Action Completeness Modeling with Background Aware Networks for Weakly-Supervised Temporal Action Localization

Md Moniruzzaman
Stony Brook University
Stony Brook, NY, USA
mmoniruzzama@cs.stonybrook.edu

Zhaozheng Yin
Stony Brook University
Stony Brook, NY, USA
zyin@cs.stonybrook.edu

Zhihai He
University of Missouri
Columbia, MO, USA
hezhi@missouri.edu

Ruwen Qin
Stony Brook University
Stony Brook, NY, USA
ruwen.qin@stonybrook.edu

Ming C Leu
Missouri University of Science and Technology
Rolla, MO, USA
mleu@mst.edu

ABSTRACT

The state-of-the-art of fully-supervised methods for temporal action localization from untrimmed videos has achieved impressive results. Yet, it remains unsatisfactory for the weakly-supervised temporal action localization, where only video-level action labels are given without the timestamp annotation on when the actions occur. The main reason comes from that, the weakly-supervised networks only focus on the highly discriminative frames, but there are some ambiguous frames in both background and action classes. The ambiguous frames in background class are very similar to the real actions, which may be treated as target actions and result in false positives. On the other hand, the ambiguous frames in action class which possibly contain action instances, are prone to be false negatives by the weakly-supervised networks and result in a coarse localization. To solve these problems, we introduce a novel weakly-supervised Action Completeness Modeling with Background Aware Networks (ACM-BANets). Our Background Aware Network (BANet) contains a weight-sharing two-branch architecture, with an action guided Background aware Temporal Attention Module (B-TAM) and an asymmetrical training strategy, to suppress both highly discriminative and ambiguous background frames to remove the false positives. Our action completeness modeling contains multiple BANets, and the BANets are forced to discover different but complementary action instances to completely localize the action instances in both highly discriminative and ambiguous action frames. In the i-th iteration, the i-th BANet discovers the discriminative features, which are then erased from the feature map. The partially-erased feature map is fed into the $(i+1)$-th BANet of the next iteration to force this BANet to discover discriminative features different from the i-th BANet. Evaluated on two challenging untrimmed video datasets, THUMOS14 and ActivityNet1.3, our approach outperforms all the current weakly-supervised methods for temporal action localization.

MM '20, October 12–16, 2020, Seattle, WA, USA

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3413687

CCS CONCEPTS

• **Understanding multimedia content** → Media interpretation.

KEYWORDS

Temporal action localization, Weakly-supervised learning, Background aware networks, Action completeness modeling

ACM Reference Format:
Md Moniruzzaman, Zhaozheng Yin, Zhihai He, Ruwen Qin, and Ming C Leu. 2020. Action Completeness Modeling with Background Aware Networks for Weakly-Supervised Temporal Action Localization. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3413687

1 INTRODUCTION

In the past two decades, impressive progress [5, 11, 29, 32, 34] has been reported on human action recognition from manually *trimmed videos* (i.e., the videos do not contain unrelated frames) such as those from the datasets of HMDB-51 [14] and UCF-101 [31]. But, the methods for recognizing actions from manually trimmed videos are unrealistic in real-world scenarios, because the daily videos are usually *untrimmed video* streams (i.e., the videos contain one or multiple action instances from one or multiple action classes with many unrelated frames) such as the videos from the datasets of THUMOS14 [10] and ActivityNet [1]. Therefore, besides recognizing actions in temporally-trimmed videos, the research community pays a significant amount of attention to develop techniques for *temporal action localization* in untrimmed videos [3, 4, 26, 28, 36, 40], where the task is to not only recognize the action classes, but also localize the temporal window (the start time and end time) of each action instance in the untrimmed video.

Most of the previous temporal action localization algorithms are fully-supervised, which require the ground truth video-level action labels along with the detailed temporal annotations for each action instance within the training videos. Acquiring such detailed annotations for large-scale datasets is expensive and requires tremendous time. In practice, it is much easier to collect untrimmed videos with

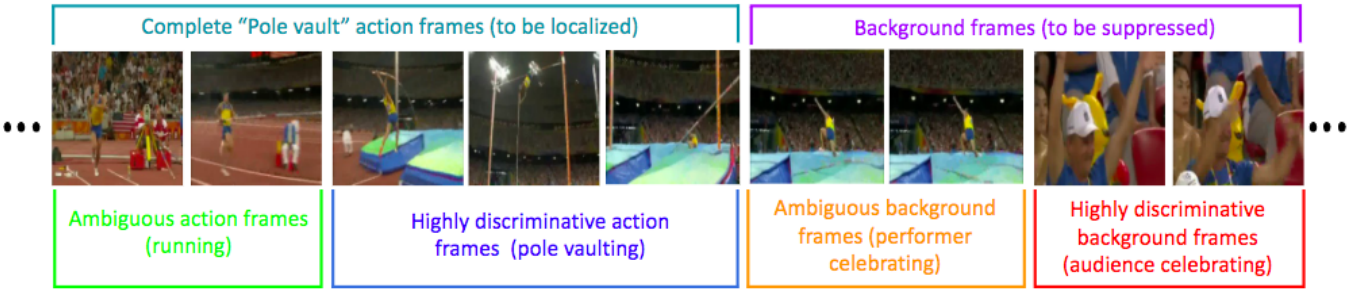

Figure 1: An illustration of highly discriminative and ambiguous action/background frames: (i) Highly discriminative action frames: highly related to the target action class (e.g., pole vaulting frames for the pole vault action), (ii) Highly discriminative background frames: not related to the target action (e.g., audience celebrating frames), (iii) ambiguous action frames: possible to contain action instances (e.g., running frames for the pole vault action), and (iv) ambiguous background frames: nearby frames that do not belong to the target action (e.g., performer celebrating frames after the pole vault action). Usually, the weakly-supervised networks can easily localize and suppress the highly discriminative action and background frames, respectively, while it is hard to localize the ambiguous action frames, and suppress the ambiguous background frames. If we can localize the action instances in both highly discriminative and ambiguous action frames, and suppress both highly discriminative and ambiguous background frames, we can improve the accuracy of temporal action localization.

weak labels (i.e., video-level action class labels only). Recently, several neural networks [21, 23, 25, 27, 33] were developed to localize action instances in untrimmed videos using the video-level labels.

