Learning Temporal Co-Attention Models for Unsupervised Video Action Localization

Guoqiang Gong, Xinghan Wang, Yadong Mu *
Wangxuan Institute of Computer Technology, Peking University
{gonggq,xinghan_wang,myd}@pku.edu.cn

Qi Tian
Noah's Ark Lab, Huawei
tian.qi1@huawei.com

Abstract

Temporal action localization (TAL) in untrimmed videos recently receives tremendous research enthusiasm. To our best knowledge, this is the first attempt in the literature to explore this task under an unsupervised setting, hereafter referred to as action co-localization (ACL), where only the total count of unique actions that appear in the video set is known. To solve ACL, we propose a two-step "clustering + localization" iterative procedure. The clustering step provides noisy pseudo-labels for the localization step, and the localization step provides temporal co-attention models that in turn improve the clustering performance. Using such two-step procedure, weakly-supervised TAL can be regarded as a direct extension of our ACL model. Technically, our contributions are two-folds: 1) temporal co-attention models, either class-specific or class-agnostic, learned from video-level labels or pseudo-labels in an iterative reinforced fashion; 2) new losses specially designed for ACL, including action-background separation loss and cluster-based triplet loss. Comprehensive evaluations are conducted on 20-action THUMOS14 and 100-action ActivityNet-1.2. On both benchmarks, the proposed model for ACL exhibits strong performances, even surprisingly comparable with state-of-the-art weakly-supervised methods. For example, previous best weakly-supervised model achieves 26.8% *under mAP@0.5 on THUMOS14, our new records are* 30.1% *(weakly-supervised) and* 25.0% *(unsupervised).*

1. Introduction

Temporal action localization (or action detection) [54, 39, 57, 48, 23, 30, 56] is a fundamental challenge in video understanding. The goal of temporal action localization is to precisely find the starting and ending time for each action instance from a long, untrimmed video. It has a variety of potential applications in real-world scenarios, including video summarization, video highlight detection, surgical skill assessment and others. To learn an effective action localization model, it is crucial to collect a sufficient number of annotated videos. In comparison with the benchmarks mainly designed for image recognition (*e.g.*, ImageNet[35]) or video classification (*e.g.*, Kinetics[3]), the largest video benchmarks for action localization, known as ActivityNet v1.3[10], has only about 20,000 annotated videos. This partly attributes to the fact that labeling an action instance is more tedious and prone to errors, since accurately delimiting the temporal boundary of an action instance is both time-consuming and subjective to different annotators. The scarcity of instance-level annotation has inspired recent endeavors on weakly-supervised temporal action localization methods. Specifically, for every training video, only a rough video-level action category yet not frame-wise labels are available.

This main scope of this work is an unexplored problem setting of unsupervised temporal action localization. In

Figure 1. Illustration of two co-attention models. Class-agnostic attention helps to find the important frames of a video. Class-specific attention models the temporal distribution of actions, which can be used for action localization. Note that actions are anonymous (obtained via co-attention piloted clustering) in ACL.

*Corresponding Author.

the unsupervised case, all we know regarding the training videos is an integer C which totals unique actions appearing in the video collection. For ease of statement, we term this new problem as *action co-localization* (ACL). To our best knowledge, our work is the first to address unsupervised temporal action localization.

To solve ACL, we propose a two-step "clustering + localization" iterative procedure. In the unsupervised case, true semantic annotations are missing, so we use clustering algorithm to group videos into C clusters, each of which defines a pseudo-action. Each unlabeled untrimmed video is assigned with a pseudo action class label based on clustering results. Then, an action localization model will be learned based on these noisy video-level pseudo-labels, which is capable of detecting action instances and predicting their pseudo-labels. The core of our proposed solution to ACL are two kinds of temporal co-attentions illustrated in Figure 1, optimized with action-background separation loss and cluster-based triplet loss respectively:

1) Inspired by classic image co-segmentation techniques [34, 15], we regard that videos of the same action (here approximated via action pseudo-label) share a common class-specific co-attention model. We get the class-specific action feature representation by the class-specific co-attention score. In particular, for videos belonging to the same cluster, we want to satisfy the following two criteria: high inter-video class-specific action feature representation similarity and high intra-video action-background feature distinctness. Based on these criteria, we design action-background separation loss to train the class-specific co-attention model. Once accurately learned, such co-attention models can be used to generate and rank action-specific proposals.

2) Since untrimmed videos usually contain a large portion of irrelevant backgrounds, we design a dataset-level class-agnostic co-attention model to learn the importance score of each frame. We get the class-agnostic video feature representation by the class-agnostic co-attention model. To pull the feature representation of the same cluster closer and push the video features belonging to different clusters further in the feature space, we design cluster-based triplet loss to train the class-agnostic co-attention model.

The clustering step and localization step reinforce each other. The clustering step provides noisy pseudo-labels for the localization step. All temporal co-attentions are then updated in the localization step. The class-agnostic co-attention model is in turn used to modulate the first video clustering step, ensuring that video frames with high attention scores play a more important role during clustering.

Importantly, weakly-supervised TAL, where video-level action class label is available, can be regarded as a special case of ACL and solved via our temporal co-attention models. In particular, videos with weak annotation can be grouped into C clusters according to video-level labels, therefore the first clustering is skipped.

