Person-level Action Recognition in Complex Events via TSD-TSM Networks

Yanbin Hao
City University of Hong Kong
haoyanbin@hotmail.com

Zi-Niu Liu
Fudan University
ziniuliu@outlook.com

Hao Zhang[§]
City University of Hong Kong
zhanghaoinf@gmail.com

Bin Zhu
City University of Hong Kong
binzhu4-c@my.cityu.edu.hk

Jingjing Chen
Fudan University
chenjingjing@fudan.edu.cn

Yu-Gang Jiang
Fudan University
ygj@fudan.edu.cn

Chong-Wah Ngo
City University of Hong Kong
cscwngo@cityu.edu.hk

ABSTRACT

The task of person-level action recognition in complex events aims to densely detect pedestrians and individually predict their actions from surveillance videos. In this paper, we present a simple yet efficient pipeline for this task, referred to as TSD-TSM networks. Firstly, we adopt the TSD detector for the pedestrian localization on each single keyframe. Secondly, we generate the sequential ROIs for a person proposal by replicating the adjusted bounding box coordinates around the keyframe. Particularly, we propose to conduct straddling expansion and region squaring on the original bounding box of a person proposal to widen the potential space of motion and interaction and lead to a square box for ROI detection. Finally, we adapt the TSM classifier on the generated ROI sequences to perform action classification and further adopt late fusion to promote the prediction. Our proposed pipeline achieved the 3rd place in the ACM-MM 2020 grand challenge, i.e., Large-scale Human-centric Video Analysis in Complex Events (Track-4), obtaining final 15.31% wf-mAP@avg and 20.63% f-mAP@avg on the testing set.

CCS CONCEPTS

• **Computing methodologies** → **Activity recognition and understanding**.

KEYWORDS

Human action recognition; pedestrian detection; complex events

ACM Reference Format:
Yanbin Hao, Zi-Niu Liu, Hao Zhang[§], Bin Zhu, Jingjing Chen, Yu-Gang Jiang, and Chong-Wah Ngo. 2020. Person-level Action Recognition in Complex Events via TSD-TSM Networks. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3394171.3416276

[§] Hao Zhang is the corresponding author.

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3416276

1 INTRODUCTION

We present elaboration and discussion of our solution for the action recognition track (Track-4) in ACM-MM 2020 grand challenge on large-scale human-centric video analysis in complex events. The used dataset is a new large-scale human-centric dataset, named Human-in-Events (HiEve), which is built for understanding a hierarchy of human motions, poses, and actions in a variety of realistic complex events, especially crowded & complex events [8]. The action recognition track is to simultaneously detect specific individuals and assign correct action labels on every sampled frame (an interval of 20) from a long-length surveillance video. This task differs from the related AVA challenge [4] in that it mainly focuses on the actions of one identity under complex event conditions, such as "walking-together", "queuing", "standing-alone", etc., rather than only human-human and human-object interactions.

The two major issues of action recognition track are the localization of pedestrians and prediction of actions. Currently, approaches address the former by applying an object detector, e.g., region proposal network (RPN) [10], and the latter by using a classification network, e.g., I3D [1]. The general pipeline mainly contains three processes: (1) to detect the person regions on the center frame (or its feature map) of a video clip, (2) to generate a ROI sequence for each person by replicating the box proposal around the center frame, and (3) to pass the ROI sequence through a 3D CNN to obtain the action label. Regarding the ROI detection, those approaches can be broadly divided into two categories. One is to detect the ROIs on the feature map of the center frame, as shown in Figure 1(a). The other is to detect the ROIs on the original keyframe, as shown in Figure 1(b). The aligned ROIs in time are thus wrapped into a person-level feature/frame cube, based on which the action prediction can be performed. By comparing the two strategies, the second one can produce ROIs with larger resolution [14, 15] and performs better in this track.

