Learning Background Suppression Model for Weakly-supervised Temporal Action Localization

Mengxue Liu, Xiangjun Gao*, Fangzhen Ge, Huaiyu Liu, and Wenjing Li

***Abstract*—Weakly-supervised temporal action localization aims to identify all action instances and their corresponding categories in the untrimmed videos. Since it involves only video-level labels during training, resulting in this problem being more challenging. Existing attention-based action localization methods use the attention module to identify action segments and assign them to the appropriate action categories. However, such methods inevitably suffer from many background segments that are similar to the target actions, being recognized as actions. To address this issue, we propose a new weakly-supervised temporal action localization network using background suppression (BS-WTAL). The network defines the filtering module, which can suppress the activation of the background regions, classification module, which identifies the activity categories, and generative attention module, which is learned to model a segment-wise representation. This enables BS-WTAL to accurately distinguish actions from the background. Furthermore, we conduct ablation studies from different perspectives. Extensive experiments are performed on two datasets: THUMOS14 and ActivityNet1.2. Our approach exhibits better performance on these two datasets and achieves performance comparable to the state-of-the-art fully-supervised methods.**

***Index Terms*—Weak Supervision, Temporal Action Localization, Filtering Module, Background Suppression**

I. Introduction

TEMPORAL action localization is a challenging subject in the field of computer vision. It aims to determine the start and end times of each action instance in the untrimmed videos, and classify each action instance. Owing to the rapid development of deep learning and its wide application in various fields, including medical diagnosis [1], [2], stock price prediction [3], etc., many methods have been proposed. And these methods have achieved remarkable performance under a fully-supervised definition [4], [5], [6], [7], [8], [9], [10], which require precise temporal boundaries and action categories of all action instances contained in the training videos. However, with the influx of voluminous online video data, annotating temporal boundaries for massive videos has become time-consuming, expensive and subjective. Relatively speaking, it is easier to label these online videos with video-level activity categories. Therefore, weakly-supervised temporal action localization (WTAL) using only video-level labels during training is more practical.

Currently, based on the video features processing techniques followed by the models, WTAL methods can be roughly divided into two types. The first type is the top-down method [11], [12], [13], [14], which uses the generated class activation mapping (TCAM) [15] to obtain a degree of correlation between each video segment and all action categories. This correlation is regarded as a class-aware attention map. The second type is the bottom-up method [16], [17], [18], which directly learns the correlation, a type of class-agnostic attention map, from the original video through the attention module. The segments with high attention values are regarded as actions; those with low attention values are considered the background. Both methods need to optimize the segment-level attention weights using the classification module. In previous studies, these two methods have achieved good performance in WTAL tasks. However, even with thorough observation and analysis of the localization results, we still encounter the problem of background regions similar to the target actions being recognized as actions. Taking the video clip of the long jump in Fig. 1 as an example, the attentions refer to the segment-level attention weights generated by the attention module in DGAM [18]. The following two lines are the ground truth of action localization and localization results in DGAM [18]. As shown in the figure, there is a small difference between the weights of some background regions and action regions. During action localization, these background regions are also selected, which are the red regions in the figure.

To solve the problem of background regions being recognized as actions, we devise a filtering module that amplifies the attention weights from the action regions and penalizes the ones from the background regions. Thus, we distinguish the action from the background more accurately. The optimization of the filtering module is inseparable from the classification and generative attention modules. Therefore, our BS-WTAL model consists of three parts: the filtering, classification, and generative attention modules. The classification module can identify the action categories contained in a video by training a classifier. Following [18], we integrate the generative attention module, which can generate segment-wise feature representation under the condition of segment-level attention weights, which is very beneficial to solve the problem of action and context confusion. By maximizing the attention weights of action segments and minimizing the classification loss, our proposed approach is optimized. We conduct experiments using our model on two very popular

Manuscript received February 9, 2021; revised August 23, 2021. This work is supported by Key Research and Development Projects in Anhui Province under Grant 201904a05020072. Natural Science Research Projects in Anhui Universities under Grant KJ2019A0606 and KJ2019A0603.

*Corresponding auther

Mengxue Liu is a graduate student of the College of Computer Science and Technology, Huaibei Normal University, Huaibei 235000, China (e-mail: 2364235879@qq.com).

Xiangjun Gao is a Professor of the College of Computer Science and Technology, Huaibei Normal University, Huaibei 235000, China (e-mail: xjgao75@163.com).

Fangzhen Ge is a Professor of the College of Computer Science and Technology, Huaibei Normal University, Huaibei 235000, China (e-mail: gfz203377@163.com).

Huaiyu Liu is a Professor of the College of Computer Science and Technology, Huaibei Normal University, Huaibei 235000, China (e-mail: hbant@163.com).

Wenjing Li is a graduate student of the College of Computer Science and Technology, Huaibei Normal University, Huaibei 235000, China (e-mail: 941266407@qq.com).

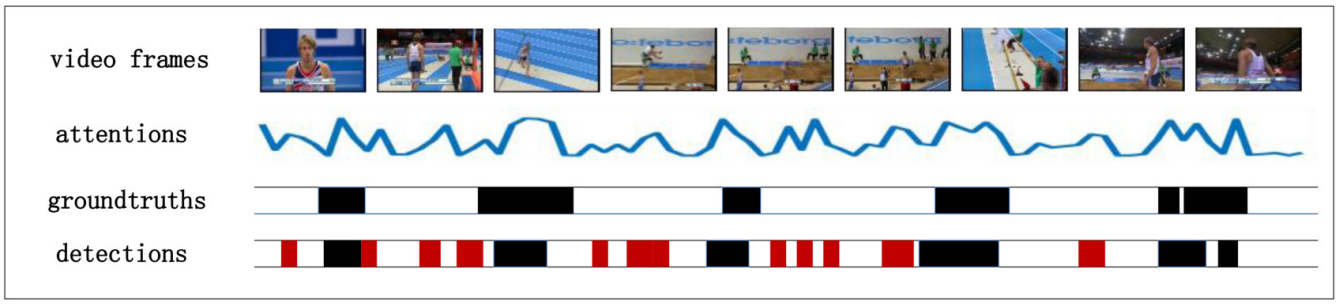

Fig. 1. Localization Results of DGAM Model.

datasets, THUMOS14 [19] and ActivityNet1.2 [20]. The results of the experiments prove that our BS-WTAL model can more effectively distinguish actions from the background than other state-of-the-art methods.