The technical contributions of this work can be summarized as below: 1) To our best knowledge, it is the first work that explores unsupervised temporal action co-localization (ACL) in the literature; 2) This paper presents a novel two-step "clustering + localization" solution to the task of unsupervised ACL. In particular, we devise class-agnostic and class-specific temporal co-attentions, which are iteratively reinforced to gradually elevate the accuracy. We propose action-background separation loss and clustering-based triplet loss combined with cross-entropy loss to train both co-attention models; 3) Our comprehensive experiments on 20-action THUMOS14 and 100-action ActivityNet-1.2 have established first baselines and evaluation protocol for ACL. Surprisingly, the proposed model for ACL exhibits competitive performances to state-of-the-art weakly-supervised methods on both benchmarks. For example, our record on THUMOS14 is 25.0% in an unsupervised setting under mAP@0.5. Besides, our new record on THUMOS14 in a weakly-supervised setting under mAP@0.5 is 30.1% while the previous best is 26.8%.

2. Related Work

Fully-supervised action localization: This refers to the problem setting where all true action instances are labeled in detail, including temporal boundaries and action categories. One of the key challenges in supervised action localization is the vast number of candidate temporal windows that are drawn from varying scales and locations. Inspired by the R-CNN family of image object detectors [9], early development of action localization models [39, 48, 37, 49, 4, 24, 44, 22] adopt a two-stage "generate action proposals + rank proposals" paradigm. The first stage draws a large pool of possible temporal windows from videos (*e.g.*, by sliding windows [5, 39, 2, 8], temporal action grouping [57, 7], or boundary point detection [18, 16]) and quickly filters out a majority of them which are least likely to contain any action. The remaining proposals further go through more fine-grained inspection, such as temporal stage-aware SSN [57]. The most confident proposals are outputted as the predicted action instances. Besides the two-stage methods, there also exist other methods which adopt a framework of reinforcement learning [50, 11] or single-shot detection [1, 17] inspired by their counterparts in image object detection (*e.g.* YOLO [32] and SSD [21]).

Weakly-supervised action localization: Video-level labeling is more straightforward compared with that in segment-level, which is referred to as weakly-supervised action localization [43]. Most of the relevant approaches are strongly inspired by multiple instance learning (MIL) [25] or visual attention models [51]. Representative approaches include

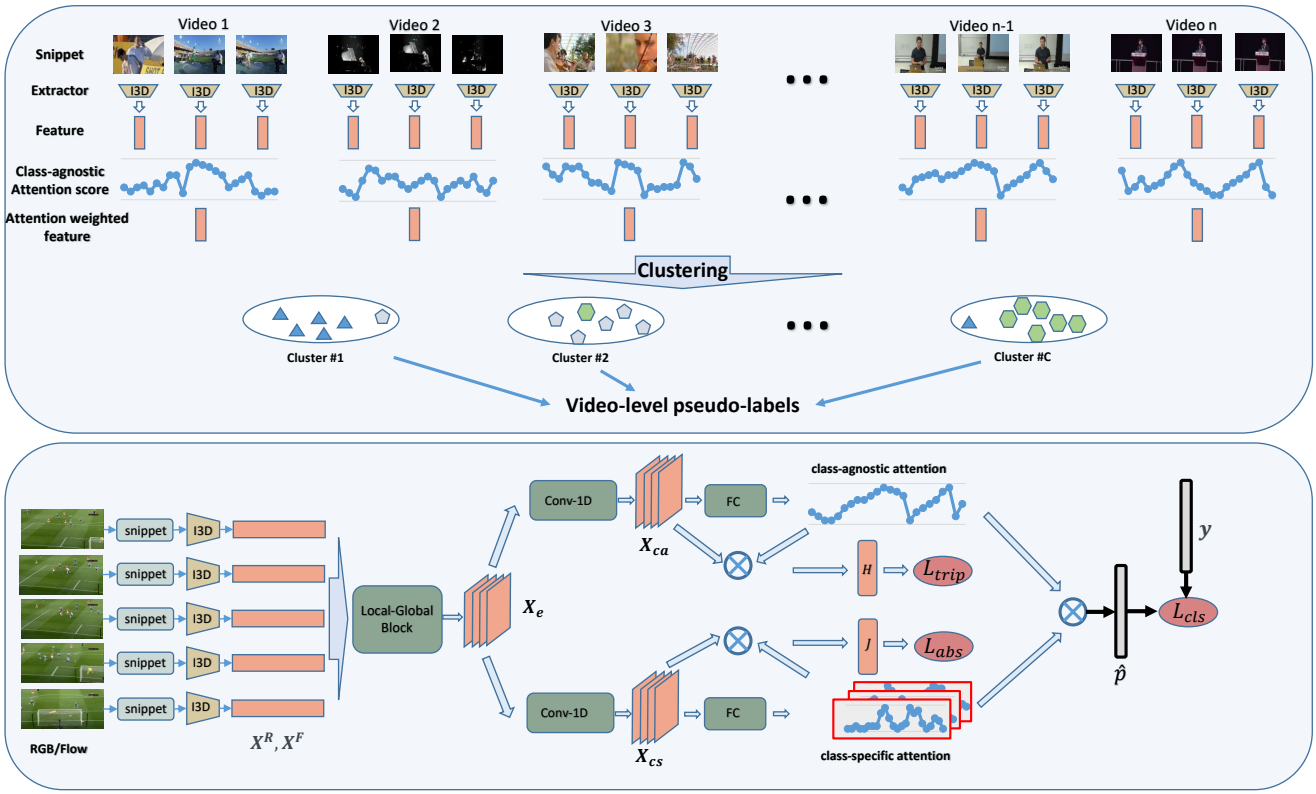

Figure 2. Overall architecture of our proposed model. The upper part is the clustering module and the lower part is the localization module. See section 3 for more details.

