The Thirty-Fourth AAAI Conference on Artificial Intelligence (AAAI-20)

Fast Learning of Temporal Action Proposal via Dense Boundary Generator

Chuming Lin,[*1] **Jian Li,**[*1] **Yabiao Wang,**[1] **Ying Tai,**[1]
Donghao Luo,[1] **Zhipeng Cui,**[1] **Chengjie Wang,**[1] **Jilin Li,**[1] **Feiyue Huang,**[1] **Rongrong Ji**[2]
[1]Youtu Lab, Tencent, [2]Xiamen University, China
{chuminglin, swordli, caseywang, yingtai, michaelluo, zhipengcui, jasoncjwang, jerolinli, garyhuang}@tencent.com
rrji@xmu.edu.cn
https://github.com/TencentYoutuResearch/ActionDetection-DBG

Abstract

Generating temporal action proposals remains a very challenging problem, where the main issue lies in predicting precise temporal proposal boundaries and reliable action confidence in long and untrimmed real-world videos. In this paper, we propose an efficient and unified framework to generate temporal action proposals named Dense Boundary Generator (DBG), which draws inspiration from boundary-sensitive methods and implements boundary classification and action completeness regression for densely distributed proposals. In particular, the DBG consists of two modules: Temporal boundary classification (TBC) and Action-aware completeness regression (ACR). The TBC aims to provide two temporal boundary confidence maps by low-level two-stream features, while the ACR is designed to generate an action completeness score map by high-level action-aware features. Moreover, we introduce a dual stream BaseNet (DSB) to encode RGB and optical flow information, which helps to capture discriminative boundary and actionness features. Extensive experiments on popular benchmarks ActivityNet-1.3 and THUMOS14 demonstrate the superiority of DBG over the state-of-the-art proposal generator (e.g., MGG and BMN).

Introduction

Generating temporal action proposals in video is a fundamental task, which serves as a crucial step for various tasks, like action detection and video analysis. In an optimal case, such proposals should well predict action intervals, with precise temporal boundaries and reliable confidence in untrimmed videos. Despite the extensive endeavors (Lin et al. 2018; 2019; Liu et al. 2019), temporal action proposal generation retains as an open problem, especially when facing action duration variability, activity complexity, blurred boundary, camera motion, background clutter and viewpoint changes in real-world scenarios.

Previous works in temporal action proposals can be roughly divided into two categories: *anchor based* (Buch et al. 2017; Heilbron, Niebles, and Ghanem 2016; Gao et al. 2017; Shou, Wang, and Chang 2016) and *boundary based* (Zhao et al. 2017a; Lin et al. 2018; 2019). Anchor-based methods design a set of anchors at different scale for each video segment, which are regularly distributed over the video sequence. These candidate anchors are then evaluated by a binary classifier. However, anchor-based methods can not predict precise boundaries and are not flexible to cover multi-duration actions.

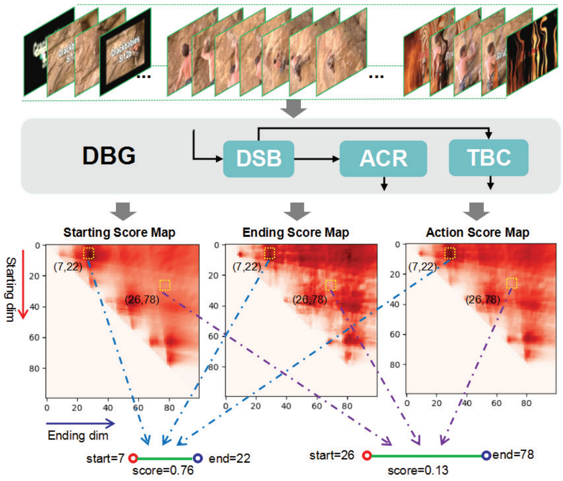
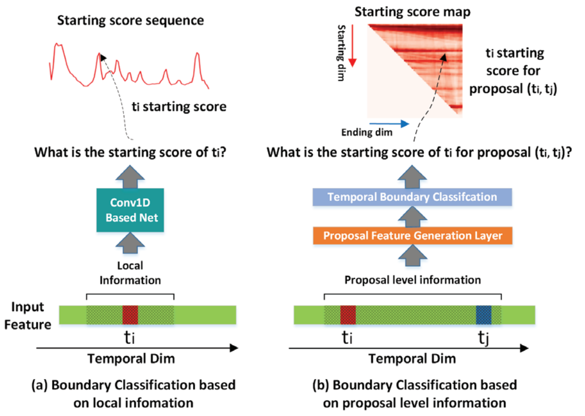


Figure 1: Overview of our proposed method. Given an untrimmed video, DBG densely evaluates all proposals by producing simultaneously three score maps: starting confidence score map, ending confidence score map and action completeness score map.

Boundary-based methods evaluate each temporal location over the video sequence. Such local information helps to generate proposals with more precise boundaries and more flexible durations. As one of the pioneering works (Zhao et al. 2017a) groups continuous high-score regions as proposal by actionness scores. (Lin et al. 2018) adopts a two-stage strategy to locate locally temporal boundaries with high probabilities, and then evaluate global confidences of candidate proposals generated by these boundaries. To explore the rich context for evaluating all proposals, (Lin et al.

[*]indicates equal contributions. This work was done when Chuming Lin was an intern at Tencent Youtu Lab. Yabiao Wang is the corresponding author.


Starting score sequence
ti starting score
What is the starting score of ti?
Conv1D Based Net
Local Information
Input Feature
ti
Temporal Dim
Starting score map
Starting dim
ti starting score for proposal (ti, tj)
Ending dim
What is the starting score of ti for proposal (ti, tj)?
Temporal Boundary Classifcation
Proposal Feature Generation Layer
Proposal level information
ti
tj
Temporal Dim

(a) Boundary Classification based on local infomation

(b) Boundary Classification based on proposal level information

Figure 2: Boundary prediction comparison of (a) local information based and (b) global proposal information based methods.

