Few-Shot Common Action Localization via Cross-Attentional Fusion of Context and Temporal Dynamics

Juntae Lee[1], Mihir Jain[*], and Sungrack Yun[1]
[1]Qualcomm AI Research[†]
{juntlee, sungrack}@qti.qualcomm.com

Abstract

The goal of this paper is to localize action instances in a long untrimmed query video using just meager trimmed support videos representing a common action whose class information is not given. In this task, it is crucial to mine reliable temporal cues representing a common action from handful support videos. In our work, we develop an attention mechanism using cross-correlation. Based on this cross-attention, we first transform the support videos into query video's context to emphasize query-relevant important frames, and suppress less relevant ones. Next, we summarize sub-sequences of support video frames to represent temporal dynamics in coarse temporal granularity, which is then propagated to the fine-grained support video features through the cross-attention. In each case, the cross-attentions are applied to each support video in the individual-to-all strategy to balance heterogeneity and compatibility of the support videos. In contrast, the candidate instances in the query video are lastly attended by the resulting support video features, at once. In addition, we also develop a relational classifier head based on the query and support video representations. We show the effectiveness of our work with the state-of-the-art (SOTA) performance in benchmark datasets (ActivityNet1.3 and THUMOS14), and analyze each component extensively.

1. Introduction

Temporal action localization [2, 4, 33, 23, 31, 18, 15, 16, 20] have been widely studied in fully or weakly-supervised manner. However, they require collecting massive videos labeled by action classes, and also only detect the action classes observed in training. Whereas, we aim to temporally localize action instances in a long untrimmed query video based on the few trimmed support videos describing a common action class. As the testing action class is unseen in training and no ground-truth class cue is given, the only cue is the commonality of the few support videos.

Figure 1: Attention represents the relationship between a moment of query and support videos. (a) Vanilla: attention of a query proposal is obtained simultaneously for the frames of all the support videos. (b) Ours: the attention is computed for the frames of one support video, at a time. While important frames of support video #2 cannot be attended by the related query proposal in (a), they are appropriately transformed to the context of the query in (b).

In this task, the alignment between query and support videos is important, which can be attained by representing the common action cues of interest from the support videos considering the query video's context. For better alignment, we divide this problem into two points: re-calibrating support video features under query video's context, and enhancing temporal dynamics and compatibility of the re-calibrated support video features.

Existing methods [39, 22, 7, 3] have mainly focused on the former point, handling the multiple support videos as a whole. However, as exemplified in Fig. 1, though the support videos represent a common action class, their context (*e.g* background, camera angle) can be different. Hence,

[*]Work completed during employment at Qualcomm Technologies, Inc.
[†]Qualcomm AI Research is an initiative of Qualcomm Technologies, Inc.

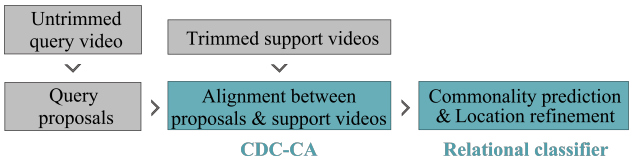

Figure 2: Overall process of few-shot common action localization in our work.

when all the support videos are transformed to query context simultaneously, the support videos cluttered by background are overly suppressed although they include useful information (Fig. 1(a)). Whereas, in Fig. 1(b), we can leverage the support videos by attending each individually.

Regarding the latter point, we pay attention to that the temporal dynamics can be enriched from multiple temporal granularities. Hence, we try to collaboratively fuse the support video features in different temporal granularities, considering the compatibility of different support videos as well. Based on the insight for each point, we propose a novel few-shot common action localization method. The overall process is briefly outlined in Fig. 2. First, the long untrimmed query video is split to query proposals, and they are aligned with the support videos through our three-stage cross-attention. Then, the proposals representing the common action class are detected with temporal location refinement.

For the query-support alignment, we develop a three-stage cross-attention (CA) mechanism. In each stage, cross-correlation between two different features (e.g. support and query, supports in different temporal granularity) is exploited with a learnable weight matrix. The 1st stage, query-to-support Context-CA transforms the support videos into the query video's context by cross-correlating them with the query proposal features. Each support video is individually attended to emphasize its most informative features. In 2nd stage of Dynamics-CA, low-temporal granularity features summarize the tuples of snippets of each support video, and they attend the fine-grained snippet-level features. Here, each support video is attended by the entire support videos in low-temporal granularity. It helps the complementary use of all the individually attended support video features as well as enhancing temporal dynamics. Finally, in the last stage (support-to-query Context-CA), the query video features are simultaneously attended with all the resulting support video features. We call this CA process (Context, Dynamics, Context-CAs) CDC-CA.

Also, for the commonality prediction and refinement, we design a relational classifier with an action classifier and an auxiliary relational module. To prevent overfitting and boost performance, the latter matches the support and query video features using pseudo-label cues as we avoid the use of class labels. In testing, this module is not used.

Major contributions: (i) We suggest a three-stage CA to enhance the representation of query video by support videos and vice versa. (ii) We attend each support video individually to increase its discriminative ability in the first two stages. (iii) We develop a relational classifier including an action classifier and an auxiliary relational module. The latter is only needed during training. (iv) Extensive experimental analyses are done on two benchmark datasets, where we achieve SOTA performance.

