Pyramid Region-based Slot Attention Network for Temporal Action Proposal Generation

Shuaicheng Li*[1]
lishuaicheng@sensetime.com

Feng Zhang*[2]
fzhang20@fudan.edu.cn

Rui-Wei Zhao[3]
rwzhao@fudan.edu.cn

Kunlin Yang[1]
yangkunlin@sensetime.com

Lingbo Liu[4]
liulingbo918@gmail.com

Rui Feng†[2]
fengrui@fudan.edu.cn

Jun Hou[1]
houjun@sensetime.com

[1] Sensetime Research

[2] Shanghai Key Laboratory of Intelligent
Information Processing
Fudan University
Shanghai, China

[3] Academy for Engineering and
Technology Fudan University
Shanghai, China

[4] The Hong Kong Polytechnic University
Hong Kong, China

Abstract

It has been found that temporal action proposal generation, which aims to discover the temporal action instances within the range of the start and end frames in the untrimmed videos, can largely benefit from proper temporal and semantic context exploitation. The latest efforts were dedicated to considering the temporal context and similarity-based semantic context through self-attention modules. However, they still suffer from cluttered background information and limited contextual feature learning. In this paper, we propose a novel Pyramid Region-based Slot Attention (PRSlot) modules to address these issues. Instead of using the similarity computation, our PRSlot module directly learns the local relations in an encoder-decoder manner and generates the representation of a local region enhanced based on the attention over input features called *slot*. Specifically, upon the input snippet-level features, PRSlot module takes the target snippet as *query*, its surrounding region as *key* and then generates slot representations for each *query-key* slot by aggregating the local snippet context with a parallel pyramid strategy. Based on PRSlot modules, we present a novel Pyramid Region-based Slot Attention Network termed PRSA-Net to learn a unified visual representation with rich temporal and semantic context for better proposal generation. Extensive experiments are conducted on two widely adopted THUMOS14 and ActivityNet-1.3 benchmarks. In particular, we improve

* Equal contribution

† Corresponding author

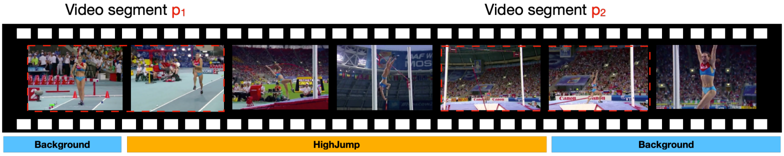

Figure 1: The boundaries can hardly distinguished from background due to cluttered content and action-unrelated scenes. Our tailor-modified PRSA-Net is designed to selectively contextualize local semantic information instead of pairwise similarity.

the AR@100 from the previous best 50.67% to 56.12% for proposal generation and raise the mAP under 0.5 tIoU from 51.9% to 58.7% for action detection on THUMOS14. Code is available at https://github.com/handhand123/PRSA-Net.

1 Introduction

Temporal action detection is a popular and fundamental problem for video content understanding. This task aims to predict the precise temporal boundaries and categories of each action instance in the videos. Similar to two-stage object detection in images, most temporal action detection methods [17, 18, 26, 31] follow a two-stage strategy: temporal action proposal generation and action proposal classification. Action classification [29] has achieved convincing performance, but temporal action detection accuracy is still unsatisfactory on mainstream benchmarks. It indicates that the quality of generated proposals is the bottleneck of the final action detection performance.

Existing proposal generation approaches make sorts of efforts to exploit rich semantic information for the sake of better high-quality proposal generation. The majority of previous works [16, 17, 18] embed the temporal representation by stacked temporal convolutions. G-TAD [31] proposes to explicitly model temporal contextual information by graph convolutional networks (GCNs). More recently, like DETR [4] in object detection, transformer based methods [21, 26] are introduced to provide long-range temporal context for proposal generation and action detection.

Despite impressive progress, these aforementioned approaches still confront two challenges that remain to be addressed: 1) limited contextual representation learning and 2) sensitivity to the complicated background. In the former case, although video snippets contain richer information than a single image such as temporal dynamical content, offering useful cues for generating proposals, much fewer efforts are spent on semantic temporal content modeling. In the latter case, due to the complicated background in untrimmed videos, [31] adopts a similarity-based attention mechanism to selectively extract the most relevant information and relationships. However, it is sub-optimal to exploit the dependencies only based on computing the pairwise similarity. Because there exists highly similar content for consistent frames, where the pairwise relations may introduce spurious noise such as cluttered background and inaccurate action-related content. As illustrated in Fig.1, the action instances surrounded by the background content, such as *video segments* p_1, p_2, can hardly be discriminated due to the camera motion and progressive action transitions. It demonstrates that applying traditional pairwise similarity-based relations such as the operation *dot product, Euclidean distance* are insufficient to deliver complete contextual information.