Challenges and motivation: Despite the recent advance, the weakly-supervised temporal action localization task is still challenging from a few aspects:

(1) As in weakly-supervised settings, the fine-grained temporal annotations are not available, the temporal boundary of action instances are usually localized from the intermediate layers of action classification networks, which may obtain a good performance by highlighting the *highly discriminative action frames* and suppressing the *highly discriminative background frames*. But, there are some *ambiguous background frames* around but not belonging to the target actions, as shown in Figure 1. It is hard to distinguish these ambiguous background frames with such weakly-supervised algorithms, which results in false positives. Therefore, the motivated research question is: *how to design and train a weakly-supervised network that can suppress both highly discriminative and ambiguous background frames to remove the false positives, eventually leading to improve the performance of temporal action localization?*

(2) Another challenging aspect comes from that, the networks tend to focus on the highly discriminative action frames, but some *ambiguous action frames* which are possible to contain action instances, are not localized by the weakly-supervised networks. Rather than localizing the action instances in only highly discriminative action frames, the action instances are supposed to be localized in both highly discriminative and ambiguous action frames. This challenge arises another research question: *given only video-level labels, how to design a temporal action localization framework to discover action instances in both highly discriminative and ambiguous action frames for localizing the complete action instances?*

Our proposal and contribution: Motivated by the above challenges, we propose a novel Action Completeness Modeling with Background Aware Networks (ACM-BANets) to localize the human actions in untrimmed videos. Our main contribution has three folds:

- We design a novel Background Aware Network (BANet) to suppress both highly discriminative and ambiguous background frames to significantly reduce the false positive rate.
- We propose an action completeness modeling framework that contains multiple BANets, where the BANets are forced to localize different but complementary action instances in both highly discriminative and ambiguous action frames.
- Our weakly-supervised ACM-BANets outperforms all the latest weakly-supervised temporal action localization methods on the challenging THUMOS14 and ActivityNet1.3 datasets.

2 RELATED WORKS

Deep learning for action recognition: With the recent availability of big data and powerful GPUs, after the breakthrough in image classification [13] with Convolutional Neural Networks (CNN), video-based human action recognition has achieved significant progresses recently. CNN-based models for human action recognition broadly follow three main directions: *(1) Multi-stream networks* [29, 34]: CNNs are trained on multiple input modalities (e.g., optical flow and warped flow in addition to RGB). Given a test video, the predictions from all CNNs are fused to get the final video-level prediction; *(2) 3D CNN* [2, 32]: taking short video clips as inputs, 3D convolution and 3D pooling are performed to extract spatio-temporal feature maps; and (3) *CNN + LSTM (Long Short Term Memory)* [5]: recurrent neural networks are built on top of CNN features to capture the long term dynamics for action recognition.

Fully-supervised temporal action localization: Temporal action localization task identifies the start time and end time as well as the action label for each action instance in the untrimmed video. Some of the previous works conducted temporal sliding windows over the input video, which were followed by a classification network to classify the action within each window [38]. Recently, motivated by the success of region-based CNN [8, 24] in object detection, several recent works [26, 28, 40] addressed

the temporal action localization problem by adopting a two-stage framework: the first stage generated segment proposals as either action or background, and the second stage classified the action proposals to the corresponding action classes. More recently, several works [3, 4, 16, 17, 36] developed trainable proposal architecture and Gaussian temporal modeling [19] to localize the temporal boundary for the target action.

Weakly-supervised temporal action localization: Recently, several works tried to adapt weakly-supervised learning methods into the temporal action localization task. A weakly-supervised action detection and recognition technique called UntrimmedNet was introduced by [33], which did not use temporal annotations during training. Nguyen et al. [21] introduced a sparse temporal pooling network for weakly-supervised action localization that employed sparsity loss. AutoLoc [27] introduced a contrastive loss function based on the Class Activation Sequence (CAS) for weakly-supervised temporal action localization. Paul et al. [23] utilized pairwise video similarity constraints to localize the target action in the temporal domain. Narayan et al. [20] utilized counting loss and center loss in addition to the classification loss for weakly-supervised temporal action localization. None of these weakly-supervised algorithms attempted to model background frames during training. More recently, some works [15, 22] tried to model background frames during training by introducing weight-sharing architectures. Although, these algorithms successfully model the highly discriminative action and background frames, they suffer from modeling ambiguous action and background frames, which result in false positives and incomplete localization, respectively.

Completeness modeling strategy: Several works [9, 35, 39] proposed adversarial erasing strategy to model the completeness of objects in object detection task. In addition to object detection, some works tried to localize complete action instances in untrimmed videos. Hide-and-Seek [30] hid random frame sequences to force the network to look at different relevant parts for the temporal action localization task. Recently, multi-branch networks [18] were used to localize action instances by inserting a diversity loss, which forced each branch to focus on different action parts.

Differently, in this paper, we propose a novel weakly-supervised temporal action localization framework that suppresses both highly discriminative and ambiguous background frames to remove the false positives, and localizes the action instances in both highly discriminative and ambiguous action frames to completely localize the action instances.

3 METHODOLOGY

In this section, we first present the feature extraction mechanism in Sec. 3.1, then our proposed Background Aware Network (BANet) in Sec. 3.2. Thereafter, we present our action completeness modeling mechanism, illustrated in Sec. 3.3. Finally, we present how to perform the temporal action localization during testing in Sec. 3.4.

3.1 Feature Extraction

In our weakly-supervised setting, we only have the information about the action labels for the entire untrimmed video. But, the untrimmed videos may contain many unrelated frames and the action instances may occur in different time instants of a video. As the action can be recognized from untrimmed videos by identifying a set of discriminative frames, we divide the video into short video segments. Formally, for a given video V, we conduct a temporal sliding window of I frames to generate video segments with the size of $I \times h_1 \times h_2 \times r$, where h_1, h_2, and r are the height, width, and number of color channels of each frame, respectively.

After generating the video segments, we use the I3D network [2] pretrained on the Kinetics dataset to extract features from every video segment. Formally, for a given video V with a set of video segments $S = \{s_t\}_{t=1}^{T}$, where T is the number of segments obtained from a video, we extract the feature as $F_s \in \mathbb{R}^D$ for each segment s, where D is the dimension of the feature representation. At the end of the feature extraction procedure, for each input video consisting of T segments, we obtain a feature map $X \in \mathbb{R}^{T \times D}$, which provides a high-level representation of the appearance and motion of the input video and is fed into the following layers in the network.

