Boundary-sensitive Pre-training: An Application of Generic Event Boundary

Mengmeng Xu[1,2*] Juan-Manuel Pérez-Rúa[1] Victor Escorcia[1] Brais Martínez[1]
Xiatian Zhu[1] Li Zhang[1] Bernard Ghanem[2] Tao Xiang[1,3]

[1] Samsung AI Centre Cambridge, UK

{j.perez-rua,v.castillo,brais.a,xiatian.zhu,li.zhang1,tao.xiang}@samsung.com

[2] King Abdullah University of Science and Technology (KAUST), Saudi Arabia

{mengmeng.xu,bernard.ghanem}@kaust.edu.sa

[3] University of Surrey, UK

Abstract

Many video analysis tasks require temporal localization thus detection of event changes. However, most existing models developed for these tasks are pre-trained on general video action classification tasks. This is because large scale annotation of temporal boundaries in untrimmed videos is expensive. Recently, a new benchmark, Kinetics-GEBD, is introduced for Generic Event Boundary Detection (GEBD) task, which has the potential to solve the mismatching problem. To reveal the importance of generic event boundary information, we investigate model pre-training for temporal localization by introducing a novel boundary-sensitive pretext (BSP) task. Instead of relying on costly manual annotations of generic event boundaries (e.g. GEBD), we propose to synthesize temporal boundaries in existing video action classification datasets. With the synthesized boundaries, BSP can be simply conducted via classifying the boundary types. This enables the learning of video representations that are much more transferable to downstream temporal localization tasks. We experimentally show that our model pre-training is beneficial to long video under-standing task, yielding compelling or new state-of-the-art performance.

1. Introduction

Recently, the focus on video analysis has shifted from trimmed video action classification to video temporal localization. This is because in many real-world applications, instead of short (*e.g.*, few seconds long) video clips, *long, untrimmed* videos are often presented (*e.g.*, from social media websites as YouTube, Instagram) with both non-interesting background and foreground contained *e.g.*, a particular action of interest. This requires a video model to conduct *temporal localization tasks*. Examples of these tasks include temporal action localization [44, 4], video grounding [1, 31], and step localization [50].

As in most other visual recognition tasks, existing models designed recently for video temporal localization are deep learning based. As such, model pre-training is critical. In particular, the two-staged model training strategy is commonly adopted [14, 39, 13, 3]: first pre-training a video encoder on a large action classification datasets (*e.g.*, Kinetics [5], Sports-1M [21] or HowTo100M [30]) followed by training a temporal localization head on the target small-scale temporal localization dataset. So there is a clear mismatch between the pre-training and downstream tasks. Ideally, model pre-training should be carried out on temporal boundary-sensitive tasks. However, this is not possible due to the lack of large scale video datasets with temporal boundary annotation. Recently, a new benchmark, Kinetics-GEBD, is introduced by [33] for Generic Event Boundary Detection (GEBD) task, which has the potential to solve the mismatching problem.

In this report, we investigate the under-studied yet critical problem of *model pre-training for temporal localization in videos*. Instead of relying on costly manual annotations of generic event boundaries (*e.g.* GEBD), we propose to synthesize large-scale untrimmed videos with generic event boundary annotations by transforming the existing trimmed video action classification datasets. Once the pre-training data problem is tackled, we focus on defining and evaluating a number of pretext tasks capable of exploiting the particularities of the synthesized data in a self-supervised manner.

The following **contributions** are made in this work: (I) We investigate the problem of model pre-training for temporal localization tasks in videos, which is largely understudied yet practically significant to video analysis. (II) We propose a scalable video synthesis method that can generate a large number of videos with temporal boundary information. This approach not only solves the key challenge of

*Work done during an internship at Samsung AI Centre.

lacking large pre-training data, but also facilitates the design of model pre-training. (III) We experimentally show that our model pre-training is beneficial to long video understanding task such as temporal action localization, yielding compelling or new state-of-the-art performance.

Please find the long version of this work in [43].

