Learning to Refactor Action and Co-occurrence Features for Temporal Action Localization

Kun Xia[1] Le Wang[1*] Sanping Zhou[1] Nanning Zheng[1] Wei Tang[2]

[1]Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University

[2]University of Illinois at Chicago

xiakun@stu.xjtu.edu.cn; {lewang, spzhou, nnzheng}@xjtu.edu.cn; tangw@uic.edu

Abstract

The main challenge of Temporal Action Localization is to retrieve subtle human actions from various co-occurring ingredients, e.g., context and background, in an untrimmed video. While prior approaches have achieved substantial progress through devising advanced action detectors, they still suffer from these co-occurring ingredients which often dominate the actual action content in videos. In this paper, we explore two orthogonal but complementary aspects of a video snippet, i.e., the action features and the co-occurrence features. Especially, we develop a novel auxiliary task by decoupling these two types of features within a video snippet and recombining them to generate a new feature representation with more salient action information for accurate action localization. We term our method RefactorNet*, which first explicitly factorizes the action content and regularizes its co-occurrence features, and then synthesizes a new action-dominated video representation. Extensive experimental results and ablation studies on THUMOS14 and ActivityNet v1.3 demonstrate that our new representation, combined with a simple action detector, can significantly improve the action localization performance.*

1. Introduction

Temporal Action Localization (TAL) aims to locate the start and end times of action instances from long untrimmed videos as well as to classify their categories. As a fundamental task in video understanding, it has attracted great attention in recent years and facilitated various applications such as security surveillance [36,43,50] and human behavior analysis [9,39,53].

A common first step of current TAL approaches [12,33, 46,52] is to extract features from each video snippet via a pre-trained two-stream network. They are then aggregated, e.g., via max pooling, to obtain the representation of a proposal
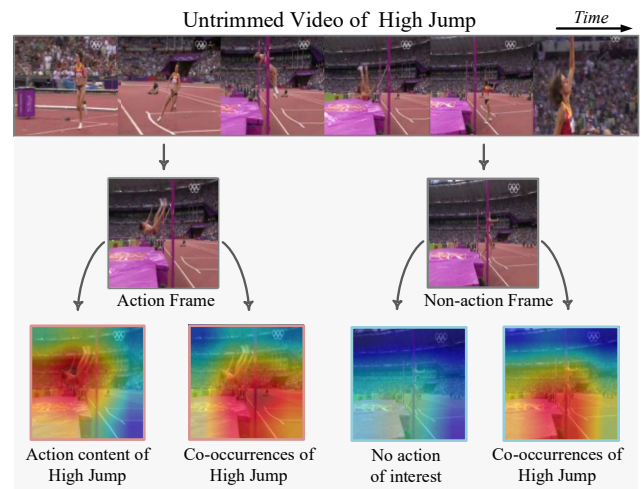

Figure 1. Visualization of the action component and the co-occurrence component decoupled by our method from a snippet representation. Co-occurrence components help reduce action ambiguity and uncertainty, but they often dominate the action components in a video, and thus adversely affect action detection. How to balance these two components in a snippet representation is an important yet under-explored problem.

or an anchor for action classification and temporal boundary regression. These snippet-level features characterize two aspects of a video snippet, and we term them the *action component* and the *co-occurrence component*, respectively. The **action component** refers to features characterizing the action occurring in a snippet, including the motion pattern of one or more persons and their interaction with objects. The **co-occurrence component** refers to features not characterizing any actions but often co-occurring with them in a frame or a snippet. This includes class-specific *context*, which only co-occurs frequently with certain actions, e.g., a track field, and class-agnostic *background*, whose occurrence is less relevant to the action categories, e.g., the sky. Figure 1 illustrates that an untrimmed video of *High Jump* contains a

---

*Corresponding author.

subtle action component and a richer co-occurrence component surrounding the action.

It is critical to treat both the action and co-occurrence components carefully to achieve robust TAL. On the one hand, while the action component directly characterizes an action, it can be ambiguous and uncertain because the motion and appearance of humans and their interactions with each other and with objects are complex. Thus, only retaining the action component from the snippet-level features is insufficient. On the other hand, while some co-occurrence components are useful to reduce the action ambiguity and uncertainty, e.g., a swimming pool distinguishing "diving" from "trampoline", some others are more like nuisance noise, e.g., audience and random persons in the scene, and over-relying on the co-occurrence component also blurs the action boundaries. Therefore, it is necessary to find an appropriate balance between the action component and the co-occurrence component in a feature representation, especially as the latter often dominates the former in a video. This problem has been largely ignored by prior work.

This paper investigates *feature refactoring* for TAL. It means to *decouple* the snippet-level features as the action component and the co-occurrence component, and then *recombine* them into a more appropriate representation to achieve effective TAL.

We propose a novel *Feature Refactoring Network* or *RefactorNet* to achieve this goal. It consists of a feature decoupling module and a feature recombining module. Since the annotations of co-occurrence components are unavailable, we collect action samples and their *coupling samples* from the whole video for the decoupling process. Coupling samples refer to any video snippets containing co-occurring elements of the action but do not involve the actual action. In other words, an action sample and its coupling sample share similar co-occurrence components but differ in whether the action component is present. Taking the action samples and coupling samples as supervision, the feature decoupling module is trained to separate action and co-occurrence components. Then, the feature recombining module synthesizes these two components into a new snippet representation containing a more salient action component and a more suitable co-occurrence component for accurate action detection.

Both quantitative and qualitative results demonstrate that our RefactorNet can effectively decouple the action and co-occurrence components, and the recombined snippet representations improve both action classification and temporal boundary regression. Combined with a simple action detector, our RefactorNet achieves state-of-the-art performance on two benchmarks, THUMOS14 and ActivityNet v1.3.

Our contributions can be summarized as follows:

- Co-occurrence components help reduce action ambiguity and uncertainty, but they often dominate the action components in a video, and thus adversely affect TAL.

How to balance these two components in a snippet representation is an important yet under-explored problem. Our proposed RefactorNet is the first approach to explicitly refactor, *i.e.*, decouple and recombine, these two components to obtain a new snippet representation containing a more salient action component and a more suitable co-occurrence component for accurate action detection.