3.2 Background Aware Network (BANet)

Most of the weakly-supervised temporal action localization algorithms are capable of suppressing the highly discriminative background frames, but not ambiguous background frames. To suppress both highly discriminative and ambiguous background frames, we introduce a Background Aware Network (BANet) that contains two branches: base branch, and attention branch, as shown in Figure 2. Both branches take the same feature map as input and also share the weights of the convolutional layers to produce the Class Activation Sequences (CAS) with three main differences:

(1) The attention branch contains an action-guided Background aware Temporal Attention Module (B-TAM) in its front to suppress highly discriminative and ambiguous background frames.
(2) The attention branch contains a self-attention weighted top-K mean at the end to select the highly discriminative video segments to predict the video-level scores, while the base branch directly performs the top-K mean for the video-level prediction.
(3) The training objectives of these two branches are different. Since every untrimmed video naturally contains some background frames that do not belong to any action classes, we consider an additional background class in addition to the action classes. The convolutional layers of the base branch directly take the feature map as input, which contains the feature representations for both action and background frames. Therefore, the training objective for the video-level prediction of this branch is set to be positive for the target action classes as well as for the additional background class. On the other hand, since the convolutional layers of the attention branch take the feature map as input in which the background frames are suppressed by B-TAM, the training objective for this branch is set to be positive for the target action classes and zero for the background class. The weight-sharing strategy with these contrasting objectives for the background class ensures that the shared parameters learn to distinguish actions from the background.

3.2.1 Base branch: The base branch loads the feature map X into temporal 1D convoluational layers and predicts segment-level classification scores by generating Class Activation Sequences (CAS), where the segments have their class scores:

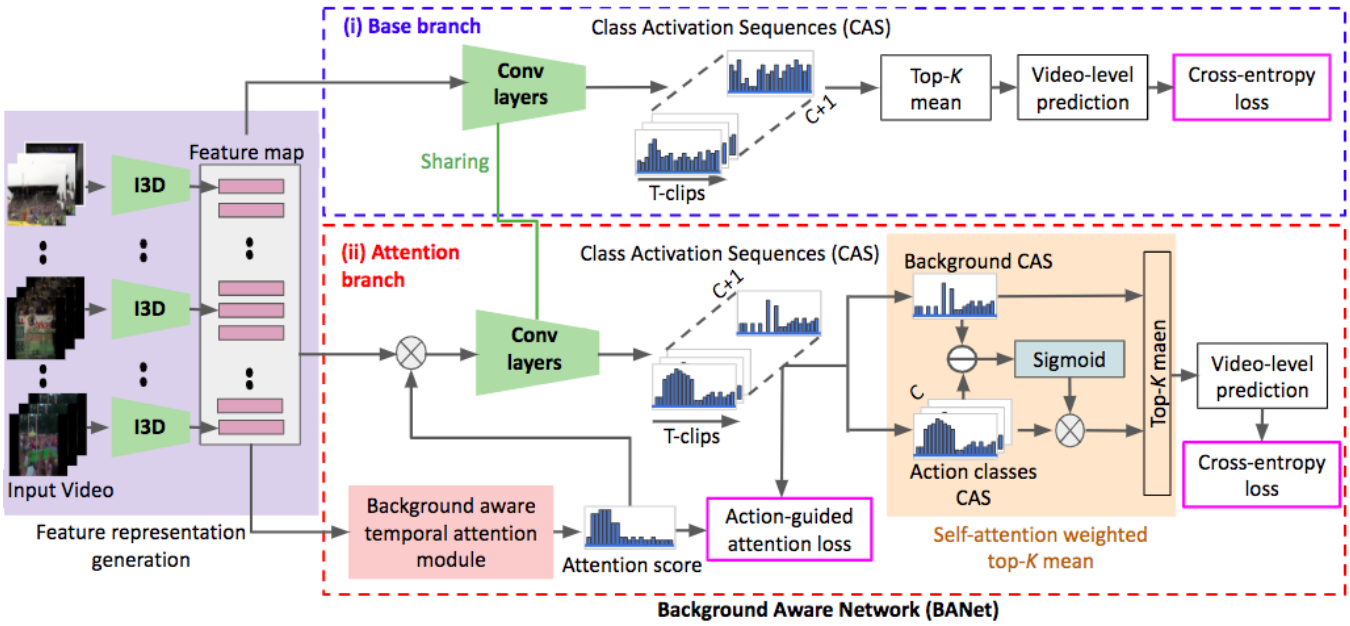

Figure 2: The architecture of our proposed Background Aware Network (BANet). Using a pre-trained network, we extract the feature representation for short video segments, which are then fed into BANet. BANet contains weight-sharing asymmetrical two-branch architecture: (i) base branch, and (ii) attention branch. The training objective of the base branch is set to be positive for the target action classes as well as for the background class. Differently, the attention branch contains an action-guided Background aware Temporal Attention Module (shown in Figure 3) in its front and a self-attention weighted top-K mean at the end, where the training objective is set to be positive for the target action classes and zero for the background class.

$$\mathbf{P}^{base} = f_{conv}(X;\theta) \tag{1}$$

where $\mathbf{P}^{base} \in \mathbb{R}^{T\times(C+1)}$ is the CAS of base branch and θ denotes trainable parameters in the convolutional layers. The first C columns in $\mathbf{P}^{base}$ represent the scores for the C action classes and the last column is the score for the background class.

Since we have the ground truth for the video as a whole, we aggregate the segment-level classification scores, using class-wise top-K mean technique [23], to generate the classification score for the entire video regarding to each action class, $\mathbf{p}^{base} \in \mathbb{R}^{C+1}$. In other words, denoting the video-level class score for class c as p_c^{base}, we have the video-level scores for all classes as $\mathbf{p}^{base} = ([p_1^{base}, ..., p_c^{base}, ..., p_{C+1}^{base}])$. After applying the softmax on $\mathbf{p}^{base}$, we get the normalized classification score vector: $\tilde{\mathbf{p}}^{base} \in [0,1]^{C+1}$.