UntrimmedNet [46], Hide-and-Seek [41], W-TALC [29], Sparse Temporal Pooling Network [27], AutoLoc [38], and CleanNet[14], etc. Among them, UntrimmedNet [46] is comprised of classification/selection models and sparsity-encouraging regularization. The work in [58] identifies a key challenge as over-sparse supporting frames along the temporal scale. The authors thus propose to iteratively erase predecessor classifiers and enforce a new classifier to learn some complementary pieces. More recently, in [19], a multi-branch network with MIL loss and diversity loss is proposed to model action completeness. 3C-Net [26] propose to use multi-label center loss and action counting loss to reduce intra-class variations and enhance separability of adjacent action instances. In [28], the authors propose a background-aware loss to explicitly model background content. TSM [53] views each action as a multi-phase process and finds an optimal phase transition path to localize the actions.

Unsupervised action clustering / localization: Most relevant works to ours are [13, 42]. They sequentially did two jobs: unsupervised action clustering that group videos of similar human actions into separate action classes (*e.g.*, by spectral clustering and dominant set selection [42]), and localize the video tube-let that contains the actors. We would emphasize that those works have fundamental differences to the task of ACL: those methods mainly target trimmed videos and the "localization" is essentially spatial.

3. The Proposed Approach

In this section, we present our proposed method for ACL. Its extension to the weakly-supervised case can be trivially obtained by skipping the first clustering step, since we know the action category of each video in the case of weak supervision. Consider that we are provided with a training set of untrimmed videos $V = \{v_i\}_{i=1}^{N}$, where N is the number of videos. In the unsupervised case, we know the number of action categories C in the entire training set, but we don't know the specific category of each video.

3.1. Video Feature Extraction

Given an untrimmed video, we first divide it into a set of snippets, each consisting of several consecutive frames. Following the common practice in previous works, we extract the RGB and flow video features for each snippet. Let $X^R, X^F \in \mathbb{R}^{T \times D}$ denote the snippet-wise RGB and flow feature sequence respectively, where T denotes the number of snippets and D denotes dimension of feature.

3.2. Architecture Overview

The overall architecture is illustrated in Figure 2. Given the RGB or flow feature $X \in \mathbb{R}^{T\times D}$ of an input video v, we first use our proposed Local-Global Feature Aggregation Block to obtain an embedding feature $X_e \in \mathbb{R}^{T\times D_1}$. Then the network breaks into two branches, each composed of a convolution layer and a fully-connected layer. The outputs of the two branches are the class-agnostic attention weights $S \in \mathbb{R}^{T\times 1}$ and the class-specific attention weights $A \in \mathbb{R}^{T\times C}$ respectively. We denote the feature vectors before the fully-connected layers of the two branches as $X_{ca}, X_{cs} \in \mathbb{R}^{T\times D_2}$. Then we combine class-agnostic and class-specific attention weights to obtain class probability distribution $\hat{p}$ for the video.

In ACL, since ground-truth video label is not available, we perform clustering on the training data to assign each video with a pseudo label and use it to calculate a cross-entropy loss $\mathcal{L}_{cls}$. Meanwhile, we use the class-agnostic attention weight S to get a cluster-based triplet loss $\mathcal{L}_{trip}$, and use the class-specific attention weight A to get an action-background separation loss $\mathcal{L}_{abs}$. Combining the above losses, we get our overall loss function

$$\mathcal{L} = \mathcal{L}_{cls} + \alpha\mathcal{L}_{trip} + \beta\mathcal{L}_{abs}, \tag{1}$$

where α and β are coefficients. Details will be described in the following sections.

3.3. Co-Attention Piloted Video Clustering

This section presents using the acquired class-agnostic attention for video clustering. In ACL, we only know the number of action categories C of the training set. To get the video-level pseudo label for each video, we utilize the spectral clustering algorithm [36, 45, 52] on the training set to obtain C clusters, so that each video can be assigned with a pseudo label according to the cluster it belongs to. For each video v with T_v snippets, let $X_v^R, X_v^F \in \mathbb{R}^{T_v\times D}$ denote its RGB and flow stream features, respectively. Let $S_{v,i}^R, S_{v,i}^F \in \mathbb{R}^{T_v\times 1}$ denote its RGB and flow stream class-agnostic attention weights of v at iteration i, respectively. Since we don't have the importance scores of each video snippet at the first iteration, we set

$$S_{v,1}^R[j,1] = S_{v,1}^F[j,1] = \frac{1}{T_v}\,(1 \le j \le T_v)\,. \tag{2}$$

Due to untrimmed videos usually contain a large proportion of background frames, the video representation which is generated by average pooling along the temporal dimension is not discriminative. To extract the action-related video representation, at iteration $i\,(i>1)$, we use the class-agnostic temporal attention $S_{v,i}^R, S_{v,i}^F$ which is generated by action localization model at iteration $i-1$. Then RGB feature f_v^R and flow feature f_v^F of video v at iteration i is generated by:

$$f_v^R = L_2Norm((X_v^R)^\intercal S_{v,i}^R),\ f_v^F = L_2Norm((X_v^F)^\intercal S_{v,i}^F),$$

where $L_2Norm(\cdot)$ denotes L_2 normalization. Afterwards, f_v^R and f_v^F are concatenated to generated the two-stream feature representation f_v of video v. Given all training videos $\{v_i\}_{i=1}^N$ and their features $\{f_i\}_{i=1}^N$, we build a fully connected affinity graph $G = \{V, E\}$, where V denotes the collection of vertices, *i.e.*, training set videos, and E denotes the collection of edges. The edge weight w_{ij} of v_i and v_j is computed as $w_{ij} = \exp\left(-\frac{\|f_i-f_j\|_2^2}{2\sigma^2}\right)$,where $\sigma = \frac{1}{N^2}\sum_{i=1}^N\sum_{j=1}^N \|f_i - f_j\|_2$, $\|\|_2$ denotes the Euclidean distance. Then $W = [w_{ij}]$ is the affinity matrix of the graph. Based on the constructed affinity graph, we use spectral clustering algorithm [36] to group untrimmed videos into C clusters, each of which defines a pseudo-action. Each unlabeled untrimmed video is assigned with a pseudo action class label based on clustering results. Then, these video-level pseudo-labels are used to train action localization model. For the weakly-supervised extension, video-level labels are available for each video, thus clustering is skipped.