2. Related work

Temporal Action Localization. Temporal localization in videos encompasses tasks such as temporal action localization (TAL), video grounding and step localization. Specifically, TAL focuses on predicting the temporal boundaries and class of an action instance in untrimmed videos [18]. Most of the TAL solutions follow either a two-stage or a one-stage approach. Two-stage methods first generate candidate action segments (e.g., proposals) [4, 19, 9, 27, 10], and then use a classifier on each proposal to obtain a class score [35, 34, 46, 48, 25]. One-stage methods predict the temporal action boundaries or generate the proposals, and classify them in a shared network [17, 6, 42, 45, 44, 28, 24, 2].

In order to show the benefit of our proposed boundary-sensitive pre-training, we adopt the publicly-available G-TAD. In order to keep a fair comparison with prior work, we do not fine-tune the video encoding network on the downstream dataset. We feed our BSP features into this model with default configurations, and report the performance from the public evaluation scripts.

Video encoding networks. It is common among state of the art methods to use a pre-trained network as the video encoder. Such network is trained on a *classification* task using standard cross-entropy, typically on large-scale datasets like ImageNet [8, 11] or Kinetics [22, 47]. For example, it is common for TAL, *e.g.*, [25, 24, 44, 2], to use features extracted with a two-stream [36] TSN model [40, 49]. That is, the model comprises two TSN networks, one with ResNet50 [16] backbone trained on RGB, and another with a BN-Inception backbone [20] trained on Optical Flow. Other methods use 3D CNN-based models, such as Pseudo-3D [32], *e.g.*, [28], and two-stream I3D models [5], *e.g.*, [15, 46]. Alternatively, some methods exploit the temporal segment annotations on the downstream temporal localization datasets to define a classification task, and use it to pre-train the video encoder [24, 44, 7, 31]. This results in less of a domain gap, at the cost of large-scale training.

A few methods further add an end-to-end fine-tuning stage directly on the downstream tasks, e.g. R-C3D [42], PBR-net [26]. However, end-to-end training is achieved by compromising other important aspects, *e.g.*using batch size of 1, resulting in lower performance in practice.

Although an action classifier trained through cross-entropy can represent the overall content of a video segment, a feature extractor trained in this manner is not tuned to be sensitive to specific temporally-localized structures, such as the start or end of an action. Instead, we propose a boundary-sensitive self-supervised pre-training that results in features with the desired temporally-localized sensitivity.

3. Method

3.1. Problem context

We consider the model pre-training problem for temporal localization in videos. This is the first stage of the common *pre-training-then-fine-tuning* paradigm [12, 38, 29], and the downstream tasks in this context. The *vanilla pre-training* method simply conducts supervised learning on a large video dataset (*e.g.*, Kinetics) $D_{tr} = \{\mathbf{V}_i\}_{i=1}^{N}$ with action class labels as ground-truth supervision. This brings about action content awareness to the model. As a result, such a pretrained model is usually superior to those with random initialization and pretrained on image data (*e.g.*, ImageNet). However, this method is limited in capturing *temporal boundary information* as required by temporal localization tasks, because event boundary annotations are not available in existing large video datasets.

Given trimmed video data with action class labels, we introduce four temporal boundary concepts including *diff-class boundary, same-class boundary, diff-speed boundary, same-speed boundary*. They all require *zero* extra annotation in video synthesis and hence enable us to generate an arbitrary number of video samples with boundary labels. Next, we will describe the proposed *boundary-sensitive video* synthesis method.

3.2. Boundary-sensitive video synthesis

Similar to [33], temporal boundary refers to a transition of shots or scenes, or a change of action content. In this work, we consider two perspectives of the video source: class semantics and motion speed, both of which are available in Kinetics (i.e., class label and frame rate). Four different boundary classes are then formulated as detailed below.

(1) Diff-class boundary. This boundary is defined as the edge between two action instances from *different classes*. It is the most intuitive boundary, as typically presented in untrimmed videos with different actions taking place continuously. To synthesize a video with this boundary, we use two videos $\mathbf{V}_1$ and $\mathbf{V}_2$ sampled randomly from different action categories. The output video S_{dc} eliminates abrupt content change, making the following model pre-training meaningful without trivial solution.