The classification loss of the base branch is defined by the cross-entropy loss:

$$L_{base} = \frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C+1} -y_{c,n}^{base}\log\tilde{p}_{c,n}^{base} \tag{2}$$

where $\tilde{p}_{c,n}^{base}$ is the classification score of the base branch on the n-th training video regarding to class c (please excuse us to abuse the math notation by adding a subscript n to denote the n-th training video), and $\mathbf{y}_n^{base} = [y_{1,n}^{base}, ...y_{C,n}^{base}, 1]$ is the video-level label for this video, in which $y_{c,n}^{base}$ is set to 1 if this video contains the action class c (note that one video may contains multiple action classes). One additional positive label for the background class is set at the end of the label, considering that all untrimmed videos in training dataset contain background frames. N is the number of training videos.

3.2.2 Attention branch: Different from the base branch, the attention branch in BANet contains an action-guided Background aware Temporal Attention Module (B-TAM) in its front and a self-attention weighted top-K mean at the end, where the training objective for the video-level prediction is set to be positive for the target action classes and zero for the background class.

We design B-TAM to suppress both highly discriminative and ambiguous background frames. Figure 3 shows the block diagram of our proposed B-TAM. B-TAM loads the feature map $X \in \mathbb{R}^{T\times D}$ into a temporal attention module to compute the attention score vector $\mathbf{w}_{att} \in [0,1]^T$. The temporal attention module consists of two convolutional layers and a LeakyReLU layer located between the two convolutional layers. The output of the second convolutional layer is passed through a sigmoid function that enforces the attention scores to be between 0 and 1. Usually, the output of the temporal attention module $\mathbf{w}_{att}$ provides low scores (close to 0) for the highly discriminative background frames and high scores (close to 1) for the highly discriminative action frames. But, it may also show a significant amount of high attention scores for the ambiguous background frames, yielding false positives. To overcome these false positives, at the training time, we calculate the high attention score vector $\mathbf{w}_{att}^H \in \mathbb{R}^T$ by zeroing out all the elements of $\mathbf{w}_{att}$

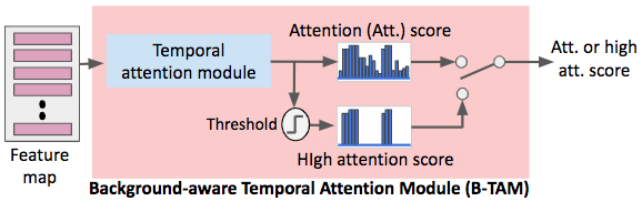

Figure 3: The block diagram of our proposed Background-aware Temporal Attention Module (B-TAM).

whose score values are less than a threshold α. Therefore, $\mathbf{w}_{att}^{H}$ only contains the attention scores for the highly discriminative video segments, which encourages the model to look at highly discriminative action frames. However, if we persistently use $\mathbf{w}_{att}^{H}$ at every training step, some ambiguous action frames may be overlooked, which affects the complete action localization. Therefore, the output of B-TAM, $\mathbf{w}^{R}$, randomly selects either $\mathbf{w}_{att}$ or $\mathbf{w}_{att}^{H}$ with equal chances at every training step. Then, the element-wise multiplication is performed between the selected attention score vector and the input feature map $X \in \mathbb{R}^{T\times D}$ to get the attention-weighted feature map $X_{att} \in \mathbb{R}^{T\times D}$ for the remaining layers. During the testing phase, we directly use $\mathbf{w}_{att}$ as the output of B-TAM.

The attention-weighted feature map X_{att} is fed into 1D temporal convolutional layers, which share the weights with the base branch. The CAS of attention branch is computes as:

$$\mathbf{P}^{att} = f_{conv}(X_{att};\theta), \quad \text{where} \quad \mathbf{P}^{att} \in \mathbb{R}^{T\times(C+1)} \tag{3}$$

Now, we aim to compute the classification score of the entire video for the attention branch. Different from the base branch, we use self-attention weighted top-K mean, which aims to select top-K scores of the action classes based on their differences with the background class. To compute this, first, we compute the self-attention scores from the difference between the action classes and the background class for each video segment:

$$W_{t,c}^{self-att} = \sigma(P_{t,c}^{att} - P_{t,C+1}^{att}), \;\; t = 1,...,T \text{ and } c = 1,...,C \tag{4}$$

where $W_{t,c}^{self-att}$ is the self-attention score for segment t regarding to action class c, $P_{t,c}^{att}$ and $P_{t,C+1}^{att}$ represents the class activation score for segment t regarding to action class c and the background class, respectively, and $\sigma(.)$ is the sigmoid operation. Then, we compute the self-attention weighted scores for the action classes as:

$$A_{t,c} = P_{t,c}^{att} W_{t,c}^{self-att}, \;\; t = 1,...,T \text{ and } c = 1,...,C \tag{5}$$

where $\mathbf{A} = [A_{t,c}] \in \mathbb{R}^{T\times C}$ is the self-attention weighted CAS for the action classes, in which $A_{t,c}$ represents the self-attention weighted score for segment t regarding to action class c. We concatenate $\mathbf{A} \in \mathbb{R}^{T\times C}$ with the CAS of the background class $\mathbf{P}_{C+1}^{att} \in \mathbb{R}^{T}$ (note that we consider the last column in $\mathbf{P}^{att} \in \mathbb{R}^{T\times(C+1)}$ as the CAS of the background class) to obtain the concatenated CAS as $[\mathbf{A}\, \mathbf{P}_{C+1}^{att}]$, then we apply top-$K$ mean on it to get the video-level classification scores, $\mathbf{p}^{att} \in \mathbb{R}^{C+1}$. After applying the softmax on $\mathbf{p}^{att}$, we get the normalized classification score vector: $\tilde{\mathbf{p}}^{att} \in [0,1]^{C+1}$.

The classification loss of the attention branch is defined by cross-entropy loss:

$$L_{att} = \frac{1}{N}\sum_{n=1}^{N}\sum_{c=1}^{C+1} -y_{c,n}^{att} \log \tilde{p}_{c,n}^{att} \tag{6}$$

where $\tilde{p}_{c,n}^{att}$ is the classification score of the attention branch for class c regarding to the n-th training video (please excuse us to abuse the math notation by adding a subscript n to denote the n-th training video), and $\mathbf{y}_n^{att} = [y_{1,n}^{att}, ...y_{C,n}^{att}, 0]$ is the video-level label for this video, in which $y_{c,n}^{att}$ is set to 1 if this video contains action class c. Since the attention branch suppresses the background frames, the label for the background class is set to 0.

