Proposal-Free Video Grounding with Contextual Pyramid Network

Kun Li, Dan Guo*, Meng Wang*
Key Laboratory of Knowledge Engineering with Big Data (HFUT), Ministry of Education
School of Computer Science and Information Engineering, Hefei University of Technology (HFUT)
School of Artificial Intelligence
Intelligent Interconnected Systems Laboratory of Anhui Province (Hefei University of Technology)
kunli.hfut@gmail.com, guodan@hfut.edu.cn, eric.mengwang@gmail.com

Abstract

The challenge of video grounding - localizing activities in an untrimmed video via a natural language query - is to tackle the semantics of vision and language consistently along the temporal dimension. Most existing proposal-based methods are trapped by computational cost with extensive candidate proposals. In this paper, we propose a novel proposal-free framework named Contextual Pyramid Network (CPNet) to investigate multi-scale temporal correlation in the video. Specifically, we propose a pyramid network to extract 2D contextual correlation maps at different temporal scales ($T*T$, $\frac{T}{2}*\frac{T}{2}$, $\frac{T}{4}*\frac{T}{4}$), where the 2D correlation map (past $\rightarrow$ current & current $\leftarrow$ future) is designed to model all the relations of any two moments in the video. In other words, CPNet progressively replenishes the temporal contexts and refines the location of queried activity by enlarging the temporal receptive fields. Finally, we implement a temporal self-attentive regression (*i.e.*, proposal-free regression) to predict the activity boundary from the above hierarchical context-aware 2D correlation maps. Extensive experiments on ActivityNet Captions, Charades-STA, and TACoS datasets demonstrate that our approach outperforms state-of-the-art methods.

Introduction

Video understanding has attracted increasing attention in the past few years; many challenges still exist in various video-based tasks, such as video classification (Wang et al. 2016) and temporal activity detection (Buch et al. 2017a). However, these action related tasks are restricted to a collection of pre-defined action classes. In real applications, people always give a query and require the model to localize the correct video segment corresponding to the query's core textual semantics. Therefore, the task of video grounding was (Gao et al. 2017; Anne Hendricks et al. 2017) proposed. There is a semantic gap between vision and language. Video grounding is a fundamental task in the field of vision-language understanding ($e.g.$, other tasks - video captioning (Zhang et al. 2020c), video question answering (Kim et al. 2020)), which has recently rapidly developed (Zhang et al. 2019a; Mun et al. 2020).

In the early works, video grounding was performed in an *"propose-and-rank"* manner, which first generated extensive video segment proposals, then calculated the confidence scores of these proposals with heuristic tactics, finally ranked them to select the appropriate candidate. Using this manner, (Gao et al. 2017; Anne Hendricks et al. 2017; Liu et al. 2018b,a; Ge et al. 2019) retrieved the entire video with various sliding windows to cover activity instances, calculated boundary regression and realized clip-sentence alignment. Nevertheless, the sliding window with a fixed-length setting is insufficient to generate rich proposals with high IoU (Intersection over Union). Meanwhile, the sliding strategy neglects the boundary sensitiveness. Besides, without sliding windows, (Buch et al. 2017b) densely generated multi-scale proposals with several times [1, k] of interval segment sampling at each time step. (Chen et al. 2018; Yuan et al. 2019; Wang, Ma, and Jiang 2020; Chen and Jiang 2019) introduced the cross-entropy regression to assign each proposal with a confidence score. (Zhang et al. 2020a) gen-