2019) propose a boundary-matching mechanism for the confidence evaluation of proposals in an end-to-end pipeline. However, it drops actionness information and only adopts the boundary matching to capture low-level features, which can not handle complex activities and clutter background. Besides, different from our method shown in Fig. 1, it employs the same methods of (Lin et al. 2018) to generate boundary probability sequence instead of map, which lacks a global scope for action instances with blurred boundaries and variable temporal durations. Fig. 2 illustrates the difference between local information and our global proposal information for boundary prediction.

To address the aforementioned drawbacks, we propose dense boundary generator (DBG) to employ global proposal features to predict the boundary map, and explore action-aware features for action completeness analysis. In our framework, a dual stream BaseNet (DSB) takes spatial and temporal video representation as input to exploit the rich local behaviors within the video sequence, which is supervised via actionness classification loss. DSB generates two types of features: Low-level dual stream feature and high-level actionness score feature. In addition, a proposal feature generation (PFG) layer is designed to transfer these two types of sequence features into a matrix-like feature. And an action-aware completeness regression (ACR) module is designed to input the actionness score feature to generate a reliable completeness score map. Finally, a temporal boundary classification (TBC) module is designed to produce temporal boundary score maps based on dual stream feature. These three score maps will be combined to generate proposals.

The main contributions of this paper are summarized as:

- We propose a fast and unified dense boundary generator (DBG) for temporal action proposal, which evaluates dense boundary confidence maps for all proposals.
- We introduce auxiliary supervision via actionness classification to effectively facilitate action-aware feature for the action-aware completeness regression.
- We design an efficient proposal feature generation layer to capture global proposal features for subsequent regression and classification modules.
- Experiments conducted on popular benchmarks like ActivityNet-1.3 (Heilbron et al. 2015) and THUMOS14 (Idrees et al. 2017) demonstrate the superiority of our network over the state-of-the-art methods.

Related Work

Action recognition. Early methods for video action recognition mainly relied on hand-crafted features such as HOF, HOG and MBH. Recent advances resort to deep convolutional networks to promote recognition accuracy. These networks can be divided into two patterns: *Two-stream networks* (Feichtenhofer, Pinz, and Zisserman 2016; Simonyan and Zisserman 2014; Wang et al. 2015; 2016), and *3D networks* (Tran et al. 2015; Qiu, Yao, and Mei 2017; Carreira and Zisserman 2017). Two-stream networks explore video appearance by passing RGB image and stacked optical flow through ConvNet pretrained on ImageNet separately. Instead, 3D methods directly create hierarchical representations with spatio-temporal filters.

Temporal action proposal. Temporal action proposal aims to detect action instances with temporal boundaries and confidence in untrimmed videos. *Anchor-based methods* generate proposals by designing a set of multi-scale anchors with regular temporal interval. The work in (Shou, Wang, and Chang 2016) adopts C3D network (Tran et al. 2015) as the binary classifier for anchor evaluation. (Heilbron, Niebles, and Ghanem 2016) proposes a sparse learning framework for scoring temporal anchors. (Gao et al. 2017) proposes to apply temporal regression to adjust the action boundaries. *Boundary-based methods* evaluate each temporal location in video. (Zhao et al. 2017a) groups continuous high-score region to generate proposals by temporal watershed algorithm. (Lin et al. 2018) locates locally temporal boundaries with high probabilities and evaluate global confidences of candidate proposals generated by these boundaries. (Lin et al. 2019) proposes a boundary-matching mechanism for confidence evaluation of densely distributed proposals in an end-to-end pipeline. MGG (Liu et al. 2019) combines anchor based method and boundary based method to accurately generate temporal action proposal.

Temporal action detection. The temporal action detection includes generating temporal proposal generation and recognizing actions, which can be divided into two patterns, *i.e.*, one-stage (Lin, Zhao, and Shou 2017; Long et al. 2019) and two-stage (Shou, Wang, and Chang 2016; Gao, Yang, and Nevatia 2017; Zhao et al. 2017b; Xu, Das, and Saenko 2017; Chao et al. 2018). The *two-stage* method first generates candidate proposals, and then classifies these proposals. (Chao et al. 2018) improves two-stage temporal action detection by addressing both receptive field alignment and context feature extraction. For *one-stage* method, (Lin, Zhao, and Shou 2017) skips the proposal generation via directly detecting action instances in untrimmed video. (Long et al. 2019) introduces Gaussian kernels to dynamically optimize temporal scale of each action proposal.

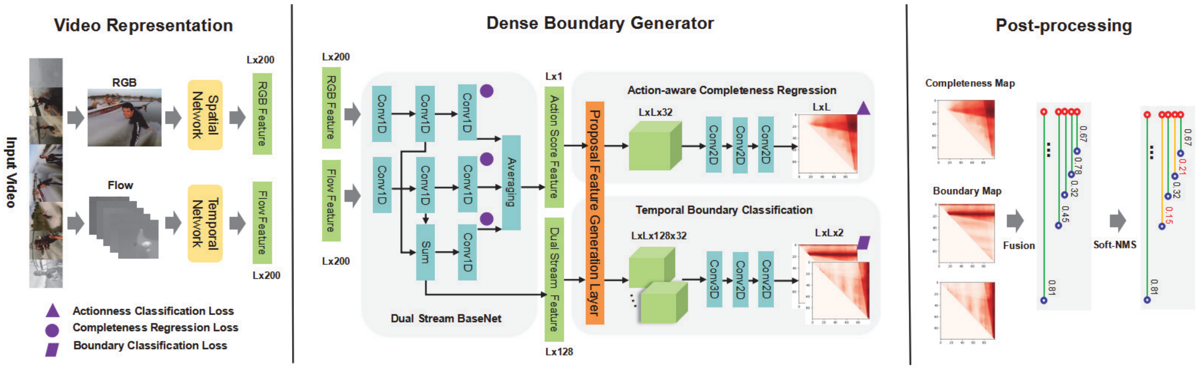


Figure 3: *(a)Video Representation:* Spatial & temporal network is used to encode video visual contents. *(b)Dense Boundary Generator:* It contains Dual Stream BaseNet, Action-aware Completeness Regression branch and Temporal Boundary Classification branch. *(c)Post-processing:* In this step, three score maps are fused and Soft-NMS is leveraged to generate proposals.