3.2.3 Optimization. The loss function in the proposed BANet is composed of three terms:

$$L_{Total} = L_{base} + L_{att} + \gamma L_{AAL} \tag{7}$$

where L_{base} and L_{att} have the same coefficients, and γ is the hyper-parameter to control the corresponding weights between the losses. Since B-TAM generates $\mathbf{w}^{R}$ ($\mathbf{w}_{att}$ or $\mathbf{w}_{att}^{H}$) without considering the specific action class information, it more likely responds to generic cues, which may not be specific to the target action classes. Therefore, we introduce an Action-guided Attention Loss L_{AAL} to refine $\mathbf{w}^{R}$ by the scores of the ground truth action classes.

To compute L_{AAL}, first we apply softmax operation on CAS of the attention branch $\mathbf{P}^{att} \in \mathbb{R}^{T\times(C+1)}$ along the class dimension to achieve normalized class activation sequences $\tilde{\mathbf{P}}^{att} \in \mathbb{R}^{T\times(C+1)}$. Then, we obtain the maximum class activation scores of the ground truth action classes from $\tilde{\mathbf{P}}^{att}$ for each video segment:

$$p_t^* = \max_{y_c>0, c\in[1,C]} \tilde{P}_{t,c}^{att} \tag{8}$$

where $\mathbf{p}^* = [p_1^*, ..., p_t^*, ..., p_T^*]$ is the maximum class activation score vector of the ground truth action classes. We define action-guided attention loss as:

$$L_{AAL} = \frac{1}{NT}\sum_{n=1}^{N}\sum_{t=1}^{T} |w_{t,n}^{R} - p_{t,n}^{*}| \tag{9}$$

where $w_{t,n}^{R}$ and $p_{t,n}^{*}$ are the output of B-TAM and maximum class activation score of the target action classes, respectively, for segment t of n-th training video (please excuse us to abuse the math notation by adding a subscript n to denote the n-th training video).

3.3 Action Completeness Modeling

Usually, deep nerual networks learn the unique pattern of a specific class for recognition and the localized action frames may highlight a small portion of the target action instead of the entire action instance. In contrast, we aim at localizing the complete action period in the temporal dimension by action completeness modeling, which includes multiple BANets that can iteratively discover different but complementary action instances in an untrimmed video. The framework of our action completeness modeling is shown in Figure 4. In our action completeness modeling, at the first iteration, BANet takes the original feature map as input, while for the remaining iterations, each BANet takes a partially erased feature map as input, where the discriminative features that are identified from the

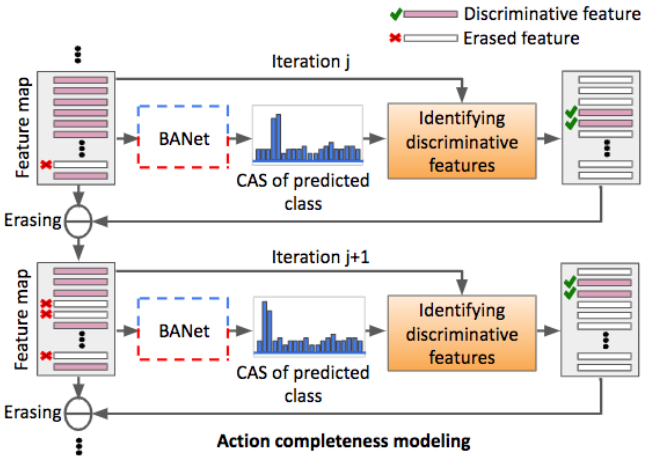

Figure 4: Overview of our action completeness modeling. It contains multiple Background Aware Networks (BANets), which can iteratively discover different but complementary action instances for the complete temporal action localization. During each iteration, BANet identifies discriminative features based on the corresponding input feature map and CAS of the predicted class. These discriminative features are then erased and fed into the BANet of the next iteration.

previous iteration are erased to force the current BANet to identify different discriminative features related to the action instances.

More specifically, let $(\mathcal{M}_1, ..., \mathcal{M}_j, ..., \mathcal{M}_J)$ be the BANets and J is the number of iterations, where each BANet contains a base branch and an attention branch. At iteration j ($j > 1$), to compute the input feature map X_j for the BANet $\mathcal{M}_j$, first, we obtain the maximum class activation score of the predicted action classes from $\mathbf{P}^{att}_{j-1}$ of $\mathcal{M}_{j-1}$ for each video segment:

$$(\bar{p}^*_t)_{j-1} = \max_{\bar{y}_c > \delta, c \in [1,C]} (P^{att}_{t,c})_{j-1}, \quad j > 1 \tag{10}$$

where $\bar{\mathbf{p}}^*_{j-1} = [(\bar{p}^*_1)_{j-1}, ..., (\bar{p}^*_t)_{j-1}, ..., (\bar{p}^*_T)_{j-1}]$ is the maximum class activation score vector of the predicted action classes obtained from $\mathcal{M}_{j-1}$, and $\bar{y}_c$ is the predicted action class which has the video-level classification score in $\tilde{\mathbf{p}}^{att}_{j-1}$ above a certain threshold δ. Then, we identify the discriminative features from X_{j-1} by zeroing out all the elements of X_{j-1} whose corresponding class activation score in $\bar{\mathbf{p}}^*_{j-1} \in \mathbb{R}^T$ are less than a threshold β. Finally, we erase these identified discriminative feature vectors from X_{j-1} and generate a new feature map X_j as input for $\mathcal{M}_j$. With such an action completeness modeling, the sequential BANets are forced to discover different but complementary action instances at different iterations, and can jointly generate the complete temporal action localization.