- Decoupling the two components is indeed very challenging because they co-occur frequently, and more severely, their annotations are unavailable. To address these difficulties, we carefully design the learning objective, and introduce action samples and their coupling samples, which can be obtained from standard TAL annotations, to supervise the decoupling process. They together help our feature decoupling module separate the two components effectively.

- Our RefactorNet outperforms all state-of-the-art methods on two benchmark datasets. Extensive ablation study and visualizations are provided to show in-depth analysis of the decoupling process and validate how it improves TAL.

2. Related Work

Temporal Action Localization. Recently, TAL has made substantial progress. CDC [35] performs convolution and de-convolution on the input video for dense score prediction, and combines segment proposals to detect action instances. BSN [24] densely predicts the start and end probabilities of actions to generate high-quality action proposals with precise boundaries. Yang *et al.* [47] propose the TPC network to preserve the temporal resolution, enabling frame-level granularity action localization with minimal loss of temporal information. GTAN [30] proposes a one-stage TAL framework to model action instances with various lengths through learning a set of Gaussian kernels. AFSD [22] presents a purely anchor-free TAL model to predict the temporal distance of the start and end from each location and the action class. In addition, other works [10, 16, 21, 48, 55] have attractive ideas that advance the TAL field further. Especially, some researchers realize that accurately inferring human actions requires understanding video context. To this end, they model relations between proposals or between snippets, *e.g.*, P-GCN [51], G-TAD [46], TCANet [33], ContextLoc [57], to enhance their features. In contrast, this paper addresses a different problem that the co-occurrence component within a video snippet often dominates the actual action, through explicitly decoupling and recombining these two components within each individual snippet. Therefore, our feature refactoring is complementary to relation modeling [33, 46, 51, 57], which is also validated by our experiments.
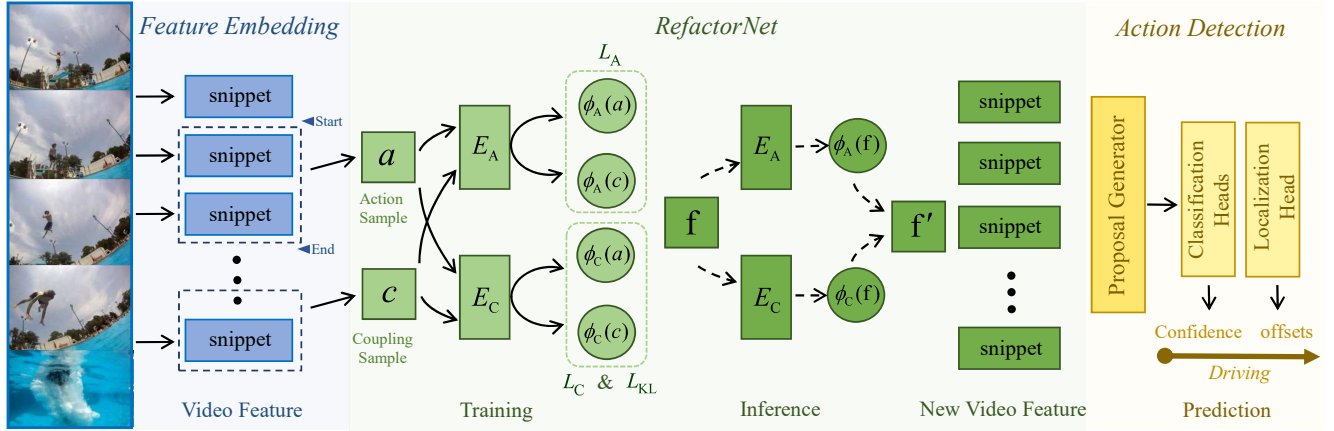
Figure 2. The overall pipeline of the proposed framework. It consists of three stages, *i.e.*, *Feature Embedding*, *RefactorNet*, and *Action Detection*. First, the feature embedding stage extracts original snippet features through a two-stream network. Subsequently, our proposed RefactorNet is trained to decouple a snippet representation into an action component and its co-occurrence component, supervised by pairs of action samples and coupling samples. During the inference process, each snippet feature vector $\mathbf{f}$ is decoupled into an action component $\phi_{\mathrm{A}}(\mathbf{f})$ and a co-occurrence component $\phi_{\mathrm{C}}(\mathbf{f})$, which are then recombined to generate a new snippet feature vector $\mathbf{f}'$. Finally, the action detector locates actions from the video via the new snippet representations.

Other related works extract the action frames of interest from false positive frames in Weakly-Supervised Temporal Action Localization (WSTAL), where only the video-level labels are available. BaSNet [20] introduces an auxiliary background class to suppress the interference of background frames to actual action frames. Liu *et al.* [29] divide video snippets into positives and negatives to learn action and context sub-spaces. Different with them, we do not introduce additional supervision for the co-occurrence component, as it may require detailed annotations and it is difficult to generalize to larger datasets. Moreover, we do not consider the co-occurrence component as negative samples, which often contain contextual information useful to action classification.

Disentangled Representation Learning. The objective of disentangled representation learning is to extract interesting features or synthesize new visual representation, which has developed significantly in recent years [6, 7, 32, 41, 49] and has been widely applied in various fields, *e.g.*, image deblurring [31], face identification [28, 44] and semantic segmentation [25], etc. Villegas *et al.* [40] decompose the inputs for video prediction into motion and content. Thereafter, predicting the next frame reduces to converting the extracted content features into the next frame content by the identified motion features. Hsieh *et al.* [14] decompose the high-dimensional video into low-dimensional temporal dynamics to predict future video frames. Wang *et al.* [42] exploit action samples and the corresponding conjugate samples to deliberately separate action from context for human action recognition. Eom *et al.* [8] present a GAN to disentangle identity-related and -unrelated features using identification labels for person re-identification. Hamaguchi *et al.* [13] use disentangled representation learning to disentangle variant

and invariant factors that represent trivial events and image contents for the rare event detection task. Bahng *et al.* [1] develop a novel algorithm, ReBias. It solves a min-max problem, where the target is to promote the independence between the network prediction and all biased predictions. Singh *et al.* [37] focus on addressing context biases for visual classifier by explicitly learning a robust feature subspace of a category. Huang *et al.* [15] design a self-supervised video representation learning method to decouple the context and motion representations from compressed videos. Inspired by these literature, in this paper, we introduce disentangled representation learning into the field of TAL.