(2) Same-class boundary. Complementary to diff-class boundary, this aims to simulate the scenarios where the same action happens repeatedly and continuously. This is

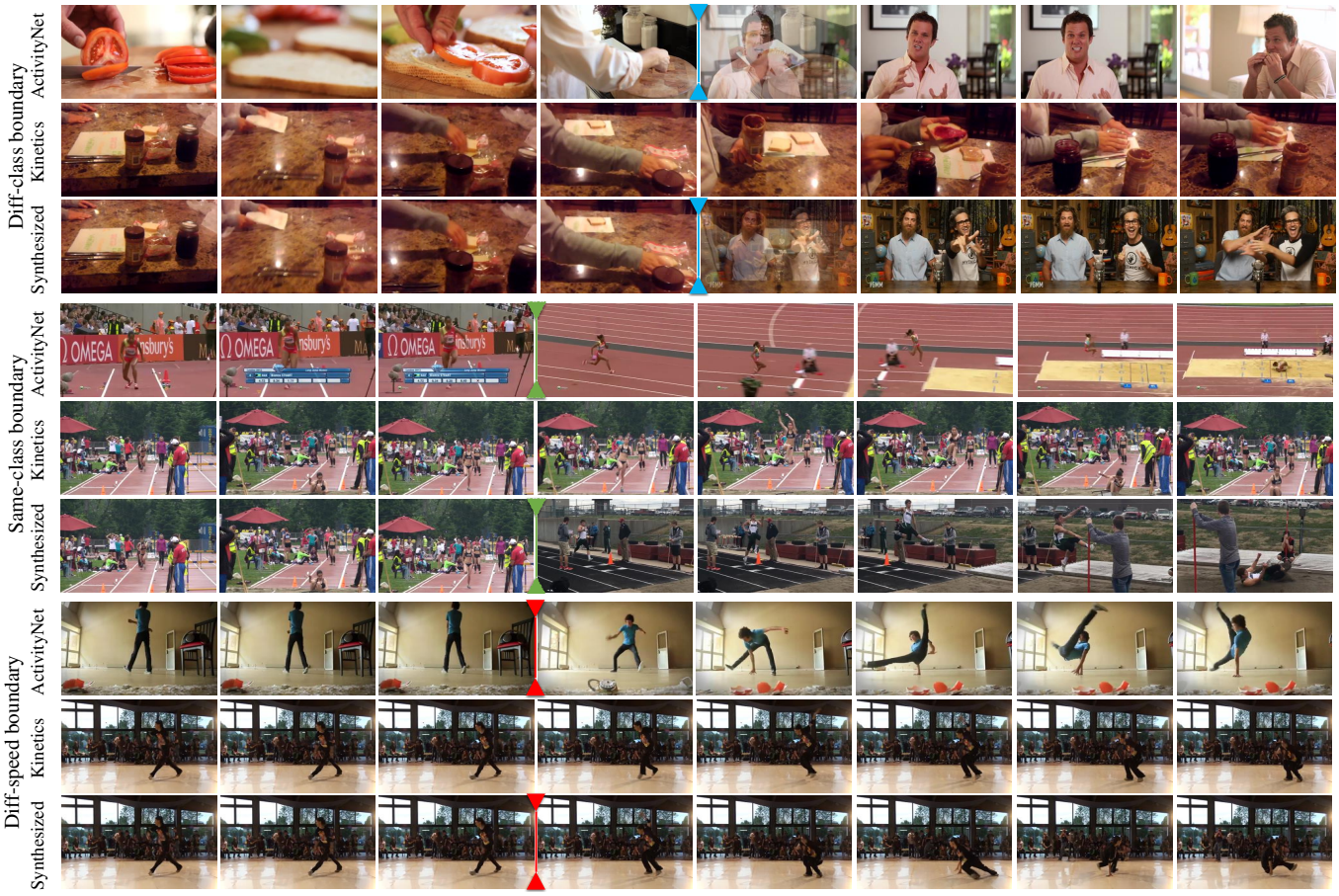

Figure 1. **Illustration of our *boundary-sensitive video* synthesis.** For each group of three rows we show from top to bottom: (a) a **real clip** from the ActivityNet dataset with real action class boundary; (b) a sampled clip from Kinetics-400 with no boundaries; and (c) a **synthesized video** by one of the proposed methods using samples from Kinetics-400 with synthetic boundaries. **(Top)** Diff-class boundary (▼); two clips from different categories are smoothly merged around frame #5. **(Middle)** Same-class boundary (▼); two clips from the same Kinetics category are stitched together from frame #4. **(Bottom)** Diff-speed boundary (▼); a clip from Kinetics is artificially sped up from frame #4.

frequently observed in untrimmed videos with multiple different shots of the same action class in a row. A new video S_{sc} is synthesized by concatenation of two videos from the same action class. Please be noted that transition is not applied in this case, since the semantic content is similar in the two input videos.

(3) Diff-speed boundary. This boundary class is motivated by an observation that the speed of content change varies from background (*e.g.*, without action) to foreground (*e.g.*, with action) and from one action instance to the other. Hence speed change entraps potentially useful temporal boundary information. Formally, we start with sampling a random video from the source data. Then, a new video S_ds with two different speed rates is generated.

(4) Same-speed boundary. This is introduced to serve as a *non-boundary* class for conceptual completion. For this class, the same videos from the source set are used with the coherent original speed rate throughout all the frames in each video. For notation consistency, we denote the videos of this type as S_{ss}.