In DGAM, it is common to categorize background regions similar to the target acts as actions. The main contribution of our study is the proposed BS-WTAL model to address this issue. The model adopts a filtering module that suppresses the background features and amplifies the action features, which are more conducive to modeling the background, to more accurately distinguish actions from the background.

II. Related Works

A. Fully-supervised temporal action localization

They require precise temporal boundaries and action categories of all action instances during training. With the rapid rise of deep learning, many object detection algorithms have achieved excellent performance, such as R-CNN [21] and its variants [22], [23]. Inspired by these methods, the fully-supervised action localization methods have also made extraordinary progress. SCNN [5] proposes the three segment-based 3D ConvsNets for action localization. SSN [6] executes a watershed algorithm on the complemented actionness values to generate more accurate proposals, then constructs a temporal structure for each proposal, and evaluates the category and completeness of each proposal through two classifiers. CDC [7] proposes a CDC filiter to retain time granularity to boost the accuracy of localization results. BSN [9] proposes a new proposal generation algorithm that locally predicts the temporal boundaries of proposals and globally evaluates the confidence scores. Yeung et al. [24] use REINFORCE to train a single coherent model that can directly predict the start and end locations of action instances. SSAD [25] adopts a single-shot action detection model that directly predicts the categories and localizes each action instance. These reported methods have a beneficial effect on the performance of action localization.

B. WTAL

The WTAL method with video-level labels and no segment-level annotations during training was first introduced in [26]. As explained in Section 1, the WTAL methods can be thought of as containing both top-down and bottom-up methods. Among top-down methods, W-TALC [11] proposes that the correlations between videos with the similar labels are conducive to the improvement of localization performance. UntrimmedNets [12] adopts a soft selection module to obtain segment-level weights and thresholds the weights to select foreground segments to perform action localization. Narayan et al. [27] propose 3C-Net, which defines center and counting losses for weakly-supervised action localization. Different from the top-down approaches, the bottom-up methods directly generate attention weights for each segment from raw videos. AutoLoc [16] trains a boundary predictor using OIC loss. STPN [17] proposes to identify action-related segments using the attention module and produce video-level features by temporal average pooling. MAAN [28] proposes a marginalized average aggregation module to solve the problem of recognizing only the most discriminative regions. Nguyen et al. [29] devise to use the complementary factors of the attention weight values generated by the attention module to model the background. DGAM [18] proposes a new generative attention model that models the segment-wise representation distribution under the segment-attention-weights condition to address the action-context confusion problem.

In addition, approaches that combine the two aforementioned methods have been developed to achieve better localization performance. For example, Bas-Net [30] introduces an additional class to represent the background and devises a two-branch architecture to suppress activation from background regions. Clean-Net [31] proposes a new contrast-based evaluation and localization network. Gong et al. [32] utilize the class-agnostic attention module to determine action segments and integrate the class-specific attention module to obtain the action probability distribution.

III. Our Approach

We introduce the proposed BS-WTAL model in great detail. During training, we provide N videos with video-level labels $y \in \{0, 1, \cdots, C\}$, where C denotes the total number of activity categories involved in this datasets and 0 represents the background class. For each training video X, we first extract RGB frames and optical flow frames between two adjacent RGB frames, then we divide them into multiple segments consisting of non-overlapping 16 frames. Next, we input these both stream segments into the pre-trained feature extractor to obtain segment-level feature vectors. For $t-th$ segment, we denote its feature as x_t and its feature dimension is set to d. When training the model, we choose the features of T segments from the both streams as the training data of the proposed BS-WTAL model. Therefore, We can represent the training data as $X = (x_t)_{t=1}^{T}$.

During testing, we use a trained model to localize all action instance occurred in testing videos. They are formatted

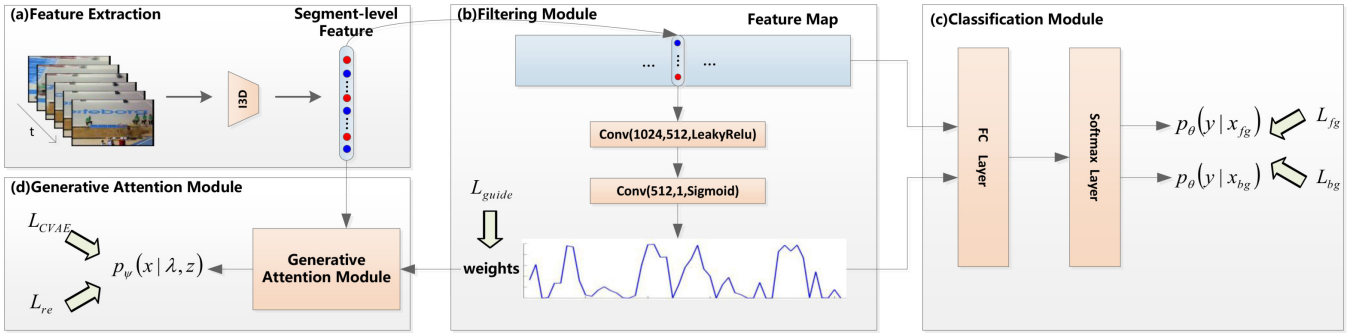

Fig. 2. Framework Overview. (a) Feature Extraction (b) Filtering Module (c) Classification Module (d) Generative Attention Module

as $\{(s_i, e_i, c_i, p_i)\}_{i=1}^{n}$, where s_i and e_i denote the starting and ending times of the localized action regions. And c_i and n denote action categories and total number of proposals generated for a video, respectively, while p_i represents the confidence score of the predicted action category.

