Exploring Stronger Feature for Temporal Action Localization

Zhiwu Qing[1,2] Xiang Wang[1,2] Ziyuan Huang[2] Yutong Feng[2] Shiwei Zhang[2*]
Jianwen Jiang[2] Mingqian Tang[2] Changxin Gao[1] Nong Sang[1*]
[1]Key Laboratory of Image Processing and Intelligent Control
School of Artificial Intelligence and Automation, Huazhong University of Science and Technology
[2]Alibaba Group
{qzw, wxiang, cgao, nsang}@hust.edu.cn
{pishi.hzy, yutong.fyt, zhangjin.zsw, jianwen.jjw, mingqian.tmq}@alibaba-inc.com

Abstract

Temporal action localization aims to localize starting and ending time with action category. Limited by GPU memory, mainstream methods pre-extract features for each video. Therefore, feature quality determines the upper bound of detection performance. In this technical report, we explored classic convolution-based backbones and the recent surge of transformer-based backbones. We found that the transformer-based methods can achieve better classification performance than convolution-based, but they cannot generate accuracy action proposals. In addition, extracting features with larger frame resolution to reduce the loss of spatial information can also effectively improve the performance of temporal action localization. Finally, we achieve 42.42% in terms of mAP on validation set with a single SlowFast [9] feature by a simple combination: BMN [16]+TCANet [19], which is 1.87% higher than the result of 2020 [20]'s multi-model ensemble.

1. Introduction

Temporal action localization is a challenging task, especially for HACS dataset [27], which contains complex relationships between actors and scenes in long videos. In this technical paper, we explore two kinds of backbones, *i.e.*, Transformer-based ViViT [1] and Timesformer [3], CNN-based SlowFast [9] and CSN [21]. From the experiment results, we draw several following conclusions: 1) The features extracted by the network with remarkable classification performance may not necessarily generate high-quality proposals for temporal action localization. Since the action classification task does not have to be sensitive to the background. For instance, the action that occurs on the football field is likely to play football. The network may only focus on the football field, and there is no need for the act of playing football. 2) Most videos are rectangular rather than square. When training the network, the input video frames to network are always square. If the same shape is still employed in extracting features, the spatial content will be lost in the rectangular video frames, which is crucial for temporal action detection.

2. Our Approach

The overall architecture of our approach is visualized in Figure 1. The video features are first extracted from video frames by convolution-based and transformer-based backbones. Then the video features are employed to generate proposals and classify action categories. Finally, the action localization results are generated by fusing proposals with classification scores.

2.1. Training Backbones

The existing mainstream pre-training methods can be divided into two types: supervised [21, 9, 1, 8] and unsupervised [13, 11]. Supervised methods can achieve stronger performance, but need to provide labels for each video. Unsupervised methods can make full use of unlabeled data. We utilize the supervised strategy for pre-training to achieve better performance.

All backbones we employed are first pre-trained on the large-scale Kinetics-700 [7] dataset or Kinetics-600 [6] dataset to improve the generalization ability, and then fine-tuned on the HACS [27] dataset. We explored four backbones with different architectures. As shown in Figure 1, the SlowFast [9] and CSN [21] are based on convolution, and ViViT [1] and Timesformer [3] are based on trans-

* Corresponding authors.

This work is supported by Alibaba Group through Alibaba Research Intern Program.

This work is done when Z. Qing and X. Wang (Huazhong University of Science and Technology), Z. Huang (National University of Singapore) and Y. Feng (Tsinghua University) are interns at Alibaba Group.