3.4 Temporal Action Localization in Testing

During the test time, we only use the class activation sequences of attention branches in BANets, since attention branches suppress the background frames. We localize the action instances in the temporal domain from a fused Class Activation Sequences $\mathbf{P}^{att}_{fuse} \in \mathbb{R}^{T \times (C+1)}$, which is calculated by element-wise maximization:

$$\mathbf{P}^{att}_{fuse} = \max(\mathbf{P}^{att}_1, ..., \mathbf{P}^{att}_j, ..., \mathbf{P}^{att}_J) \tag{11}$$

For the localization, first, we compute the predicted classes from the video-level prediction $\tilde{\mathbf{p}}^{att}_1$ of BANet $\mathcal{M}_1$, since $\mathcal{M}_1$ looks at the most highly discriminative features and provides more accurate video-level prediction. Then, we select the CAS of the predicted classes from $\mathbf{P}^{att}_{fuse}$. Thereafter, we threshold the CAS of the predicted classes with a set of thresholds λ_{loc} (ranging from 0 to 0.25 with the step 0.025) to get the candidate segments from the T segments, where each sequence of consecutive candidate segments becomes a proposal. We compute the confidence score for each proposal by averaging the class activation scores of the segments within that proposal. Finally, we perform Non-Maximum Suppression (NMS) with threshold 0.7 to keep the highly overlapped proposals with the high confidence scores to get the final proposals, which are one-dimensional connected components in temporal domain.

4 EXPERIMENTS

4.1 Datasets

THUMOS14 [10]: THUMOS14 has temporal boundary annotations for 200 validation videos and 213 testing videos, which belong to 20 classes. Following rules in the literature [15, 18, 21–23, 25, 27, 33], we use 200 validation videos without using the temporal annotations for training and 213 videos for testing.

ActivityNet1.3 [1]: ActivityNet1.3 dataset covers 200 action classes, which has temporal boundary annotations for 10,024 videos for training, 4926 videos for validation, and 5044 videos for testing. Since the labels of the testing set are withheld, following the rules in the literature [15, 18, 21, 22], we use the training set without using the temporal annotations to train our network and validation set for the evaluation.

Evaluation metrics: We follow the standard evaluation protocol based on mean average precision (mAP) values at different levels of intersection over union (IoU) thresholds. The results are calculated using the benchmark code provided by ActivityNet.

4.2 Implementation Details

First, we generate video segments by sliding a non-overlapping temporal window of 16 frames, then the video segments are resized to have a tensor size of $16 \times 224 \times 224 \times 3$ for RGB and $16 \times 224 \times 224 \times 2$ for optical flow (OF), which are fed to the appearance and motion streams of pre-trained I3D network, respectively, to obtain features of dimension 1024 in each stream. The extracted 1024 dimensional RGB and OF feature vectors are concatenated for each video segment, which produces the feature map of size $T \times 2048$ for an input video, where each video segment has 16 frames, and T depends on the length of the video. We optimize the loss function in Eq.(7) using Adam [12] optimizer. The hyper-parameter γ in Eq.(7) is set as 0.1. We set the thresholds $\alpha = 0.7 \times \max(\mathbf{w}_{att})$ to get $\mathbf{w}^H_{att}$ for the B-TAM in BANet, $\delta = 0.25$ (Eq.(10)) to get the predicted classes from the video-level prediction, and $\beta = 0.9 \times \max(\bar{\mathbf{u}})$ to identify the highly discriminative feature for the ACM-BANets. We use PyTorch to implement our network.

Table 1: Comparing the temporal action localization performance of our algorithm with other state-of-the-art methods in terms of mAP (%) under different IoU thresholds on the THUMOS14 test set. Algorithms are separated regarding the level of supervision. $^+$ indicates the use of additional labels, *e.g.*, the number of action instances in videos, used in some methods.

Supervision	IoU →	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Full	PSDF [38]	51.4	42.6	33.6	26.1	18.8	-	-	-	-
	CDC [26]	-	-	40.1	29.4	23.3	13.1	7.9	-	-
	R-C3D [36]	54.5	51.5	44.8	35.6	28.9	-	-	-	-
	SSN [40]	60.3	56.2	50.6	40.8	29.1	-	-	-	-
	TURN-TAP [6]	54.0	50.9	44.1	34.9	25.6	-	-	-	-
	CBR [7]	60.1	56.7	50.1	41.3	31.0	19.1	9.9	-	-
	TAL-Net [3]	59.8	57.1	53.2	**48.5**	**42.8**	**33.8**	**20.8**	-	-
	GTAN [19]	**69.1**	**63.7**	**57.8**	47.2	38.8	-	-	-	-
Weak$^+$	STAR [37]	**68.8**	**60.0**	48.7	34.7	23.0	-	-	-	-
	3C-Net [20]	59.1	53.5	44.2	34.1	26.6	-	8.1	-	-
	Nguyen et al. [22]	64.2	59.5	**49.1**	**38.4**	**27.5**	**17.3**	**8.6**	**3.2**	**0.5**
Weak	UntrimmedNet [33]	44.4	37.7	28.2	21.1	13.7	-	-	-	-
	AutoLoc [27]	-	-	35.8	29.0	21.2	13.4	5.8	-	-
	STPN (I3D) [21]	52.0	44.7	35.5	25.8	16.9	9.9	4.3	1.2	0.1
	W-TALC (I3D) [23]	55.2	49.6	40.1	31.1	22.8	-	7.6	-	-
	CMCS (I3D) [18]	57.4	50.8	41.2	32.1	23.1	15.0	7.0	-	-
	3C-Net (I3D) [20]	56.8	49.8	40.9	32.3	24.6	-	7.7	-	-
	Nguyen et al. (I3D) [22]	60.4	56.0	46.6	37.5	26.8	17.6	9.0	3.3	0.4
	BaS-Net (I3D) [15]	58.2	52.3	44.6	36.0	27.0	18.6	10.4	3.9	0.5
	DGAM (I3D) [25]	60.0	54.2	46.8	38.2	28.8	19.8	11.4	3.6	0.4
	ACM-BANet (ours) (I3D)	**64.6**	**57.7**	**48.9**	**40.9**	**32.3**	**21.9**	**13.5**	**5.9**	**0.9**

Table 2: Results on ActivityNet1.3 dataset. The column Avg. indicates the average mAP at IoU thresholds 0.5:0.05:0.95.