3. RefactorNet

The pipeline of our entire framework is presented in Figure 2. The proposed RefactorNet is inserted between the feature embedding module and the action detection module to refactor the snippet feature representation. We elaborate the proposed network and action detection framework below.

3.1. Problem Setting

Given an untrimmed video, we extract the visual features of every a few consecutive frames as a snippet feature vector $\mathbf{f} \in \mathbb{R}^C$, where C is the feature dimension. Then, we can represent the video feature sequence as $\mathbf{F} \in \mathbb{R}^{C \times L}$, where L is the number of snippets in the video. The ground truth annotation is a set of action instances $\Psi = \{\varphi_n = (t_{s,n}, t_{e,n}, c_n)\}_{n=1}^N$, where $t_{s,n}$, $t_{e,n}$ and c_n represent the start time, end time and action class of an action instance φ_n, respectively. N is the total number of action instances in the video.

3.2. Motivation and Overview

Prior methods perform TAL upon the original video features $\mathbf{F}$ extracted by a two-stream network. However, the co-occurrence component, including context and background, within each video snippet often dominates the subtle action component. While some co-occurrence components are useful to reduce the action ambiguity and uncertainty, a TAL model can over-rely on the co-occurrence component after training, and confuses it with the actual action content during testing. We seek an appropriate balance between action and co-occurrence components, through amplifying the action signal within the video snippet and regularizing the co-occurrence component, to improve action localization.

To this end, we propose a novel RefactorNet, as illustrated in Figure 2. It aims at synthesizing new snippet features through decoupling and recombining the action component and the co-occurrence component based on action samples and coupling samples. Finally, the new feature representation is used for action localization. Specifically, our approach consists of three steps. (1) Collect action samples and their coupling samples. (2) Explicitly decouple action and co-occurrence components based on pairs of action samples and coupling samples. (3) Effectively recombine these two components and synthesize new snippet features for video action detection.

3.3. Collecting action samples and coupling samples

For an untrimmed training video with its annotation Ψ, we can divide all snippets of the video into action snippets and non-action snippets, depending on whether they are within the extent of an action instance. Subsequently, we treat action snippets of an action instance as an action sample, denote its feature vector as $\boldsymbol{a}$, and record its category label. We collect all action samples from the video. To enable feature decoupling, we collect high-quality coupling samples through retrieving the non-action snippets that have high cosine similarities with at least one action sample. Accordingly, the feature vector of a coupling sample is denoted as $\boldsymbol{c}$. Finally, we obtain a collection of action samples $\mathbb{A}$ and a collection of coupling samples $\mathbb{C}$ from all training videos. We report some visual examples of action samples and coupling samples in Figure 3.

3.4. Decoupling Action and Co-occurrence Components

We now decouple the action component and the co-occurrence component from the snippet features. A coupling sample does not include any actual action information as it is a non-action snippet. But, by construction, its features are similar to those of the corresponding action sample, indicating the coupling sample includes co-occurrence information of the human action, e.g., Figure 3. In other words, an action sample and its coupling sample share similar co-occurrence
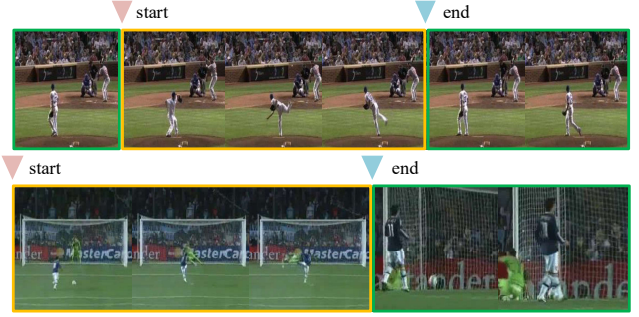

Figure 3. Examples of action samples (yellow) and coupling samples (green) for "Baseball Pitch" (top) and "Soccer Penalty" (bottom).

components but differ in whether the action component is present. Therefore, we propose to exploit both the similarity and difference between action samples and their coupling samples to factorize action contents from the dominating co-occurrence features.

As illustrated in Figure 2, we design two encoders ϕ_{A} and ϕ_{C} for feature decoupling. The two encoders have the same network structure but do not share weights. They take an action sample $\boldsymbol{a} \in \mathbb{A}$ and its coupling sample $\boldsymbol{c} \in \mathbb{C}$ as input, and output an action feature vector and a co-occurrence feature vector for each sample, respectively. They are learned by minimizing two loss functions below:

$$\mathcal{L}_{\mathrm{A}} = \max\left\{0, \cos\left\langle\phi_{\mathrm{A}}(\boldsymbol{a}), \phi_{\mathrm{A}}(\boldsymbol{c})\right\rangle\right\}, \tag{1}$$

$$\mathcal{L}_{\mathrm{C}} = \left(1 - \cos\left\langle\phi_{\mathrm{C}}(\boldsymbol{a}), \phi_{\mathrm{C}}(\boldsymbol{c})\right\rangle\right), \tag{2}$$

where ϕ_{A} extracts the action component of a sample, and ϕ_{C} extracts the co-occurrence component of a sample. $\mathcal{L}_{\mathrm{A}}$ is used to minimize the similarity between the action component in an action sample, i.e., $\phi_{\mathrm{A}}(\boldsymbol{a})$, and the action component in its coupling sample, i.e., $\phi_{\mathrm{A}}(\boldsymbol{c})$. This is because an action sample contains the actual action content while its coupling sample does not. $\mathcal{L}_{\mathrm{C}}$ is used to maximize the similarity between the co-occurrence component of an action sample, i.e., $\phi_{\mathrm{C}}(\boldsymbol{a})$, and the co-occurrence component of its coupling sample, i.e., $\phi_{\mathrm{C}}(\boldsymbol{c})$. This is because an action sample shares the similar co-occurrence component with its coupling sample. By minimizing these two losses over all training videos, the two encoders are learned to decouple the action component and the co-occurrence component of an arbitrary snippet.