2. Related work

Temporal action localization The goal is to predict the temporal boundary and the label of action instances in untrimmed videos. In the fully-supervised case where temporal annotations are given during training, several works tried to obtain better temporal proposals [5, 4, 19], while others improved temporal searching [41, 42] or classifiers [29]. R-C3D [35] proposed an end-to-end trainable activity detector based on Faster-RCNN [27]. GNNs [45, 37] are recently used to capture the temporal relationship among proposals/snippets. In the weakly-supervised setting, as only video-level annotations are given, most methods have aimed to obtain discriminative snippet-level activations and conduct post-processing to localize action instances. A co-activity similarity loss to enforce the feature similarity for video pairs with a common class was introduced in [25], and videos are segmented into interpretable fragments in [10]. Some works focused on distinguishing action and near-action/background snippets exploiting variational auto-encoder [28], entropy maximization [20], and an additional class-agnostic model [21]. Also, to better localize difficult snippets, multi-modal (audio-visual) fusion [14] or a contrastive loss with easy foreground/background snippets [43] are devised. Although those fully- and weakly-supervised methods attained large progress in this field, the learned models can only localize activity categories observed in the training dataset.

Few-shot temporal action localization [38] pioneered few-shot action localization, where a few (or only one) positive and several negative labeled videos steer the localization via an end-to-end meta-learning strategy. [36] also temporally localized an action from a few positive labeled and several negative labeled videos. They adopted a region proposal network [27] to produce proposals with flexible boundaries. [44] performed few-shot action localization where video-level annotations are needed. They constructed a multi-scale feature pyramid to directly produce temporal features at variable scales. Unlike those works, few-shot common action localization is less tied up with the need for labels. It localizes the action instances in a long untrimmed query video according to the commonality between the query and support inputs. Assuming a query with only one common action instance, [3] computed the probabilities for starting, ending, inside, or outside of ac-

tion instances at every time-step, and decided the window with the highest joint probability as the action instance. Extending to the query videos with multiple action instances of the same action class, [39] generated action proposals, and then classified the proposals and regressed their temporal locations. [22] set a linear classifier itself as a prototype, which needs fine-tuning by support videos for every target action, then the prototype is cross-attended by query proposals using multiple self-attentions. [7] boosted [39] adding a single-head transformer where two different affinity matrices are used for the cross-attention weight between query and support videos.

Cross-attention To leverage the relationship between two heterogeneous representations, diverse cross-attention schemes have been devised. Inspired by self-attention (scaled dot-product of key, query, and value), [34] applied the scaled dot-product operation for the concatenation of image and text features for VQA. Also, [12] attended student using key and value of teacher for knowledge propagation. Several works exploited the cross-correlation between the heterogeneous representation as the attention weights for image and sentence matching in visual question answering (VQA) [17, 13], query and prototype matching in prototypical few-shot learning [6], and audio-visual fusion [26]. Here, we exploit the cross-attention to improve the matching between query and support video clips as well as to better utilize multiple support videos in the context of few-shot action localization.

3. Method

In this section, we first describe the three-stage cross-attention, CDC-CA, which consists of query-to-support Context, support-to-support (in different temporal granularity) temporal Dynamics, support-to-query Context CAs. Then, we explain the relational classifier.

Problem statement Given an untrimmed query video V_Q and L trimmed support videos $\{V_S^1, V_S^2, \ldots, V_S^L\}$, the goal is to train a network (which consists of backbone g, proposal-net h, CDC-CA and relational classifier f) to temporally localize action instances in the query video based on the commonality of the support videos whose action classes are same and unseen during training. Note that any ground-truth class cues are not given even during training.

In training, we resort to a meta-learning strategy. Here, the action classes in the training set (C_{train}) and those of the testing set (C_{test}) have no overlapping. Also, to simulate the few-shot configuration of support and query videos that will be encountered at the testing phase, we exploit episode-based training. Specifically, in a training iteration, we compose an episode as a set of a query and L support videos $\{(V_Q, V_S^1, V_S^2, \ldots, V_S^L)\}$ from a randomly selected class of C_{train}. In our work, we set L to 1 or 5. Formally, the objective function is represented by

$$\arg\min_{g,h,f} \mathbb{E}_{(V_Q,\{V_S^l\}_{l=1}^L)\sim C_{\text{train}}} [\mathcal{L}(\mathcal{Y}, f(h(g(V_Q)), g(\{V_S^l\}_{l=1}^L))], \quad (1)$$

where $\mathcal{Y}$ denotes the set of temporal positions for the ground-truth action segments of the interest in V_Q. $\mathcal{L}$ is the loss function.

3.1. Overall framework

Fig. 3 depicts the overall framework of our method. To obtain the initial representations of the input query and support videos, we follow the preprocessing of [39]. The backbone network g [32] generates video representations for each input. Then, for the query video, the proposal subnet h [35] yields potential temporal action instances $Q = \{q_i\}_{i=1}^{N_Q}$, called action proposals, with diverse temporal lengths (details are in Sec. S-4 of supplementary materials). N_Q is the number of the proposals, and q_i denotes ith action proposal representation. For every lth support video, we uniformly split each support video into N_S fine-grained temporal snippets by $S^{(l)} = \{s_i^{(l)}\}_{i=1}^{N_S}$. Then, CDC-CA aligns Q and $\{S^{(l)}\}_{l=1}^L$. Next, in the relational classifier, the action classifier detects the action proposals with the target common action, and yields their temporal offsets to refine the start and end times of the proposals. In training, learning an auxiliary relational module in parallel with the action classifier is beneficial. In testing, the auxiliary relational module is discarded, and the localization head suppresses redundant proposals which is explained in Sec. 4.1.