To relieve the above issues, we propose a novel Pyramid Region-based Slot Attention

(PRSlot) to contextualize action-matter representation. It is building upon the recently-emerged slot attention [20], which learns the object-centric representations for image classification task. Our well-design PRSlot module takes the snippet-level features as input and maps them to a set of output vectors by aggregating local region context that we refer to as *slots*. **First**, our PRSlot module is enhanced by a Region-based Attention (RA) which directly estimates the confidence from inputs and its local region to the slots. Unlike the similarity-based attention mechanism, our RA restricts the scope of slot interactions to local surroundings and learns the semantic attention directly using an encoder-decoder architecture. **Second**, instead of applying recurrent function to update slots over multiple iterations in original slot attention [20], our tailor-modified PRSlot module presents a parallel pyramid slot representation updating strategy with multi-scale local regions. **Finally**, the complementary action-matter representation generated by the PRSlot modules are used to produce boundaries scores and proposal-level confidence respectively by linear layers. Based on the above components, we present a novel Pyramid Region-based Slot Attention Network called PRSA-Net to capture abundant semantic representation for high-quality proposal generations. Experimental results show our PRSA-Net is superior to the SOTA performance on two widely used benchmarks. The main contributions of this paper are therefore as follows,

- A newly Region-based Attention is proposed to directly generate relation scores from the slot representations and its surroundings using encoder-decoder manner instead of similarity-based operation.

- We propose a novel Pyramid Region-based Slot Attention module, which incorporates a region-based attention mechanism and a parallel pyramid iteration strategy to effectively capture contextual representations for better boundaries discrimination.

- We perform extensive experiments on the THUMOS14 and ActivityNet-1.3 benchmarks and evaluate the performances of temporal action proposal generation and detection. The results show that our proposed approach outperforms other SOTA methods. Our ablation study also highlights the importance of both architecture components.

2 Related Work

Temporal Action Proposals and Detection. Based on the extracted video features from the feature extractor, numbers of recent approaches solve the action detection problems in two steps: (1) proposal generation; (2) proposal classification and refinement. In the first step, the proposals can be generated in a top-down fashion, which is based on preset sliding temporal windows or anchors [6, 10, 11, 23]. Or alternatively, the proposals can be generated in a bottom-up fashion, which directly predicts proposal locations based on video frames or snippets features [2, 8, 9, 24, 32]. More recently, many methods [22, 33, 35] pay attention to refine the proposals. In particular, the video features and proposals generated by other proposal generation methods, *e.g.*, BSN[17] or BMN [18], are all processed to enhance the proposal-level representations. However, our proposal generation method only inputs video features and generates proposals by ours. Apart from the above-mentioned ones, some other approaches may even exploit the context information at object-level [12, 30]. However, those methods are beyond the scope of this paper since we mainly focus on the snippet-level feature learning and proposal generation while not proposal refinement.

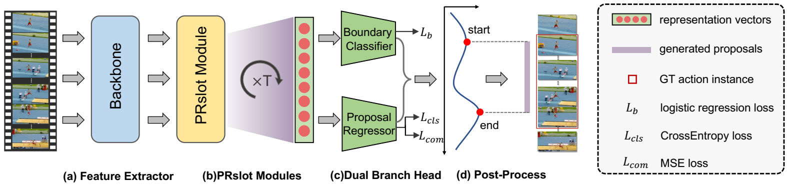

Figure 2: The overall architecture of the proposed PRSA-Net. The snippet-level features of videos are extracted by a CNN network in Feature Extractor. Then PRSlot modules are employed to contextualize slot representations with T times. Next, Boundary Classifier and Proposal Regressor are utilized to generate the predicted proposals. Finally, post-process is utilized to suppress redundant proposals.

Attention Mechanism. The vanilla attention mechanism is proposed by [27] and contributes more to capturing long-range dependencies. Recently, attention mechanism has been widely applied in the computer vision field, *e.g.*, image classification ViT [7], activity recognition [15], and object detection [4]. Original Slot Attention [20] is proposed to exploit object-centric representation. Different from the previous attention mechanism, our proposed region-based attention is estimated directly by high-level correlation features instead of similarity operations, which can concentrate on the local context better.

3 Approach

3.1 Problem Formulation

Given an untrimmed video $V = \{v_t\}_{t=1}^{T}$ contains a collection of N action instances $\Psi = \{\psi_n = (t_{s,n}, t_{e,n})\}_{n=1}^{N}$, where ψ_n refers to the n-th action and $(t_{s,n}, t_{e,n})$ correspond to its annotated start, end time. The aim of temporal action proposal generation task is to predict a set of action proposals $\Phi = \{\phi_w = (t_{s,w}, t_{e,w}, p_w)\}_{w=1}^{W}$ as close to the GT annotations as possible based on the content of V. Here p_w is the w-th proposal confidence. We propose a PRSANet to generate temporal action proposals precisely. The pipeline of PRSA-Net is illustrated in Fig.2. We provide detailed descriptions of PRSA-Net below.