Methods IoU →	0.5	0.75	0.95	Avg.
STPN [21]	29.3	16.9	2.6	-
CMCS [18]	34.0	20.9	5.7	21.2
Nguyen et al. [22]	36.4	19.2	2.9	-
BaS-Net [15]	34.5	22.5	4.9	22.2
ACM-BANet (ours)	**37.6**	**24.7**	**6.5**	**24.4**

4.3 Comparisons with the State-of-the-art

Table 1 summarizes the performance on the THUMOS14 dataset for action localization methods in the past few years. Regarding the level of supervision, we separate the methods into three categories: (i) full supervision: use precise temporal annotations; (ii) weak$^+$ supervision: use video-level labels AND exploit additional annotations (e.g., the number of action instances in videos); and (iii) weak supervision: use only video-level labels. As shown in Table 1, our algorithm outperforms the other state-of-the-art weakly-supervised methods by a large margin, and establishes a new state-of-the-art on weakly-supervised temporal action localization on the challenging THUMOS14. It is important to note that our method performs better than the latest weakly-supervised methods [15, 18, 20–23, 25], which use the same pre-trained I3D features as us. At the same time, although without the precise temporal annotation on the action instance in untrimmed videos, our approach achieves competitive performance compared to several recent fully-supervised methods such as [3, 6, 7, 40], on the coarse localization of action instances. Even without the additional annotations such as the number of action instances in videos, our approach achieves superior performance compared to the methods in weak$^+$ supervision, on the challenging fine localization of action instances for IoU ≥ 0.4.

Table 2 presents the performance of our algorithm on the validation set of ActivityNet1.3 dataset, showing the superior performance of the proposed method on temporal action localization, compared to other state-of-the-arts.

4.4 Ablation Studies

We perform ablation studies on THUMOS14 to investigate the contribution of different branches, modules, and loss functions of ACM-BANets, as summarized in Table 3.

(i) Base branch: The first experiment in Table 3 shows the performance of the base branch. We design the base branch without any attention module and set this branch with only L_{base} loss, which aims to classify an input video into action classes and the background class. We use the segment-level class activation sequences of the predicted classes to localize the action instances.

(ii) Attention branch: The second set of experiments in Table 3 shows the performance of the attention branch, which is trained to recognize action classes in videos. We conduct several experiments on attention branch to see the effectiveness of our proposed Background-aware Temporal Attention Module (B-TAM) and Action-guided Attention Loss (L_{AAL}). First, we design the attention branch with conventional Temporal Attention Module (TAM), and L_{att} loss. Then, we include the L_{AAL} loss in the attention branch.

Table 3: Ablation study of different architectures and loss functions on THUMOS14 dataset.

Approach	TAM	B-TAM	L_{base}	L_{att}	L_{AAL}	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Base branch			✓			35.5	28.3	23.1	18.5	13.4	8.9	4.8	1.9	0.1
Attention branch	✓			✓		40.2	33.2	27.0	22.3	15.7	10.9	5.8	2.2	0.2
Attention branch	✓			✓	✓	52.4	45.4	36.6	27.1	20.1	13.5	7.1	2.7	0.3
Attention branch		✓		✓	✓	54.6	47.8	38.8	29.7	22.3	15.2	9.0	3.2	0.5
BANet		✓	✓	✓	✓	63.1	56.0	47.3	39.3	30.8	20.4	12.1	4.9	0.8
ACM-BANets (iteration=2)		✓	✓	✓	✓	64.2	57.1	48.1	40.2	31.9	21.5	12.9	5.3	0.9
ACM-BANets (iteration=3)		✓	✓	✓	✓	64.6	57.7	48.9	40.9	32.3	21.9	13.5	5.9	0.9
ACM-BANets (iteration=4)		✓	✓	✓	✓	64.5	57.4	48.7	40.6	32.1	21.8	13.3	5.8	0.9

Table 4: Ablation study of different aggregating techniques used in the attention (att) branch of BANet on THUMOS14 dataset. mAP is computed at IoU=0.5.

Methods	mAP
BANet (top-k mean in att. branch)	29.7
BANet (self-att. weighted top-k mean in att. branch)	30.8

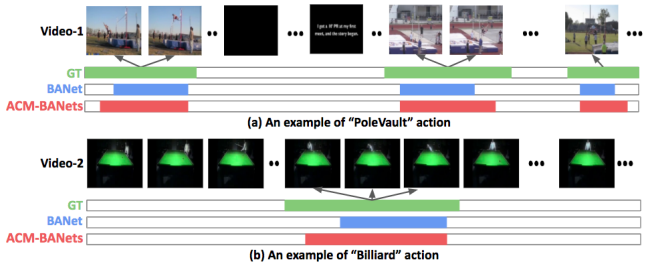

Figure 5: Qualitative results on untrimmed testing videos from THUMOS14. (a) Multiple occurrences of "PoleVault" action appear in a single video. (b) The appearance of all the frames remain similar from the beginning to the end in the video with the action of "Billiard". Our proposed BANet focuses on the true actions and neglects the background frames, and our ACM-BANets tries to further completely localize the temporal intervals for the target class.

Finally, we configure the attention branch with B-TAM, and L_{att} and L_{AAL} loss functions. The second set in Table 3 summarizes that our L_{AAL} loss leads to achieve better performance and our B-TAM provides superior performance compared to TAM. Therefore, we choose the third configuration of the second set in Table 3 as the attention branch in our proposed ACM-BANets.

(iii) BANet: The third set in Table 3 shows the performance of our proposed BANet, which includes both base and attention branches, and is jointly trained with the corresponding loss functions to suppress both ambiguous and highly discriminative background frames. Table 3 shows that our proposed BANet outperforms the individual base branch and attention branch by a large margin.

(iv) ACM-BANets: The fourth set in Table 3 shows the performance of our proposed ACM-BANets across different iterations, which aim to discover different but complementary action instances in both highly discriminative and ambiguous action frames for the complete temporal action localization. ACM-BANets obtain a significant increase over BANet and the highest performance is achieved for the third iteration. Therefore, we choose $J = 3$ as the number of iteration for our ACM-BANets.

We also perform the ablation studies on THUMOS14 for different aggregating techniques used in the attention branch of BANet for the video-level prediction, as shown in Table 4, which summarizes that we achieve better performance for self-attention weighted top-K mean compared to top-K mean aggreagating technique.

4.5 Qualitative Analysis

We present some qualitative results on the test set of THUMOS14 in Figure 5. Figure 5(a) shows the example of the "PoleValut" action, which appears in a single video multiple times with different action context and background frames. Figure 5(b) shows the "Billiard" action, where the appearance of all the frames remain similar from the beginning to the end in the video with the action of "Billiard". In both examples, our BANet focuses on true actions and neglects the background frames, and our ACM-BANets tries to further completely localize the temporal intervals for the target class.