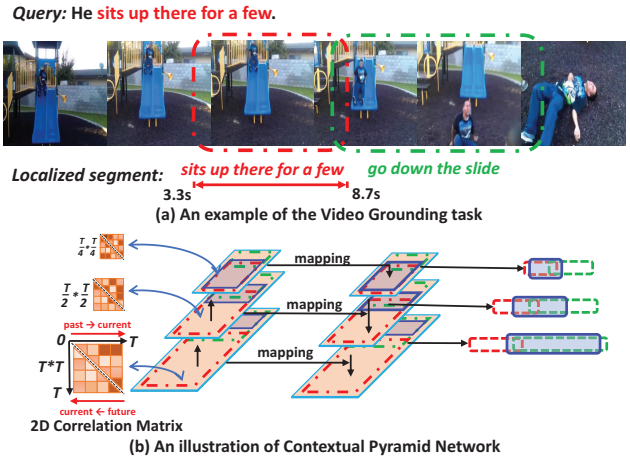

Figure 1: Illustration of an example for Video Grounding. There are two actions (red and green regions) in the video; which one is to be queried? To address this issue, we focus on the fine-grained temporal correlation. We employ a contextual pyramid network to learn 2D temporal correlation, inspecting different temporal scales to refine the predicted location. In the 2D correlation matrices, we investigate two cases of temporal clues: "past $\rightarrow$ current & current $\leftarrow$ future".

*Corresponding author.

erated candidate proposals by learning the boundary probability curve. Despite the success of these works, the enumerate candidate proposals are time-consuming and computational memory consuming.

To address these problems, (He et al. 2019; Wang, Huang, and Wang 2019; Wu et al. 2020) applied reinforcement learning for video grounding. The key idea is to design an agent to observe the video sequence, then learn a decision-making paradigm to regress the activity location. Recently, proposal-free based methods are proposed for video grounding and have achieved comparable progress due to the merit of model training efficiency. (Yuan, Mei, and Zhu 2019) was the first proposal-free method for video grounding, which employed a co-attention regression to predict the starting and ending times directly. (Ghosh et al. 2019; Rodriguez et al. 2020) investigated multi-modal fusion with various attention mechanisms. (Mun et al. 2020) inspected a local-to-global multi-modal interaction to locate the activity. These works are devoted to multi-modal modeling (*i.e.*, fusion or interaction), whereas the fine-grained temporal clues in videos have not been explored with enough attention.

As shown in Figure 1, the video contains easily confused semantic units (*e.g.*, multiple actions - *sit up there for a few* and *go down the slide* - occurring at different time stamps in the sequence). Observing salient frames, actions "*sit up*" and "*go down*" are vaguely to differentiate the boundaries. We attempt to address the location issue from the temporal clues. Different from score matching between candidate proposals and the query, we prefer the location regression optimization based on fine-grained temporal correlation between any two moments in the videos under the query (*e.g.*, the true action '*sit up*"). As shown in Figure 2, we first learn the multi-modal feature encoding of video and query; then we believe that fine-grained temporal semantics benefit the activity boundary location, we leverage a hierarchical pyramid to realize multi-scale temporal interaction (2D temporal correlation maps). In the pyramid, residual learning is applied to augment the temporal contexts between different temporal scales. Besides, a core component of the pyramid is the context-aware module (CAM) unit, which calculates the forward and backward temporal validation (from the past and the future to the current time step), *i.e.*, the fine-grained 2D correlation modeling. With the help of comprehensively hierarchical temporal clues, our method localizes activity with accurate boundaries. The contributions are summarized as follows:

- We present a novel proposal-free based framework named Contextual Pyramid Network (CPNet) for video grounding, which effectively captures multi-scale temporal correlation maps to recalibrate the temporal range of queried activity.
- Except for the context-aware hierarchy in CPNet, the CAM operation enhances discriminative activity regions through the fine-grained 2D correlation modeling (*i.e.*, past $\rightarrow$ current & current $\leftarrow$ future), and the temporal self-attentive regression performs effectively for proposal-free video grounding without any candidate proposals.
- Extensive experiments are conducted on three benchmark datasets and demonstrate the effectiveness of the proposed CPNet. Ablation studies and qualitative visualizations also verify each component of CPNet.

Related Work

Video grounding originates from temporal activity localization (Shou, Wang, and Chang 2016). The classical temporal activity localization task (Shou, Wang, and Chang 2016) targets to locate the starting and ending time of actions ($i.e.$, action detection) and identify the action labels ($i.e.$, action classification) in an untrimmed video. By contrast, the video grounding task (Gao et al. 2017) retrieves the required action in videos, but has to understand the textual semantics in query first and identify the required action. Previous works of action location preferred temporal sliding windows to cover candidate proposals (proposal-based methods) (Wang et al. 2011; Wang and Schmid 2013) and end-to-end deep learning frameworks (Simonyan and Zisserman 2014; Shou, Wang, and Chang 2016; Tran et al. 2015; Lin, Zhao, and Shou 2017). Related works of video grounding can be divided into two parts: proposal-based (candidate deep proposal generation) and proposal-free (end-to-end deep learning) methods.