In this paper, we follow the second strategy (Figure 1(b)) to generate person regions and go into improving the system performance

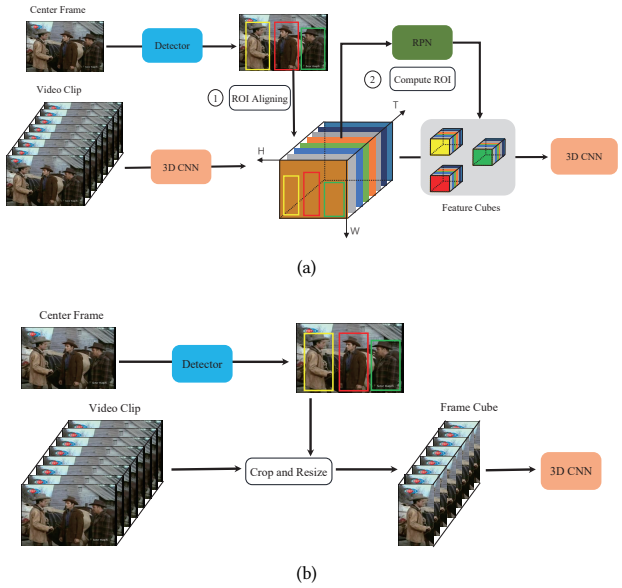


Figure 1: Pipeline architectures. In (a), ROIs are extracted on the feature map and the bounding box is detected by either ① or ②. In (b), both ROIs and bounding boxes are extracted on the keyframe.

by introducing sophisticated solutions. In particular, these solutions are for addressing three important issues, including accurate pedestrian localization, appropriate person-level ROI cube generation, and efficient action prediction. More specifically, we detect pedestrians on keyframes by finetuning the much effective task-aware spatial disentanglement (TSD) [12] model on the HiEve dataset. Considering the interactions and occlusions among individuals, as well as the profile change of a moving person over time, we appropriately expand the bounding box of a proposal by such as straddling expansion and region squaring to involve more context information into the ROIs and lead to square regions. Afterwards, square cropping and black padding operations are for producing two frame-level ROI cubes for each person proposal, on which the temporal shift module (TSM) [7] is thus applied for action classification. The final predicted action label of each proposal is based on the combination of the two resulting confidence scores.

2 METHOD

Our person-level action recognition system is equipped with TSD and TSM. Figure 1(b) illustrates the architecture of the TSD-TSM system. In the followings, we explain the proposed pipeline in detail on aspects of pedestrian detection by TSD, straddling expansion and region squaring for box adjustment and ROIs detection, and action prediction by TSM.

2.1 Pedestrian detection

We adopt a variant of the Faster-RCNN network named Task-Aware Spatial Disentanglement (TSD) networks as the pedestrian detector [9, 12].

Compared with the common faster-rcnn pipeline that recognizes an object's category and regresses its location using the same ROI feature yielded by a sibling head, TSD-networks conduct recognition and regression on two separate modified ROI features based on original ROI area. Particularly, the content classification and shape regression have different preferences for regional features: features in salient areas contain rich information for classification, while features around boundary benefits bounding box regression. By separating the gradient flows of category classification and location regression, TSD-networks achieves a higher categorization score and a more accurate box than before. Experimental results verify that TSD networks can accurately localize crowded tiny pedestrians in surveillance videos.

2.2 Straddling expansion and region squaring

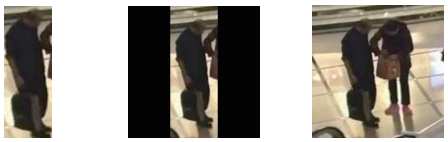
Straddling expansion. We observe that directly extending the bounding box of a proposal in time by replicating may cause to disappear from the extreme front and back boxes. Also, we consider that the surroundings of a person in complex scenes should be involved in the action prediction, as this person-level action recognition track contains more interaction and occlusion among individuals. For example, to identify the action of "walking-together", gathering information cannot be well recognized only through a single person. Based on above, we first introduce a straddling expansion method for bounding box refinement. Specifically, letting $\{w_1, h_1, w_2, h_2\}$ denote axis coordinates of a bounding box proposal, b, obtained from TSD, where (w_1, h_1) is for the upper-left corner and (w_2, h_2) is for the lower-right corner, the straddling expansion computes a new bounding box $\hat{b}$ with a new set of coordinates, $\left\{\hat{w}_1, \hat{h}_1, \hat{w}_2, \hat{h}_2\right\}$, as follows:

$$\hat{w}_1 = \max(0, w_1 - \alpha\,(w_2 - w_1)), \tag{1}$$
$$\hat{h}_1 = \max(0, h_1 - \alpha\,(h_2 - h_1)), \tag{2}$$
$$\hat{w}_2 = \min(W_f, w_2 + \alpha\,(w_2 - w_1)), \tag{3}$$
$$\hat{h}_2 = \min(H_f, h_2 + \alpha\,(h_2 - h_1)), \tag{4}$$

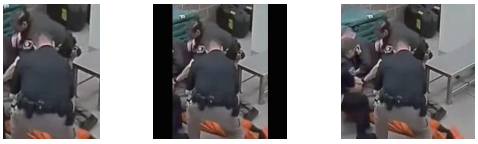
where α is the expansion coefficient, and W_f and H_f are the width and height of the video frame respectively. The straddling expansion enlarges the bounding box in both width and height, and coefficient α controls the expansion degree.