A. Architecture Overview

The overall architecture of our proposed BS-WTAL model is shown in Fig. 2. It is composed of the feature extraction, the filtering module, the classification module and the generative attention module. Given a video feature $X = (x_t)_{t=1}^{T}$, our algorithm first generate the attention weights $\lambda = (\lambda_t)_{t=1}^{T}$, which denote the correlation between each video segment and action categories. Then we perform temporal average pooling to calculate video-level foreground features x_{fg} from segment-level features and their corresponding attention weights λ. Similarly, we use $1-\lambda$ to compute the video-level background representation, denoted by x_{bg}. Next, these video-level representations are fed into the classification module to produce the class scores. The generative attention module, that is, the conditional Variational Auto-Encoder (VAE), utilizes the segment-level feature and its attention weight to generate a segment-level representation, denoted by $\widehat{x}$, which aims to further refine the attention weights. We will discuss these details in the following sections.

B. Filtering Module

In a localization model using the attention mechanism, the main objective is to optimize the segment-level attention weight λ. By observing the results of action localization in DGAM [18], we realize the following problem: many background regions that are similar to the target actions in appearance or motion patterns tend to be recognized as actions, which causes performance degradation. We analyze attention weights and find that the corresponding attention weights of background regions mistaken as actions are not significantly different from the attention weights corresponding to action regions. To address this issue, we devise a filtering module that can suppress the attention weights of background regions and encourage the attention weights of foreground regions.

We experiment with features extracted from I3D [33]. For a training video, we first extract the RGB frames and the optical flow frames between two adjacent RGB frames using the TV-L1 algorithm [34]. Then, we divide two streams into segments consisting of non-overlapping 16 frames. Next, these segments are inputted to I3D to obtain segment-level RGB and flow features. Note that both stream features are sent into the filtering module.

We show the composition of the attention module and the filtering module in Figure 3. The attention module consists of fully connected layers while the filtering module is implemented by convolutional layers. Both perform sigmoid operations on the output of the final layer to ensure that that the generated attention weight values are between 0 and 1. Thereafter, we use these attention weights λ to weight segment-level features to obtain video-level foreground features dominated by action features. Thus, we can obtain more accurate class scores.

We take the long jump shown in Fig. 1 as an example to analyze the attention weights generated by the attention and filtering modules, as shown in Fig. 4. In this diagram , we can see the localization results of DGAM and our model on the same video clip. In Fig. 4(a), the first line is the attention weights generated by the attention module of DGAM, the following line is action instances labeled in this video clip, and the last line is the localization results predicted by DGAM. In Fig. 4(b), the first line is the attention weights generated by the filtering module of our model, the middle line is the same action instances labeled in this video clip, as in Fig. 4(a). Finally, the third line is the results of our algorithm prediction. From this graph we can conclude that our model can generate better attention weights. In the results of DGAM, many background regions are recognized as actions. Owing to the integration of the filtering module, our algorithm successfully suppresses the attention weights from background segments and solves the problem of distinguishing action instances from background regions.

C. Classification Module

To predict video-level class scores of foreground features, we utilize attention weights λ over all segments and their corresponding segment-level features to perform temporal average pooling, so as to obtain a video-level foreground feature x_{fg} dominated by action features. We can define it as follows:

$$x_{fg} = \frac{\sum_{t=1}^{T} \lambda_t x_t}{\sum_{t=1}^{T} \lambda_t}. \tag{1}$$

Then, the video-level foreground features are input to the classifier to produce video-level prediction $p_{fg}[c]$ and compute the cross-entropy loss L_{fg} between the prediction $p_{fg}[c]$

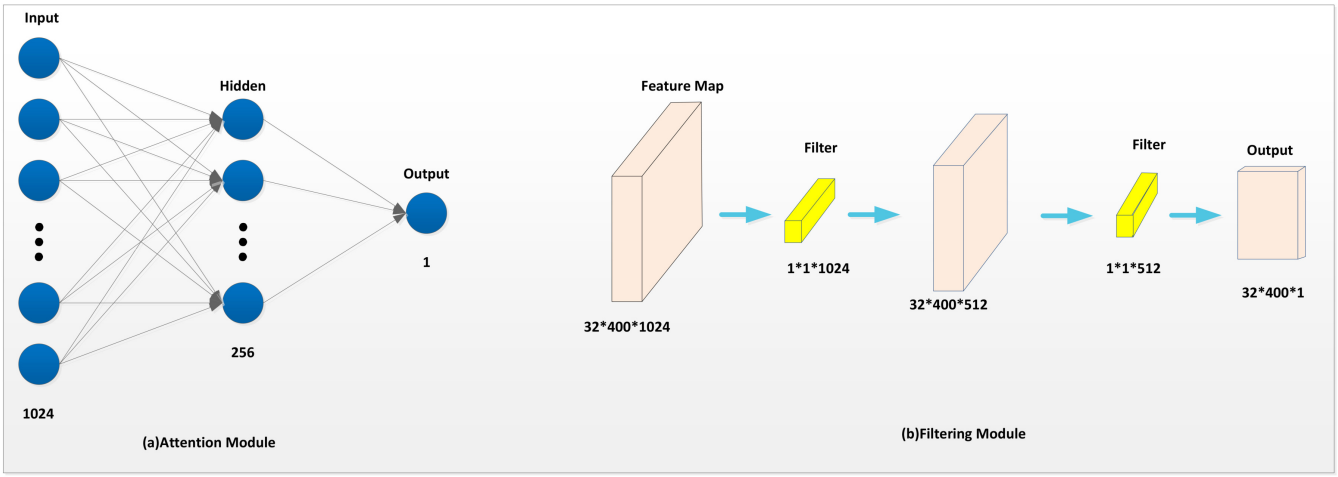

Fig. 3. Structure comparison between attention module and filtering module.