3.2 Feature Extractor

Given an untrimmed video V, we utilize the Kinetics [14] pre-trained I3D [5] to extract video features. We consider RGB and optical flow features jointly as they can capture different motion aspects. Following the implementation of previous methods [17, 31], each frame feature is flattened into a C-dimensional feature vector and then we group every consecutive σ frames into $L = \lceil T/\sigma \rceil$ snippets. In this work, these generated video snippets are the minimal units for further feature modeling. For convenience, we denote the extracted snippet features as $X \in \mathbb{R}^{L \times C}$, where C is the feature dimension and L is the total number of video snippets. In the end, a 1d convolution is applied to transform the channels into $C_{\texttt{input}}$.

3.3 Pyramid Region-based Slot Attention

The original slot attention module is firstly proposed for updating the object-centric representations (*slots*) by similarity-based attention. Slots produced by slot attention bind to any

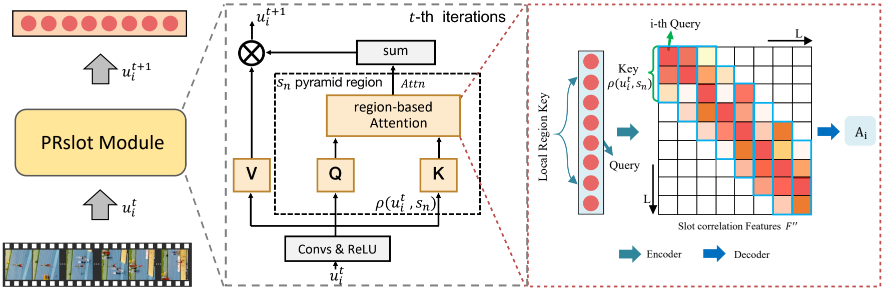

Figure 3: Overview of our Pyramid Region-based Slot Attention (PRSlot) module. For input snippet (slot) features u_i, we take x_i as *query* and its local context snippet (slots) feature as *key*. An encoder-decoder operation is applied to map the correlation features into the local region score vector, and then generate the output slot features through the weighted sum of the input value (V).

object in the input embedding and are updated via a learned recurrent function for object-centric learning. Inspired by this, we propose a tailor-modified Pyramid Region-based Slot Attention (PRSlot) module for capturing action-matter representations by a novel attention mechanism. We adopt dense slots to capture boundary-aware representation. The insight is that the motion content in boundaries context transfers rapidly and can be distinguished from the background. The slot is initialized with snippet features X and then is updated by the proposed PRSlot module for T Times. For convenience, the i-th slot representation refined t times denotes as $u_i^{(t)}$. In particular, slot features are processed through convolutional layers with stride 1, window size 3 and padding 1, followed by a ReLU activation layer, and produce the slot features shaped as $L \times C_{\texttt{embed}}$. Then, based on the proposed Region-based attention mechanism (*will describe below*), we update our slot representations by the weighted sum of input and then apply batchnorm to the output vectors.

3.3.1 Region-based Attention Mechanism.

As shown in Fig. 3, to focus on the crucial local region relations, we design a newly region-based attention mechanism to learn the snippets relationships instead of using the similarity computation (*cosine*, *dot product* or *Euclidean distance*). Our region-based attention directly estimates the snippets interactions by the content of the snippet and its local surroundings. Mathematically, given the target input feature u_i and its surrounding features denoted as $\rho(u_i, s) = \{u_{i-s}, ..., u_{i+s}\}$, where s represents the window size of its local region, we formulate the calculation of the interaction scores as

$$A_i = f_{\text{region}}(u_i, \rho(u_i, s)). \tag{1}$$

Here the output A_i is defined to be the attention vector for the i-th snippet (slot), and its length exactly equals $2s+1$, which is the temporal length of the covered surrounding $\rho(u_i, s)$. f_{region} is a series of operations to map the local region features into the interaction confidence scores for the target snippet (slot). The value of j-th element in A_i, here denoted as $A_{i,j}$, is expected to indicate the relation score between j-th slot and the target slot i. We take every input feature $u_i \in \{u_l\}_{l=0}^{L-1}$ as *query*, and the local context slot of u_i as *key*, our designed region-based attention operation constructs a learnable score vector, which is used to quantitatively measure the importance between the target *query* and *key*. In detail, this attention mechanism consists of an encoder to embed local contextual information and a decoder to estimate the relevant attention. Due to the permutation invariant of temporal snippets, we additionally add the position embedding to the input features. After this, the output slot representation can be described based on the attention over the feature map in the local region.