3.2. CDC-CA

The green box of Fig. 3 illustrates three stages of CDC-CA. We first introduce the cross-attention mechanism to individually attend the fine-grained (snippet-level) support video features by the query's context (QtoS Context-CA). Next, we explain the cross-attention to enhance the temporal dynamics of each support video from all the support videos in different temporal granularity (Dynamics-CA). Then, we present the way to attend the features of query proposals via the cross-attention with all the resulting support features (StoQ Context-CA). These enhanced features serve as crucial inputs to the following relational classifier.

QtoS Context-CA: We first enhance the support features in consideration of the context of the query video (i.e. query to support, dubbed QtoS). To this end, we develop a cross-attention mechanism. When we suppose a vanilla approach such as Fig. 1(a), the attention weight is obtained at once based on the relationship between the query proposals and all the support videos. In this case, a support video, which is relatively less relevant to the query video than the other support videos, gets tiny attention weights and cannot

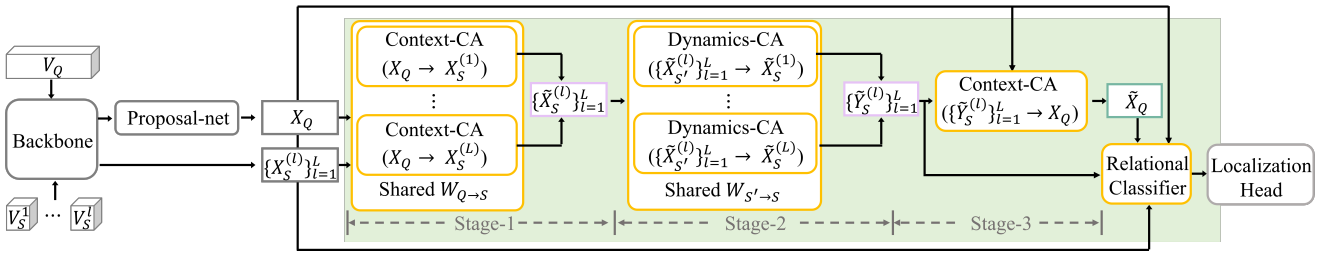

Figure 3: An overview of the proposed method. lth support video, $X_S^{(l)}$, is attended by query proposals X_Q, individually. Next, the temporal dynamics of all L the support videos are propagated to each support video. The resulting support videos $\{\tilde{Y}_S^{(l)}\}_{l=1}^L$ give attention to the query proposals. $\tilde{X}_Q$ is the resulting attended query proposals. Then, the relational classifier predicts the commonality scores of the proposals for the action of support videos, and the localization head finalizes action localization removing redundant proposals.

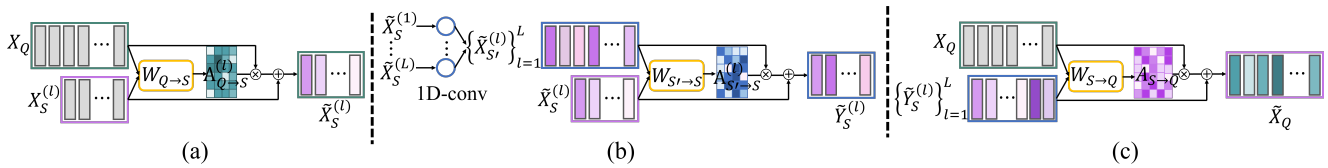

Figure 4: Cross-attention (best view in color). (a) QtoS Context-CA for lth support video by the query proposals. (b) Dynamics-CA for lth support video by L support videos in different temporal granularity. (c) StoQ Context-CA of all the query proposals by the attended support videos, at once.

represent its action information in the context of the query video. As depicted in Fig. 4(a), to utilize the query video's context faithfully, we individually attend the support videos.

In specific, for lth support video, we first encode the backbone features of the query proposals Q to $X_Q \in \mathbb{R}^{d\times N_Q}$ and the support snippets $S^{(l)}$ to $X_S^{(l)} \in \mathbb{R}^{d\times N_S}$, with a linear layer. The columns of X_Q and $X_S^{(l)}$ represent the encoded d-dimensional features of the query proposal and support snippets, respectively. After that, we compute the cross-correlation of X_Q and $X_S^{(l)}$ to measure their relevance. To reduce the gap of the heterogeneity between the query and support videos, we use a learnable weight matrix $W_{Q\to S} \in \mathbb{R}^{d\times d}$ and compute the cross-correlation as

$$\Lambda_{Q\to S}^{(l)} = X_Q^T W_{Q\to S} X_S^{(l)} \tag{2}$$

where $\Lambda_{Q\to S}^{(l)} \in \mathbb{R}^{N_Q\times N_S}$. Note that each column of X_Q and $X_S^{(l)}$ are l_2-normalized before the cross-correlation computation, and the learnable weight $W_{Q\to S}$ and the linear layer are shared across all l *i.e.* all the support videos.

In the cross-correlation matrix, a high correlation coefficient means that the corresponding proposal and snippet features are highly relevant. Accordingly, ith row of $\Lambda_{Q\to S}^{(l)}$ corresponds to the relevance of ith query proposal to the N_S support snippets. Then, from row-wise soft-max of $\Lambda_{Q\to S}^{(l)}$, we can obtain cross-attention weights $A_{Q\to S}^{(l)}$ where each row represents the relative relevance of a query proposal to the support snippets.