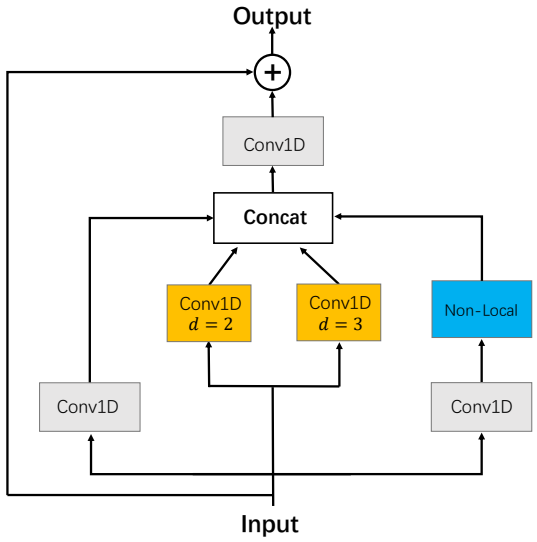

Figure 3. Local-Global Feature Aggregation Block. d is the dilation rate of temporal 1D convolution.

3.4. Local-Global Feature Aggregation Block

Since the feature of each snippet contains only the information of the current snippet, there is a lack of temporal context information. To improve the discriminability of each snippet feature, a Local-Global Feature Aggregation Block (FAB) is proposed to extract both local and global context information. The architecture is illustrated in Figure 3. FAB consists of three parts: a 1D temporal con-

volution branch, a dilated temporal pyramid branch, and a global context branch. The dilated temporal pyramid branch is composed of 2 parallel dilated convolutions with different dilation rates to aggregate local temporal context. The global context branch uses a non-local block [47] to capture the temporal correlation between all frames. A 1D temporal convolution with kernel size 1 is added before the global context branch to reduce the computation cost. The outputs of all branches are concatenated and fused via 1D temporal convolution. A skip connection is inserted after 1D temporal convolution.

3.5. Class-Specific Temporal Attention Module

In an untrimmed video, actions usually take place in a part of the video. We design Class-Specific Temporal Attention Module to get the probabilities of different action categories appearing at different times. These class-specific probabilities are further used for action localization. In this module, we take the class-specific feature X_{cs} as input and output a sequence of class-specific temporal attention scores $A \in \mathbb{R}^{T\times C}$. Then A is passed through a softmax along the temporal dimension, yielding the normalized class-specific temporal attention scores $\hat{A} = softmax(A)$.

In the task of unsupervised or weakly supervised temporal action localization, since there is no temporal annotation, the temporal boundary of action localization result is usually inaccurate. To get a more precise temporal boundary of action, we designed the action-background separation loss.

For a batch of training videos, we randomly sample videos from Z clusters and K videos of each cluster. Let $V_z = \{v_k\}_{k=1}^{K}$ denotes all videos belonging to the same cluster $z(1 \leq z \leq C)$ in a batch. The design of action-background separation loss is based on two criteria, namely high inter-video action similarity and high intra-video action-background distinctness. For each video v_k, we extract the action feature

$$J_k = {X_{cs,k}}^{\intercal} \hat{A}_k [:, z], \tag{3}$$

and background feature

$$B_k = \frac{1}{T_k - 1} {X_{cs,k}}^{\intercal} \left(\mathbf{1} - \hat{A}_k [:, z]\right), \tag{4}$$

where $\hat{A}_k$ is the normalized temporal attention score of v_k, $X_{cs,k}$ is the class-specific feature of v_k and T_k is the number of snippets of v_k. Assume we have a pair of videos v_m and v_n belonging to V_z. Let d denote the cosine distance function, τ_1 and τ_2 denote two positive margins respectively. To ensure the high inter-video action similarity, we use the following equation to enforce this requirement:

$$d(J_m, J_n) \leq \tau_1. \tag{5}$$

To satisfy high intra-video action-background distinctness, we use the following equations:

$$d(J_m, B_m) - d(J_m, J_n) \geq \tau_2, \tag{6}$$

$$d(J_n, B_n) - d(J_m, J_n) \geq \tau_2. \tag{7}$$

For videos belonging to the same cluster z, we calculate action-background separation loss of cluster z as follows:

$$\mathcal{L}_{inter,z} = \sum_{m=1}^{K} \sum_{n=1, n\neq m}^{K} max\{d(J_m, J_n) - \tau_1, 0\}, \tag{8}$$

$$\mathcal{L}_{intra,z} = \sum_{m=1}^{K} \sum_{n=1, n\neq m}^{K} max\{d(J_m, J_n) - d(J_m, B_m) + \tau_2, 0\}. \tag{9}$$

For a batch of videos, we sum the loss of all cluster in a batch as follows:

$$\mathcal{L}_{abs} = \sum_{z=1}^{Z} (L_{inter,z} + \theta \cdot L_{intra,z}), \tag{10}$$

where θ is a weight coefficient.