Collectively, we denote all four types of boundary-sensitive videos as $S = \{S_{dc}, S_{sc}, S_{ds}, S_{ss}\}$.

3.3. Boundary-sensitive pre-training

Given the boundary-sensitive video data S as generated in Sec. 3.2, we now describe how to use them for video model pre-training with the hope that the pretrained model can benefit the temporal localization downstream tasks.

An intuitive pre-training method is supervised classification by treating each type of synthetic video as a unique class. That is, a four-way classification task. Formally, we first label a boundary class $y \in \{0, 1, 2, 3\}$ to each video $x \in S$ according to their boundary type. With a target video model θ, we predict the boundary classification vec-

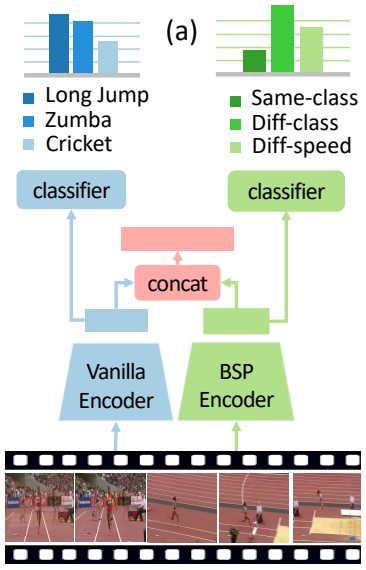

Figure 2. **Integrating BSP with vanilla action-classification pre-training.** Two independently-trained feature streams produce vanilla action and BSP features that are concatenated as output.

tor $\mathbf{p} = \{p_0, p_1, p_2, p_3\}$ for a given training video x. To pre-train the model, we use the cross-entropy loss function. The main merit of this method is to easily accommodate different types of boundary supervision in a principled manner. This allows the effective use of our synthetic video data.

We integrate our method with the classification-based pre-training features for enhancing boundary awareness as required for temporal localization downstream tasks. We designed Two-stream integration that consists of two steams in parallel, one for action classification-based pre-training and one for our boundary-sensitive pre-training (Fig. 2(a)). For simplicity, we use the same backbone for both. To integrate their information, feature concatenation is adopted at the penultimate layer.