Proposal-based methods. Early works (Gao et al. 2017; Anne Hendricks et al. 2017) addressed this task as a multi-modal matching problem, which performed in an "*propose-and-rank*" manner. They adopted sliding windows to extract candidate proposals and measured the distance (*i.e.*, matching score) between the candidates and query (Ge et al. 2019; Liu et al. 2018b); the proposal with the highest score was selected as the prediction result. Based on this manner, candidate proposals-based works are rapidly developed, *e.g.*, (Ge et al. 2019; Chen et al. 2018; Xu et al. 2019; Yuan et al. 2019; Wang, Ma, and Jiang 2020; Chen et al. 2020a; Zhang et al. 2020a; Zeng et al. 2020). Besides, a 2D temporal map describing all the proposal solutions is proposed and embedded into the deep learning framework (Zhang et al. 2020b).

Proposal-free methods. To alleviate the extensive computation of enumerated candidates in the above mentioned proposal-based methods, researchers applied *reinforcement learning* to predict activity boundaries in the end-to-end proposal-free manner (He et al. 2019; Wu et al. 2020). In these works, various agents regulated the temporal boundaries progressively based on its learning policy. (He et al. 2019) proposed the first reinforcement learning-based framework for video grounding; (Wu et al. 2020) and (Wang, Huang, and Wang 2019) respectively proposed a tree-structured policy and a semantic policy based reinforcement framework.

Apart from reinforcement learning, several proposal-free *regression methods* in an end-to-end manner have been proposed. (Ghosh et al. 2019) encoded video and text modalities into a joint representation and directly predicted the starting and ending times. (Yuan, Mei, and Zhu 2019) and (Rodriguez et al. 2020) utilized different query-video attentions to fuse multi-modal features. (Mun et al. 2020) further exploited both local and global contexts by bi-modal interaction. In this paper, we devote to the joint cross-modal representation learning with proposal-free regression

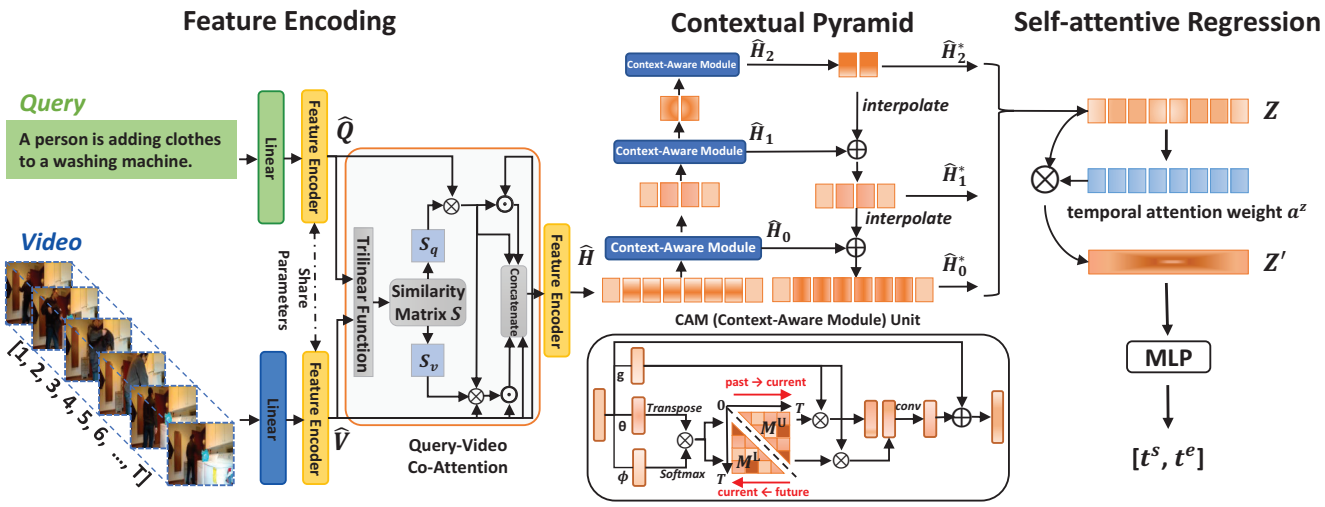

Figure 2: Overview of proposed Contextual Pyramid Network (CPNet) for video grounding. First, we solve Query-Video co-attention in a multi-modal encoding (embedding) stage. Subsequently, we investigate explicitly temporal context (past → current & current ← future) to refine the awareness of activity boundary. The contextual pyramid is used to explore and integrate multi-scale 2D temporal correlations. Finally, the integrated contextual-aware feature is fed to a self-attentive regression module to predict the starting and ending times of the required activity.