3.6. Class-Agnostic Temporal Attention Module

Untrimmed videos usually contain irrelevant backgrounds. For example, each video in the THUMOS14 validation set has 71% background on average. To alleviate the confusion caused by excessive background information, we hope to select the foreground part in which actions take place from the untrimmed video. So we designed the Class-Agnostic Temporal Attention Module to learn the attention score of each snippet. This module takes the class-agnostic feature X_{ca} as input and outputs a sequence of class-agnostic temporal attention score $S \in \mathbb{R}^{T\times 1}$.

To pull the video feature representation of the same cluster closer and push the video feature representation of different clusters further in the feature space, we utilize the triplet loss. Same with section 3.5, we randomly sample videos from Z clusters and K videos of each cluster. Then we extract the class-agnostic video feature representation H by $H = {X_{ca}}^{\intercal} S$.

Let d denote the cosine distance and m denote the positive margin. In cluster z, for each anchor video v_a, suppose v_n is the video which is not in cluster z and has the minimal distance to v_a, v_p is the video in cluster z and has the maximum distance to v_a. Suppose H_a, H_n, H_p are their class-agnostic video features respectively, then the following condition should be satisfied:

$$d(H_a, H_n) - d(H_a, H_p) \geq m \tag{11}$$

then we calculate the cluster-based triplet loss as follows:

$$\mathcal{L}_{trip} = \sum_{z=1}^{Z} \sum_{a=1}^{K} max\{d(H_a, H_p) - d(H_a, H_n) + m, 0\}. \tag{12}$$

To predict the action category of each video, we first get the normalized class-agnostic temporal attention score $\hat{S}$ by taking softmax operation along the temporal dimension on S. Then we calculate the weighted average $p = A\hat{S} \in \mathbb{R}^{C\times 1}$. We get probabilistic distribution over action classes $\hat{p}$ by performing softmax along the category dimension on p. The cross-entropy loss for a batch of videos is calculated by:

$$\mathcal{L}_{cls} = -\sum_{n=1}^{ZK}\sum_{i=1}^{C} y_{n,i}\log \hat{p}_{n,i}, \tag{13}$$

where y_n denotes the label of video v_n and $\hat{p}_n$ denotes the predicted label of video v_n.

3.7. Iterative optimization

In this paper, we propose a novel iterative optimization method to tackle the problem of unsupervised action localization. The method consists of two iterative steps: video clustering and temporal action localization.

Since there are no true semantic annotations, we first perform spectral clustering algorithm [36] on the training set, then assign each video with a pseudo label according to the cluster it belongs to. Pseudo video labels are then employed as the supervision information to train the localization part and temporal co-attentions are updated. As discussed in section 3.3, the class-agnostic co-attention model guides the video clustering step, ensuring video frames with high attention scores play a important role during clustering.

On the one hand, a better co-attention model S helps to find the important frames of the video and yield a better feature representation f_v for the video. On the other hand, a more precise feature representation leads to more precise pseudo labels obtained from the clustering process, and in turn provides better supervision for the localization. Our experiments show this iterative optimization process indeed gradually improves the performance of both steps.

3.8. Action Localization by Class-Specific Co-Attentions

Given a test video, we first use the trained localization network to obtain the class-specific attention A and video-level classification score $\hat{p}$. Then we threshold on $\hat{p}$, and find all the categories c satisfying $\hat{p}_c > \tau$. Let $[\alpha_0, ..., \alpha_r]$ be a set of threshold values. Each α_j is used to threshold on $A[:, c]$ and obtain a set of localization proposals.Each proposal has the form of (b_i, e_i, c_i), where b_i, e_i denote the start and end time of the i^{th} detected action and c_i is the predicted category. Following [19], we combine the Outer-Inner Contrastive loss in [38] and the video-level class score $\hat{p}_c$ to score each action proposal:

$$score_i = avg(A[inner, c]) - avg(A[outer, c]) + \gamma\hat{p}_c, \tag{14}$$

	Methods	mAP@IoU (%)				
		0.3	0.4	0.5	0.6	0.7
FS	SLM-mgram [33]	30.0	23.2	15.2	-	-
	Glimpse [50]	36.0	26.4	17.1	-	-
	PSDF [54]	33.6	26.1	18.8	-	-
	S-CNN [39]	36.3	28.7	19.0	10.3	5.3
	SSAD [17]	43.0	35.0	24.6	-	-
	CDC [37]	40.1	29.4	23.3	13.1	7.9
	R-C3D [48]	44.8	35.6	28.9	-	-
	SSN [57]	51.9	41.0	29.8	-	-
	TAL-Net [4]	53.2	48.5	42.8	33.8	20.8
WS	Hide-and-seek [41]	19.5	12.7	6.8	-	-
	UntrimmedNet [46]	28.2	21.1	13.7	-	-
	STPN [27]	35.5	25.8	16.9	9.9	4.3
	Autoloc [38]	35.8	29.0	21.2	13.4	5.8
	W-TALC [29]	40.1	31.1	22.8	-	7.6
	MAAN [55]	41.1	30.6	20.3	12.0	6.9
	CMCS [20]	41.2	32.1	23.1	15.0	7.0
	3C-Net [26]	44.2	34.1	26.6	-	8.1
	BM [28]	46.6	37.5	26.8	17.6	9.0
	TSM [53]	39.5	-	24.5	-	7.1
	CleanNet [14]	37.0	30.9	23.9	13.9	7.1
	Ours	**46.9**	**38.9**	**30.1**	**19.8**	**10.4**
US	Ours	39.6	32.9	25.0	16.7	8.9

Table 1. Comparisons on the THUMOS14 test set for action detection. We denote fully-supervised, weakly-supervised and unsupervised as FS, WS and US respectively.

where $inner$ denotes the predicted action boundary (b_i, e_i), and $outer$ denotes the inflated region $(b_i-(e_i-b_i)/4, b_i)\cup(e_i, e_i+(e_i-b_i)/4)$. γ is a trade-off coefficient. Notice that the proposals for different α_i may overlap, so we perform non-maximum suppression(NMS) on all these proposals and get the final localization output.