Region squaring. The obtained bounding box is mostly a rectangular box. Instead of cropping rectangular regions on a frame, we propose to further square the adjusted bounding box with the original ratio of width/height kept. Given the width of $\hat{b}$ as $\hat{W}_b = \hat{w}_2 - \hat{w}_1$ and its height as $\hat{H}_b = \hat{h}_2 - \hat{h}_1$, $\hat{b}$ is further adjusted as follows:

$$\hat{\hat{W}}_b, \hat{\hat{H}}_b = \max(\hat{W}_b, \hat{H}_b), \tag{5}$$
$$\hat{\hat{w}}_1 = \max(0, \hat{w}_1 - \frac{1}{2}\left(\hat{\hat{W}}_b - \hat{W}_b\right)), \tag{6}$$
$$\hat{\hat{h}}_1 = \max(0, \hat{h}_1 - \frac{1}{2}\left(\hat{\hat{H}}_b - \hat{H}_b\right)), \tag{7}$$
$$\hat{\hat{w}}_2 = \min(W_f, \hat{w}_2 + \frac{1}{2}\left(\hat{\hat{W}}_b - \hat{W}_b\right)), \tag{8}$$
$$\hat{\hat{h}}_2 = \min(H_f, \hat{h}_2 + \frac{1}{2}\left(\hat{\hat{H}}_b - \hat{H}_b\right)). \tag{9}$$



(a) Action: walking-together



(b) Action: crouching-bowing

Figure 2: Examples of region croppings. The left cropping is based on the original bounding box, the center cropping is based on black padding operation, and the right cropping is based on the square cropping operation.

This operation results in a (approximately) square region for each person proposal, corresponding to the newly computed bounding box $\hat{\hat{b}}$ with its coordinates as $\{\hat{\hat{w}}_1, \hat{\hat{h}}_1, \hat{\hat{w}}_2, \hat{\hat{h}}_2\}$.

Based on $\hat{\hat{b}}$, we can directly crop ROIs on a sequence of frames and generate a ROI sequence for the person proposal. This cropping strategy in fact may expand the proposal region in a large scale, especially for ones with very high/low width-height ratio, such as "a standing person", "a lying person", etc. We infer this can benefit much the action understanding in complex events through involving in more potential contexts. Besides, we also propose to build a black image with the same size of $\hat{\hat{b}}$ and insert $\hat{b}$ to the center of the black image. This strategy is expected to focus much on the person proposal itself when predicting actions. For description convenience, we refer the first cropping operation to as **square cropping** and the second one to as **black padding**. Figure 2 gives example results of the above operations.

2.3 Action prediction

Both the square cropping and black padding produce a ROI for each person proposal. We solely extend each of the two ROIs from the center frame (keyframe) to the front/back T frames, and thus obtain two ROI cubes for every person proposal. Let $\mathbf{C}^{sq}$ and $\mathbf{C}^{bp}$ denote the cube by using square cropping and black padding respectively, where the number of frames in $\mathbf{C}^{sq}$ and $\mathbf{C}^{bp}$ is $2T$+1. Afterwards, we adopt the TSM [7] equipped with non-local operator [13] to perform action classification separately on the two ROI cubes. Hence, we have

$$\boldsymbol{s}^{sq} = TSM(\mathbf{C}^{sq}) \tag{10}$$

$$\boldsymbol{s}^{bp} = TSM(\mathbf{C}^{bp}). \tag{11}$$

Both the two confidence score vectors, $\boldsymbol{s}^{sq}$ and $\boldsymbol{s}^{bp}$, are the action predictions of one person proposal. We sum them with equal weights (0.5), and have the final confidence score vector $\boldsymbol{s}$ as

$$\boldsymbol{s} = 0.5\boldsymbol{s}^{sq} + 0.5\boldsymbol{s}^{bp}. \tag{12}$$

The bounding box coordinates of b and scores $\boldsymbol{s}$ are finally involved into the challenge submissions.