4. Experiments

4.1. Experiment setup

As discussed in Sec. 2, temporal action localization task targets to recognize the specific point in time where the semantic content of the video changes. Solutions to the task use the two-step training paradigm: first pre-training the video encoder with the BSP method, followed by training the task-specific (TAL) model with our BSP model *frozen* as video feature extractor. This allows to explicitly examine the quality and efficacy of model pre-training.

Datasets. We used *ActivityNet-1.3* [18] dataset to evaluate the performance of temporal action localization task. ActivityNet-1.3 is a popular benchmark for temporal action localization. It contains 19,994 annotated untrimmed videos with 200 different action classes. The split ratio of train:val:test is 2:1:1. Each video has an average of 1.65 action instances. Following the common practice, we train and test the models on the training and validation set.

Evaluation metrics. We adopted the standard performance metrics specific for *temporal action localization* task: mean Average Precision (mAP) at varying temporal Intersection over Union (tIoU) thresholds was used. Following the official evaluation setting, we reported mAP scores at three tIoU thresholds of $\{0.5, 0.75, 0.95\}$ and the average mAP over ten thresholds of $[0.05 : 0.95]$ with step at 0.05 for ActivityNet-1.3.

Implementation details. Throughout the experiments, we only use RGB input to compute the video representation since Optical Flow is computationally expensive and adds complexity to the feature extraction model. However, current standard features rely on TSN[40], which is insensitive to time. Thus, removing the Optical Flow stream can have a very negative impact. In order to alleviate this issue, we adopt the Temporal Shift Module (TSM) architecture [23].

Given a variable-length video, we firstly sample every 8 consecutive frames as a snippet. Then we feed the snippet into our pre-trained models, and save the features before the fully connected layer. Thus, we obtain a set of snippet-level feature for the untrimmed video.

For the state-of-the-art *temporal localization models*, we selected G-TAD [44] for temporal action localization on Activity-1.3.

4.2. Comparison to the state-of-the-art

In this section, we compare the performance of the proposed BSP features under different tasks and different temporal localization networks.

On the TAL task, our BSP feature can significantly boost the performance of G-TAD, see Tab. 1, without relying on an Optical Flow stream. Adding BSP features increases performance by 0.93% at 0.5 IoU, and by 0.5% average mAP. Compared to the features originally used by G-TAD, our method does not required time-consuming computations to extract optical flow, neither do we fine-tune our video encoder on ActivityNet.

4.3. Visualization Result

We visualize some qualitative results in Fig. 3. Top shows that only BSP predicts the temporal boundary of *PlayingSquash* precisely by successfully detecting the scene change between 2nd and 3rd frames. In the bottom, both methods fail to find the start point for *ScubaDiving*, but BSP's result (*stepping into water*) makes more sense.

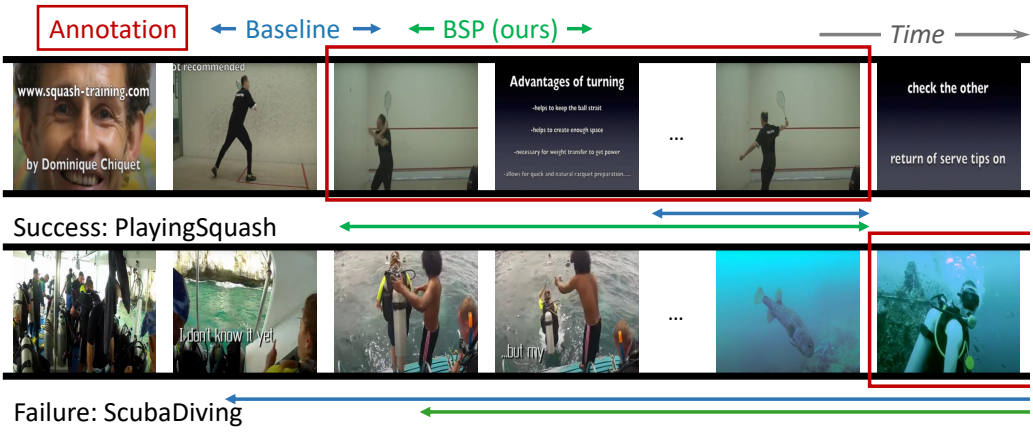

Figure 3. **Success and failure cases of BSP (ours) and baseline.** Top shows that only BSP predicts the temporal boundary of *PlayingSquash* precisely by successfully detecting the scene change between 2nd and 3rd frames. In the bottom, both methods fail to find the start point for *ScubaDiving*, but BSP's result (*stepping into water*) makes more sense.