Approach

Suppose there are a set of untrimmed video frames $F = \{f_t\}_{t=1}^{l_f}$, where f_t is the t-th RGB frame and l_f is the number of frames in the video V. The annotation of V can be denoted by a set of action instances $\psi_g = \{\varphi_i = (ts_i, te_i)\}_{i=1}^{N_g}$, where N_g is the number of ground truth action instances in video V, and ts_i, te_i are starting and ending points of action instance φ_i. The generation of temporal action proposal aims to predict proposals $\psi_p = \{\varphi_i = (ts_i, te_i, p_i)\}_{i=1}^{N_p}$ to cover ψ_g with high recall and overlap, where p_i is the confidence of φ_i.

Pipeline of our framework

Fig. 3 illustrates the proposed pipeline. In the phrase of video representation, spatial and temporal network are employed to encode video visual contents. The output scores of the two-stream network are used as RGB and flow features separately, which are fed into our dense boundary generator (DBG). DBG contains three modules: dual stream BaseNet (DSB), action-aware completeness regression (ACR) and temporal boundary classification (TBC). DSB can be regarded as a DBG backbone to exploit the rich local behaviors within the video sequence. DSB will generate two types of features: low-level dual stream feature and high-level actionness score feature. Actionness score feature is learned under auxiliary supervision of actionness classification loss, while dual stream feature is generated by late fusion of RGB and flow information. The proposal feature generation (PFG) layer transfers these two types of sequence features into a matrix-like feature. ACR will take actionness score features as input to produce an action completeness score map for dense proposals. TBC will produce temporal boundary confidence maps based on the dual stream features. ACR and TBC are trained by completeness regression loss and binary classification loss simultaneously. At last, the post-processing step generates dense proposals with boundaries and confidence by score map fusion and Soft-NMS.

Video Representation

To explore video appearance and motion information separately, we encode the raw video sequence to generate video representation by (Wang et al. 2016), which contains spatial network for single RGB frame and temporal network for stacked optical flow field. We partition the untrimmed video frame sequence $F = \{f_t\}_{t=1}^{l_f}$ into snippets sequence $S = \{s_t\}_{t=1}^{l_s}$ by a regular frame interval δ, where $l_s = l_f/\delta$. A snippet s_t contains 1 RGB frame and 5 stacked optical flow field frames. We use output scores in the top layer of both spatial and temporal network to formulate the RGB feature S_t and flow feature T_t. Thus, a video can be represented by a two-stream feature sequence $\{S_t, T_t\}_{t=1}^{l_s}$. We set $l_s = L$ to keep the length of two-stream video feature sequence a constant.

Dense Boundary Generator

Dual stream BaseNet. The DBG backbone receives the spatial and temporal video feature sequences as input, and outputs actionness score feature and dual stream feature for ACR and TBC separately. DSB serves as the backbone of our framework, which adopts several one-dimensional temporal convolutional layers to explore local semantic information for capturing discriminative boundary and actionness features. As show in Tab. 1, we use two stacked one-dimensional convolutional layers to exploit spatial and temporal video representation respectively, written by $sf = F_{conv12}(F_{conv11}(S))$, $tf = F_{conv22}(F_{conv21}(T))$. Then, following (Li, Qian, and Yang 2017), we fuse sf, tf by element-wise sum to construct low-level dual stream feature, denoted by $dsf = F_{sum}(sf, tf)$. Three convolutional layers will be adopted for sf, tf, dsf separately to generate three actionness feature sequences $P^a = (F_{conv13}(sf), F_{conv23}(tf), F_{conv33}(dsf))$. In training, we use three auxiliary actionness binary classification loss to supervise P^a. In inference, three actionness feature sequence are averaged to generate high-level actionness score feature, which can be defined by $asf = F_{avg}(P^a)$.

Proposal feature generation layer. The PFG layer is an ef-

Table 1: The detail design of dual stream BaseNet (DSB), action-aware completeness regression (ACR) module and temporal boundary classification (TBC) module.

DSB					
layer	kernel	output	layer	kernel	output
Conv1D$_{11}$	3	L×256	Conv1D$_{21}$	3	L× 256
Conv1D$_{12}$	3	L×128	Conv1D$_{22}$	3	L× 128
Sum	_	L×128	Conv1D$_{33}$	1	L×1
Conv1D$_{13}$	1	L×1	Conv1D$_{23}$	1	L× 1
Averaging	_	L×1			

ACR			TBC		
layer	kernel	output	layer	kernel	output
PFG	_	L×L×32	PFG	_	L×L×32×128
Conv2D$_{11}$	1×1	L×L×256	Conv3D$_{21}$	1×1×32	L×L×512
Conv2D$_{12}$	1×1	L×L×256	Conv2D$_{22}$	1×1	L×L×256
Conv2D$_{13}$	1×1	L×L×1	Conv2D$_{23}$	1×1	L×L×2

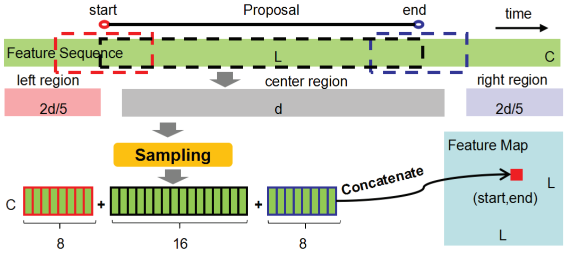


Figure 4: Details of the proposal feature generation layer. Given a feature sequence, we concatenate the sampled feature regions to construct proposal context feature map.

ficient and differentiable layer that is able to generate temporal context feature for each proposal and make our framework be end-to-end trainable. For an arbitrary input feature f^{in} whose shape is $L \times C$, the PFG layer is able to produce the proposal feature tensor whose shape is $L \times L \times N \times C$, which contains $L \times L$ proposal features f^p whose size is $N \times C$.