The attention-weighted proposal features are summed to the corresponding support snippet feature. This is to ensure that the meaningful action information of the support video is well-preserved while applying the cross-attention. Formally, the attended support snippet features $\tilde{X}_S^{(l)}$ are obtained by

$$\tilde{X}_S^{(l)} = X_Q A_{Q\to S}^{(l)} + X_S^{(l)}. \tag{3}$$

Note that, through the attention weights, the query proposal injects its context to the support snippets proportionally to their relevance *i.e.* more to the highly relevant support snippets. With this, the support snippet features are enhanced to better guide the information about the target action to the query proposals in the later stage of attending queries of Fig. 4(a).

Dynamics-CA: Temporal dynamics is important to understand the actions, which is well-represented in consecutive snippets (tuple) rather than a single snippet. Hence, as illustrated in Fig. 4(b), to extract temporal dynamics of the attended support video features, we apply 1-dimensional temporal convolution on each $\tilde{X}_S^{(l)}$ as

$$\tilde{X}_{S'}^{(l)} = \tilde{X}_S^{(l)} \odot \mathbf{w}_k \tag{4}$$

where $\odot$ denotes the convolution operation along temporal axis, $\mathbf{w}_k$ denotes the weight of 1D-conv layer. k is the kernel size of $\mathbf{w}_k$ ($k < L$). The resulting $\tilde{X}_{S'}^{(l)}$ is in different

temporal granularity with $\tilde{X}_S^{(l)}$, and its components are the temporally-summarized features for the tuples of k neighboring snippets, respectively.

Next, we propagate the temporal dynamics from tuple-level features to the fine-grained snippet features. Here, we also consider the compatible use of L support videos. In other words, as the support videos represent a common action, the support segment features should be closely located in a latent feature space. However, since the QtoS Context-CA are applied individually for each support video, the heterogeneity may be overly increased, which can degenerate the effect of attention on query videos in the following last stage. To compensate this, we attend the snippet-level features of lth support video $\tilde{X}_S^{(l)}$ from the tuple-level features of all L support videos, i.e, $\tilde{X}_{S'} = \{\tilde{X}_{S'}^{(l)}\}_{l=1}^{L}$.

Concretely, like eq. (2), we first compute cross-correlation between $\tilde{X}_{S'}$ and $\tilde{X}_S^{(l)}$ by

$$\Lambda_{S'\to S}^{(l)} = \tilde{X}_{S'}^T W_{S'\to S} \tilde{X}_S^{(l)} \tag{5}$$

Then, the temporal dynamics-attended support video feature $\tilde{Y}_S^{(l)}$ is generated by

$$\tilde{Y}_S^{(l)} = \tilde{X}_{S'} A_{S'\to S}^{(l)} + \tilde{X}_S^{\prime(l)}, \tag{6}$$

where $A_{S'\to S}^{(l)}$ is the cross-attention weight from $\Lambda_{S'\to S}^{(l)}$.

StoQ Context-CA: In this stage, we attend query videos with the support videos using the cross-attention. This support-to-query context attention is called StoQ Context-CA described in Fig. 4(c). To mine the richer information for target action from all the support videos, we attend query proposal features using the entire stabilized support segment features $\tilde{Y}_S = \{\tilde{Y}_S^{(1)}, \tilde{Y}_S^{(2)}, \ldots, \tilde{Y}_S^{(L)}\} \in \mathbb{R}^{d\times LN_S}$. Similar to attending support features, we exploit the cross-attention to obtain the attended query proposals $\tilde{X}_Q$ with a learnable weight matrix $W_{S\to Q} \in \mathbb{R}^{d\times d}$ by

$$\tilde{X}_Q = \tilde{Y}_S A_{S\to Q} + X_Q \tag{7}$$

where $A_{S\to Q}$ is attention weight computed by row-wise soft-max of cross-correlation, $\Lambda_{S\to Q} = \tilde{Y}_S^T W_{S\to Q} X_Q$. From this cross-attention, the query proposals, which have high relevance to crucial support segments, better delineate the target action.

3.3. Relational classifier

Up to this point, we described how CDC-CA generates better representations of query proposals and support videos. Here, we explain the way to obtain the final decision from the attended representations. Fig. 5 depicts our relational classifier including an action classifier and an auxiliary relational module. The auxiliary module facilitates learning how to understand the relationship between query proposals and a target action with pseudo action class cues. As the auxiliary relational module is discarded in the deployed network, no action class label is required in testing.

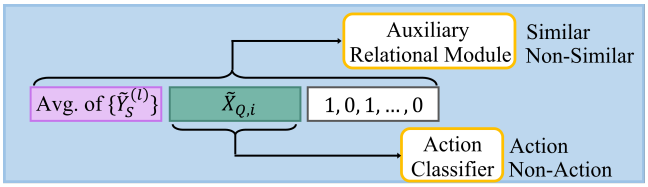

Figure 5: Relational classifier consists of auxiliary relational module and action classifier.

Auxiliary relational module: In episodic few-shot learning, an auxiliary classifier can be co-trained to categorize an input into one of the ground-truth classes rather than the target classes of an episode. This is beneficial for feature extractors to prevent overfitting, stabilize the training, and boost the performance [24, 11].