Encoder: Summarizing the surrounding slot context for the target slot is critical to augment slot representation. The encoder is utilized to exploit contextual information for the surrounding slots of the target slot. The detailed implementation is described below. Upon the input slot features $U \in \mathbb{R}^{C_{\texttt{embed}} \times L}$ (we omit the batch dimension for clarity), we repeat the temporal length L in last dimension for sampling local windows, and then generate a 2D feature map $F \in \mathbb{R}^{C_{\texttt{embed}} \times L \times L}$. A 1×1 convolutional layer is adopted to transform the $C_{\texttt{embed}}$-dimension channel into $C_{\texttt{out}}$-dimension channel. Next, we apply the 2D convolution layer to augment the target slot using the local surrounding context shaped as $2s+1$. Specifically, the 2D convolutional layer is implemented for target slot feature embedding, where we denote the kernel size as $(2s+1,1)$, padding size $(s,0)$ and stride $(1,1)$, indicating that we capture local context with size $2s+1$ from each column slot. It generates the feature map $F' \in \mathbb{R}^{C_{\texttt{out}} \times L \times L}$, and we regard this feature map as the high-level correlation matrix. For this content-based feature map F', we only consider the local contextual information for each target slot. The process can be formulated as,

$$F''_{i,j} = \begin{cases} F'_{i,j}, & u_i \in \rho(u_j, s), \\ 0, & \text{Otherwise}, \end{cases} \tag{2}$$

where $F'_{i,j} \in \mathbb{R}^{C_{\texttt{embed}}}$ denotes the slot correlation features between i-th slot and j-th slot.

Decoder: For the sparse relation features F'', a decoder is deployed to map the relation features into the correspond relation scores. It is noted that we only consider the local surroundings of the target slot and operate scores mapping, which can be formulated as,

$$A_{i,j} = \begin{cases} f^{\text{dec}}_{\text{region}}(F''_{i,j}), & u_i \in \rho(u_j, s), \\ 0, & \text{Otherwise}, \end{cases} \tag{3}$$

Here $f^{\text{dec}}_{\text{region}}$ represents the decoding functions which is used to map the slots relation features into scores confidence. In practice, a 2D convolutional layer is applied to distribute the relation features into relation scores, where we set kernel size to $(2s+1,1)$, padding size to $(s,0)$, out channel to 1 and stride to $(1,1)$. This convolutional layer is only deployed for the local slots $\rho(u_i,s)$ of target slot i and formulate a slot confidence vector. Finally, a Softmax operation is followed to normalize the score matrix. In this way, each interaction score $A_{i,j}$ is actually dependent on the content of u_j and $\rho(u_j,s)$.

3.3.2 Iteration and update strategy.

The original slot attention updates representations via recurrent function at each iteration $t = 1...T$. However, the recurrent function (*e.g.*GRU) is time-consuming and achieves limited performance boost. It will be demonstrated in ablation study. In order to tackle the issues and make the PRSlot specialize in the temporal action-matter representation exploiting, we develop a parallel pyramid iteration strategy for slot representations updating. Due to the variants of video duration, we apply a variety of local regions to exploit the slot representation completely. In detail, $|S|$ region-based attention using different scale surroundings $s \in S$ are deployed in parallel, where S is a collection of snippet region size and $|S|$ denotes the carnality of the set. In the end, we fuse the slot attention by element sum-wise and aggregate the input values to their assigned slots.

Method	Backbone	@50	@100	@200	@500
MGG [19]	TSN	39.93	47.75	54.65	61.36
BSN [17] + SNMS	TSN	37.46	46.06	53.21	60.64
BMN [18] + SNMS	TSN	39.36	47.72	54.70	62.07
BC-GNN [1] + NMS	TSN	41.15	50.35	56.23	61.45
BU-TAL [34]	I3D	44.23	50.67	55.74	-
BSN++ [25] +SNMS	TSN	42.44	49.84	57.61	65.17
RTD-Net [26]	I3D	41.52	49.32	56.41	62.91
Ours + NMS	TSN	47.49	55.14	60.18	63.53
Ours + SNMS	TSN	44.11	52.52	59.19	65.12
Ours + NMS	I3D	**49.06**	**56.12**	**61.30**	63.20
Ours + SNMS	I3D	45.81	53.13	59.32	**66.32**

Table 1: Comparison of the action proposal generation performances with state-of-the-arts on THUMOS14 in terms of AR@AN(%).

Method	AR@1	AR@50	AUC
BSN [17]	32.17	74.16	66.17
MGG [19]	-	74.54	66.43
BMN [18]	-	75.01	67.10
BC-GNN [1]	-	76.73	68.05
BU-TAL [34]	-	75.27	66.51
RTD-Net [26]	33.05	73.21	65.78
Ours	35.37	76.90	69.21

Table 2: Comparison of the action proposal generation performances with state-of-the-arts on ActivityNet-1.3 in terms of AR@AN(%) and AUC.