3.5. Recombining Action and Co-occurrence Components

Till now, we have trained two encoders ϕ_{A} and ϕ_{C} to decouple the action component and the co-occurrence component of a snippet representation. However, only using the action component for TAL is suboptimal because the relevant contextual information included in the co-occurrence

component often serves as useful clues for action classification, *e.g.*, the tennis court of "Tennis Swing". This is also validated in our experiment. So it is necessary to effectively recombine them after explicit decoupling. Given a snippet representation $\mathbf{f}$, we could obtain $\phi_A(\mathbf{f})$ and $\phi_C(\mathbf{f})$ from the two encoders, which are then synthesized into a new feature representation, denoted as $\mathbf{f}'$:

$$\mathbf{f}' = \phi_A(\mathbf{f}) \oplus \phi_C(\mathbf{f}), \tag{3}$$

where $\oplus$ represents concatenation. Consequently, we can obtain the new video feature sequence $\mathbf{F}'$ via RefactorNet.

In addition, we expect that the recombined feature representation contains salient action signals. To avoid overwhelming co-occurrence features from degenerating the performance of action detectors, we adopt a regularization based on the KL divergence to encourage the co-occurrence component obtained from ϕ_C to be close to the normal distribution $\mathcal{N}(0, 1)$. Inspired by [31], we formulate the KL divergence loss as follows:

$$\mathcal{L}_{\mathrm{KL}} = \frac{1}{2} \sum_{i=1}^{D} \left(\mu_i^2 + \sigma_i^2 - \log\left(\sigma_i^2\right) - 1 \right), \tag{4}$$

where μ and σ represent the mean and standard deviation of a co-occurrence feature, and D is the feature dimension. The KL divergence loss regularizes co-occurrence features by limiting the distribution range. As a result, the network will suppress overwhelming context or background information.

Finally, the network will output the new snippet representation containing the salient action component, which leads to more effective video action detection.

Summary. Our RefactorNet aims to obtain an appropriate snippet representation with both a salient action component and a suitable co-occurrence component for TAL. We expect it to reduce the action ambiguity and also avoid over-relying on the co-occurrence component. This refactoring mechanism distinguishes the proposed approach from previous approaches for TAL.

3.6. Action Detection

After feature refactoring, we need to locate action instances based on the new video representation. We adopt a boundary-based proposal generator for candidate proposal generation. For each proposal, we extract its features from snippet features within it. Afterwards, we employ a multi-layer perceptron (MLP) as a localization head to refine the boundaries of each proposal, and two other MLPs as classification heads to predict the category and the completeness of the proposal. The product of the classification score and the completeness score is used as the confidence score of each proposal. Finally, we adopt Soft-NMS (soft non-maximum suppression) to suppress redundant proposals with high overlaps and obtain detection results.

4. Training and Inference

4.1. Training

During the training phase, we minimize the sum of the aforementioned losses for RefactorNet:

$$\mathcal{L}_{\mathcal{R}} = \alpha(\mathcal{L}_A + \mathcal{L}_C) + \beta\mathcal{L}_{\mathrm{KL}}, \tag{5}$$

where α and β are the weighting hyper-parameters.

Next, we apply a boundary-based proposal generator, *e.g.*, BSN [24], to the new video representation to produce a set of proposals. Especially, we define the following loss function for training the start and end action boundary predictors:

$$\mathcal{L}_{\mathcal{P}} = \mathcal{L}_{bl}^{\mathrm{s}} + \gamma\mathcal{L}_{bl}^{\mathrm{e}}, \tag{6}$$

where γ is a hyper-parameter. We adopt the binary logistic regression loss function as $\mathcal{L}_{bl}$. Then, we group probability peaks of starts and ends to produce candidate proposals. The features of each action proposal are extracted by RoI Pooling [11]. In addition, the overall loss function for training the proposal refinement network is as follows:

$$\mathcal{L}_{\mathcal{D}} = \mathcal{L}_{cls} + \lambda_1\mathcal{L}_{com} + \lambda_2\mathcal{L}_{reg}, \tag{7}$$

where λ_1 and λ_2 are hyper-parameters to trade-off these losses. We adopt the cross-entropy loss as $\mathcal{L}_{cls}$ for action classification, the hinge loss as $\mathcal{L}_{com}$ to predict the completeness score of each proposal, and the smooth-$\mathcal{L}_1$ loss as $\mathcal{L}_{reg}$ to predict the offset of center coordinate and duration for each proposal.

4.2. Inference

During the inference phase, given a feature vector $\mathbf{f}$ of a snippet in an untrimmed video, our RefactorNet takes it as input and outputs the new snippet feature vector $\mathbf{f}'$ with refactored action and co-occurrence components. Subsequently, we perform action boundary regression via the new snippet representations $\mathbf{F}'$ to find start and end locations with high responses and group them into action proposals. The localization head and classification heads produce the offsets, category score and completeness score for each proposal, respectively. We multiply the classification score and completeness score to get its confidence score. The final action localization result is retrieved by soft-NMS in the post-processing process.

5. Experiments

5.1. Datasets and Metrics

THUMOS14 [17] is a standard benchmark for TAL. It contains 200 validation videos and 213 testing videos, including 20 action categories. It is very challenging since each video has more than 15 action instances. Following the common