Fig. 4 shows the detail of our PFG layer. First, for each candidate proposal $\varphi = (t_s, t_e)$, we sample N_l locations from the left region $r^s = [t_s - d_g/k, t_s + d_g/k]$, N_c locations from the center region $r^a = [t_s, t_e]$ and N_r locations from the right region $r^e = [t_e - d_g/k, t_e + d_g/k]$ by linear interpolation, respectively, where $d_g = t_e - t_s, k = 5$ and $N = N_l + N_c + N_r$. Then, with these sampling locations, we concatenate the corresponding temporal location features to produce the context proposal feature. Therefore, it is obvious to generate each proposal feature $f^p_{t_s,t_e}$ from the input feature f^{in} through the following formula:

$$f^p_{t_s,t_e,n,c} = w_l f^{in}_{t_l,c} + w_r f^{in}_{t_r,c}, \tag{1}$$

where

$$t_l = \begin{cases} \lfloor t_s - \frac{d_g}{k} + \frac{2d_g}{k(N_l-1)} n \rfloor, & n < N_l, \\ \lfloor t_s + \frac{d_g}{N_c-1} n \rfloor, & N_l \leq n < N_l + N_c, \\ \lfloor t_e - \frac{d_g}{k} + \frac{2d_g}{k(N_r-1)} n \rfloor, & n \geq N_l + N_c, \end{cases} \tag{2}$$

$$w_l = \begin{cases} t_r - t_s + \frac{d_g}{k} - \frac{2d_g}{k(N_l-1)} n, & n < N_l, \\ t_r - t_s - \frac{d_g}{N_c-1} n, & N_l \leq n < N_l + N_c, \\ t_r - t_e + \frac{d_g}{k} - \frac{2d_g}{k(N_r-1)} n, & n \geq N_l + N_c, \end{cases} \tag{3}$$

$$t_r = 1 + t_l, w_r = 1 - w_l. \tag{4}$$

When calculating gradient for training PFG layer, $f^p_{t_s,t_e}$ is differentiable for f^{in}, and its differential formulas are:

$$\frac{\partial f^p_{t_s,t_e,n,c}}{\partial f^{in}_{t_l,c}} = w_l, \frac{\partial f^p_{t_s,t_e,n,c}}{\partial f^{in}_{t_r,c}} = w_r. \tag{5}$$

In our experiments, we set $N_l = N_r = 8$ and $N_c = 16$, thus $N = 32$. Note that if $t_s \geq t_e$, then the proposal feature $f^p_{t_s,t_e}$ will be zero.

Action-aware completeness regression. The ACR branch receives actionness score feature as input and outputs action completeness map P^c to estimate the overlap between candidate proposals and ground truth action instances. In ACR, we employ the PFG layer and several two-dimensional convolutional layers for each proposal to explore semantic information in the global proposal level. As show in Tab. 1, the PFG layer can transfer temporal actionness score features asf to three-dimensional proposal feature tensors, which are fed into multi two-dimensional convolutional layers to generate $L \times L$ action completeness maps, denoted as $P^c = F_{(Conv11,Conv12,Conv13)}(F_{PFG}(asf))$. For each location or proposal in the action completeness map, we use a smooth L1 regression loss to supervise P^c to generate reliable action completeness score.

Temporal boundary classification. The TBC branch receives dual stream feature as input and outputs boundary confidence map $P^{s,e}$ to estimate the starting and ending probabilities for dense candidate proposals. Similar with ACR, TBC includes the PFG layer, a three-dimensional convolutional layer and several two-dimensional convolutional layers. As show in Tab. 1, dual stream features dsf from DSB is transfered by the PFG layer to four-dimensional proposal tensors. Multi convolutional layers are stacked to generate $L \times L \times 2$ boundary confidence maps written by $P^{s,e} = F_{(Conv21,Conv22,Conv23)}(F_{PFG}(dsf))$. For each location or proposal in the boundary confidence map, we use the binary classification loss to supervise $P^{s,e}$ to predict precise temporal boundaries.

Training and Inference

To jointly learn action completeness map and boundary confidence map, a unified multi-task loss is further proposed. In inference, with three score maps generated by DBG, a score fusion strategy and Soft-NMS can generate dense proposals with confidence.

Label and Loss

Given the annotation $\psi_g = \{\varphi_i = (ts_i, te_i)\}_{i=1}^{N_g}$ of a video V, we compose actionness label g^a for auxiliary DSB actionness classification loss, boundary label g^s, g^e for TBC boundary classification loss, and action completeness label g^c for ACR completeness regression loss. For a given ground truth action instance $\varphi = (t_s, t_e)$, we define its action region as $r^a_g = [t_s, t_e]$, starting region as $r_g^s = [t_s - d_t, t_s + d_t]$ and ending region as $r_g^e = [t_e - d_t, t_e + d_t]$, where d_t is the two temporal locations intervals.