However, since the ground-truth action classes are not available in our task, we utilize pseudo action classes. As in Fig. 5, we develop the auxiliary relational module which takes as an input the concatenation of the attended query proposal (green), support prototype (purple), and a pseudo-class identifier (white). The support prototype is the average of all the attended support segments. The pseudo-class identifier is a multi-hot vector with 1 for the pseudo action classes present in the support videos and 0 otherwise. As action class annotations are unavailable, the pseudo action classes are simply obtained by k-means clustering of the pre-trained backbone features extracted from all the ground-truth positive action instances in the training set. Then, these pseudo action classes can provide the cue to capture the commonality of the support videos with the same action across episodes. Hence, joint training with the auxiliary module assists the action classifier to learn to more correctly find the common actions in the query video without getting distracted by the non-target actions. The auxiliary relational module predicts whether the query proposal and the prototype are similar or not. Note that this module is only needed for training and discarded in the testing phase as the training and testing classes are mutually exclusive.

Action classifier: The action classifier consists of two parallel linear layers. Taking only the query proposal features $\tilde{X}_Q$ as inputs, the action classifier computes two outputs. The first is soft-max activated and decides if a query proposal contains target action or not. The second regresses the temporal offsets from the corresponding ground-truth.

3.4. Loss functions

To optimize the entire network, as in [39], we use loss terms for both support-agnostic and -conditioned parts. In

the support-agnostic part, the binary cross-entropy loss $\mathcal{L}_{\text{cls}}^{\text{ag}}$ predicts if the proposal contains any activity or not, and the smooth l_1 regression loss $\mathcal{L}_{\text{reg}}^{\text{ag}}$ optimizes the relative displacement between proposals and ground-truths.

In the auxiliary relational module of the support-conditioned part, we simply use the binary cross-entropy loss $\mathcal{L}_{\text{aux}}$ to predict if the proposal and support videos are similar or not. In the action classifier of the support-conditioned part, similar to the support-agnostic part, the classification loss $\mathcal{L}_{\text{cls}}^{\text{co}}$ predicts if the proposal includes the same common action of support videos, and the regression loss $\mathcal{L}_{\text{reg}}^{\text{co}}$ optimizes the relative displacement between the proposals and the ground-truths. Also, we design a pairwise ranking loss to add constraints to the action classifier. Considering a pair of proposals q_i and q_j, where at least one of them is a positive query proposal, we let the proposal with the larger IoU (intersection over union) for the ground-truth action instance have a higher soft-max score for the action class. Formally, the pairwise ranking loss $\mathcal{L}_{\text{rank}}$ is represented by

$$\mathcal{L}_{\text{rank}} = \frac{1}{N_{\text{pair}}} \sum_{(i,j)} (\Delta\text{IoU}_{ij} - \Delta p_{ij}^{\text{action}})^2 \qquad (8)$$

where N_{pair} is the number of considered proposal pairs. ΔIoU_{ij} is the difference between IoUs of q_i and q_j with their corresponding ground-truths, and $\Delta p_{ij}^{\text{action}}$ is the difference of soft-max predictions for the proposals on the action class from the action classifier. Note that to relax the relationship between IoU and soft-max prediction, we give temperature to the soft-max predictions in $\mathcal{L}_{\text{rank}}$. Finally, total loss (more details in Sec. S-1 of supplementary materials) is given by

$$L = L_{\text{cls}}^{\text{ag}} + L_{\text{reg}}^{\text{ag}} + L_{\text{cls}}^{\text{co}} + L_{\text{reg}}^{\text{co}} + L_{\text{aux}} + \lambda L_{\text{rank}}. \qquad (9)$$

4. Experiments

4.1. Datasets and evaluation

To evaluate few-shot common action localization, we use the revised versions [3, 39] of ActivityNet1.3 [1] and THUMOS14 [9]. In the revised, there are two cases depending on queries: single-instance and multi-instance. We address 1- or 5-shot settings in both cases. For meta-learning strategy, the entire action classes of each dataset are split into 80% for training, 10% for validation, and 10% for testing. For a fair comparison, we follow the data configuration of [39], which will be described in the following paragraphs. Further details are in Sec. S-4 of supplementary material.

Common single-instance: For both datasets, videos with multiple actions are divided into independent videos where each video contains just one action instance and background. Then, videos longer than 768 frames are discarded. If a video is selected as a support video, its foreground action instance is only used as the support input. For ActivityNet1.3, there are 10,035 and 2,483 videos for training and validation+testing, each. The average video length is 89.0s. For THUMOS14, there are 3,580 and 775 training and validation+testing videos, respectively. The average length is 11.4s.

Common multi-instance: In real-world scenarios, the lengths of query videos are usually not constrained, and the query videos may contain multiple action instances. Hence, we exploit the original videos of ActivityNet1.3 and THUMOS14 without any processing for query videos. Support videos are still trimmed ones. For ActivityNet1.3, the numbers of videos are 6,747 and 1,545 for training and validation+testing, respectively. The average video length is 148.2s. For THUMOS14, there are 1,664 training videos and 323 validation+testing videos. The average video length is 230.6s.

Inference: In testing, the backbone-generated query proposals are refined by non-maximum suppression (NMS) with a threshold 0.7. Also, following [39], if the query video is longer than 768 frames, we generate the multi-scale segments [30]. We slide windows with sizes of 512 and 768 frames along the temporal axis with 75% overlap. The generated proposals of the windows go through the NMS to remove redundant proposals. Then, the selected proposals are fed into our CDC-CA and relational classifier. Finally, we perform NMS (threshold 0.3) for the regressed proposals based on the outputs of the relational classifier to remove redundant ones.

Evaluation: We measure the temporal action localization performance with mean Average Precision (mAP). A prediction is correct when it has the correct foreground/background classification and has IoU with its ground-truth larger than a threshold. The threshold is set to 0.5 unless specified.