4. Evaluations

4.1. Data Description and Evaluation Protocol

We evaluate our method on two large-scale benchmark datasets: THUMOS14 and ActivityNet-1.2. The videos contained in both datasets are untrimmed, implying that videos contain some frames that are not from any target action.

THUMOS14 [12]. A subset of the THUMOS14 dataset contain videos with temporal annotations from 20 action classes. Following the previous convention [4, 19], we train our model with 200 untrimmed videos in the validation set and evaluate our method on the test set of 212 videos.

ActivityNet-1.2 [10]. To facilitate comparisons, we conduct experiments on ActivityNet-1.2 which contains 4,819 training videos, 2,383 validation videos and 2,480 testing videos from 100 activity classes. Since the test set annotations of this dataset are withheld, we train our model on the training set and perform evaluations on the validation set as in previous works [29, 26].

Cluster evaluation protocol. To measure the clustering performance, we use three criteria, *i.e*., purity, normalized mutual information score (NMI) and adjust rand index (ARI), which are widely used in the clustering task [31]. Bigger values of these criteria indicate better clustering performance.

Action localization evaluation protocol. For temporal action localization task, we report the traditional mean Average Precision (mAP) [6] at different temporal intersection over union(IoU). The IoU thresholds are 0.3, 0.4, 0.5, 0.6, 0.7 on THUMOS14. The IoU thresholds are 0.5, 0.75, 0.95 on ActivityNet-1.2. The average mAP with IoU thresholds [0.5:0.95:0.05] is used to compare different methods on ActivityNet-1.2.

4.2. Implementation Details

We utilize two-stream architecture [40] to extract features for video frames. In our experiment, two separate I3D [3] models are trained from consecutive frames and flow on Kinetics [3], respectively. I3D takes non-overlapping snippets of 16 stacked RGB or optical flow frames as input and extracts 1024-dimensional feature for each stream. We adopt the late fusion of the RGB and optical flow streams to generate the final action localization results.

Our action localization model is implemented in PyTorch. It is trained with a mini-batch size of 24 using Adam optimizer with weight decay 0.001. For a batch of training data, we randomly sample videos from 12 clusters and 2 videos of each cluster. The learning rates are set as 0.001 and 0.0001 for ActivityNet-1.2 and THUMOS14 respectively. We set both α and β in Eq.1 to 0.5. For action-background separation loss, we set τ_1, τ_2 as 0.0001 and 0.25 respectively. θ in Eq.10 is set as 1. For triplet loss, we set margin parameter m as 0.5 in Eq.12. When generating action localization results, we only keep classes whose video-level probabilities are above 0.1. For class c with $\hat{p}_c > 0.1$, we use a set of threshold values ranging from $[0.1 : 1.0 : 0.1] \times mean\,(A[:, c])$, where $mean$ get the mean value of $A[:, c]$. γ in Eq.14 is empirically set as 0.1. Proposals generated by different thresholds are combined, then NMS is used to remove duplicate localization results.

For ACL, we only know the cluster index that each proposal belongs to. To make comparisons with other fully-supervised or weakly-supervised TAL methods, we need to further map the cluster indices to action classes of THUMOS14 or ActivityNet-1.2 to get the class label for each proposal in the testing step. Specifically, we map each cluster to the action class which occurs most frequently in this cluster. In addition, since some videos may contain multiple actions of different classes (*e.g*., some videos in THUMOS14 contain both diving and cliff diving), sometimes we should map one cluster to more than one action. In our experiment, suppose in cluster z, C_A is the most frequently occurred action class and appears N_A times, then for any action class C_B whose number of occurrence in cluster z is greater than $\frac{N_A}{2}$, we also map cluster z to action C_B.

	Methods	mAP@IoU (%)			
		0.5	0.75	0.95	Average
WS	FC-CRF [58]	27.3	14.7	2.9	15.6
	AutoLoc [38]	27.3	15.1	3.3	16.0
	W-TALC [29]	37.0	-	-	18.0
	CMCS [20]	36.8	22.0	**5.6**	22.4
	3C-Net [26]	37.2	-	-	21.7
	TSM [53]	28.3	17.0	3.5	-
	CleanNet [14]	37.1	20.3	5.0	21.6
	Ours	**40.0**	**25.0**	4.6	**24.6**
US	Ours	35.2	21.4	3.1	21.1

Table 2. Comparisons on the ActivityNet1.2 dataset for action detection.We denote weakly-supervised and unsupervised as WS and US respectively.

Iteration	Purity↑	ARI↑	NMI↑
1	0.645	0.445	0.726
2	0.740	0.569	0.788
3	0.780	0.612	0.811

Table 3. Comparing video clustering results of different iterations on THUMOS14 validation set.

4.3. Comparisions with state-of-the-art

Table 1 summarizes the results of the THUMOS14 test set when the IoU threshold varies from 0.3 to 0.7. Specifically, for mAP@0.5, the result achieved by our method in the unsupervised case is comparable to the results obtained by the state-of-the-art weakly-supervised methods [26, 28] and outperform all other recent weakly-supervised methods. In weakly supervised case, we improve the mAP@0.5 from previous state-of-the-art 26.8% to 30.1%.