and video-label y :

$$L_{fg} = -\log p_{fg}\left[y\right]. \tag{2}$$

Similarly, we use the complementary factor $1-\lambda$ to compute the background features x_{bg} [29]:

$$x_{bg} = \frac{\sum_{t=1}^{T}\left(1-\lambda_t\right)x_t}{\sum_{t=1}^{T}\left(1-\lambda_t\right)}. \tag{3}$$

Next, the video-level background features are sent into the classification module to obtain the class scores $p_{bg}\left[c\right]$ corresponding to category c and calculate the cross-entropy loss L_{bg}, which forces the class score on the background class close to 1.

$$L_{bg} = -\log p_{bg}\left[0\right]. \tag{4}$$

D. Training

1) Loss of the generative attention module: Following [18], we optimize CVAE using pseudo-labels, which are provided by the attention weights generated by the filtering module. By minimizing the variational loss L_{CVAE}, the generative attention module can be optimized. The L_{CVAE} can be defined as following:

$$\begin{aligned} L_{CVAE} = &-\frac{1}{L}\sum_{l=1}^{L}\log p_{\psi}\left(x_t\left|\lambda_t\right., z_t^{(l)}\right) \\ &+\beta\cdot KL\left(q_{\varphi}\left(z_t|x_t,\lambda_t\right)\|p_{\psi}\left(z_t\left|\lambda_t\right.\right)\right). \end{aligned} \tag{5}$$

where $z_t^{(l)}$ represents $l-th$ example sampled from q_{φ}. And q_{φ} is defined as a Gaussian distribution, which can be formulated as $q_{\varphi}\left(z_t|x_t,\lambda_t\right) = N\left(z_t|\mu_{\varphi},\Sigma_{\varphi}\right)$. We can obtain μ_{φ} and Σ_{φ} from the encoder and they are mean value and the standard deviation of Gaussian distribution respectively. Furthermore, the prior probability distribution $p_{\psi}\left(z_t|\lambda_t\right)$ is approximated as a Gaussian function and is written as $p_{\psi}\left(z_t|\lambda_t\right) = N\left(z_t|r\lambda_t\cdot 1, I\right)$, where 1 is a vector whose element values are all one and γ is a super parameter that control the difference between the attention weights. In addition, β is a super parameter that control the weight between the reconstruction quality and sampling accuracy.

CVAE generates feature representation $\hat{x}$ for each segment. In order to further optimize the attention weights λ, we calculate the the reconstruction Loss L_{re}, which forces the generated segment-level features close to the original feature representations. We define this loss function as follows:

$$L_{re} = -\sum_{t=1}^{T}\log\left\{\frac{1}{L}\sum_{l=1}^{L}p_{\psi}\left(x_t|\lambda_t, z_t^{(l)}\right)\right\}. \tag{6}$$

where $z_t^{(l)}$ is $l-th$ example sampled from the prior distribution $p_{\psi}\left(z_t|\lambda_t\right)$. In our setting, we let L equal to 1. Therefore, we can rewrite Eq.(6) as follows:

$$L_{re} = \sum_{t=1}^{T}\left\|x_t - f_{\psi}\left(\lambda_t, z_t\right)\right\|^2. \tag{7}$$

2) Loss of the filtering module: To further optimize attention weights λ, we first obtain top-down class-wise attention weights from temporal class activation maps (TCAM) [17]. Then, we minimize the distance between the class-wise and the class-agnostic attention weights to obtain more reliable attention values. Therefore, L_{guide} can be calculated as following:

$$L_{guide} = \frac{1}{T}\sum_{t=1}^{T}\left|\lambda_t - \widehat{\lambda}_t^{fg}\right| + \left|\lambda_t - \widehat{\lambda}_t^{bg}\right|. \tag{8}$$

where $\widehat{\lambda}_t^{fg}$ and $\widehat{\lambda}_t^{bg}$ are the top-down class-aware foreground and background TCAMs. The calculation method is as follows:

$$\widehat{\lambda}_t^{fg} = G\left(\sigma_s\right) * \frac{\exp^{w_y^T x_t}}{\sum_{c=0}^{C}\exp^{w_c^T x_t}}, \tag{9}$$

$$\widehat{\lambda}_t^{bg} = G\left(\sigma_s\right) * \frac{\sum_{c=1}^{C}\exp^{w_c^T x_t}}{\sum_{c=0}^{C}\exp^{w_c^T x_t}}. \tag{10}$$

where G is a Gaussian smooth filter with standard deviations σ_s, and * is a convolution operation.

3) Training objective: Following [18], we train our model using an alternate training method, which requires two steps:

Step 1 Update filtering Module and classification module:

$$L = L_{fg} + \gamma_1 L_{bg} + \gamma_2 L_{re} + \gamma_3 L_{guide}. \tag{11}$$

Step 2 Optimize CVAE with loss L_{CVAE}.