3 EVALUATION

3.1 Dataset and metrics

The HiEve dataset [8] contains 32 video sequences with most of them longer than 900 frames. The total time length is 33 minutes and 18 seconds. And it has 56,643 action annotations corresponding to 14 action categories. Among of these videos, the first 19 (1-19) videos are for training and the left 13 (20-32) videos are for testing.

This challenge track [8] uses two groups of metrics to measure performance. The first group contains f-mAP@α and f-mAP@avg. f-mAP@α ($\alpha \in \{0.5, 0.6, 0.75\}$) evaluates spatial action detection accuracy on a single frame with an IOU threshold α and f-mAP@avg is the mean value of all f-mAP@α. The second group contains wf-mAP@α and wf-mAP@avg. Slightly different to f-mAP@α, wf-mAP@α ($\alpha \in \{0.5, 0.6, 0.75\}$) assigns appropriate weights to different action categories and wf-mAP@avg is the mean value of all wf-mAP@α.

3.2 Implementation

Implementation details of the presented three operations are elaborated in the followings separately. There is only one parameter α used in straddling expansion and region squaring, and we empirically set α to 0.1. Below, the experimental settings in pedestrian detection and action prediction are given in detail.

In pedestrian detection, we adopt TSD networks[1] with FPN-ResNext101-64-4d backbone in training pedestrian detector. We randomly split training videos into 1:3 val/train sets to find optimal settings, then fix them for a complete training. An optimal settings are presented as below. We fine-tune TSD networks on the HiEve dataset from pre-trained weights on the CrowdHuman dataset[11], the total number of training epoch is 5 with lr=10e-5. Multi-scale training [800, 1200] and testing augmentation (flip/scales) are adopted. Evaluations on val set shows that TSD pedestrian detector achieve 41.4% mAP@0.5:095, 80.4% mAP@0.5. During inference, we keep person boxes with confidence score above 0.85.

In action prediction, the number of frames in each unidirectional extension is set as $T = 10$ and as a result there is a total of 21 frames in each ROI cube. The action predictor, TSM networks[2] (with non-local operator [13]) under ResNet-50 [5], is first pretrained on Kinetics-400 dataset [6] and then finetuned on the ROI cubes extracted from the training videos of HiEve. During the finetuning, we averagely segment a cube into 8 sub-cubes and randomly sample a frame from each sub-cube. The 8 sampled frames are further resized with the shorter-side set as 256 and the original aspect ratio is kept. We also conduct random cropping (in range $[1, 0.875, 0.75]$) for data augmentation and then resize each frame to 224×224. The initial learning rate is set to 0.001 for SGD training and the total epoch is 25 (decays by 0.1 at epochs 5 and 15). In testing, center cropping in range 1 is adopted.

[1] https://github.com/Sense-X/TSD

[2] https://github.com/mit-han-lab/temporal-shift-module

Table 1: Performance comparison for different methods on HiEve dataset. "SC" and "BP" indicate square cropping and black padding respectively. The best performance is boldfaced.

Group	Method	wf-mAP@avg	wf-mAP@0.5	wf-mAP@0.6	wf-mAP@0.75	f-mAP@avg	f-mAP@0.5	f-mAP@0.6	f-mAP@0.75
I	RPN+I3D	6.88	9.65	7.91	3.07	8.31	11.01	9.65	4.26
II	Faster R-CNN+I3D	10.13	13.35	11.57	5.49	10.95	14.50	12.33	6.01
	Transformer+I3D	7.28	9.88	8.32	3.65	7.03	9.32	8.10	3.66
III	TSD-TSM (with original ROI)	6.98	9.21	8.12	3.62	8.66	11.16	10.09	4.72
	TSD-TSM+SC	14.85	19.45	17.51	7.60	20.03	25.99	23.39	10.72
	TSD-TSM+BP	13.06	17.22	15.33	6.64	16.76	21.82	19.22	9.23
	TSD-TSM+SC+BP (final submission)	**15.31**	**19.88**	**17.97**	**8.07**	**20.63**	**26.45**	**24.14**	**11.30**

3.3 Results

We group the compared methods into (I) baseline-1: PRN+I3D [2] that detects ROIs on the feature map and generates feature-level ROI cubes for proposals; (II) baseline-2: Faster R-CNN+I3D [8] (an improved version of PRN+I3D) and Transformer+I3D [3] that detect ROIs on the keyframe and also generate feature-level ROI cubes for proposals; (III) the proposed methods: TSD-TSM versions that detect ROIs on the keyframe and generate frame-level ROI cubes for proposals.