Table 1. **TAL on ActivityNet-1.3 validation set**. "*" indicates RGB-only Kinetics pre-trained TSM feature without fine-tuning.

Method	0.5	0.75	0.95	Average
Singh *et al.* [37]	34.47	-	-	-
Wang *et al.* [41]	43.65	-	-	-
Chao *et al.* [6]	38.23	18.30	1.30	20.22
SCC [17]	40.00	17.90	4.70	21.70
CDC [34]	45.30	26.00	0.20	23.80
R-C3D [42]	26.80	-	-	-
BSN [25]	46.45	29.96	8.02	30.03
P-GCN [46]	48.26	33.16	3.27	31.11
BMN [24]	50.07	34.78	8.29	33.85
BC-GNN [2]	50.56	34.75	**9.37**	34.26
G-TAD [44]	50.36	34.60	9.02	34.09
G-TAD*	50.01	35.07	8.02	34.26
G-TAD*+**BSP (Ours)**	**50.94**	**35.61**	7.98	**34.75**

5. Conclusion

In this work we have investigated the under-studied problem of model pre-training for temporal localization tasks in videos by introducing a novel Boundary-Sensitive Pretext (BSP) task. Beyond vanilla pre-training on trimmed video data, we additionally synthesized a large number of videos with different types of temporal boundary information, and explored a number of pretext task designs using these boundary-sensitive videos. We evaluated the BSP model on temporal action localization task with different input modalities and motion complexity. The results demonstrate that our BSP can strongly enhance the vanilla model with boundary-sensitive feature representations, yielding competitive or new state-of-the-art performance. Please find the long version of this work in [43].

References

[1] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing Moments in Video With Natural Language. In *ICCV*, 2017. 1

[2] Yueran Bai, Yingying Wang, Yunhai Tong, Yang Yang, Qiyue Liu, and Junhui Liu. Boundary Content Graph Neural Network for Temporal Action Proposal Generation. In *ECCV*, 2020. 2, 5

[3] Sagie Benaim, Ariel Ephrat, Oran Lang, Inbar Mosseri, William T Freeman, Michael Rubinstein, Michal Irani, and Tali Dekel. SpeedNet: Learning the speediness in videos. In *CVPR*, 2020. 1

[4] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. SST: single-stream temporal action proposals. In *CVPR*, 2017. 1, 2

[5] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the Kinetics dataset. In *CVPR*, 2017. 1, 2

[6] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A. Ross, Jia Deng, and Rahul Sukthankar. Rethinking the faster R-CNN architecture for temporal action localization. In *CVPR*, 2018. 2, 5

[7] Cristian Rodriguez-Opazo, Edison Marrese-Taylor, Fatemeh Sadat Saleh, Hongdong Li and Stephen Gould. Proposal-free Temporal Moment Localization of a Natural-Language Query in Video using Guided Attention. In *WACV*, 2020. 2

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 2

[9] Victor Escorcia, Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. DAPs: Deep action proposals for action understanding. In *ECCV*, 2016. 2

[10] Jiyang Gao, Kan Chen, and Ramakant Nevatia. CTAP: Complementary temporal action proposal generation. *ECCV*, 2018. 2

[11] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. TALL: Temporal activity localization via language query. In *ICCV*, 2017. 2

[12] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2

[13] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, 2020. 1

[14] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. In *NeurIPS*, 2020. 1

[15] Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking ImageNet pre-training. In *CVPR*, 2019. 2

[16] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[17] Fabian Caba Heilbron, Wayner Barrios, Victor Escorcia, and Bernard Ghanem. SCC: Semantic context cascade for efficient action detection. In *CVPR*, 2017. 2, 5