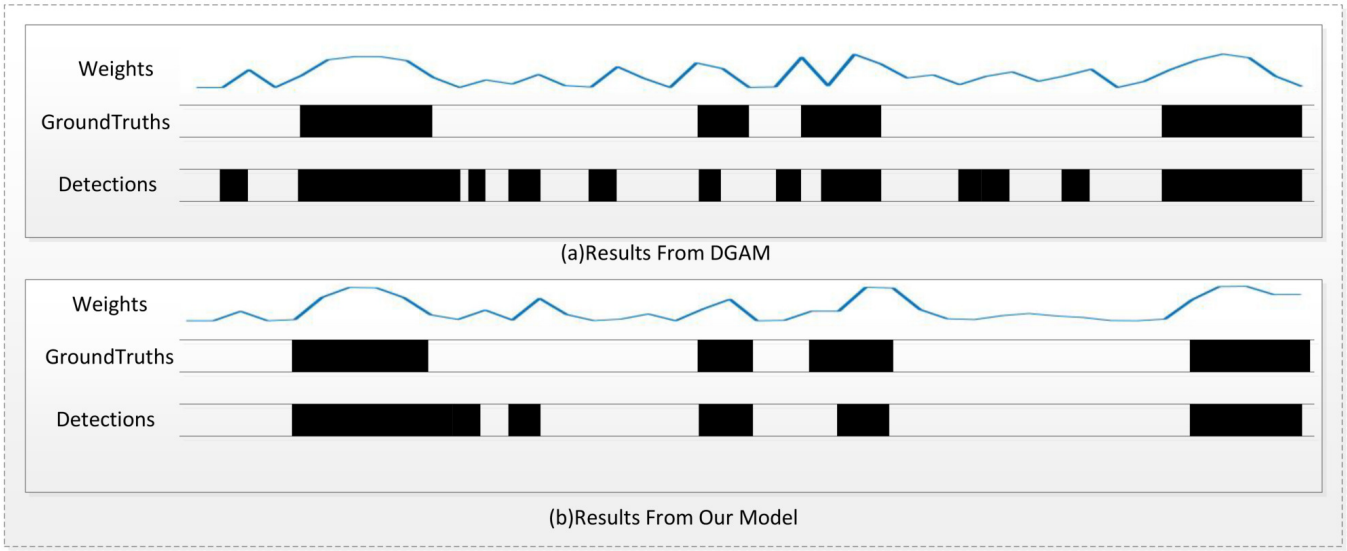

Fig. 4. Visualization of the effectiveness of our method.

E. Action Localization

During inference, we use only the filtering and classification modules. Given a video, we feed it to our model and obtain the attention weights λ and class scores. Following [18], we threshold the class scores to identify the action categories that occur in this video. For each of the remaining categories, we calculate the W-TCAM and threshold it to extract the connected components. The proposals are generated by integrating adjacent components. We further follow [14], [16] to obtain the confidence score of each proposal.

IV. Experiments

A. Data Description and Evaluation Protocol

1) Datasets: To measure the performance of our method on the task of WTAL, we conducted extensive experiments on two datasets, THUMOS14 [19] and ActivityNet1.2 [20]. They contain multiple actions and have a large proportion of non-action regions, which is an obstacle to locating action instances but also improves the generalization ability of our model.

THUMOS14. This dataset provides 200 validation and 212 testing videos. Following the convention, we train using the validation videos and acquire only their video-level labels, and evaluate our algorithm using the testing videos. This dataset contains 20 activity categories. Moreover, this dataset is quite long and involves more than one action category in some videos. The lengths of action instances vary greatly, from less than a second to minutes. These all pose challenges to action localization.

ActivityNet1.2. This dataset involves an enormous number of videos contained a variety of human activities, which contains 4819 training videos and 2383 testing videos. There are 200 action categories in this dataset.

2) Evaluation metrics: In our experiments, we evaluate the performance of our proposed model by calculating the mean average precision (mAP) under different intersection over union (IoU). we utilize the benchmark code to compute the mAP and the code is provided by ActivityNet official codebase.

B. Implementation Details

For each video in these two datasets, we extract RGB and optical flow features using the pre-trained feature extractor I3D [33], which is trained on the Kinetics dataset [35]. Note that this feature extractor does not make any fine-tuning in our experiment. Specifically, both RGB and optical flow frames extracted from RGB data are divided into many segments consisting of non-overlapping 16 frames. Segment-level RGB and flow features with a dimension of $d = 1024$ are obtained by feeding these segments into I3D. Following [18], during training, we set T to 400. Furthermore, if the number of video segments is more than 400, T is set to 400 for THUMOS14, 200 for ActivityNet1.2, and the entire video will be used as input otherwise.

In the filtering module, we devise two convolutional layers with convolutional kernel sizes of $1 \times 1 \times 1024$ and $1 \times 1 \times 512$. This is expected to suppress the activation from the background regions. Thus, our model can effectively identify action regions.

During training, we set the batch size to 32. In Eq.(11), we set γ_1 to 0.03 and γ_3 to 0.1. γ_2 is set to 0.5 for the RGB stream and 0.3 for flow stream. β=0.1 in Eq.(5). During testing, if the maximum score is greater than 0.1, we identify that the video contains at least one action category. Then, we threshold the classification score to obtain the classes that this video contains. Next, the process of temporal action localization is conducted on the remaining classes to generate action proposals. Finally, we perform a non maximum suppression operation with an IoU threshold of 0.5 on the candidate proposals to remove duplicated action proposals. We apply deep learning framework pytorch [36] to implement our model and optimize using the Adam optimizer whose learning rate is set to 0.001.