Table 2 presents the results on the benchmark ActivityNet-1.2. We compare our method with other recent state-of-the-art weakly-supervised action localization methods. Even without video class annotations, our method in unsupervised case achieves average mAP of 21.1%, showing competitive performance comparing to several recent weakly supervised methods. In weakly-supervised case, our method outperforms the state-of-the-art weakly-supervised methods by 2.2% in terms of average mAP.

4.4. Effectiveness of iterative optimization

To demonstrate the effectiveness of our iterative optimization method, we evaluate the clustering and action localization performance at each iteration.

Clustering performance w.r.t. iterations: Table 3 compares the clustering performance of different iterations on THUMOS14 validation set. As the number of iterations in-

Iteration	mAP@IoU (%)				
	0.3	0.4	0.5	0.6	0.7
1	21.4	17.1	13.0	8.1	4.0
2	33.6	27.8	20.8	13.1	7.1
3	39.6	32.9	25.0	16.7	8.9

Table 4. Comparing action localization results of different iterations on THUMOS14 test set.

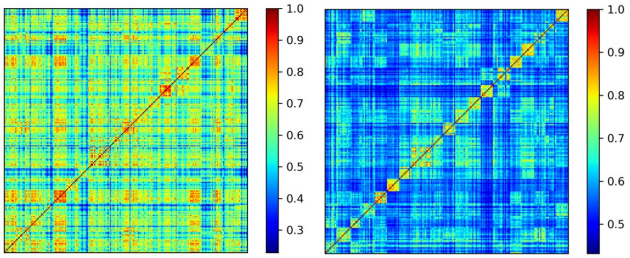

Figure 4. Visualize affinity matrices used for spectral clustering [36] of iteration 1(left) and iteration 3(right) on THUMOS14 validation set. For clarification, videos are arranged according to their classes.

creases, the performance of clustering is getting better. Figure 4 shows the visualization of affinity matrices used for spectral clustering in section 3.3. It can be seen that with the help of class-agonic temporal attention pooling method, the video representation concentrates more on frames related to video actions and is more discriminative. Therefore we can get better clustering results and reduce the noise of pseudo-labels.

mAP w.r.t. iterations: Action localization results of different iterations on THUMOS14 test set are shown in Table 4. Since the performance of clustering increases with the number of iterations, the error rate of pseudo label decreases in the meantime. As the quality of pseudo-labels improves, our ACL model can learn more precise attention weights, so the performance of temporal action localization can be improved by a non-trivial margin. As demonstrated in Table 4, we achieved an mAP of 13.0% when the IoU threshold is 0.5 at iteration 1. As the number of iterations increase, the mAP gets improved. We finally get an mAP of 25.0% under the unsupervised setting. Figure 5 shows the qualitative action localization results of different iterations. We can observe that as the number of iteration increases, the action localization results are more precise in Figure 5.

4.5. Ablation studies

To analyze the contribution of each model component, we perform ablation studies on the THUMOS14 test set in the unsupervised case. The average mAP at the IoU threshold from 0.3 to 0.7 is used to compare different settings. Results in Table 5 show the comparison of all settings. We use the model in which the Local-Global feature aggregation block is replaced with standard 1D temporal convolutions combined with only classification loss as our baseline. We add Local-Global feature aggregation block to the baseline model and improve the average mAP from 21.39% to 22.16%. Based on the model with Local-Global feature aggregation block, we explore the contributions of each loss function by adding the proposed losses. As we can see in Table 5, the action-background loss improves the performance by 1.32%, and the triplet loss improves the performance by 0.89%. Finally, we use all losses to train the action localization model and achieve an mAP of 24.68%, implying that each loss contributes to the overall performance.

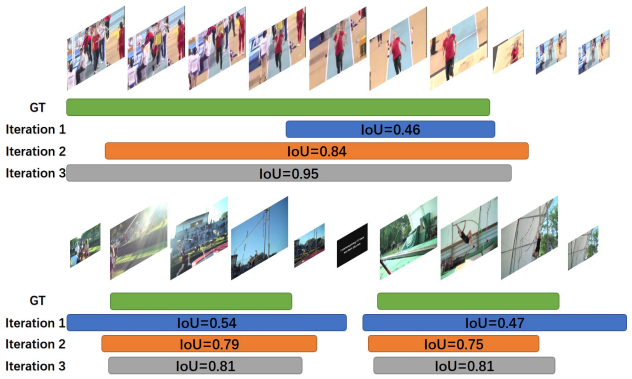

Figure 5. Qualitative examples of localization results by our method on THUMOS14 dataset. GT denotes the ground truth segments. Iteration 1,2 and 3 denote localization results of different iterations respectively.

Methods	Avg(0.3:0.7)
Conv1D+$\mathcal{L}_{cls}$	21.39
LG+$\mathcal{L}_{cls}$	22.16
LG+$\mathcal{L}_{cls}$ + $\mathcal{L}_{abs}$	23.48
LG+$\mathcal{L}_{cls}$ + $\mathcal{L}_{trip}$	23.05
LG+$\mathcal{L}_{cls}$ + $\mathcal{L}_{abs}$ + $\mathcal{L}_{trip}$	24.68

Table 5. Ablation studies results on THUMOS14 test set.

5. Conclusions

We address the problem of unsupervised action localization for the first time. A two-step "clustering + localization" framework is developed and validated on two large-scale benchmarks using our proposed evaluation metrics. The major contributions include novel temporal co-attention models and several loss function specially designed for this new task. Moreover, our formulation can be easily extended to the weakly-supervised case. Our experiments re-calibrate the state-of-the-art performances under the weakly-supervised setting and achieved surprisingly competitive results under the unsupervised setting. **Acknowledgement**: This work is supported by Beijing Municipal Commission of Science and Technology (Z181100008918005), Beijing Natural Science Foundation (Z190001), and Tencent AI Lab Rhino-Bird Focused Research Program (JR202021).