5 CONCLUSION

In this paper, we introduce a novel ACM-BANets for the weakly-supervised temporal action localization, which addresses two main challenges: (1) how to design and train a weakly-supervised network that can suppress both highly discriminative and ambiguous *background* frames to remove the false positives? and (2) how to design a temporal action localization framework to discover action instances in both highly discriminative and ambiguous *action* frames for the complete localization? Our proposed ACM-BANets that suppresses both highly discriminative and ambiguous background frames and discovers action instances in both highly discriminative and ambiguous action frames outperforms the current weakly-supervised temporal action localization methods on THUMOS14 and ActivityNet1.3 datasets.

ACKNOWLEDGMENTS

This project was supported by the National Science Foundation via NRI-1954548 and CPS-1646162.

REFERENCES

[1] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the ieee conference on computer vision and pattern recognition*. 961–970.

[2] Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6299–6308.

[3] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. 2018. Rethinking the faster r-cnn architecture for temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1130–1139.

[4] Xiyang Dai, Bharat Singh, Guyue Zhang, Larry S Davis, and Yan Qiu Chen. 2017. Temporal context network for activity localization in videos. In *Proceedings of the IEEE International Conference on Computer Vision*. 5793–5802.

[5] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2625–2634.

[6] Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia. 2017. Turn tap: Temporal unit regression network for temporal action proposals. In *Proceedings of the IEEE international conference on computer vision*. 3628–3636.

[7] Jiyang Gao, Zhenheng Yang, and Ram Nevatia. 2017. Cascaded boundary regression for temporal action detection. *arXiv preprint arXiv:1705.01180* (2017).

[8] Ross Girshick. 2015. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 1440–1448.

[9] Qibin Hou, PengTao Jiang, Yunchao Wei, and Ming-Ming Cheng. 2018. Self-erasing network for integral object attention. In *Advances in Neural Information Processing Systems*. 549–559.

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

[11] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei. 2014. Large-scale video classification with convolutional neural networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 1725–1732.

[12] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[14] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. 2011. HMDB: a large video database for human motion recognition. In *2011 International Conference on Computer Vision*. IEEE, 2556–2563.

[15] Pilhyeon Lee, Youngjung Uh, and Hyeran Byun. 2020. Background Suppression Network for Weakly-Supervised Temporal Action Localization.. In *AAAI*. 11320–11327.

[16] Tianwei Lin, Xu Zhao, and Zheng Shou. 2017. Single shot temporal action detection. In *Proceedings of the 25th ACM international conference on Multimedia*. 988–996.

[17] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. 2018. Bsn: Boundary sensitive network for temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 3–19.

[18] Daochang Liu, Tingting Jiang, and Yizhou Wang. 2019. Completeness modeling and context separation for weakly supervised temporal action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1298–1307.

[19] Fuchen Long, Ting Yao, Zhaofan Qiu, Xinmei Tian, Jiebo Luo, and Tao Mei. 2019. Gaussian temporal awareness networks for action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 344–353.

[20] Sanath Narayan, Hisham Cholakkal, Fahad Shahbaz Khan, and Ling Shao. 2019. 3c-net: Category count and center loss for weakly-supervised action localization. In *Proceedings of the IEEE International Conference on Computer Vision*. 8679–8687.

[21] Phuc Nguyen, Ting Liu, Gautam Prasad, and Bohyung Han. 2018. Weakly supervised action localization by sparse temporal pooling network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6752–6761.

[22] Phuc Xuan Nguyen, Deva Ramanan, and Charless C Fowlkes. 2019. Weakly-supervised action localization with background modeling. In *Proceedings of the IEEE International Conference on Computer Vision*. 5502–5511.

[23] Sujoy Paul, Sourya Roy, and Amit K Roy-Chowdhury. 2018. W-talc: Weakly-supervised temporal activity localization and classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 563–579.

[24] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*. 91–99.

[25] Baifeng Shi, Qi Dai, Yadong Mu, and Jingdong Wang. 2020. Weakly-Supervised Action Localization by Generative Attention Modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1009–1019.

[26] Zheng Shou, Jonathan Chan, Alireza Zareian, Kazuyuki Miyazawa, and Shih-Fu Chang. 2017. Cdc: Convolutional-de-convolutional networks for precise temporal action localization in untrimmed videos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5734–5743.

[27] Zheng Shou, Hang Gao, Lei Zhang, Kazuyuki Miyazawa, and Shih-Fu Chang. 2018. Autoloc: Weakly-supervised temporal action localization in untrimmed videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 154–171.

[28] Zheng Shou, Dongang Wang, and Shih-Fu Chang. 2016. Temporal action localization in untrimmed videos via multi-stage cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1049–1058.

[29] Karen Simonyan and Andrew Zisserman. 2014. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*. 568–576.

[30] Krishna Kumar Singh and Yong Jae Lee. 2017. Hide-and-seek: Forcing a network to be meticulous for weakly-supervised object and action localization. In *2017 IEEE international conference on computer vision (ICCV)*. IEEE, 3544–3553.

[31] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* (2012).

[32] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2015. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*. 4489–4497.

[33] Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. 2017. Untrimmednets for weakly supervised action recognition and detection. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 4325–4334.

[34] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*. Springer, 20–36.

[35] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. 2017. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1568–1576.

[36] Huijuan Xu, Abir Das, and Kate Saenko. 2017. R-c3d: Region convolutional 3d network for temporal activity detection. In *Proceedings of the IEEE international conference on computer vision*. 5783–5792.

[37] Yunlu Xu, Chengwei Zhang, Zhanzhan Cheng, Jianwen Xie, Yi Niu, Shiliang Pu, and Fei Wu. 2019. Segregated temporal assembly recurrent networks for weakly supervised multiple action detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9070–9078.

[38] Jun Yuan, Bingbing Ni, Xiaokang Yang, and Ashraf A Kassim. 2016. Temporal action localization with pyramid of score distribution features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3093–3102.

[39] Xiaolin Zhang, Yunchao Wei, Jiashi Feng, Yi Yang, and Thomas S Huang. 2018. Adversarial complementary learning for weakly supervised object localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1325–1334.

[40] Yue Zhao, Yuanjun Xiong, Limin Wang, Zhirong Wu, Xiaoou Tang, and Dahua Lin. 2017. Temporal action detection with structured segment networks. In *Proceedings of the IEEE International Conference on Computer Vision*. 2914–2923.