optimization for video grounding. And the main contribution is that we propose a contextual pyramid network with proposal-free regression, which investigates more fine-grained temporal correlations for video grounding.

Proposed Method

In this paper, we solve video grounding as an activity boundary regression problem. Given an untrimmed video $\boldsymbol{V}$ and a textual query $\boldsymbol{Q}$, the goal of video grounding is to identify a video segment with exact starting and ending times (t^s, t^e), which semantically corresponds to the query. Specifically, let visual features be $\boldsymbol{V} = \{v_1, \ldots, v_T\} \in \mathbb{R}^{d_v \times T}$ and textual features be $\boldsymbol{Q} = \{w_1, \ldots, w_L\} \in \mathbb{R}^{d_w \times L}$, where T and L are the respective length of video and query; we aim to learn a model $\mathcal{F}$ to locate the exact starting and ending times of the queried activity in the video.

$$< t^s, t^e >= \mathcal{F}(\boldsymbol{V}, \boldsymbol{Q}, \Theta), \tag{1}$$

where Θ is a collection of parameters of the model $\mathcal{F}$.

The pipeline of the proposed CPNet model is illustrated in Figure 2, which consists of three key components: feature encoding, contextual pyramid, and temporal self-attentive regression. Firstly, the feature encoding phase is responsible for encoding the features of video and query; it performs a multi-modal embedding. Subsequently, the contextual pyramid is employed to exploit valuable temporal clues, *i.e.*, to refine the activity region by learning multi-scale 2D-dim temporal correlation matrices ($T*T$, $\frac{T}{2}*\frac{T}{2}$, $\frac{T}{4}*\frac{T}{4}$). At last, a self-attentive regression is utilized to predict the activity boundary. To summarize, CPNet progressively replenishes the 2D temporal correlation (past → current & current ← future) using the contextual pyramid network in hierarchy ($T*T$, $\frac{T}{2}*\frac{T}{2}$, $\frac{T}{4}*\frac{T}{4}$), which refines the accuracy of activity localization.

Feature Encoding

For cross-modal feature encoding, we adopt a simplified QANet (Seo et al. 2017) as a backbone to encode visual features of $\boldsymbol{V}$ and textual features of $\boldsymbol{Q}$, where the parameters of QANet encoder are shared for both visual and textual encoding. The **Feature Encoder** module in Figure 2 mainly consists of positional encoding (PE), a stacked convolution block, multi-head self-attention, and layer normalization. Firstly, on the positional encoding (Vaswani et al. 2017) layer, the position information is added to the original features. After that, four depthwise separable convolution layers (Chollet 2017) in a stacked block are implemented for local context modelling; and then multi-head self-attention (Seo et al. 2017) is employed to build the long-range dependence in each feature sequence. Then, we obtain new visual features $\hat{\boldsymbol{V}} = \{\hat{v}_1, \ldots, \hat{v}_T\} \in \mathbb{R}^{d \times T}$ and textual features $\hat{\boldsymbol{Q}} = \{\hat{w}_1, \ldots, \hat{w}_L\} \in \mathbb{R}^{d \times L}$ with the same dimension d.