References

[1] Shyamal Buch, Victor Escorcia, Bernard Ghanem, Li Fei-Fei, and Juan Carlos Niebles. End-to-end, single-stream temporal action detection in untrimmed videos. In *BMVC*, 2017.

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

[3] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, 2017.

[4] 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.

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

[6] Mark Everingham, Luc Van Gool, Christopher K. I. Williams, John M. Winn, and Andrew Zisserman. The pascal visual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010.

[7] Jiyang Gao, Kan Chen, and Ram Nevatia. CTAP: complementary temporal action proposal generation. In *ECCV*, 2018.

[8] Jiyang Gao, Zhenheng Yang, Chen Sun, Kan Chen, and Ram Nevatia. TURN TAP: temporal unit regression network for temporal action proposals. In *ICCV*, 2017.

[9] Ross B. Girshick. Fast R-CNN. In *ICCV*, 2015.

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

[11] Jingjia Huang, Nannan Li, Tao Zhang, Ge Li, Tiejun Huang, and Wen Gao. SAP: self-adaptive proposal model for temporal action detection based on reinforcement learning. In *AAAI*, 2018.

[12] Haroon Idrees, Amir Roshan 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.

[13] Tian Lan, Yang Wang, and Greg Mori. Discriminative figure-centric models for joint action localization and recognition. In *ICCV*, 2011.

[14] Ziyi Liu Le Wang, Qilin Zhang, Zhanning Gao, Zhenxing Niu, Nanning Zheng, and Gang Hua. Weakly supervised temporal action localization through contrast based evaluation networks. 2019.

[15] Weihao Li, Omid Hosseini Jafari, and Carsten Rother. Deep object co-segmentation. In *ACCV*, 2018.

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

[17] Tianwei Lin, Xu Zhao, and Zheng Shou. Single shot temporal action detection. In *ACM Multimedia*, 2017.

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

[19] Daochang Liu, Tingting Jiang, and Yizhou Wang. Completeness modeling and context separation for weakly supervised temporal action localization. In *CVPR*, 2019.

[20] Daochang Liu, Tingting Jiang, and Yizhou Wang. Completeness modeling and context separation for weakly supervised temporal action localization. In *CVPR*, 2019.

[21] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott E. Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: single shot multibox detector. In *ECCV*, 2016.

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

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

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

[25] Oded Maron and Tomás Lozano-Pérez. A framework for multiple-instance learning. In *NIPS*, 1997.

[26] Sanath Narayan, Hisham Cholakkal, Fahad Shabaz Khan, and Ling Shao. 3c-net: Category count and center loss for weakly-supervised action localization. 2019.

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

[28] Phuc Xuan Nguyen, Deva Ramanan, and Charless C Fowlkes. Weakly-supervised action localization with background modeling. *ICCV*, 2019.

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

[30] A. J. Piergiovanni and Michael S. Ryoo. Temporal gaussian mixture layer for videos. In *ICML*, 2019.

[31] William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

[32] Joseph Redmon, Santosh Kumar Divvala, Ross B. Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *CVPR*, 2016.

[33] Alexander Richard and Juergen Gall. Temporal action detection using a statistical language model. In *CVPR*, 2016.

[34] Carsten Rother, Thomas P. Minka, Andrew Blake, and Vladimir Kolmogorov. Cosegmentation of image pairs by histogram matching - incorporating a global constraint into mrfs. In *CVPR*, 2006.

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Fei-Fei Li. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[36] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(8):888–905, 2000.

[37] 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.

[38] Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. Autoloc: Weakly-supervised temporal action localization in untrimmed videos. In *ECCV*, 2018.

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

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

[41] 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.

[42] Khurram Soomro and Mubarak Shah. Unsupervised action discovery and localization in videos. In *ICCV*, 2017.

[43] Haisheng Su, Xu Zhao, and Tianwei Lin. Cascaded pyramid mining network for weakly supervised temporal action localization. *CoRR*, abs/1810.11794, 2018.

[44] Rui Su, Wanli Ouyang, Luping Zhou, and Dong Xu. Improving action localization by progressive cross-stream cooperation. In *CVPR*, 2019.

[45] Ulrike von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17(4):395–416, 2007.

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

[47] Xiaolong Wang, Ross B. Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018.

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

[49] Ke Yang, Peng Qiao, Dongsheng Li, Shaohe Lv, and Yong Dou. Exploring temporal preservation networks for precise temporal action localization. In *AAAI*, 2018.

[50] 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.

[51] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *CVPR*, 2016.

[52] Stella X. Yu and Jianbo Shi. Multiclass spectral clustering. In *ICCV*, pages 313–319, 2003.

[53] Tan Yu, Zhou Ren, Yuncheng Li, Enxu Yan, Ning Xu, and Junsong Yuan. Temporal structure mining for weakly supervised action detection. In *ICCV*, 2019.

[54] Jun Yuan, Bingbing Ni, Xiaokang Yang, and Ashraf A. Kassim. Temporal action localization with pyramid of score distribution features. In *CVPR*, 2016.

[55] Yuan Yuan, Yueming Lyu, Xi Shen, Ivor W. Tsang, and Dit-Yan Yeung. Marginalized average attentional network for weakly-supervised learning. In *ICLR*, 2019.

[56] 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.

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

[58] Jia-Xing Zhong, Nannan Li, Weijie Kong, Tao Zhang, Thomas H. Li, and Ge Li. Step-by-step erasion, one-by-one collection: A weakly supervised temporal action detector. In *ACM Multimedia*, 2018.