Model	Conference	THUMOS14 (%)						ActivityNet v1.3 (%)			
		0.3	0.4	0.5	0.6	0.7	Avg.	0.5	0.75	0.95	Avg.
TAL-Net [5]	CVPR'18	53.2	48.5	42.8	33.8	20.8	39.8	38.2	18.3	1.3	20.2
BSN [24]	ECCV'18	53.5	45.0	36.9	28.4	20.0	36.8	46.5	30.0	8.0	30.0
GTAN [30]	CVPR'19	57.8	47.2	38.8	—	—	—	52.6	34.1	8.9	34.3
MGG [27]	CVPR'19	53.9	46.8	37.4	29.5	21.3	37.8	—	—	—	—
P-GCN [51]	ICCV'19	63.6	57.8	49.1	—	—	—	42.9	28.1	2.5	27.0
BMN [23]	ICCV'19	56.0	47.4	38.8	29.7	20.5	38.5	50.1	34.8	8.3	33.9
G-TAD [46]	CVPR'20	54.5	47.6	40.2	30.8	23.4	39.3	50.4	34.6	9.0	34.1
BC-GNN [2]	ECCV'20	57.1	49.1	40.4	31.2	23.1	40.2	50.6	34.8	**9.4**	34.3
BU-MR [55]	ECCV'20	53.9	50.7	45.4	38.0	28.5	43.3	43.5	33.9	9.2	30.1
TCANet [33]	CVPR'21	60.6	53.2	44.6	36.8	26.7	44.4	54.3	39.1	8.4	37.6
MUSES [26]	CVPR'21	68.9	64.0	56.9	46.3	31.0	53.4	50.0	35.0	6.6	34.0
AFSD [22]	CVPR'21	67.3	62.4	55.5	43.7	31.1	52.0	52.4	35.3	6.5	34.4
ContextLoc [57]	ICCV'21	68.3	63.8	54.3	41.8	26.2	50.9	56.0	35.2	3.6	34.2
RTD-Net [38]	ICCV'21	68.3	62.3	51.9	38.8	23.7	49.0	47.2	30.7	8.6	30.8
VSGN [54]	ICCV'21	66.7	60.4	52.4	41.0	30.4	50.2	52.4	36.0	8.4	35.1
RefactorNet	CVPR'22	**70.7**	**65.4**	**58.6**	**47.0**	**32.1**	**54.8**	**56.6**	**40.7**	7.4	**38.6**

Table 1. Performance comparison on THUMOS14 and ActivityNet v1.3 in terms of mAP at different IoU thresholds. The "Avg" columns denote the average mAP in $[0.3 : 0.1 : 0.7]$ on THUMOS14 and $[0.5 : 0.05 : 0.95]$ on ActivityNet v1.3, respectively.

setting [52], we use the validation set for training and evaluate on the testing set.

ActivityNet v1.3 [3] is a large-scale benchmark for video-based action localization. It contains 10k training videos and 5k validation videos corresponding to 200 different actions. Following the standard practice [26], we train our method on the training set and test it on the validation set.

Evaluation Metrics. We use the mean Average Precision (mAP) as the evaluation metric. The tIoU thresholds are $[0.3 : 0.1 : 0.7]$ for THUMOS14 and $[0.5 : 0.05 : 0.95]$ for ActivityNet v1.3. We also report the average mAP of the IoU thresholds between 0.5 and 0.95 with the step of 0.05 on ActivityNet v1.3.

5.2. Implementation Details

We divide each input video into 16-frame snippets and exploit a two-stream I3D network [4] pre-trained on the Kinetics dataset [18] to extract original snippet features. For RefactorNet, we adopt the Adam [19] optimizer to train the network. Similar to the training scheme [8], we train RefactorNet in two stages. In the first stage, we train the two encoders ϕ_A and ϕ_C with the corresponding losses $\mathcal{L}_A$ and $\mathcal{L}_C$ for 30 epochs on the training data with a learning rate 0.001. In the second stage, we train the whole network end to end with the learning rate of 0.001 for 20 epochs. We use the similarity score between a pair of an action sample and a coupling sample as α and empirically set $\beta = 0.001$. For the action detector, we adopt BSN [24] on THUMOS14 and ActivityNet v1.3 to produce action proposals. For each proposal, a feature representation is extracted via RoI pool-

ing [34]. Then, we apply three different MLPs on each proposal for boundary regression, action classification and completeness prediction, as in [26, 51]. For $\mathcal{L}_D$, we set $\gamma = 1$ and $\lambda_1 = \lambda_2 = 0.5$, as in [24, 26]. We train the model for 20 epochs with an initial learning rate 0.001. For fair comparison, we combine our proposals with video-level classification results from [45] on ActivityNet v1.3.

5.3. Comparison with State-of-the-art Methods

THUMOS14. Our method is compared with state-of-the-art methods in Table 1. We report mAP at different tIoU thresholds as well as average mAP between 0.3 and 0.7 with the step of 0.1. Especially, the mAP of our method is 1.7% higher than that of MUSES [26] when tIoU=0.5. Our method also achieves significant improvement on the average mAP. It indicates the effectiveness of our method for accurate action localization.

ActivityNet v1.3. We also report the action localization results of various methods in Table 1. At tIoU 0.5, our method outperforms all the other methods. In addition, our improvement is particularly notable at tIoU=0.75, where we outperform the state-of-the-art methods by a margin of 1.6%. These experimental results further demonstrate the effectiveness of our method.

5.4. Ablation Study

In this section, we conduct comprehensive ablation studies in two folds. On the one hand, we verify the effects of main components within the proposed model. On the other hand, we verify the effectiveness of our method for

13879

Method	mAP@tIoU (%)					
	0.3	0.4	0.5	0.6	0.7	Avg.
Baseline	68.2	62.7	56.0	45.9	30.1	52.6
Deep Baseline	68.5	63.0	55.5	45.3	29.2	52.3
Baseline + RefactorNet	**70.7**	**65.4**	**58.6**	**47.0**	**32.1**	**54.8**

Table 2. Ablation study of the effect of feature refactoring on THUMOS14. The baseline is constructed by removing RefactorNet from our framework. The deep baseline is constructed by replacing RefactorNet in our framework with temporal 1D convolutional layers so that its model size and depth are the same as those of our framework.

Method	Class (%)	tIoU (%)
Baseline	76.7	79.6
Deep Baseline	77.1	78.4
Baseline + RefactorNet	**85.9**	**83.5**

Table 3. Ablation study of action classification accuracy (Class) and localization accuracy (tIoU) of high-quality proposals, whose tIoU w.r.t. ground truth are greater than 0.7, on THUMOS14.

temporal action localization and report both qualitative and quantitative experimental results and analyses.

Effect of Feature Refactoring. We construct a baseline by removing feature refactoring from our model. Table 2 shows our model outperforms the baseline by a large margin. For fair comparison, we further construct a deep baseline by adding more layers (*i.e.*, temporal 1D convolutional layers) to the baseline so that its model size and depth are the same as those of our model. Table 2 shows naively increasing the model size and network depth does not improve the baseline. This indicates that our performance improvement comes from effectively refactoring feature representations rather than simply adding more layers.