C. State-of-the-Art Comparisons

Table I summarizes the experimental results of our proposed model and other methods on the THUMOS14 testing dataset. We calculate mAP values at different IoU thresholds of 0.1:0.1:0.9. Temporal action localization conducted under

TABLE I
WTAL PERFORMANCE COMPARSION ON THUMOS14 TESTING SET. ("OURS" IS OUR REALIZATION)

Supervision	Method	mAP@IoU								
		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
Full	SCNN [5]	47.7	43.5	36.3	28.7	19.0	10.3	5.3	-	-
	SSN [6]	66.0	59.4	51.9	41.0	29.8	-	-	-	-
	CDC [7]	-	-	40.1	29.4	23.3	13.1	7.9	-	-
	Yeung et al. [24]	48.9	44.0	36.0	26.4	17.1	-	-	-	-
	BSN [9]	-	-	53.5	45.0	36.9	28.4	20.0	-	-
	BMN [10]	-	-	56.0	47.4	38.8	29.7	20.5	-	-
Weak	Hide-and-seek [13]	36.4	27.8	19.5	12.7	6.8	-	-	-	-
	UntrimmedNets [12]	44.4	37.7	28.2	21.1	13.7	-	-	-	-
	STPN [17]	52.0	44.7	35.5	25.8	16.9	9.9	4.3	1.2	0.1
	AutoLoc [16]	-	-	35.8	29.0	21.2	13.4	5.8	-	-
	W-TALC [11]	49.0	42.8	32.0	26.0	18.8	-	6.2	-	-
	MAAN [28]	59.8	50.8	41.1	30.6	20.3	12.0	6.9	2.6	0.2
	CleanNet [31]	-	-	37.0	30.9	23.9	13.9	7.1	-	-
	3C-Net [27]	59.1	53.5	44.2	34.1	26.6	-	8.1	-	-
	Bas-Net [30]	58.2	52.3	44.6	36.0	27.0	18.6	10.4	**3.9**	**0.5**
	Nguyen et al. [29]	60.4	56.0	46.6	37.5	26.8	17.6	9.0	3.3	**0.4**
	DGAM [18]	60.0	54.2	46.8	38.2	28.8	19.8	11.4	3.6	**0.4**
	ours	**61.9**	**56.0**	**48.1**	**38.7**	**30.0**	**20.0**	**11.4**	**3.8**	0.2

TABLE II
THE RESULT OF THE EXPERIMENT ON THE ACTIVITYNET1.2 VALIDATION SET. ("OURS" IS OUR REALIZATION)

Supervision	Method	mAP@IoU										
		0.5	0.55	0.6	0.65	0.7	0.75	0.8	0.85	0.9	0.95	AVG
Full	SSN [6]	41.3	38.8	35.9	32.9	30.4	27.0	22.2	18.2	13.2	6.1	26.6
Weak	UntrimmedNets* [12]	7.4	6.1	5.2	4.5	3.9	3.2	2.5	1.8	1.2	0.7	3.6
	AutoLoc* [16]	27.3	24.9	22.5	19.9	17.5	15.1	13.0	10.0	6.8	3.3	16.0
	W-TALC [11]	37.0	33.5	30.4	25.7	14.6	12.7	10.0	7.0	4.2	1.5	18.0
	CleanNet [31]	37.1	33.4	29.9	26.7	23.4	20.3	17.2	13.9	9.2	5.0	21.6
	3C-Net [27]	37.2	-	-	-	23.7	-	-	-	9.2	-	21.7
	Liu et al. [14]	36.8	-	-	-	-	22.0	-	-	-	5.6	22.4
	Bas-Net [30]	38.5	-	-	-	-	24.2	-	-	-	5.6	24.3
	DGAM [18]	**41.0**	37.5	33.5	30.1	26.9	23.5	19.8	15.5	10.8	5.3	24.4
	ours	**40.6**	**37.5**	33.6	30.4	27.2	24.2	20.4	16.4	11.8	6.0	24.8

fully-supervised settings utilizes a training set with segment-level labels during training. However, in WTAL task, we use only video-level labels to train our model. Nevertheless, our algorithm is still able to achieve competitive performance. In addition, compared to other WTAL methods, the performance of our proposed BS-WTAL model is superior to those of the others for IoU less than 0.8, indicating that our method addresses the issue at hand by suppressing attention weights of background segments. However, for IoU values at 0.8 and 0.9, our method does not perform as well as Bas-Net [30], which proves that our method also suppresses the attention weights of the boundaries of the target action when suppressing the attention weights of the background regions. Although this network accurately determines the localization of the target action, it cannot precisely pinpoint the boundaries of the target action, which shows us the next direction of our research.

We also conduct experiments on the ActivityNet1.2 dataset to validate the performance of our BS-WTAL model. We compared the performance of our method with other state-of-the-art methods and the results are presented in Table II. According to the evaluation criteria, we calculate the mAPs under different IoU thresholds. Moreover, we compute the mean of the mAP values for thresholds between 0.5 and 0.95 in increments of 0.05, which is denoted as mAP@IoU. As far as the average mAP is concerned, our method outperforms all other WTAL methods. Simultaneously, our method is competitive compared to the fully-supervised method. Note that * indicates that the corresponding models use a weaker feature extractor to extract features than others.