3.4 Dual branch head

Next, slot embedding provided by PRSlot modules serves as the latent action-aware representations for the proposal estimation.

Boundary Classifier. A lightweight 1d convolution layer is introduced to transform the input channels into 2 for (*start, end*) detection and a non-linear sigmoid function is followed to form the start/end probabilities $\mathcal{P}^s/\mathcal{P}^e$ separately.

Proposal Confidence Regressor. Following the conventional proposal regression method [18], we use pre-defined dense anchors to generate densely distributed proposals shaped as $D \times L$ and then apply 1DAlignLayer [31] to extract the proposal-level features for corresponding proposal anchors shaped as $D \times L \times C_{\mathtt{out}}$. Finally, 2 FC layers are used to predict the proposal-level completeness maps M^{com} and classification map M^{cls}.

3.5 Training and Inference

Label Assignment. We first generate temporal boundary label G_s and G_e followed by BSN [17]. Next, we generate the dense proposal label map $G^c \in \mathbb{R}^{D \times L}$. As described in [18], for a proposal $G^c_{i,j}$ with start frame i and end frame $i+j$, we calculate the intersection over union (IoU) with all ground-truth and denote the maximum IoU as the value of $G^c_{i,j}$.

Training. We define the following loss function to train our proposed model

$$\mathcal{L} = \mathcal{L}_b + \mathcal{L}_p + \lambda \mathcal{L}_{norm} \tag{4}$$

Here the boundary classification loss $\mathcal{L}_b$ is a weighted binary logistic regression loss used to determine the starting and ending scores $\mathcal{P}^s \in \{\mathcal{P}^s_l\}^L_{l=1}$ or $\mathcal{P}^e \in \{\mathcal{P}^e_l\}^L_{l=1}$. $\mathcal{L}_{norm}$ is the regularization term for network weights, we set $\lambda = 0.0002$. We also construct the proposal confidence losses $\mathcal{L}_p$ which is the combination of cross-entropy loss and mean square error loss to optimize dense proposals confidence scores M^{com} and M^{cls}. It can be formulated as,

$$\mathcal{L}_p = \mathcal{L}_{cls}(M^{cls}, G^c) + \lambda_c \mathcal{L}_{com}(M^{com}, G^c) \tag{5}$$

where $\mathcal{L}_{cls}$ is binary logistic regression loss and $\mathcal{L}_{com}$ is the MSE loss, usually we set the balance term $\lambda_c = 10$.

Inference. Based on the outputs of the boundaries classifier, we select the valid starting snippets from $\{\mathcal{P}^s_l\}^L_{l=1}$ by two conditions: (1) $\mathcal{P}^s_{l-1} < \mathcal{P}^s_l ; \mathcal{P}^s_l > \mathcal{P}^s_{l+1}$; (2) $\mathcal{P}^s_l > 0.5 \times$

Attention Mechanism		Iteration Strategy		AR@AN (*testing set*)		
`SA`	`RA`	`original`	`ours`	@50	@100	@200
✓		✓		35.6	43.1	50.9
✓			✓	43.5	52.3	56.5
	✓	✓		42.9	52.8	55.6
	✓		✓	**49.1**	**56.1**	**61.3**

Table 3: Comparisons of different combinations in our model. Evaluated on THUMOS14 in terms of AR@AN(%). `SA` and `RA` denote the implementation of similarity-based and our region-based attention separately. `original` and `ours` denote the original and our parallel pyramid strategy respectively.

Iteration times	@50	@100	@200	@500
1	48.4	55.3	60.1	61.6
2	**49.1**	**56.1**	61.3	**63.2**
3	48.9	55.7	**61.9**	62.3

Table 4: Different setting choices for the number of pyramid iterations. Evaluated on THUMOS14 in terms of AR@AN (%)

$\max_{n=1}^{L}\{\mathcal{P}_n^s\}$. We apply the same rule for recording ending snippets. Then, we combine these valid starting and ending snippets and obtain the candidate proposals denoted as $\Phi = \{\phi_w = (t_{s,w}, t_{e,w}, p_w, M_w\}_{w=1}^{W}\}$, where $\mathcal{P}_w = \mathcal{P}_{t_{s,w}}^s \cdot \mathcal{P}_{t_{s,w}}^e$ and proposal-level confidence $M_w = M_{t_{s,w},t_{e,w}}^{cls} \cdot M_{t_{s,w},t_{e,w}}^{com}$. Finally, we fuse the proposal confidence scores and output the predicted proposals $\Phi = \{\phi_w = (t_{s,w}, t_{e,w}, S_w\}_{w=1}^{W}\}$, here $S_w = p_w \times M_w$.