Analysis of Performance Improvements. In order to further explore the reasons behind performance improvements, we report the accuracy of action classification and the average tIoU of high-quality proposals in Table 3. High-quality proposals refer to action proposals after the regression whose tIoU w.r.t. ground truth is greater than 0.7. The experimental results show that the refactored snippet representations benefit both action classification and temporal boundary regression, compared with the baseline and deep baseline.

Loss Functions. To verify the effectiveness of our different loss functions, we conduct an ablation experiment by using different losses to train our model. In Table 4, we measure mAP and report results on THUMOS14. The first row demonstrates the performance of our baseline, which does not include our RefactorNet. The second row indicates that RefactorNet is guided by $\mathcal{L}_A$ and $\mathcal{L}_C$ to explicitly decouple the actual action from its co-occurrence components. The experimental results show that explicit feature refactoring can

$\mathcal{L}_A \& \mathcal{L}_C$	$\mathcal{L}_{KL}$	mAP@tIoU (%)					
		0.3	0.4	0.5	0.6	0.7	Avg.
		68.2	62.7	56.0	45.9	30.1	52.6
✓		69.6	64.2	57.8	46.1	31.4	53.8
✓	✓	**70.7**	**65.4**	**58.6**	**47.0**	**32.1**	**54.8**

Table 4. Ablation study of different losses on THUMOS14. $\mathcal{L}_A$ and $\mathcal{L}_C$ are used to decouple action and co-occurrence components and $\mathcal{L}_{KL}$ is used to regularize the co-occurrence component.
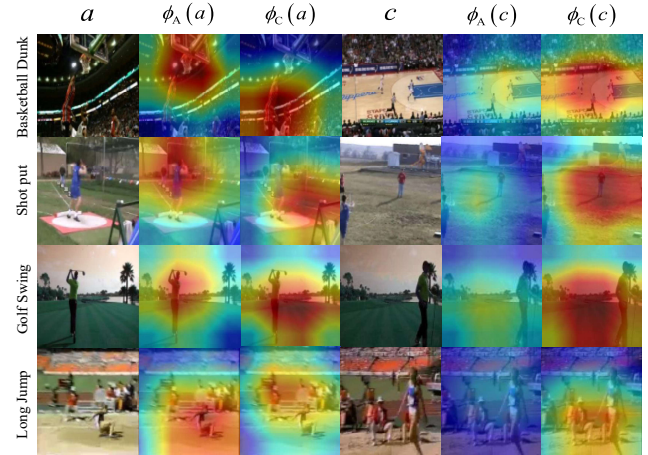

Figure 4. Visualization of the action content and the corresponding co-occurring component decoupled by our RefactorNet from action samples and coupling samples. The decoupled action and co-occurrence components respectively focus on the action and co-occurring visual elements.

significantly facilitate action localization. In addition, $\mathcal{L}_{KL}$ further regularizes the co-occurrence features by limiting their distribution range. The third row in Table 4 demonstrates that $\mathcal{L}_{KL}$ will enforce the network to use more action information when synthesizing the new video representation for action localization. With the combination of these loss functions, our RefactorNet is driven to combine the action component and the co-occurrence component within each snippet in a more balanced way. As a result, the action detector can achieve remarkable performance.

Visual Analysis for Decoupled Features. To validate the ability of our method to decouple the action component and the co-occurrence component, we conduct spatial visualization for the outputs of two encoders, *i.e.*, $\phi_A(a)$, $\phi_C(a)$, $\phi_A(c)$ and $\phi_C(c)$, by class activation maps (CAM [56]). As illustrated in Figure 4, the outputs of two encoders are visualized by heatmaps imposed on the original frames. It can be observed that the action representation obtained by the encoder ϕ_A focuses on the action area, *e.g.*, dunk of "Basketball Dunk" or swing of "Golf Swing", and the co-occurrence representation obtained by the encoder ϕ_C focuses on the background scene, *e.g.*, lawn or sand. As a result, our method
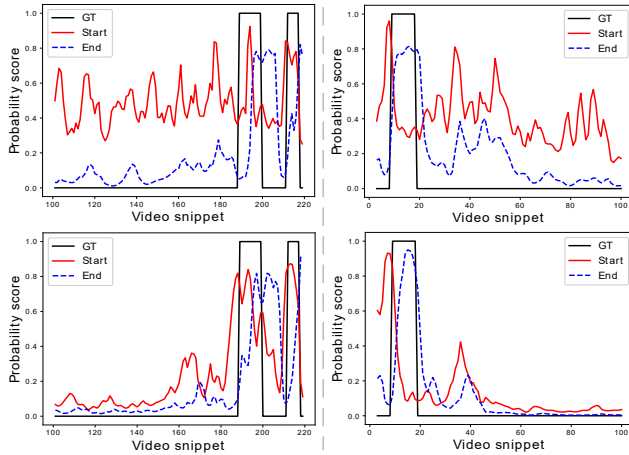
Figure 5. Comparison of temporal boundary predictions produced by the baseline (top) and our method (bottom). Particularly, the average precision of action boundary prediction of our method is 19.3%, while 13.4% for the baseline. It can be observed that our RefactorNet has better robustness and higher precision.

Figure 6. Qualitative detection results on ActivityNet v1.3 (top) and THUMOS14 (bottom).

| Modality | mAP@tIoU (%) | | | Class (%) |
	0.3	0.5	0.7	
RGB & Flow	68.2	56.0	30.1	76.7
Only Action	69.6	57.2	30.9	81.3
Action & Co-occ.	**70.7**	**58.6**	**32.1**	**85.9**

Table 5. Ablation study of using only the action component for TAL in terms of mAP at different IoU thresholds and the classification accuracy (Class) on THUMOS14. "RGB & Flow" represents the original video features. "Action" and "Co-occ." represent the action component and the co-occurrence component, respectively.

can effectively decouple action and co-occurrence components, and recombine them to facilitate action localization.

Visual Analysis for Temporal Boundary Prediction. Co-occurrence components often dominate action components in video snippets. This causes unreliable boundary predictions that degenerate detection performance. In order to verify whether our method can facilitate robust boundary prediction, we visualize some prediction results in Figure 5. Compared with the baseline, our method can help accurately predict boundary locations as well as reduce false alarms caused by the distraction of co-occurrence components. Visualization examples demonstrate that regulating co-occurrence components is an effective solution for robust temporal boundary prediction.