DSB actionness classification. For each temporal location i within actionness score feature sequence P^a, we denote its region as $r_i = [i - d_t/2, i + d_t/2]$. Then, we calculate maximum overlap ratio IoR for r_i with r_g^a, where IoR is defined as the overlap ratio with ground truth proportional to the duration of this region. If this ratio is bigger than an overlap threshold 0.5, we set the actionness label as $g_i^a = 1$, else we have $g_i^a = 0$. With three actionness probability sequences P^a, we can construct DSB actionness classification loss using binary logistic regression:

$$\mathcal{L}^a_{DSB} = \frac{1}{3L}\sum_{j=1}^{3}\sum_{i=1}^{L} g_i^a log(p_i^{a_j}) + (1 - g_i^a) log(1 - p_i^{a_j}). \tag{6}$$

TBC boundary classification. For each location (i, j) within starting confidence map P^s or ending confidence map P^e, we denote its starting region as $r_{i,j}^s = [i - d_t/2, i + d_t/2]$ and its ending region as $r_{i,j}^e = [j - d_t/2, j + d_t/2]$. Similar with above actionness label, we calculate the starting label $g_{i,j}^s$ for $r_{i,j}^s$ with r_g^s and the ending label $g_{i,j}^e$ for $r_{i,j}^e$ with r_g^e. We also adopt binary logistic regression to construct the classification loss function of TBC for the starting and ending separately:

$$\mathcal{L}^s_{TBC} = \frac{1}{L^2}\sum_{i=1}^{L}\sum_{j=1}^{L} g_{i,j}^s log(p_{i,j}^s) + (1 - g_{i,j}^s) log(1 - p_{i,j}^s), \tag{7}$$

$$\mathcal{L}^e_{TBC} = \frac{1}{L^2}\sum_{i=1}^{L}\sum_{j=1}^{L} g_{i,j}^e log(p_{i,j}^e) + (1 - g_{i,j}^e) log(1 - p_{i,j}^e). \tag{8}$$

ACR completeness regression. For each location or proposal (i, j) within action completeness map P^c , we denote its region as $r_{i,j} = [i, j]$. For $r_{i,j}$, We caculate the maximum Intersection-over-Union (IoU) with all r_g^c to generate completeness label $g_{i,j}^c$. With the action completeness map P^c from ACR, we simply adopt smooth L1 loss to construct the ACR loss function:

$$\mathcal{L}^c_{ACR} = \frac{1}{L^2}\sum_{i=1}^{L}\sum_{j=1}^{L} smooth_{L1}(p_{i,j}^c - g_{i,j}^c). \tag{9}$$

Following BSN, we balance the effect of positive and negative samples for the above two classification losses during training. For regression loss, we randomly sample the proposals to ensure the ratio of proposals in different IoU intervals [0,0.2],[0.2,0.6] and [0.6,1] that satisfies 2:1:1. We use the above three-task loss function to define the training objective of our DGB as:

$$\mathcal{L}_{DBG} = \lambda\mathcal{L}^a_{DSB} + \mathcal{L}^s_{TBC} + \mathcal{L}^e_{TBC} + \mathcal{L}^c_{ACR}, \tag{10}$$

where weight term λ is set to 2 to effectively facilitate the actionness score features.

Prediction and Post-processing

In inference, different from BSN, three actionness probability sequences from DSB will not participate in computation of the final proposal results. Based on three score maps from ACR and TBC, we adopt post-processing to generate dense proposals with confidence.

Score map fusion. To make boundaries smooth and robustness, we average boundary probability of these proposals sharing the same starting or ending location. For starting and ending score map P^s, P^e from TBC, we compute each location or proposal boundary probability $P_{i,j}^s$ and $P_{i,j}^e$ by:

$$P_{i,j}^s = \frac{1}{L}\sum_{k=1}^{L} P_{i,k}^s, P_{i,j}^e = \frac{1}{L}\sum_{k=1}^{L} P_{k,j}^e. \tag{11}$$

For each proposal (i, j) whose starting and ending locations are i and j, we fuse boundary probability with completeness score map P^c to generate the final confidence score $P_{i,j}$:

$$P_{i,j} = P_{i,j}^c \times P_{i,j}^s \times P_{i,j}^e. \tag{12}$$

For the fact that the starting location is in front of the ending location, we consider the upper right part of the score map, and then get the dense candidate proposals set as $\psi_p = \{\varphi_i = (i, j, P_{i,j})\}_{i=1,j=1}^{i<=j<=L}$.

Proposal retrieving. The above proposal generation will produce dense and redundant proposals around ground truth action instances. Subsequently, we need to suppress redundant proposals by Soft-NMS, which is a non-maximum suppression by a score decaying function. After Soft-NMS step, we employ a confidence threshold to get the final sparse candidate proposals set as $\psi_p = \{\varphi_i = (s_i, e_i, P_i)\}_{i=1}^{N}$, where N is the number of retrieved proposals.

Experiments

Evaluation Datasets

ActivityNet-1.3. It is a large-scale dataset containing 19,994 videos with 200 activity classes for action recognition, temporal proposal generation and detection. The quantity ratio of training, validation and testing sets satisfies 2:1:1.

THUMOS14. This dataset has 1,010 validation videos and 1,574 testing videos with 20 classes. For the action proposal or detection task, there are 200 validation videos and 212 testing videos labeled with temporal annotations. We train our model on the validation set and evaluate on the test set.

Implementation Details

For video representation, we adopt the same two-stream network (Xiong et al. 2016) pretrained on ActivityNet-1.3 and parameter setting by following (Lin et al. 2019; 2018) to encode video features. For ActivityNet-1.3, we resize video feature sequence by linear interpolation and set $L = 100$. For THUMOS14, we slide the window on video feature sequence with $overlap = 0.5$ and $L = 128$. When training DBG, we use Adam for optimization. The batch size is set to 16. The learning rate is set to 10^{-3} for the first 10 epochs, and we decay it to 10^{-4} for another 2 epochs. For Soft-NMS, we set the threshold 0.8 on the ActivityNet-1.3 and 0.65 on the THUMOS14. ϵ in Gaussian function is set to 0.75 on both temporal proposal generation datasets.