Based on new features $\hat{\boldsymbol{V}}$ and $\hat{\boldsymbol{Q}}$, we propose a Query-Video co-attention mechanism by integrating $\hat{\boldsymbol{Q}}$, $\hat{\boldsymbol{V}}$, and $\hat{\boldsymbol{Q}} \odot \hat{\boldsymbol{V}}$ to generate a similarity matrix $\boldsymbol{S}$:

$$\boldsymbol{S} = \mathbf{W}^s[\hat{\boldsymbol{Q}}, \hat{\boldsymbol{V}}, \hat{\boldsymbol{Q}} \odot \hat{\boldsymbol{V}}] \in \mathbb{R}^{T \times L}. \tag{2}$$

Each element $s_{i,j}$ in $\boldsymbol{S}$ represents the relationship of visual feature $\hat{v}_i$ with word $\hat{w}_j$. We conduct the $softmax$ function on each row of $\boldsymbol{S}$, and obtain attention map $\boldsymbol{S}_q$ which indicates the relevance of each word $\hat{w}$ to all the visual features $\hat{\boldsymbol{V}}$. Similarly, we conduct $softmax$ on each column of $\boldsymbol{S}$, and obtain attention map $\boldsymbol{S}_v$ which indicates the relevance of each visual feature $\hat{v}$ to all the words $\hat{\boldsymbol{Q}}$.

As the task is to locate temporal boundary in the video, we map all the visual and textual clues to the visual dimension. We explore the intra-relation of video itself $\mathbb{C}_{\boldsymbol{V}\rightarrow\boldsymbol{V}}$ and

the relationship between query and video $\mathbb{C}_{\boldsymbol{Q}\rightarrow\boldsymbol{V}}$. Here are $\mathbb{C}_{\boldsymbol{V}\rightarrow\boldsymbol{V}} = \boldsymbol{S}_q \cdot \boldsymbol{S}_v{}^{\mathsf{T}} \cdot \hat{\boldsymbol{V}}^{\mathsf{T}}$ and $\mathbb{C}_{\boldsymbol{Q}\rightarrow\boldsymbol{V}} = \boldsymbol{S}_q \cdot \hat{\boldsymbol{Q}}$. Up to now, we concatenate all the related visual clues as follows:

$$\begin{aligned}\boldsymbol{H} &= \mathbb{W}^h[\hat{\boldsymbol{V}}; \mathbb{C}_{\boldsymbol{Q}\rightarrow\boldsymbol{V}}; \hat{\boldsymbol{V}} \odot \mathbb{C}_{\boldsymbol{Q}\rightarrow\boldsymbol{V}}; \hat{\boldsymbol{V}} \odot \mathbb{C}_{\boldsymbol{V}\rightarrow\boldsymbol{V}}] \in \mathbb{R}^{4d\times T} \\ &= \mathbb{W}^h\Big[\hat{\boldsymbol{V}};\ \boldsymbol{S}_q\cdot\hat{\boldsymbol{Q}};\ \hat{\boldsymbol{V}}\odot\big(\boldsymbol{S}_q\cdot\hat{\boldsymbol{Q}}\big);\ \hat{\boldsymbol{V}}\odot\big(\boldsymbol{S}_q\cdot\boldsymbol{S}_v{}^{\mathsf{T}}\cdot\hat{\boldsymbol{V}}^{\mathsf{T}}\big)\Big]\end{aligned} \tag{3}$$

where $\mathbb{W}^h$ is a learnable parameter, $[;]$ is concatenate operation, $^{\mathsf{T}}$ denotes the transpose operations and $\odot$ is the element-wise multiplication. After that, we further use a fully connected layer to reduce the dimension of $\boldsymbol{H}$ to dimension $d \times T$ and implement the **Feature Encoder** module again, *i.e.*, $\hat{\boldsymbol{H}} = \mathbf{Feature\ Encoder}(\mathbf{FC}(\mathbf{H})) \in \mathbb{R}^{d\times T}$.

Contextual Pyramid

As temporal context clues in a video are crucial, we propose a contextual pyramid to mine rich temporal contexts through fine-grained hierarchical correlation at different 2D temporal scales ($T * T$, $\frac{T}{2} * \frac{T}{2}$, $\frac{T}{4} * \frac{T}{4}$). Motivated by non-local operation (Wang et al. 2018) to capture the global temporal dependencies, we focus on the temporal correlation of any two-time steps in each 2D temporal matrix. Generally, a normal non-local operation of positions t_i and t_j is given as:

$$\mathbf{y}_{t_i} = \frac{1}{\mathcal{C}(\mathbf{x})} \sum_{\forall j} f\left(\mathbf{x}_{t_i}, \mathbf{x}_{t_j}\right) g\left(\mathbf{x}_{t_j}\right), \tag{4}$$

where $\mathbf{x} \in \mathbb{R}^{d\times T}$ is an input feature sequence, and $\mathbf{y} \in \mathbb{R}^{d\times T}$ is the output feature sequence, $\mathcal{C}(\cdot)$ is a normalization function and $g(\cdot)$ is a feature embedding layer.