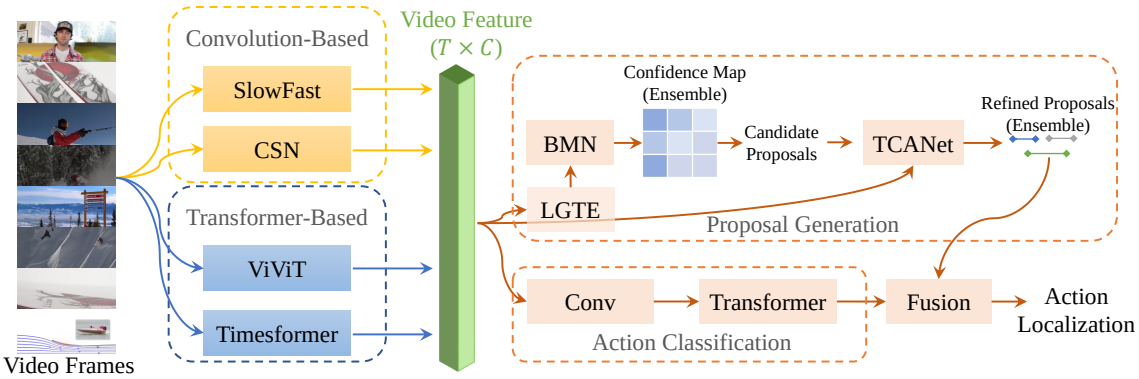

Figure 1. **The overall framework of our approach.** The input video frames are first extracted features by backbones. Then the BMN [16] and TCANet [19] are employed to generate accuracy proposals. The video-level features are utilized to perform action classification task. The fusion of proposals and action category generates action detection results.

Backbone	SlowFast [9]	CSN [21]	ViViT [1]	Timesformer [3]
Layers	101	152	**12**	12
Frames	32×2	32×2	32×2	8×32
Resolution	224×224			
BS	256			
Optimizer	SGD	AdamW	AdamW	SGD
LR	0.02	1e-4	1e-5	0.004
WD	1e-7	1e-4	0.1	1e-4
LR Policy	Cosine			
WU Epochs	5	8	2.5	1
T Epochs	50	30	30	15
Dropout	0.5	0.0	0.5	0.5

Table 1. **Training details for all backbones.** "LR" refers to Learning Rate, "WD" is weight decay, "WU" means Warm Up, "T Epochs" is Training Epochs and "BS" refers to Batch Size.

former. In fine-tuning stage, the features extracted by backbone are used to perform the classification task with $(C+1)$ categories. The fine-tuning details are introduced in the Table 1. Note that there are differences in training settings between different models. Our training strategies are for reference only. Among them, we use open source pre-trained models to initialize SlowFast, CSN, and Timesformer, and only ViViT is reprodced by us. The details of pre-training process can be referred to our EPIC-KITCHENS-100 Action Recognition report [12].

2.2. Extracting Features

Following mainstream action proposal generation methods [16, 17, 15, 25, 10, 5, 2, 26, 19, 20, 23, 24], we pre-extract features for each video. Specifically, for a video which contains l frames, the whole video can be divided into N clips uniformly. We set the stride between consecutive clips to $\delta = 8$, which can be converted to 0.267s in a 30-fps videos. In the temporal dimension, the sampling strategy for input clips is consistent with the fine-tuning process. In the spatial dimension, the transformer-based methods are also consistent with fine-tuning, while the convolution-based methods utilize the resolution of 256×320 as the input to extract features, which can save more spatial information for temporal localization.

2.3. Generating Proposals

The popular Boundary Matching Network(BMN) [16] based on dense prediction are easier to generate proposals with high recall rate. Combined with Temporal Context Aggregation Network(TCANet) [19] to further refine proposals, it can achieve impressive performance on HACS dataset.

Training BMN. The video-level features($C \times N$) are resized to 200, (e.g. $C \times 200$). Local-Global Temporal Encoders(LGTEs) [19] are also inserted into the base module in the BMN. The AdamW [18] is employed as optimizer. The batch size, learning rate, weight decay and training epochs are set to 128, 0.001, 1e-5 and 10, respectively. BMN designs Temporal Evaluation Module(TEM) and Proposal Evaluation Module(PEM) to evaluate the boundary scores and the IoU of proposals. In our implementation, we only employ the scores output by PEM, since the output of TEM lack global perceptions, which cannot improve the precision.

Training TCANet. We do not resize the features to preserve fine-grained temporal information. Three Temporal Boundary Regressors(TBRs) [19] are employed to refine the proposals generated by BMN, and the first TBR is employed to augment proposals [22] for accurate proposal distribution. Our optimizer for TCANet is Adam [14], and the batch size, learning rate and weight decay is set to 64, 0.0016 and 1e-5, respectively. We train the models for 10 epochs with cosine learning rate schedule.