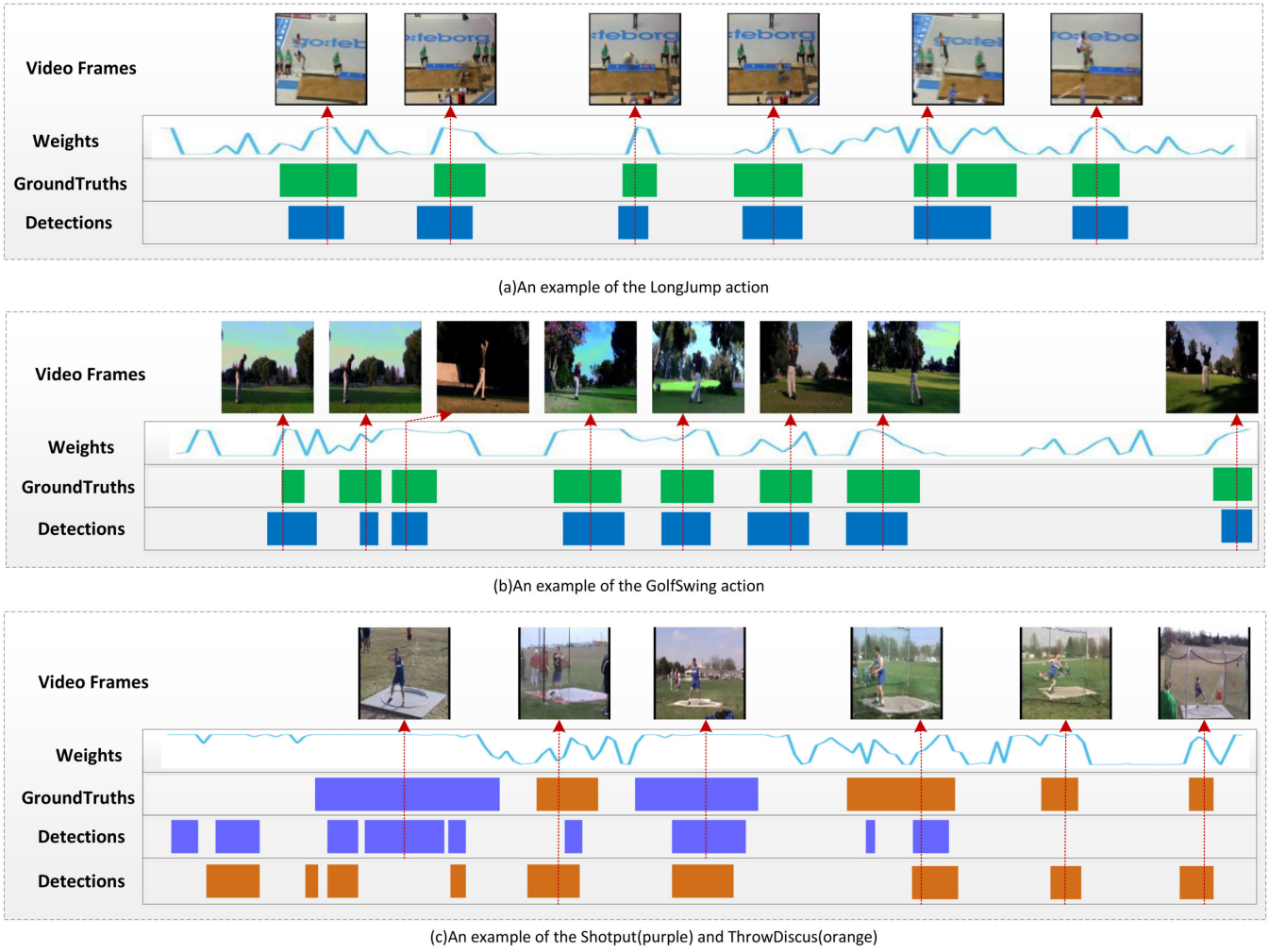

Fig. 5. Qualitative results on THUMOS14.

TABLE III
ABLATION STUDY OF THE EFFECTIVENESS OF OUR MODEL.

loss				mAP@0.5	
L_{fg}	L_{bg}	L_{guide}	L_{re}	DGAM [18]	ours
√				21.5	24.9
√	√			24.8	27.0
√	√	√		26.7	28.5
√	√	√	√	28.8	30.0

D. Ablation Study

To investigate the impact of each component of BS-WTAL model to the action localization performance, we design an ablation study on the optimization objectives of our proposed model and execute it on the THUMOS14 dataset. The baseline is set to the main pipeline with only foreground loss L_{fg}. We add background loss L_{bg} to learn modeling the background to distinguish the foreground and background better. Next, we utilize the self-guide regulation loss L_{guide} to further optimize attention. Our model also uses L_{re}, which is related to the generative attention module. We can optimize L_{CVAE} while optimizing L_{re}. Simultaneously, we also analyze the experimental results with DGAM [18] under the same conditions.

The results of the ablation study are summarized in Table III. The mAP values are reported under IoU thresholds of 0.5. The loss of foreground classification is used to model the foreground, so that the model can recognize the action instance. The loss of background classification is used to model the background, so that the model can accurately separate the action instance from the background. As a result, the performance is improved by 2.1%. Self-guide regularization is an additional loss function to further optimize attention λ , which leads to a 1.5% mAP improvement. Our BS-WTAL model finally achieved a 30.0% performance gain owing to the application of the loss function L_{re}. Moreover, our model is superior to DGAM [18], which proves that the filtering module plays a crucial role in improving the localization performance.

E. Qualitative results

Fig. 5 illustrates a comparison between the localization results of several actions and ground truths. These three examples are from the THUMOS14 dataset. Fig. 5(a) presents a sample with numerous action instances along with their attention weights and our predicted results. Before and after the target action, there are many actions, such as athletes running and waving to the audience, which are similar to the target action in their appearances. Nevertheless, our algorithm can still pinpoint the temporal boundaries of these target actions successfully. In Fig. 5(b), many incomplete

actions in the video are depicted, such as the beginning of the target action, which are parts of the target action. It is challenging to exclude these regions during inference, but our model can achieve good performance. In Fig. 5(c), two similar actions occurring at the same time in one video, and they are so similar in motion and appearance, that it is very challenging to localize the action regions. Despite these challenges, our method can accurately localize these temporal actions; moreover, it can accurately classify these actions.

V. Conclusion

We addressed the issue of many background regions that are similar to the target actions tending to be recognized as actions. A BS-WTAL model was developed and validated on two large-scale benchmarks, THUMOS14 and ActivityNet1.2. Our algorithm defined the filtering module for background suppression, the classification module for action classification, and the generative attention module for segment-wise representation modeling. Our model experimentally outperformed the state-of-the-art methods under weakly-supervised settings and achieved competitive performance under a fully-supervised setting.