4.2. Comparative assessment

We report the performance of compared methods from the literature. Due to the variance of k-means clustering of the pseudo label generation, we report the average of three runs for our method.

For AcitivityNet1.3, the left of Table 1 demonstrates the comparative results on both common single- and multi-instance cases when $L = 1$ or 5. [3, 39, 22, 7] were developed for the common temporal action localization (our task) in videos, and [8] was for the common object detection in images. Compared to them, our method shows notably higher performance for all the settings ([3] was designed to use one support video). Specifically, in the single-instance case, we outperform those methods by at least 1.4% and 5.1% in the 1- and 5-shot settings, respec-

Table 1: Comparison with state-of-the-arts in terms of mAP@0.5.

Method	*ActivityNet1.3*				*THUMOS14*			
	Single-instance		Multi-instance		Single-instance		Multi-instance	
	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot	1-shot	5-shot
Feng *et al.* [3]	43.5	n/a	31.4	n/a	34.1	n/a	4.3	n/a
Hu *et al.* [8]	41.0	45.4	29.6	38.9	-	42.2	-	6.8
Yang *et al.* [39]	53.1	56.5	42.1	43.9	48.7	51.9	7.5	8.6
Yang *et al.* [40]	57.5	60.6	47.8	48.7	-	-	-	-
Nag *et al.* [22]	55.1	63.0	44.1	48.2	49.2	54.3	7.3	10.4
Hsieh *et al.* [7]	60.7	61.2	-	-	-	-	-	-
Ours	**62.1** $\pm$ **1.0**	**66.3** $\pm$ **1.2**	**48.2** $\pm$ **1.2**	**53.5** $\pm$ **1.1**	**53.8** $\pm$ **1.2**	**59.2** $\pm$ **1.3**	**9.8** $\pm$ **0.4**	**15.7** $\pm$ **0.5**

Table 2: Ablation study on CDC-CA for the single-instance setting.

Method	QtoS Context-CA	Dynamics-CA	mAP (%)	
			1-shot	5-shot
C-0			52.0	54.3
C-1	✓		61.5	65.4
C-2		✓	61.3	65.1
C-3	✓	✓	62.1	66.3

Table 3: Ablation study on the relational classifier on the single-instance setting.

Method	$\mathcal{L}_{\text{rank}}$	Auxiliary rel. module	mAP (%)	
			1-shot	5-shot
R-0			61.8	65.8
R-1	✓		61.9	66.0
R-2		✓	61.9	66.1
R-3	✓	✓	62.1	66.3

Table 4: Analysis of the individual cross-attention strategy on QtoS Context-CA and Dynamics-CA, comparing with the aggregated cross-attention on the 5-shot setting.

		Single-inst.	Multi-inst.
QtoS Context-CA	Aggregated	65.7	52.8
	Individual (Ours)	66.3	53.5
Dynamics-CA	Aggregated	65.8	52.9
	Individual (Ours)	66.3	53.5

tively. In the multi-instance, the margins are at least 4.1% and 5.3% for 1- and 5-shot, each. Note that [40] was developed for object-level common temporal action localization. Although object-level localization is more challenging, it requires further information to learn their model such as the coordinates of foreground object bounding boxes. Hence, they have advantages in the frame-level localization task. Even so, our method yields better results than [40] in all the settings. Also, in the 5-shot setting, the score margins to the existing works are larger than those of 1-shot in both single- and multi-instance. This means that enhancing the temporal dynamics of a support video by the entire support video in Dynamics-CA and aligning individually each support video to the query video's context in QtoS Context-CA are helpful to mine the knowledge for a common action from multiple support videos. Unlike ActivityNet1.3, THUMOS14 includes shorter action instances which make correct localization more difficult. Nevertheless, we outperform the compared methods over all the settings.

4.3. Component analysis

We analyze our method on ActivitNet1.3. More studies are in supplementary materials.

Impact of cross-attention: First, to see the component-wise impact of our cross-attention, we ablate QtoS Context-CA and Dynamics-CA in Table 2, we can see that without any of them (C-0), the performance is severely degraded. Also, when comparing C-1 and C-2, both cross-attentions show meaningful improvements on C-0, but QtoS Context-CA is slightly more important. Using both shows the best performance in our final method C-3.

We can also consider changing the order of QtoS Context-CA and Dynamics-CA. In this case, we observed that the performance is slightly lowered (62.1%→61.7% in 1-shot, 66.3%→ 65.6% in 5-shot). Hence, the temporal dynamics extracted under the query video's context is more useful in our task.

Further, we evaluate the efficacy of our cross-attention, comparing it with other cross-attention mechanisms. To this end, we compare our CDC-CA itself (corresponding to R-0 in Table 3) to the progressive cross-attention [39], and the multi-head cross-attention with fine-tuned (50-100 iterations) prototype [22], light-weight transformer-based booster [7]. For a fair comparison, the pairwise ranking

Table 5: Cosine similarity between support videos and between support and query videos in embedding space before/after passing through Dynamics-CA.

	Support-Support	Support-Query		
		Pos. query	Neg. query	Δ
Cos. Sim. (before)	0.49	0.41	0.38	0.05
Cos. Sim. (after)	0.61	0.43	0.34	0.13

loss and auxiliary relational module are not used in R-0. CDC-CA yields mAP gain of at least +1.1% and +2.8% in 1-shot (60.7% [7] in Table 1 *vs* 61.8% R-0) and 5-shot (63.0% [22] in Table 1 *vs* 65.8% R-0), each. Hence, we conclude our cross-attention more effectively enhances the query and support videos in this task.