[18] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 2, 4

[19] Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. Fast temporal activity proposals for efficient detection of human actions in untrimmed videos. In *CVPR*, 2016. 2

[20] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 2

[21] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014. 1

[22] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The Kinetics human action video dataset. *arXiv preprint*, 2017. 2

[23] Ji Lin, Chuang Gan, and Song Han. Tsm: Temporal shift module for efficient video understanding. In *ICCV*, 2019. 4

[24] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. BMN: boundary-matching network for temporal action proposal generation. In *ICCV*, 2019. 2, 5

[25] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. BSN: Boundary sensitive network for temporal action proposal generation. In *ECCV*, 2018. 2, 5

[26] Qinying Liu and Zilei Wang. Progressive boundary refinement network for temporal action detection. In *AAAI*, 2020. 2

[27] Yuan Liu, Lin Ma, Yifeng Zhang, Wei Liu, and Shih-Fu Chang. Multi-granularity generator for temporal action proposal. In *CVPR*, 2019. 2

[28] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Gaussian temporal awareness networks for action localization. In *CVPR*, 2019. 2

[29] Dhruv Mahajan, Ross Girshick, Vignesh Ramanathan, Kaiming He, Manohar Paluri, Yixuan Li, Ashwin Bharambe, and Laurens van der Maaten. Exploring the limits of weakly supervised pretraining. In *ECCV*, 2018. 2

[30] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. HowTo100M: Learning a Text-Video Embedding by Watching Hundred Million Narrated Video Clips. In *ICCV*, 2019. 1

[31] Jonghwan Mun, Minsu Cho, and Bohyung Han. Local-Global Video-Text Interactions for Temporal Grounding. In *CVPR*, 2020. 1, 2

[32] Zhaofan Qiu, Ting Yao, and Tao Mei. Learning spatio-temporal representation with pseudo-3d residual networks. In *ICCV*, 2017. 2

[33] Mike Zheng Shou, Deepti Ghadiyaram, Weiyao Wang, and Matt Feiszli. Generic event boundary detection: A benchmark for event segmentation. *CoRR*, abs/2101.10511, 2021. 1, 2

[34] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. CDC: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In *CVPR*, 2017. 2, 5

[35] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*, 2016. 2

[36] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 2014. 2

[37] Gurkirt Singh and Fabio Cuzzolin. Untrimmed video classification for activity detection: submission to activitynet challenge. *arXiv preprint*, 2016. 5

[38] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *ICCV*, 2017. 2

[39] Jiangliu Wang, Jianbo Jiao, and Yun-Hui Liu. Self-supervised video representation learning by pace prediction. In *ECCV*, 2020. 1

[40] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 2, 4

[41] Ruxin Wang and Dacheng Tao. UTS at activitynet 2016. *ActivityNet Large Scale Activity Recognition Challenge*, 2016. 5

[42] Huijuan Xu, Abir Das, and Kate Saenko. R-C3D: Region convolutional 3d network for temporal activity detection. In *ICCV*, 2017. 2, 5

[43] Mengmeng Xu, Juan-Manuel Perez-Rua, Victor Escorcia, Brais Martínez, Xiatian Zhu, Li Zhang, Bernard Ghanem, and Tao Xiang. Boundary-sensitive pre-training for temporal localization in videos. *CoRR*, abs/2011.10830, 2020. 2, 5

[44] Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-TAD: Sub-graph localization for temporal action detection. In *CVPR*, 2020. 1, 2, 4, 5

[45] Ze-Huan Yuan, Jonathan C. Stroud, Tong Lu, and Jia Deng. Temporal action localization by structured maximal sums. In *CVPR*, 2017. 2

[46] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph convolutional networks for temporal action localization. In *ICCV*, 2019. 2, 5

[47] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. Dense regression network for video grounding. In *CVPR*, 2020. 2

[48] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. Temporal action detection with structured segment networks. In *ICCV*, 2017. 2

[49] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *ECCV*, 2018. 2

[50] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *CVPR*, 2019. 1