4 Experiments

In this section, we conduct extensive experiments on two challenging benchmarks, i.e., THUMOS14[13] and ActivityNet-1.3 [3] to demonstrate the effectiveness of the proposed method. We utilize the Kinetics [14] pre-trained I3D [5] to extract video features. For fair comparisons with [17, 18, 31], we also conduct experiments based on the Kinetics pre-trained TSN [28] to extract video features. More details about datasets and implementations are available in the supplementary material.

4.1 Temporal Proposal Generation

Comparisons with State-of-the-Arts. We compare our PRSA-Net with other methods on two backbones for a fair comparison. The results on THUMOS14 are summarized in Table 1. It can be observed that our proposed PRSA-Net outperforms all of the aforementioned methods by a large margin with either NMS or Soft-NMS used in post-processing. When using TSN as feature extractor, our method respectively improve the AR@50 from 42.44% to 47.49%, AR@100 from 49.84% to 55.14%, and AR@200 from 57.61% to 60.18%. It is worth noting that our model with I3D backbone outperforms all previous methods by a large margin (+ 4.83% AR@50 and +5.45% AR@100). With the well-design PRSlot and its region-based attention, our method establishes superior performance on THUMOS14. The results on ActivityNet-1.3 are displayed in Table 2. Our method shows better performances than the previous best results under both AR@100 and AUC scores. In particular, the AR@1 achieves 2.32% performance boost in this well-studied benchmark.

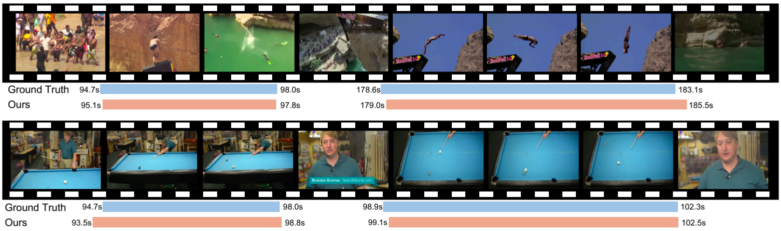

Figure 4: Some action proposal examples predicted by our PRSA-Net on THUMOS14.

Temporal Scale	@50	@100	@200	@500
100	48.4	53.3	59.1	61.6
200	**49.3**	55.4	60.6	62.1
250	49.1	**56.1**	**61.3**	**63.2**
300	48.9	55.0	61.2	62.9

Table 5: Ablation study on the choice of the temporal scales Evaluated on THUMOS14 in terms of AR@AN (%)

4.2 Ablation Studies

Variants of our PRSA-Net. To measure the importance of our proposed region-based attention and parallel pyramid iteration strategy, we conduct the ablation study with the following variants. Baseline: we replace the PRSlot with the original slot attention, which includes the similarity-based attention and recurrent iteration strategy. More baseline details can be found in our Supplementary materials. We also adopt different combinations of our PRSlot architecture components. Table 3 reports the detailed proposal generation results on THUMOS14. The ✓ symbols stand for *with* the corresponding components or strategies. We can find that our proposed region-based attention (*the last row*) improves the AR by more than 6%, indicating the effectiveness of our designed PRSlot module. Additionally, our parallel pyramid iteration strategy also contributes to performance boosts. We attribute performance differences to variations in architecture design and iteration schemes.
Sensitivity to iteration times. We also report the results of the sensitivity of our model to the different settings for iteration times in Table 4. We find that using 2 iterations can reach the best results while performance slightly degrades at using 3 times. We defer the other ablation studies to the supplementary.
Study on Temporal Scales. In Table 5, we show the effect of the temporal scales for the proposal generation. The performance benefits from the larger temporal scale. Having more than $L = 300$ temporal scales when training, performance decreases slightly.

4.3 Qualitative Results

In Fig. 4, we visualize some action detection results generated by our PRSA-Net on THUMOS14. We can find that our model can successfully detect the action instances in these examples with precise action boundaries. For the untrimmed video with multiple action instances, the foreground actions and background scenes can be well distinguished.

4.4 Action Detection with our proposals

When evaluating the quality of our proposals, we follow the conventional two-stage action detection works [17, 18, 31]. Therefore, we classify the candidate proposals using external