Table 2: Comparison between our approach and other state-of-the-art temporal action generation approaches on validation set and test set of ActivityNet-1.3 dataset in terms of AR@AN and AUC.

Method	TCN	MSRA	Prop-SSAD	CTAP	BSN	MGG	BMN	Ours
AR@100 (val)	-	-	73.01	73.17	74.16	74.54	75.01	**76.65**
AUC (val)	59.58	63.12	64.40	65.72	66.17	66.43	67.10	**68.23**
AUC (test)	61.56	64.18	64.80	-	66.26	66.47	67.19	**68.57**

Table 3: Comparison between DBG with other state-of-the-art methods on THUMOS14 in terms of AR@AN.

Feature	Method	@50	@100	@200	@500	@1000
C3D	SCNN-prop	17.22	26.17	37.01	51.57	58.20
C3D	SST	19.90	28.36	37.90	51.58	60.27
C3D	TURN	19.63	27.96	38.34	53.52	60.75
C3D	MGG	29.11	36.31	44.32	54.95	60.98
C3D	BSN+NMS	27.19	35.38	43.61	53.77	59.50
C3D	BSN+SNMS	29.58	37.38	45.55	54.67	59.48
C3D	BMN+NMS	29.04	37.72	46.79	56.07	60.96
C3D	BMN+SNMS	**32.73**	40.68	47.86	56.42	60.44
C3D	Ours+NMS	32.55	**41.07**	**48.83**	**57.58**	59.55
C3D	Ours+SNMS	30.55	38.82	46.59	56.42	**62.17**
2Stream	TAG	18.55	29.00	39.61	-	-
Flow	TURN	21.86	31.89	43.02	57.63	64.17
2Stream	CTAP	32.49	42.61	51.97	-	-
2Stream	MGG	39.93	47.75	54.65	61.36	64.06
2Stream	BSN+NMS	35.41	43.55	52.23	61.35	65.10
2Stream	BSN+SNMS	37.46	46.06	53.21	60.64	64.52
2Stream	BMN+NMS	37.15	46.75	54.84	62.19	65.22
2Stream	BMN+SNMS	39.36	47.72	54.70	62.07	65.49
2Stream	Ours+NMS	**40.89**	**49.24**	**55.76**	61.43	61.95
2Stream	Ours+SNMS	37.32	46.67	54.50	**62.21**	**66.40**

Table 4: Efficiency comparison among DBG and BMN and BSN in validation set of ActivityNet-1.3. *e2e* means the method is able to be trained end-to-end.

Method	*e2e*	AR@100	AUC	T_{pro}	T_{all}
BSN	×	74.16	66.17	0.624	0.629
BMN	✓	75.01	67.10	0.047	0.052
DBG	✓	**76.65**	**68.23**	**0.008**	**0.013**

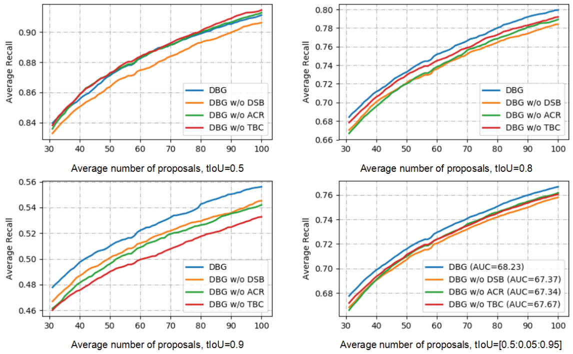


Figure 5: Ablation study of effectiveness of modules in DBG on validation set of ActivityNet-1.3 in terms of AR@AN curve.

Table 5: Performance analysis of PFG layer.

$N_l/N_c/N_r$	4/8/4	6/12/6	8/16/8	10/20/10	0/16/0	8/0/8
AR@10	57.22	**57.29**	**57.29**	57.09	55.74	56.85
AR@50	71.13	71.57	**71.59**	71.36	70.29	71.17
AR@100	76.14	76.27	**76.65**	76.50	75.53	76.13
AUC	67.91	68.14	**68.23**	68.11	66.94	67.83

Temporal Proposal Generation

To evaluate the proposal quality, we adopt different IoU thresholds to calculate the average recall (AR) with average number of proposals (AN). A set of IoU thresholds [0.5:0.05:0.95] is used on ActivityNet-1.3, while a set of IoU thresholds [0.5:0.05:1.0] is used on THUMOS14. For ActivityNet-1.3, area under the AR vs. AN curve (AUC) is also used as the evaluation metrics.

Comparison experiments. We further compare our DBG with other methods on the validation set of ActivityNet-1.3. Tab. 2 lists a set of proposal genearation methods including TCN (Dai et al. 2017), MSRA (Yao et al. 2017), Prop-SSAD (Lin, Zhao, and Shou 2017), CTAP (Gao, Chen, and Nevatia 2018), BSN (Lin et al. 2018), MGG (Liu et al. 2019) and BMN (Lin et al. 2019). Our method achieves state-of-the-art performance and improves AUC from 67.10% to 68.23%, which demonstrates that our DBG can achieve an overall performance promotion of action proposal generation. Especially, with multiple video representation networks and multi-scale video features, our ensemble DBG achieves **73.05%** AUC, which ranks top-1 on ActivityNet Challenge 2019 on temporal action proposals.