Individual *vs* aggregated: To study the effectiveness of the individual cross-attention on QtoS Context-CA and Dynamics-CA for multiple support videos, we compare our individual cross-attention approach with the aggregated cross-attention ('Aggregated') in Table 4. In 'Aggregated' of QtoS Context-CA, we concatenate all the support snippets of the entire support videos, and simultaneously attend them via query proposals. We see that 'Aggregated' degrades the localization performance. This result shows the benefit of individual cross-attention in QtoS Context-CA. Also, in Dynamics-CA, we identified a similar tendency. Accordingly, mining the commonality of temporal dynamics in multiple support videos is more beneficial when it is propagated to each support video, individually.

Compatibility in support videos: In Dynamics-CA, a snippet-level support video feature is attended by tuple-level features from all the support videos to promote compatibility among multiple support videos. To identify this, at two points (before or after passing through Dynamics-CA), we measure the cosine similarities 1) between two support videos, and 2) between a support video and a (positive or negative) query proposal. Table 5 reports the average of cosine similarities. The increased support-support similarity (0.49% $\rightarrow$ 0.61%) indicates that the support videos get closer to each other in the embedding space. Also, we can see that the discriminative ability is improved as seen by the increased difference (Δ) in the similarity of the support to positive and negative query proposals.

Further, to more clearly show the effect of individual-to-all cross-attention in Dynamics-CA, we also identified individual-to-individual Dynacmis-CA where the snippet-level features are attended by the corresponding temporally-convolutioned features for each support video. Comparison of ours with this variant is 66.3% *vs* 65.9% in 5-shot single instance, and 53.5% *vs* 53.2% in 5-shot multiple instances. Hence, we can see that our approach (individual-to-all) is more beneficial for the collaborative use of a few support videos.

Table 6: Effect of the multi-hot pseudo action indicator by varying the number of pseudo action classes or removing it.

		w/o Indicator	k			
			80	160	240	320
Single Inst.	1-shot	61.9	61.9	62.1	62.1	62.3
	5-shot	66.0	66.2	66.3	66.5	66.4
Multiple Inst.	1-shot	47.8	48.0	48.2	48.4	48.4
	5-shot	53.3	53.4	53.5	53.4	53.5

Relational classifier: Here, we verify the auxiliary relational module and the pairwise ranking loss by ablating each. To show their effect, we compare each ablated version to the baseline R-0 without any of them in Table 3. The pairwise ranking loss gives performance improvement by 0.1% and 0.2% for 1- and 5-shot, respectively (R-1). Hence, this loss lets the proposals with larger IoUs to ground-truths get higher action scores. From the result of R-2, we also see that learning the auxiliary module in parallel to the action classifier is useful to boost performance in both settings. And, combining both works the best (R-3).

Adequacy to no. of pseudo action label: In the auxiliary relational module, we use the multi-hot indicator as the pseudo-action cue in consideration that no true label is available. To see its effectiveness, in Table 6, we report the mAP scores by varying the number of pseudo action classes (k) and w/o the indicator as well. In the single instance case, compared to the R-1 of Table 3, the auxiliary relational module did not show an effect without the multi-hot indicator. Also, for all other settings, as k gets larger, the multi-hot indicator yields larger performance gains overall. Though the support videos represent a common action, there is diversity in background or action details. Hence, it is beneficial to distinguish the support videos with more pseudo labels. Considering the computational cost, we set k to 160 as the default value.

5. Conclusions

For few-shot common action localization, we proposed the three-stage cross-attention (CDC-CA) and the relational classifier. CDC-CA increases the effect of the cross-attention by individually attending each support video, enhancing the temporal dynamics of each and the compatibility of multiple support videos, and then attending the query video via all the enhanced support videos. In the relational classifier, we designed the pairwise ranking loss which makes more precise action localization of the action classifier. Learned in parallel with the action classifier, the auxiliary relational module with the pseudo-class labels prevents the network from overfitting to each training episode. This module is discarded in testing. Extensive experiments analyzed and validated each component. Finally, we achieved SOTA on ActivityNet1.3 and THUMOS14.

References

[1] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015. 6

[2] Victor Escorcia, Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. Daps: Deep action proposals for action understanding. In *ECCV*, 2016. 1

[3] Yang Feng, Lin Ma, Wei Liu, Tong Zhang, and Jiebo Luo. Video re-localization. In *ECCV*, 2018. 1, 2, 6, 7

[4] Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia. Turn tap: Temporal unit regression network for temporal action proposals. In *ICCV*, 2017. 1, 2

[5] 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

[6] Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Cross attention network for few-shot classification. In *NeurIPS*, 2019. 3

[7] He-Yen Hsieh, Ding-Jie Chen, Cheng-Wei Chang, and Tyng-Luh Liu. Aggregating bilateral attention for few-shot instance localization. In *WACV*, 2023. 1, 3, 6, 7, 8

[8] Tao Hu, Pascal Mettes, Jia-Hong Huang, and Cees GM Snoek. Silco: Show a few images, localize the common object. In *ICCV*, 2019. 6, 7

[9] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The thumos challenge on action recognition for videos "in the wild". *Computer Vision and Image Understanding*, 155:1–23, 2017. 6

[10] Mihir Jain, Amir Ghodrati, and Cees G. M. Snoek. ActionBytes: Learning from trimmed videos to localize actions. In *CVPR*. 2020. 2