Method	Backbone	Classifier	0.7	0.6	0.5	0.4	0.3
BSN [17]	TSN	UNet	20.0	28.4	36.9	45.0	53.5
MGG [19]	TSN	UNet	21.3	29.5	37.4	46.8	53.9
BMN [18]	TSN	UNet	20.5	29.7	38.8	47.4	56.0
G-TAD [31]	TSN	UNet	23.4	30.8	40.2	47.6	54.5
BU-TAL et al. [34]	I3D	UNet	28.5	38.0	45.4	50.7	53.9
BSN++ [25]	TSN	UNet	22.8	31.9	41.3	49.5	59.9
BC-GNN [1]	TSN	UNet	23.1	31.2	40.4	49.1	57.1
RTD-Net [26]	I3D	UNet	25.0	36.4	45.1	53.1	58.5
Ours	TSN	UNet	28.8	39.2	51.1	58.9	65.4
Ours	I3D	UNet	**30.9**	**44.0**	**55.0**	**64.4**	**69.1**
BSN [17]	TSN	PGCN	-	-	49.1	57.8	63.6
G-TAD [31]	TSN	PGCN	22.9	37.6	51.6	60.4	66.4
RTD-Net [26]	I3D	PGCN	23.7	38.8	51.9	62.3	68.3
Ours	I3D	PGCN	**28.4**	**47.3**	**58.7**	**73.2**	**76.3**

Table 6: Performance comparisons of temporal action detection on THUMOS14 in terms of mAP@tIoUs(%).

Method	0.5	0.75	0.95	Average
TAL-Net [6]	38.23	18.30	1.30	20.22
BSN [17]	46.45	29.96	8.02	30.03
P-GCN [33]	48.26	33.16	3.27	31.11
BMN [18]	50.07	34.78	8.29	33.85
G-TAD [31]	50.36	34.60	9.02	34.09
BSN++ [25]	51.27	35.70	8.33	34.88
BC-GNN [1]	50.56	35.35	9.71	34.68
RTD-Net [26]	47.21	30.68	8.61	30.83
Ours	**52.37**	**37.18**	**9.78**	**36.26**

Table 7: Comparison with state-of-the-arts detection methods on ActivityNet-1.3 in terms of mAP@tIoUs(%).

classifiers. We use the video classifier in UntrimmedNet [29] to assign the video-level action classes on THUMOS14. Furthermore, we also introduce the proposal-level classifier P-GCN [33] to predict action labels for every candidate proposals. We evaluate the final action detection performances and make comparisons with the state-of-the-art in Table 6. Our approach achieves significant improvements under all tIoU settings. Especially, the mAP at the typical IoU = 0.5 was boosted from 45.4% to 55.0%, reaching a considerable improvement ratio of 21.1%. Also, when proposal-level P-GCN is applied following the same implementations in [26, 31], the performance can be boosted rapidly and achieve 58.7% mAP@0.5. Table 7 shows our action detection results on the validation set of ActivityNet-1.3 comparing with previous works. Again, our method outperforms most other methods in almost all cases, including the average mAPs over different tIoUs. These experiments demonstrate that the proposals generated by our PRSA-Net are able to boost the action detection performance better.

5 Conclusion

In this paper, we propose a novel Pyramid Region-based Slot Attention Network (PRSA-Net) for temporal action proposal generation. Specifically, a Pyramid Region-based Slot Attention (PRSlot) module is introduced to capture action-aware representations, which is enhanced by the proposed region-based attention and parallel pyramid iteration strategy. The experiments show the advantages of our method both in action proposals and action detection performance.

Acknowledgments. This work was supported by National Natural Science Foundation of China (No. 62172101); This work was supported (in part) by the Science and Technology Commission of Shanghai Municipality(No. 21511100500).

References

[1] Yueran Bai, Yingying Wang, Yunhai Tong, Yang Yang, Qiyue Liu, and Junhui Liu. Boundary content graph neural network for temporal action proposal generation. In *European Conference on Computer Vision*, pages 121–137. Springer, 2020.

[2] Shyamal Buch, Victor Escorcia, Chuanqi Shen, Bernard Ghanem, and Juan Carlos Niebles. SST: Single-Stream Temporal Action Proposals. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[3] Fabian Caba Heilbron, Victor Escorcia, Bernard Ghanem, and Juan Carlos Niebles. ActivityNet: A Large-scale Video Benchmark for Human Activity Understanding. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European Conference on Computer Vision*, pages 213–229, 2020.

[5] Joao Carreira and Andrew Zisserman. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[6] Yu-Wei Chao, Sudheendra Vijayanarasimhan, Bryan Seybold, David A Ross, Jia Deng, and Rahul Sukthankar. Rethinking the Faster R-CNN Architecture for Temporal Action Localization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

[8] Victor Escorcia, Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. DAPs: Deep Action Proposals for Action Understanding. In *European Conference on Computer Vision (ECCV)*, 2016.

[9] Jiyang Gao, Zhenheng Yang, Kan Chen, Chen Sun, and Ram Nevatia. TURN TAP: Temporal Unit Regression Network for Temporal Action Proposals. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[10] Jiyang Gao, Zhenheng Yang, and Ram Nevatia. Cascaded Boundary Regression for Temporal Action Detection. In *British Machine Vision Conference (BMVC)*, 2017.