Visual Analysis for Temporal Action Localization. To verify whether the new video representation contains more favorable clues for action localization, we report some qualitative results in Figure 6. We can see that our method can help the model effectively learn indicative information to alleviate the misclassification caused by over-relying on the co-occurrence component. In addition, refactoring action and co-occurrence components can provide salient action features to reduce the action ambiguity and uncertainty.

Analysis of the Action Component. TAL not only needs to locate actions but also accurately predict their categories. Table 5 indicates that only using the action component for TAL is suboptimal because the co-occurrence component often contains contextual information useful to action classification, and helps reduce action ambiguity and uncertainty.
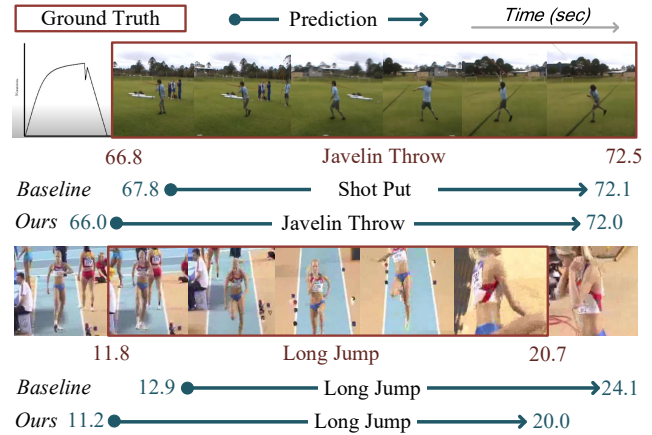
6. Conclusion

In this paper, we rethink and explore refactoring action and co-occurrence components of snippet features for TAL. We present a novel feature refactoring network, RefactorNet, inserted between the video feature extractor and the action detector. It aims at decoupling the snippet representation into the actual action content and its co-occurrence component, which are then recombined into a new snippet representation with salient action component and suitable co-occurrence component. Quantitative and qualitative experiments verify the effectiveness of the proposed method. As a result, the combination of RefactorNet with action detectors achieves great performance on THUMOS14 and ActivityNet v1.3.

7. Acknowledgment

References

[1] Hyojin Bahng, Sanghyuk Chun, Sangdoo Yun, Jaegul Choo, and Seong Joon Oh. Learning de-biased representations with biased representations. In *ICML*, pages 528–539, 2020. 3

[2] Yueran Bai, Yingying Wang, Yunhai Tong, Yang Yang, Qiyue Liu, and Junhui Liu. Boundary content graph neural network for temporal action proposal generation. In *ECCV*, pages 121–137, 2020. 6

[3] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, pages 961–970, 2015. 6

[4] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, pages 6299–6308, 2017. 6

[5] 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*, pages 1130–1139, 2018. 6

[6] Jinwoo Choi, Chen Gao, Joseph CE Messou, and Jia-Bin Huang. Why can't i dance in a mall? learning to mitigate scene bias in action recognition. In *NeurIPS*, pages 853–865, 2019. 3

[7] Emily L Denton and Vighnesh Birodkar. Unsupervised learning of disentangled representations from video. In *NIPS*, pages 4417–4426, 2017. 3

[8] Chanho Eom and Bumsub Ham. Learning disentangled representation for robust person re-identification. In *NeurIPS*, pages 5297–5308, 2019. 3, 6

[9] Lijie Fan, Wenbing Huang, Chuang Gan, Stefano Ermon, Boqing Gong, and Junzhou Huang. End-to-end learning of motion representation for video understanding. In *CVPR*, pages 6016–6025, 2018. 1

[10] Jialin Gao, Zhixiang Shi, Guanshuo Wang, Jiani Li, Yufeng Yuan, Shiming Ge, and Xi Zhou. Accurate temporal action proposal generation with relation-aware pyramid network. In *AAAI*, pages 10810–10817, 2020. 2

[11] Ross Girshick. Fast r-cnn. In *ICCV*, pages 1440–1448, 2015. 5

[12] Guoqiang Gong, Liangfeng Zheng, and Yadong Mu. Scale matters: Temporal scale aggregation network for precise action localization in untrimmed videos. In *ICME*, pages 1–6, 2020. 1

[13] Ryuhei Hamaguchi, Ken Sakurada, and Ryosuke Nakamura. Rare event detection using disentangled representation learning. In *CVPR*, pages 9327–9335, 2019. 3

[14] Jun-Ting Hsieh, Bingbin Liu, De-An Huang, Fei-Fei Li, and Juan Carlos Niebles. Learning to decompose and disentangle representations for video prediction. In *NeurIPS*, pages 515–524, 2018. 3

[15] Lianghua Huang, Yu Liu, Bin Wang, Pan Pan, Yinghui Xu, and Rong Jin. Self-supervised video representation learning by context and motion decoupling. In *CVPR*, pages 13886–13895, 2021. 3

[16] Mihir Jain, Amir Ghodrati, and Cees GM Snoek. Actionbytes: Learning from trimmed videos to localize actions. In *CVPR*, pages 1171–1180, 2020. 2

[17] Yu-Gang Jiang, Jingen Liu, A Roshan Zamir, George Toderici, Ivan Laptev, Mubarak Shah, and Rahul Sukthankar. Thumos challenge: Action recognition with a large number of classes, 2014. 5

[18] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017. 6

[19] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[20] Pilhyeon Lee, Youngjung Uh, and Hyeran Byun. Background suppression network for weakly-supervised temporal action localization. In *AAAI*, pages 11320–11327, 2020. 3

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

[22] Chuming Lin, Chengming Xu, Donghao Luo, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yanwei Fu. Learning salient boundary feature for anchor-free temporal action localization. In *CVPR*, pages 3320–3329, 2021. 2, 6

[23] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. Bmn: Boundary-matching network for temporal action proposal generation. In *ICCV*, pages 3889–3898, 2019. 6

[24] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. Bsn: Boundary sensitive network for temporal action proposal generation. In *ECCV*, pages 3–19, 2018. 2, 5, 6