[11] Dahyun Kang, Heeseung Kwon, Juhong Min, and Minsu Cho. Relational embedding for few-shot classification. 2021. 5

[12] Hanul Kim, Mihir Jain, Jun-Tae Lee, Sungrack Yun, and Fatih Porikli. Efficient action recognition via dynamic knowledge propagation. In *ICCV*, 2021. 3

[13] Jin-Hwa Kim, Kyoung-Woon On, Woosang Lim, Jeonghee Kim, Jung-Woo Ha, and Byoung-Tak Zhang. Hadamard product for low-rank bilinear pooling. In *ICLR*, 2017. 3

[14] Jun-Tae Lee, Mihir Jain, Hyoungwoo Park, and Sungrack Yun. Cross-attentional audio-visual fusion for weakly-supervised action localization. In *ICLR*. 2021. 2

[15] Jun-Tae Lee, Hyunsin Park, Sungrack Yun, and Simyung Chang. Multi-head modularization to leverage generalization capability in multi-modal networks. *AAAI*, 36(7), 2022. 1

[16] Jun-Tae Lee, Sungrack Yun, and Mihir Jain. Leaky gated cross-attention for weakly supervised multi-modal temporal action localization. In *WACV*, 2022. 1

[17] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *ECCV*, 2018. 3

[18] Pilhyeon Lee, Jinglu Wang, Yan Lu, and Hyeran Byun. Weakly-supervised temporal action localization by uncertainty modeling. In *AAAI*. 2021. 1

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

[20] Zhekun Luo, Devin Guillory, Baifeng Shi, Wei Ke, Fang Wan, Trevor Darrell, and Huijuan Xu. Weakly-supervised action localization with expectation-maximization multi-instance learning. In *ECCV*. 2020. 1, 2

[21] Junwei Ma, Satya Krishna Gorti, Maksims Volkovs, and Guangwei Yu. Weakly supervised action selection learning in video. In *CVPR*. 2021. 2

[22] Sauradip Nag, Xiatian Zhu, and Tao Xiang. Few-shot temporal action localization with query adaptive transformer. In *BMVC*, 2021. 1, 3, 6, 7, 8

[23] Phuc Nguyen, Ting Liu, Gautam Prasad, and Bohyung Han. Weakly supervised action localization by sparse temporal pooling network. In *CVPR*. 2018. 1

[24] Boris N Oreshkin, Pau Rodriguez, and Alexandre Lacoste. Tadam: Task dependent adaptive metric for improved few-shot learning. In *NeurIPS*, 2018. 5

[25] Sujoy Paul, Sourya Roy, and Amit K. Roy-Chowdhury. W-TALC: Weakly-supervised temporal activity localization and classification. In *ECCV*. 2018. 2

[26] R Gnana Praveen, Wheidima Carneiro de Melo, Nasib Ullah, Haseeb Aslam, Osama Zeeshan, Théo Denorme, Marco Pedersoli, Alessandro L Koerich, Simon Bacon, Patrick Cardinal, et al. A joint cross-attention model for audio-visual fusion in dimensional emotion recognition. In *CVPR*. 2022. 3

[27] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 2

[28] Baifeng Shi, Qi Dai, Yadong Mu, and Jingdong Wang. Weakly-supervised action localization by generative attention modeling. In *CVPR*. 2020. 2

[29] 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

[30] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*, 2016. 6

[31] Krishna Kumar Singh and Yong Jae Lee. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *ICCV*. 2017. 1

[32] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015. 3

[33] Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. UntrimmedNets for weakly supervised action recognition and detection. In *CVPR*. 2017. 1

[34] Xi Wei, Tianzhu Zhang, Yan Li, Yongdong Zhang, and Feng Wu. Multi-modality cross attention network for image and sentence matching. In *CVPR*, 2020. 3

[35] Huijuan Xu, Abir Das, and Kate Saenko. R-c3d: Region convolutional 3d network for temporal activity detection. In *ICCV*, 2017. 2, 3

[36] Huijuan Xu, Ximeng Sun, Eric Tzeng, Abir Das, Kate Saenko, and Trevor Darrell. Revisiting few-shot activity detection with class similarity control. In *arXiv preprint arXiv:2004.00137*, 2020. 2

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

[38] Hongtao Yang, Xuming He, and Fatih Porikli. One-shot action localization by learning sequence matching network. In *CVPR*, 2018. 2

[39] Pengwan Yang, Vincent Tao Hu, Pascal Mettes, and Cees GM Snoek. Localizing the common action among a few videos. In *ECCV*, 2020. 1, 3, 5, 6, 7

[40] Pengwan Yang, Pascal Mettes, and Cees GM Snoek. Few-shot transformation of common actions into time and space. In *CVPR*, 2021. 7

[41] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. End-to-end learning of action detection from frame glimpses in videos. In *CVPR*, 2016. 2

[42] Gang Yu and Junsong Yuan. Fast action proposals for human action detection and search. In *CVPR*, 2015. 2

[43] Can Zhang, Meng Cao, Dongming Yang, Jie Chen, and Yuexian Zou. Cola: Weakly-supervised temporal action localization with snippet contrastive learning. In *CVPR*, 2021. 2

[44] Da Zhang, Xiyang Dai, and Yuan-Fang Wang. Metal: Minimum effort temporal activity localization in untrimmed videos. In *CVPR*, 2020. 2

[45] Chen Zhao, Ali K Thabet, and Bernard Ghanem. Video self-stitching graph network for temporal action localization. In *ICCV*, 2021. 2