References

[1] M. A. Aslam, C. Xue, M. Liu, K. Wang, and D. Cui, "Classification and prediction of gastric cancer from saliva diagnosis using artificial neural network." *Engineering Letters*, vol. 29, no. 1, pp. 10–24, 2021.

[2] R. Sarno, S. I. Sabilla, and D. R. Wijaya, "Electronic nose for detecting multilevel diabetes using optimized deep neural network." *Engineering Letters*, vol. 28, no. 1, pp. 31–42, 2020.

[3] Y. Zhong, L. Luo, X. Wang, and J. Yang, "Multi-factor stock selection model based on machine learning." *Engineering Letters*, vol. 29, no. 1, pp. 177–182.

[4] K. Soomro, H. Idrees, and M. Shah, "Action localization in videos through context walk," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 3280–3288.

[5] Z. Shou, D. Wang, and S.-F. Chang, "Temporal action localization in untrimmed videos via multi-stage cnns," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1049–1058.

[6] Y. Zhao, Y. Xiong, L. Wang, Z. Wu, X. Tang, and D. Lin, "Temporal action detection with structured segment networks," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2914–2923.

[7] Z. Shou, J. Chan, A. Zareian, K. Miyazawa, and S.-F. Chang, "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*, 2017, pp. 5734–5743.

[8] H. Xu, A. Das, and K. Saenko, "R-c3d: Region convolutional 3d network for temporal activity detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5783–5792.

[9] T. Lin, X. Zhao, H. Su, C. Wang, and M. Yang, "Bsn: Boundary sensitive network for temporal action proposal generation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.

[10] T. Lin, X. Liu, X. Li, E. Ding, and S. Wen, "Bmn: Boundary-matching network for temporal action proposal generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3889–3898.

[11] S. Paul, S. Roy, and A. K. Roy-Chowdhury, "W-talc: Weakly-supervised temporal activity localization and classification," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 563–579.

[12] L. Wang, Y. Xiong, D. Lin, and L. Van Gool, "Untrimmednets for weakly supervised action recognition and detection," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2017, pp. 4325–4334.

[13] K. K. Singh and Y. J. Lee, "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, 2017, pp. 3544–3553.

[14] D. Liu, T. Jiang, and Y. Wang, "Completeness modeling and context separation for weakly supervised temporal action localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1298–1307.

[15] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2921–2929.

[16] Z. Shou, H. Gao, L. Zhang, K. Miyazawa, and S.-F. Chang, "Autoloc: Weakly-supervised temporal action localization in untrimmed videos," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 154–171.

[17] P. Nguyen, T. Liu, G. Prasad, and B. Han, "Weakly supervised action localization by sparse temporal pooling network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6752–6761.

[18] B. Shi, Q. Dai, Y. Mu, and J. Wang, "Weakly-supervised action localization by generative attention modeling," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1009–1019.

[19] H. Idrees, A. R. Zamir, Y.-G. Jiang, A. Gorban, I. Laptev, R. Sukthankar, and M. Shah, "The thumos challenge on action recognition for videos "in the wild"," *Computer Vision and Image Understanding*, vol. 155, pp. 1–23, 2017.

[20] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles, "Activitynet: A large-scale video benchmark for human activity understanding," in *Proceedings of the ieee conference on computer vision and pattern recognition*, 2015, pp. 961–970.

[21] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 580–587.

[22] R. Girshick, "Fast r-cnn," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1440–1448.

[23] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, pp. 91–99, 2015.

[24] S. Yeung, O. Russakovsky, G. Mori, and L. Fei-Fei, "End-to-end learning of action detection from frame glimpses in videos," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2678–2687.

[25] T. Lin, X. Zhao, and Z. Shou, "Single shot temporal action detection," in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 988–996.

[26] C. Sun, S. Shetty, R. Sukthankar, and R. Nevatia, "Temporal localization of fine-grained actions in videos by domain transfer from web images," in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 371–380.

[27] S. Narayan, H. Cholakkal, F. S. Khan, and L. Shao, "3c-net: Category count and center loss for weakly-supervised action localization," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 8679–8687.

[28] Y. Yuan, Y. Lyu, X. Shen, I. W. Tsang, and D.-Y. Yeung, "Marginalized average attentional network for weakly-supervised learning," *arXiv preprint arXiv:1905.08586*, 2019.

[29] P. X. Nguyen, D. Ramanan, and C. C. Fowlkes, "Weakly-supervised action localization with background modeling," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 5502–5511.

[30] P. Lee, Y. Uh, and H. Byun, "Background suppression network for weakly-supervised temporal action localization," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, 2020, pp. 11 320–11 327.

[31] Z. Liu, L. Wang, Q. Zhang, Z. Gao, Z. Niu, N. Zheng, and G. Hua, "Weakly supervised temporal action localization through contrast based evaluation networks," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 3899–3908.

[32] G. Gong, X. Wang, Y. Mu, and Q. Tian, "Learning temporal co-attention models for unsupervised video action localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9819–9828.

[33] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6299–6308.

[34] J. S. Pérez, E. Meinhardt-Llopis, and G. Facciolo, "Tv-l1 optical flow estimation," *Image Processing On Line*, vol. 2013, pp. 137–150, 2013.

[35] W. Kay, J. Carreira, K. Simonyan, B. Zhang, C. Hillier, S. Vijayanarasimhan, F. Viola, T. Green, T. Back, P. Natsev *et al.*, "The kinetics human action video dataset," *arXiv preprint arXiv:1705.06950*, 2017.

[36] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.