Suppressing redundant predictions. We utilize Soft-NMS [4] to remove redundant predictions. The low thresh-

Backbone	Top-1(Val)
CSN [21]	91.54%
SlowFast [9]	90.37%
ViViT [1]	**91.92%**
Timesformer [3]	91.81%

Table 2. **Comparison between different backbones for clip-level action classification.**

Feature	LGTE [19]	mAP(Val)
CSN [21]	-	38.88%
CSN [21]	x2	**40.88%**
SlowFast [9]	-	38.83%
SlowFast [9]	x2	39.77%
ViViT [1]	-	36.63%
ViViT [1]	x2	37.30%
Timesformer [3]	-	32.23%

Table 3. **Comparison between different features based on BMN.** The "x2" means that we insert 2 LGTEs into base module in BMN.

Feature	Resolution	LGTE [19]	mAP(Val)
SlowFast[9]	224x224	-	37.28%
	224x224	x2	38.39%
	256x320	-	38.83%
	256x320	x2	**39.91%**

Table 4. **Ablation studies for feature resolution.** The resolution refers to the input resolution of frames when extracting feature.

old, high threshold and alpha in Soft-NMS are set to 0.25, 0.9 and 0.4, respectively.

2.4. Generating Detection Results

Since the proposals output by BMN [16] and TCANet [19] are class-agnostic, they need to be further classified to generate detection results. Considering that almost all videos in the HACS dataset [27] have only one category, we directly fuse the video-level classification results with proposals:

$$S_{det} = S_{props} \times S_{action}. \quad (1)$$

Where the S_{det} is the final detection score for submission, the S_{props} is the score for each proposal output by BMN or TCANet, and the S_{action} is the video-level score for each category.

3. Experiments

The Table 3 explores convolution-based and transformer-based features in terms of mAP. We notice that the convolution-based methods are better than transformer-based methods for action proposals. However, as shown in the Table 2, the transformer-based achieve impressive performance on clip-level classification task. This enlightens us that the classification performance of the backbone is not always positive for proposals, especially for Timesformer [3].

Feature	Method	Top-1(Val)	mAP (val)	mAP (Test)
ViViT	BMN	94.33%	33.46%	33.04%
		95.25%	33.91%	33.38%
CSN	BMN	94.33%	38.88%	38.68%
		96.07%	39.57%	39.26%
		96.07%	41.62%	41.17%
CSN	BMN+TCA	96.07%	42.74%	42.34%
C+S+V	Ensemble	96.27%	**44.83%**	**44.29%**
2020 Winner [20]		94.33%	40.55%	40.53%

Table 5. **Performance comparison between Validation set and Test set on different settings.** The "C+S+V" in the table refers to CSN [21]+Slowfast [9]+ViViT [1], and the "BMN+TCA" is the candidate proposals output by BMN [16] are input to TCA [19] for further refine.

For LGTE [19] in the Table 4, the improvement for CSN [21] feature is greater than the SlowFast [9] and the ViViT [9]. For the ViViT feature, since the spatio-temporal attention has been employed, the role of LGTE is limited.

In Table 4, we explore the influence of frame resolution in extracting features. The 224x224 cropping area in training limits the spatial information of each frame, especially for non-square video frames. Therefore, adopting a larger area for cropping can improve the quality of the extracted features. This is convenient to implement for convolution-based networks. However, for the transformer-based networks with the fixed position embedding, the same resolution as the training process is still used to extract features.

In Table 5,we show the results of our previous submissions. It can be noted that TCANet [19] can still achieve 1.09% improvement on the validation set, even based on a better baseline. Finally, we fuse the confidence maps output by multiple BMNs [16] and the refined proposals output by multiple TCANets. Thanks to the complementarity between the models trained with different features, we reached 44.83% and 44.29% on the validation set and test set, respectively, which was 3.76% higher than the 2020 Winner [20] on the test set.

4. Conclusion

In this technical paper, we explore different features, resolution, BMN and TCANet for temporal action detection. We found that the features extracted by the network with high classification performance may not necessarily generate high-quality proposals. This may be guiding us to design a backbone that is more suitable for temporal action detection. The ablation studies for resolution prove that avoiding the loss of spatial information can effectively improve the performance of temporal detection. Finally, with these strategies, our single-model suppresses 1.81% than the multi-model fusion used by the 2020 winner.

5. Acknowledgment

This work is supported by the National Natural Science Foundation of China under grant 61871435 and the Fundamental Research Funds for the Central Universities no. 2019kfyXKJC024 and by Alibaba Group through Alibaba Research Intern Program.

