supplementary of SCOAD

Na Ye[1[0000−0001−5985−2281]], Xing Zhang[1[0000−0002−9112−4070]], Dawei Yan[1[0000−0001−5202−0255]], Wei Dong[1[0000−0003−0263−3584]], and Qingsen Yan[2[0000−0003−1010−3540⋆]]

[1] Xi'an University of Architecture and Technology
[2] Northwestern Polytechnical University

1 More details of experiment

1.1 Click-level labels obtainment

Different click-level labels will unavoidably affect the performance of our model. Benefiting previous work, we used manual click-level labels from BackTAL and Sf-Net on THUMOS14. Although annotation of click-level labels is inexpensive, performing large-scale annotations is still tricky. Therefore, we generate random click labels on the ground truth on ActivityNet1.2. We consider that each section of the action area that exceeds one-third of the total length of the video should perform once with action-click annotation. The background-click annotation should be performed at least once in each video.

Admittedly, user interactions can never be random. This is a question worth studying. When the action occurs, the click event should occur as close as possible to the time when the action is most distinguishable. Therefore, considering the visual effect of the human eyes, we divide the action instance into three regions (head, middle, and tail) and generate pseudo-click labels in each region, respectively. Intuitively, we believe that the center position should be the most recognizable for most actions. Therefore, we performed additional experiments, verified our suspicions in Table 1, and encouraged users to click as close to the center of the action as possible. Meanwhile, the effect of other characteristics of click labels (density, interval time, etc.) on model performance will be the direction of our future research.

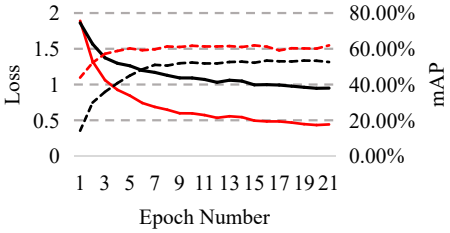
Table 1. The click event occurs in a different area of the video on the THUMOS14.

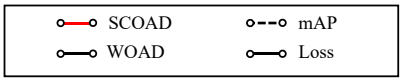
Methods	Action click area	mAP	pAP@ 1.0
SCOAD	Head	55.9	21.7
	Middle	64.1	24.0
	Tail	58.0	22.1
	Manual	61.9	24.4

⋆ Corresponding author:qingsenyan@gmail.com

Table 2. Our approach was compared with the performance on THUMOS14 under various IoU thresholds.

Methos	Threshold	Supervison	pAP@Time Threshold(Seconds)										mAP
			1.0	2.0	3.0	4.0	5.0	6.0	7.0	8.0	9.0	10.0	
SCOAD	0.1	Video-Level+Click-Level	24.9	37.0	43.3	47.7	50.1	51.2	52.3	52.8	53.4	53.6	61.1
	0.3	Video-Level+Click-Level	24.4	**39.2**	**44.8**	**49.0**	**50.7**	**51.6**	**52.4**	**53.0**	**53.6**	**54.0**	**61.9**
	0.5	Video-Level+Click-Level	25.7	37.1	43.2	46.0	48.2	49.5	50.3	50.6	51.2	51.4	61.6
	0.7	Video-Level+Click-Level	24.6	36.1	41.7	45.5	47.1	48.4	49.7	50.0	50.7	51.3	61.1





Methods	Supervision	Param	FLOPs	Infer time
WOAD	Video	110M	6.72G	0.40ms
SCOAD	Video+Click	80M	5.11G	0.32ms

Fig. 1. Compare with the state-of-the-art method for WOAD.

1.2 IoU threshold

This paper has many settings for thresholds, but we all align with WOAD and we restrict ourself here to discussing the effects of our proposed IoU filter thresholds. During training, AIM obtains pseudo-action instances through a two-stage threshold strategy. First, categories of video-level small confidence scores are filtered using thresholds. Naturally, short instances that cannot constitute an action are filtered using a threshold. Eventually, AIM generates action instances under these threshold filters. For a fair comparison with WOAD, we generate action instances with thresholds consistent with WOAD in AIM. To explore the appropriate parameters for the operation of our IoU filters, we experimented with several groups of IoU thresholds in Tabel 2.

1.3 Training costs

we compared each epoch's *loss* and mAP curves of WOAD and SCOAD. SCOAD converges faster than the state-of-the-art method for WOAD in Fig 1. Meanwhile, the $FLOPs$ as a metric that measures the importance of training time, our method was 1.61G lower than WOAD. Therefore, SCOAD has lower training time.