytelling Linking and Video Search. Proceedings of TRECVID 2018.