[25] Shilong Liu, Lei Zhang, Xiao Yang, Hang Su, and Jun Zhu. Unsupervised part segmentation through disentangling appearance and shape. In *CVPR*, pages 8355–8364, 2021. 3

[26] Xiaolong Liu, Yao Hu, Song Bai, Fei Ding, Xiang Bai, and Philip HS Torr. Multi-shot temporal event localization: a benchmark. In *CVPR*, pages 12596–12606, 2021. 6

[27] Yuan Liu, Lin Ma, Yifeng Zhang, Wei Liu, and Shih-Fu Chang. Multi-granularity generator for temporal action proposal. In *CVPR*, pages 3604–3613, 2019. 6

[28] Yu Liu, Fangyin Wei, Jing Shao, Lu Sheng, Junjie Yan, and Xiaogang Wang. Exploring disentangled feature representation beyond face identification. In *CVPR*, pages 2080–2089, 2018. 3

[29] Ziyi Liu, Le Wang, Wei Tang, Junsong Yuan, Nanning Zheng, and Gang Hua. Weakly supervised temporal action localization through learning explicit subspaces for action and context. In *AAAI*, pages 2242–2250, 2021. 3

[30] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. Gaussian temporal awareness networks for action localization. In *CVPR*, pages 344–353, 2019. 2, 6

[31] Boyu Lu, Jun-Cheng Chen, and Rama Chellappa. Unsupervised domain-specific deblurring via disentangled representations. In *CVPR*, pages 10225–10234, 2019. 3, 5

[32] Tian Pan, Yibing Song, Tianyu Yang, Wenhao Jiang, and Wei Liu. Videomoco: Contrastive video representation learning with temporally adversarial examples. In *CVPR*, pages 11205–11214, 2021. 3

13882

[33] Zhiwu Qing, Haisheng Su, Weihao Gan, Dongliang Wang, Wei Wu, Xiang Wang, Yu Qiao, Junjie Yan, Changxin Gao, and Nong Sang. Temporal context aggregation network for temporal action proposal refinement. In *CVPR*, pages 485–494, 2021. 1, 2, 6

[34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *T-PAMI*, pages 1137–1149, 2017. 6

[35] 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*, pages 5734–5743, 2017. 2

[36] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, pages 568–576, 2014. 1

[37] Krishna Kumar Singh, Dhruv Mahajan, Kristen Grauman, Yong Jae Lee, Matt Feiszli, and Deepti Ghadiyaram. Don't judge an object by its context: Learning to overcome contextual bias. In *CVPR*, pages 11070–11078, 2020. 3

[38] Jing Tan, Jiaqi Tang, Limin Wang, and Gangshan Wu. Relaxed transformer decoders for direct action proposal generation. *ICCV*, 2021. 6

[39] Praveen Tirupattur, Kevin Duarte, Yogesh S Rawat, and Mubarak Shah. Modeling multi-label action dependencies for temporal action localization. In *CVPR*, pages 1460–1470, 2021. 1

[40] Ruben Villegas, Jimei Yang, Seunghoon Hong, Xunyu Lin, and Honglak Lee. Decomposing motion and content for natural video sequence prediction. *ICLR*, 2017. 3

[41] Jinpeng Wang, Yuting Gao, Ke Li, Xinyang Jiang, Xiaowei Guo, Rongrong Ji, and Xing Sun. Enhancing unsupervised video representation learning by decoupling the scene and the motion. In *AAAI*, pages 10129–10137, 2021. 3

[42] Yang Wang and Minh Hoai. Pulling actions out of context: Explicit separation for effective combination. In *CVPR*, pages 7044–7053, 2018. 3

[43] Michael Wray, Hazel Doughty, and Dima Damen. On semantic similarity in video retrieval. In *CVPR*, pages 3650–3660, 2021. 1

[44] Xiang Wu, Huaibo Huang, Vishal M Patel, Ran He, and Zhenan Sun. Disentangled variational representation for heterogeneous face recognition. In *AAAI*, pages 9005–9012, 2019. 3

[45] Yuanjun Xiong, Limin Wang, Zhe Wang, Bowen Zhang, Hang Song, Wei Li, Dahua Lin, Yu Qiao, Luc Van Gool, and Xiaoou Tang. Cuhk & ethz & siat submission to activitynet challenge 2016. *arXiv preprint arXiv:1608.00797*, 2016. 6

[46] Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-tad: Sub-graph localization for temporal action detection. In *CVPR*, pages 10156–10165, 2020. 1, 2, 6

[47] Ke Yang, Peng Qiao, Dongsheng Li, Shaohe Lv, and Yong Dou. Exploring temporal preservation networks for precise temporal action localization. In *AAAI*, 2018. 2

[48] Le Yang, Junwei Han, Tao Zhao, Tianwei Lin, Dingwen Zhang, and Jianxin Chen. Background-click supervision for temporal action localization. *T-PAMI*, 2021. 2

[49] Li Yingzhen and Stephan Mandt. Disentangled sequential autoencoder. In *ICML*, pages 5670–5679, 2018. 3

[50] Zehuan Yuan, Jonathan C Stroud, Tong Lu, and Jia Deng. Temporal action localization by structured maximal sums. In *CVPR*, pages 3684–3692, 2017. 1

[51] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph convolutional networks for temporal action localization. In *ICCV*, pages 7094–7103, 2019. 2, 6

[52] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph convolutional module for temporal action localization in videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 2021. 1, 6

[53] Yuanhao Zhai, Le Wang, Wei Tang, Qilin Zhang, Junsong Yuan, and Gang Hua. Two-stream consensus network for weakly-supervised temporal action localization. In *ECCV*, 2020. 1

[54] Chen Zhao, Ali K Thabet, and Bernard Ghanem. Video self-stitching graph network for temporal action localization. In *ICCV*, pages 13658–13667, 2021. 6

[55] Peisen Zhao, Lingxi Xie, Chen Ju, Ya Zhang, Yanfeng Wang, and Qi Tian. Bottom-up temporal action localization with mutual regularization. In *ECCV*, pages 539–555, 2020. 2, 6

[56] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016. 7

[57] Zixin Zhu, Wei Tang, Le Wang, Nanning Zheng, and Gang Hua. Enriching local and global contexts for temporal action localization. In *ICCV*, pages 13516–13525, 2021. 2, 6