Tab. 3 compares proposal generation methods on the testing set of THUMOS14. To ensure a fair comparison, we adopt the same video feature and post-processing step. Tab. 3 shows that our method using C3D or two-stream video features outperforms other methods significantly when the proposal number is set within [50,100,200,500,1000].

We conduct a more detailed comparison on the validation set of ActivityNet-1.3 to evaluate the effectiveness and efficiency among BSN, BMN, and DBG. As shown in Tab. 4, for a 3-minute video processed on Nvidia GTX 1080Ti, our inference speed accelerates a lot. And our proposal feature generation is reduced from 47ms to 8ms, while the total inference time decreases to 13ms.

Ablation study. We further conduct detailed ablation study to evaluate different components of the proposed framework, including DSB, ACR, and TBC, include the following *DBG w/o DSB*: We discard DSB and feed concatenated spa-

Table 6: Generalization evalation on ActivityNet-1.3.

	Seen		Unseen	
Training Data	AR@100	AUC	AR@100	AUC
Seen+Unseen	73.30	66.57	67.23	64.59
Seen	72.95	66.23	**64.77**	**62.18**

Table 7: Action detection results on testing set of THUMOS14 in terms of *m*AP@*tIoU*.

Method	classifier	0.7	0.6	0.5	0.4	0.3
SST	SCNN-cls	-	-	23.0	-	-
TURN	SCNN-cls	7.7	14.6	25.6	33.2	44.1
BSN	SCNN-cls	15.0	22.4	29.4	36.6	43.1
MGG	SCNN-cls	15.8	23.6	29.9	37.8	44.9
BMN	SCNN-cls	17.0	24.5	32.2	40.2	45.7
Ours	SCNN-cls	**18.4**	**25.3**	**32.9**	**40.4**	**45.9**
SST	UNet	4.7	10.9	20.0	31.5	41.2
TURN	UNet	6.3	14.1	24.5	35.3	46.3
BSN	UNet	20.0	28.4	36.9	45.0	53.5
MGG	UNet	21.3	29.5	37.4	46.8	53.9
BMN	UNet	20.5	29.7	38.8	47.4	56.0
Ours	UNet	**21.7**	**30.2**	**39.8**	**49.4**	**57.8**

tial and temporal features into the BSN-like BaseNet.

DBG w/o ACR: We discard action-aware feature and auxiliary actionness classification loss, and adopt dual stream feature for action-aware completeness regression like TBC.

DBG w/o TBC: We discard the whole temporal boundary classification module, and instead predict boundary probability sequence like actionness feature sequence in DSB.

As illustrated in Fig. 5, the proposed DBG outperforms all its variants in terms of AUC with different IoU thresholds, which verifies the effectiveness of our contributions. The *DBG w/o ACR* results demonstrate that action-aware feature using auxiliary supervision is more helpful than dual stream feature for action completeness regression. The *DBG w/o TBC* results explain the remarkable superiority of dense boundary maps for all proposals. When the IoU threshold is strict and set to be 0.9 for evaluation, a large AUC gap between *DBG* (blue line) and *DBG w/o TBC* (red line) shows TBC can predict more precise boundaries. Fig. 6 shows more examples to demonstrate the effects of DBG on handling actions with various variations.

Analysis of PFG layer. To confirm the effect of the PFG layer, we conduct experiments to examine how different sampling locations within features affect proposal generation performance. As shown in Tab. 5, The experiments that sampling 8, 16, 8 locations from left region, center region and right region respectively within proposal features achieves the best performance. The *0/16/0* results indicate that context information around proposals are necessary for better performance on proposal generation. The *8/0/8* experiment that only adopting left or right local region features for TBC to predict starting or ending boundary confidence map shows the importance of the global proposal information.

Generalizability. Following BMN, we choose two different action subsets on ActivityNet-1.3 for generalizability analysis: "Sports, Exercise, and Recreation" and "Socializing, Relaxing, and Leisure" as seen and unseen subsets, respectively. We employ I3D network (Carreira and Zisserman 2017) pretrained on Kinetics-400 for video representation. Tab. 6 shows the slight AUC drop when testing the unseen subset, which clearly explains that DBG works well to generate high-quality proposals for unseen actions.

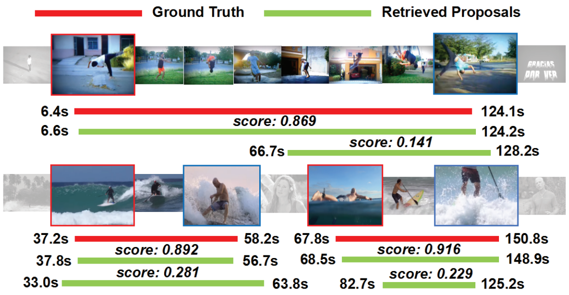


Figure 6: Visualization examples of proposals generated by DBG on ActivityNet-1.3 dataset.

Temporal Proposal Detection

To evaluate the proposal quality of DBG, we put proposals in a temporal action detection framework. We adopt mean Average Precision (*m*AP) to evaluates the temporal action detection task. We adopt a set of IoU thresholds {0.3,0.4,0.5,0.6,0.7} for THUMOS14.

We follow a two-stage "detection by classifying proposals" framework in evaluation, which feeds the detected proposals into the state-of-the-art action classifiers SCNN (Shou, Wang, and Chang 2016) and UntrimmedNet (Wang et al. 2017). For fair comparisons, we use the same classifiers for other proposal generation methods, including SST (Buch et al. 2017), TURN (Gao et al. 2017), CTAP (Gao, Chen, and Nevatia 2018), BSN (Lin et al. 2018), MGG (Liu et al. 2019) and BMN (Lin et al. 2019). The experimental results on THUMOS14 are shown in Tab. 7, which demonstrates that DBG based detection significantly outperforms other state-of-the-art methods in temporal action detection methods. Especially, with the same IOU threshold 0.7, our DBG based detection achieves an *m*AP improvements of 1.4% and 1.2% for two types of classifiers separately from BMN based methods.