References

[1] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer. *arXiv preprint arXiv:2103.15691*, 2021. 1, 2, 3

[2] Yueran Bai, Yingying Wang, Yunhai Tong, Yang Yang, Qiyue Liu, and Junhui Liu. Boundary content graph neural network for temporal action proposal generation. *arXiv preprint arXiv:2008.01432*, 2020. 2

[3] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding? *arXiv preprint arXiv:2102.05095*, 2021. 1, 2, 3

[4] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S Davis. Soft-nms–improving object detection with one line of code. In *Proceedings of the IEEE international conference on computer vision*, pages 5561–5569, 2017. 2

[5] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. Sst: Single-stream temporal action proposals. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2911–2920, 2017. 2

[6] Joao Carreira, Eric Noland, Andras Banki-Horvath, Chloe Hillier, and Andrew Zisserman. A short note about kinetics-600. *arXiv preprint arXiv:1808.01340*, 2018. 1

[7] Joao Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *arXiv preprint arXiv:1907.06987*, 2019. 1

[8] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017. 1

[9] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6202–6211, 2019. 1, 2, 3

[10] Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. In *Proceedings of the IEEE international conference on computer vision*, pages 3628–3636, 2017. 2

[11] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *arXiv preprint arXiv:2010.09709*, 2020. 1

[12] Ziyuan Huang, Zhiwu Qing, Xiang Wang, Yutong Feng, Shiwei Zhang, Jianwen Jiang, Zhurong Xia, Mingqian Tang, Nong Sang, and Marcelo Ang. Towards training stronger video vision transformers for epic-kitchens-100 action recognition. *arXiv preprint arXiv:2106.05058*, 2021. 2

[13] Ziyuan Huang, Shiwei Zhang, Jianwen Jiang, Mingqian Tang, Rong Jin, and Marcelo Ang. Self-supervised motion learning from static images. *arXiv preprint arXiv:2104.00240*, 2021. 1

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[15] Chuming Lin, Jian Li, Yabiao Wang, Ying Tai, Donghao Luo, Zhipeng Cui, Chengjie Wang, Jilin Li, Feiyue Huang, and Rongrong Ji. Fast learning of temporal action proposal via dense boundary generator. In *AAAI*, pages 11499–11506, 2020. 2

[16] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3889–3898, 2019. 1, 2, 3

[17] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 2

[18] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2

[19] Zhiwu Qing, Haisheng Su, Weihao Gan, Dongliang Wang, Wei Wu, Xiang Wang, Yu Qiao, Junjie Yan, Changxin Gao, and Nong Sang. Temporal context aggregation network for temporal action proposal refinement. *arXiv preprint arXiv:2103.13141*, 2021. 1, 2, 3

[20] Zhiwu Qing, Xiang Wang, Yongpeng Sang, Changxin Gao, Shiwei Zhang, and Nong Sang. Temporal fusion network for temporal action localization: Submission to activitynet challenge 2020 (task e). *arXiv preprint arXiv:2006.07520*, 2020. 1, 2, 3

[21] Du Tran, Heng Wang, Lorenzo Torresani, and Matt Feiszli. Video classification with channel-separated convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5552–5561, 2019. 1, 2, 3

[22] Xiaopei Wan, Zhenhua Guo, Chao He, Yujiu Yang, and Fangbo Tao. Augmenting proposals by the detector itself. *arXiv preprint arXiv:2101.11789*, 2021. 2

[23] Xiang Wang, Baiteng Ma, Zhiwu Qing, Yongpeng Sang, Changxin Gao, Shiwei Zhang, and Nong Sang. Cbr-net: Cascade boundary refinement network for action detection: Submission to activitynet challenge 2020 (task 1). *arXiv preprint arXiv:2006.07526*, 2020. 2

[24] Xiang Wang, Shiwei Zhang, Zhiwu Qing, Yuanjie Shao, Changxin Gao, and Nong Sang. Self-supervised learning for semi-supervised temporal action proposal. *arXiv preprint arXiv:2104.03214*, 2021. 2

[25] Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10156–10165, 2020. 2

[26] Shiwei Zhang, Lin Song, Changxin Gao, and Nong Sang. Glnet: Global local network for weakly supervised action localization. *IEEE Transactions on Multimedia*, 22(10):2610–2622, 2019. 2

[27] Hang Zhao, Zhicheng Yan, Lorenzo Torresani, and Antonio Torralba. Hacs: Human action clips and segments dataset for recognition and temporal localization. *arXiv preprint arXiv:1712.09374*, 2019. 1, 3