[11] Fabian Caba Heilbron, Juan Carlos Niebles, and Bernard Ghanem. Fast Temporal Activity Proposals for Efficient Detection of Human Actions in Untrimmed Videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[12] Fabian Caba Heilbron, Wayner Barrios, Victor Escorcia, and Bernard Ghanem. SCC: Semantic Context Cascade for Efficient Action Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[13] Haroon Idrees, Amir R Zamir, Yu-Gang Jiang, Alex Gorban, Ivan Laptev, Rahul Sukthankar, and Mubarak Shah. The THUMOS Challenge on Action Recognition for Videos "In the Wild". *Computer Vision and Image Understanding (CVIU)*, 155:1–23, 2017.

[14] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The Kinetics Human Action Video Dataset. *arXiv:1705.06950*, 2017.

[15] Shuaicheng Li, Qianggang Cao, Lingbo Liu, Kunlin Yang, Shinan Liu, Jun Hou, and Shuai Yi. Groupformer: Group activity recognition with clustered spatial-temporal transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13668–13677, 2021.

[16] Chuming Lin, Jian Li, Yabiao Wang, Ying Tai, Donghao Luo, Zhipeng Cui, Chengjie Wang, Jilin Li, Feiyue Huang, and Rongrong Ji. Fast Learning of Temporal Action Proposal via Dense Boundary Generator. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[17] Tianwei Lin, Xu Zhao, Haisheng Su, Chongjing Wang, and Ming Yang. BSN: Boundary Sensitive Network for Temporal Action Proposal Generation. In *European Conference on Computer Vision (ECCV)*, 2018.

[18] Tianwei Lin, Xiao Liu, Xin Li, Errui Ding, and Shilei Wen. BMN: Boundary-Matching Network for Temporal Action Proposal Generation. In *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[19] Yuan Liu, Lin Ma, Yifeng Zhang, Wei Liu, and Shih-Fu Chang. Multi-Granularity Generator for Temporal Action Proposal. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[20] Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in Neural Information Processing Systems*, 33: 11525–11538, 2020.

[21] Megha Nawhal and Greg Mori. Activity graph transformer for temporal action localization. *arXiv preprint arXiv:2101.08540*, 2021.

[22] Zhiwu Qing, Haisheng Su, Weihao Gan, Dongliang Wang, Wei Wu, Xiang Wang, Yu Qiao, Junjie Yan, Changxin Gao, and Nong Sang. Temporal context aggregation network for temporal action proposal refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 485–494, 2021.

[23] Zheng Shou, Dongang Wang, and Shih-Fu Chang. Temporal Action Localization in Untrimmed Videos via Multi-stage CNNs. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[24] Bernard Ghanem Shyamal Buch, Victor Escorcia and Juan Carlos Niebles. End-to-End, Single-Stream Temporal Action Detection in Untrimmed Videos. In *British Machine Vision Conference (BMVC)*, 2017.

[25] Haisheng Su, Weihao Gan, Wei Wu, Junjie Yan, and Yu Qiao. Bsn++: Complementary boundary regressor with scale-balanced relation modeling for temporal action proposal generation. *arXiv preprint arXiv:2009.07641*, 2020.

[26] Jing Tan, Jiaqi Tang, Limin Wang, and Gangshan Wu. Relaxed transformer decoders for direct action proposal generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13526–13535, 2021.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

[28] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. In *European Conference on Computer Vision (ECCV)*, 2016.

[29] Limin Wang, Yuanjun Xiong, Dahua Lin, and Luc Van Gool. UntrimmedNets for Weakly Supervised Action Recognition and Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[30] Jianchao Wu, Zhanghui Kuang, Limin Wang, Wayne Zhang, and Gangshan Wu. Context-Aware RCNN: A Baseline for Action Detection in Videos. In *European Conference on Computer Vision (ECCV)*, 2020.

[31] Mengmeng Xu, Chen Zhao, David S Rojas, Ali Thabet, and Bernard Ghanem. G-TAD: Sub-Graph Localization for Temporal Action Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[32] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. End-to-End Learning of Action Detection from Frame Glimpses in Videos. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[33] Runhao Zeng, Wenbing Huang, Mingkui Tan, Yu Rong, Peilin Zhao, Junzhou Huang, and Chuang Gan. Graph Convolutional Networks for Temporal Action Localization. In *IEEE International Conference on Computer Vision (ICCV)*, 2019.

[34] Peisen Zhao, Lingxi Xie, Chen Ju, Ya Zhang, Yanfeng Wang, and Qi Tian. Bottom-up Temporal Action Localization with Mutual Regularization. In *European Conference on Computer Vision (ECCV)*, 2020.

[35] Zixin Zhu, Wei Tang, Le Wang, Nanning Zheng, and Gang Hua. Enriching local and global contexts for temporal action localization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13516–13525, 2021.