We show the performance comparison for different methods in Table 1. The results of methods in Group-I and -II are from the work [8]. From this table, we can find that all the methods in Group-II and -III outperform the baseline-1 in Group-I which is a strong baseline in AVA challenge. This gives clues that detecting ROIs on keyframe is a better strategy than on feature map with respect to this dataset. We speculate that this is because the scenario in HiEve is complex and crowded and simple 3D CNN feature will downsample the resolution of ROI features. Comparing the results between methods in groups II and III, most of the TSD-TSM based methods (except the first one in Group-III) show higher performance. This observation in a sense verifies claim that generating frame-level cubes are more appropriate than feature-level cubes for this dataset. Among the four methods in Group-III, using square cropping and/or black padding to adjust ROI shows consistently much better performances than the method with the original ROI, from 6.98% to 13.06%/14.85%/15.31%. Comparing the results between TSD-TSM+SC and TSD-TSM+BP, using square cropping outperforms the one with black padding. And further fusing the two results (TSD-TSM+SC+BP) exhibits the best performance in terms of all metrics. We speculate that since the scenes in HiEve contain complex interactions between persons, appropriately enlarge the region of proposal could provide more information for understanding person-level actions.

4 CONCLUSION

In this work, we have presented our solutions for the person-level action recognition track in the challenge of large-scale human-centric video analysis in complex events. The proposed TSD-TSM pipeline adopts much effective object detector model (TSD) and video understanding model (TSM) to achieve accurate pedestrian localization and action prediction respectively. The designed straddling expansion and region squaring operations work properly to generate ROIs for person proposals, which can appropriately involve the context information. Our final submission obtains the top-3 results in this challenge and achieve a significant improvement of over 50% on wf-mAP@avg and 90% on f-mAP@avg over the best baseline (Faster R-CNN+I3D).

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China, No. 61872256.

REFERENCES

[1] Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6299–6308.
[2] Rohit Girdhar, João Carreira, Carl Doersch, and Andrew Zisserman. 2018. A better baseline for ava. *arXiv preprint arXiv:1807.10066* (2018).
[3] Rohit Girdhar, Joao Carreira, Carl Doersch, and Andrew Zisserman. 2019. Video action transformer network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 244–253.
[4] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. 2018. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6047–6056.
[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.
[6] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. 2017. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950* (2017).
[7] Ji Lin, Chuang Gan, and Song Han. 2019. Tsm: Temporal shift module for efficient video understanding. In *Proceedings of the IEEE International Conference on Computer Vision*. 7083–7093.
[8] Weiyao Lin, Huabin Liu, Shizhan Liu, Yuxi Li, Guo-Jun Qi, Rui Qian, Tao Wang, Nicu Sebe, Ning Xu, Hongkai Xiong, et al. 2020. Human in Events: A Large-Scale Benchmark for Human-centric Video Analysis in Complex Events. *arXiv preprint arXiv:2005.04490* (2020).
[9] Yu Liu, Guanglu Song, Yuhang Zang, Yan Gao, Enze Xie, Junjie Yan, Chen Change Loy, and Xiaogang Wang. 2020. 1st Place Solutions for OpenImage2019–Object Detection and Instance Segmentation. *arXiv preprint arXiv:2003.07557* (2020).
[10] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*. 91–99.
[11] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. 2018. CrowdHuman: A Benchmark for Detecting Human in a Crowd. *arXiv preprint arXiv:1805.00123* (2018).
[12] Guanglu Song, Yu Liu, and Xiaogang Wang. 2020. Revisiting the Sibling Head in Object Detector. *CVPR* (2020).
[13] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. 2018. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7794–7803.
[14] Hao Zhang and Chong-Wah Ngo. 2016. Object Pooling for Multimedia Event Detection and Evidence Localization. *Ite Transactions on Media Technology & Applications* 4, 3 (2016), 218–226.
[15] Hao Zhang and Chong-Wah Ngo. 2018. A Fine Granularity Object-Level Representation for Event Detection and Recounting. *IEEE Transactions on Multimedia* 21, 6 (2018), 1450–1463.