Conclusion

This paper introduces a novel and unified temporal action proposal generator named Dense Boundary Generator (DBG). In this work, we propose dual stream BaseNet to generate two different level and more discriminative features. We then adopt a temporal boundary classification module to predict precise temporal boundaries, and an action-aware completeness regression module to provide reliable action completeness confidence. Comprehensive experiments are conducted on popular benchmarks including ActivityNet-1.3 and THUMOS14, which demonstrates the superiority of our proposed DBG compared to state-of-the-art methods.

References

Buch, S.; Escorcia, V.; Shen, C.; Ghanem, B.; and Niebles, J. C. 2017. SST: single-stream temporal action proposals. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 6373–6382.

Carreira, J., and Zisserman, A. 2017. Quo vadis, action recognition? A new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, 4724–4733.

Chao, Y.; Vijayanarasimhan, S.; Seybold, B.; Ross, D. A.; Deng, J.; and Sukthankar, R. 2018. Rethinking the faster R-CNN architecture for temporal action localization. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, 1130–1139.

Dai, X.; Singh, B.; Zhang, G.; Davis, L. S.; and Qiu Chen, Y. 2017. Temporal context network for activity localization in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, 5793–5802.

Feichtenhofer, C.; Pinz, A.; and Zisserman, A. 2016. Convolutional two-stream network fusion for video action recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 1933–1941.

Gao, J.; Yang, Z.; Sun, C.; Chen, K.; and Nevatia, R. 2017. TURN TAP: temporal unit regression network for temporal action proposals. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 3648–3656.

Gao, J.; Chen, K.; and Nevatia, R. 2018. Ctap: Complementary temporal action proposal generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 68–83.

Gao, J.; Yang, Z.; and Nevatia, R. 2017. Cascaded boundary regression for temporal action detection. In *British Machine Vision Conference 2017, BMVC 2017, London, UK, September 4-7, 2017*.

Heilbron, F. C.; Escorcia, V.; Ghanem, B.; and Niebles, J. C. 2015. Activitynet: A large-scale video benchmark for human activity understanding. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*, 961–970.

Heilbron, F. C.; Niebles, J. C.; and Ghanem, B. 2016. Fast temporal activity proposals for efficient detection of human actions in untrimmed videos. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 1914–1923.

Idrees, H.; Zamir, A. R.; Jiang, Y.-G.; Gorban, A.; Laptev, I.; Sukthankar, R.; and Shah, M. 2017. The thumos challenge on action recognition for videos “in the wild”. *Computer Vision and Image Understanding* 155:1–23.

Li, J.; Qian, J.; and Yang, J. 2017. Object detection via feature fusion based single network. In *2017 IEEE International Conference on Image Processing (ICIP)*, 3390–3394. IEEE.

Lin, T.; Zhao, X.; Su, H.; Wang, C.; and Yang, M. 2018. BSN: boundary sensitive network for temporal action proposal generation. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part IV*, 3–21.

Lin, T.; Liu, X.; Li, X.; Ding, E.; and Wen, S. 2019. BMN: boundary-matching network for temporal action proposal generation. *CoRR* abs/1907.09702.

Lin, T.; Zhao, X.; and Shou, Z. 2017. Single shot temporal action detection. In *Proceedings of the 2017 ACM on Multimedia Conference, MM 2017, Mountain View, CA, USA, October 23-27, 2017*, 988–996.

Liu, Y.; Ma, L.; Zhang, Y.; Liu, W.; and Chang, S.-F. 2019. Multi-granularity generator for temporal action proposal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3604–3613.

Long, F.; Yao, T.; Qiu, Z.; Tian, X.; Luo, J.; and Mei, T. 2019. Gaussian temporal awareness networks for action localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 344–353.

Qiu, Z.; Yao, T.; and Mei, T. 2017. Learning spatio-temporal representation with pseudo-3d residual networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 5534–5542.

Shou, Z.; Wang, D.; and Chang, S. 2016. Temporal action localization in untrimmed videos via multi-stage cnns. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 1049–1058.

Simonyan, K., and Zisserman, A. 2014. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, 568–576.

Tran, D.; Bourdev, L. D.; Fergus, R.; Torresani, L.; and Paluri, M. 2015. Learning spatiotemporal features with 3d convolutional networks. In *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*, 4489–4497.

Wang, L.; Xiong, Y.; Wang, Z.; and Qiao, Y. 2015. Towards good practices for very deep two-stream convnets. *CoRR* abs/1507.02159.

Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; and Gool, L. V. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII*, 20–36.

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

Xiong, Y.; Wang, L.; Wang, Z.; Zhang, B.; Song, H.; Li, W.; Lin, D.; Qiao, Y.; Gool, L. V.; and Tang, X. 2016. CUHK & ETHZ & SIAT submission to activitynet challenge 2016. *CoRR* abs/1608.00797.

Xu, H.; Das, A.; and Saenko, K. 2017. R-C3D: region convolutional 3d network for temporal activity detection. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 5794–5803.

Yao, T.; Li, Y.; Qiu, Z.; Long, F.; Pan, Y.; Li, D.; and Mei, T. 2017. Msr asia msm at activitynet challenge 2017: Trimmed action recognition, temporal action proposals and densecaptioning events in videos. In *CVPR ActivityNet Challenge Workshop*.

Zhao, Y.; Xiong, Y.; Wang, L.; Wu, Z.; Tang, X.; and Lin, D. 2017a. Temporal action detection with structured segment networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2933–2942.

Zhao, Y.; Xiong, Y.; Wang, L.; Wu, Z.; Tang, X.; and Lin, D. 2017b. Temporal action detection with structured segment networks. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, 2933–2942.