In this work, for video grounding, we model a new calculation of $\frac{1}{\mathcal{C}(\mathbf{x})}\sum_{\forall j} f\left(\mathbf{x}_{t_i}, \mathbf{x}_{t_j}\right)$ in Eq. 4. We transform it into a normalized correlation matrix $\boldsymbol{M}$. As shown in Figure 2, the element $m_{i,j}$ in $\boldsymbol{M}$ correlates the influence of feature $\mathbf{x}_{t_i}$ on feature $\mathbf{x}_{t_j}$. We split the correlation matrix $\boldsymbol{M}$ to two parts: upper triangular matrix $\boldsymbol{M}^{\mathbf{U}}$ which represents the influence of past $\rightarrow$ current and lower triangular matrix $\boldsymbol{M}^{\mathbf{L}}$ which represents the influence of current $\leftarrow$ future.

To calculate matrix $\boldsymbol{M}$, we firstly design the pairwise correlation function $f(\cdot)$ as:

$$f(\mathbf{x}_{t_i}, \mathbf{x}_{t_j}) = \theta(\mathbf{x}_{t_i})^{\mathsf{T}}\phi(\mathbf{x}_{t_j}) \in \mathbb{R}^{T\times T}, \tag{5}$$

where $\theta(\mathbf{x}_{t_i}) = \mathbf{W}^{\theta}\mathbf{x}_{t_i}$ and $\phi(\mathbf{x}_{t_j}) = \mathbf{W}^{\phi}\mathbf{x}_{t_j}$ refer to feature embedding by respective two fully-connected layers; $\mathbf{W}^{\theta}, \mathbf{W}^{\phi} \in \mathbb{R}^{d\times d}$ are learnable parameters. Then we conduct $softmax$ on each row of $f(\mathbf{x}, \mathbf{x})$ to generate a normalized correlation matrix $\boldsymbol{M}$:

$$\boldsymbol{M} = \mathrm{softmax}_{\mathrm{row}}\big(f(\mathbf{x}, \mathbf{x})\big) \in \mathbb{R}^{T\times T}, \tag{6}$$

To jointly consider the influences of both past and future to current, we explore a combination of $\boldsymbol{M}^{\mathbf{U}}$ and $\boldsymbol{M}^{\mathbf{L}}$. Thus, the normal non-local operation (Eq. 4) is transformed into the following formula:

$$\mathbf{x}' = \boldsymbol{M}^{\boldsymbol{U}} \otimes g(\mathbf{x}) + \boldsymbol{M}^{\mathbf{L}} \otimes g(\mathbf{x}) \in \mathbb{R}^{d\times T}, \tag{7}$$

where $\otimes$ denotes the matrix multiplication and $g(\cdot)$ is a fully-connected layer. We further employ a convolution operation and residual connection on $\mathbf{x}$ and $\mathbf{x}'$ as follows:

$$\mathbf{x}^* = Conv_3(\mathbf{x}') + \mathbf{x} \in \mathbb{R}^{d\times T}. \tag{8}$$

Until now, we elaborate a single context-aware layer in the proposed contextual pyramid. We define the whole calculation of $\mathbf{x}^*$ as $\mathbf{x}^* = \mathbf{CAM}(\mathbf{x})$. As shown in Figure 2, in the 3-layer pyramid, we build a collection of $\{\hat{\boldsymbol{H}}_0, \hat{\boldsymbol{H}}_1, \hat{\boldsymbol{H}}_2\}$ in a bottom-up pathway (Eq. 9), and then perform a top-down pathway $\{\hat{\boldsymbol{H}}_0^*, \hat{\boldsymbol{H}}_1^*, \hat{\boldsymbol{H}}_2^*\}$ (Eq. 10), where $\hat{\boldsymbol{H}}_i, \hat{\boldsymbol{H}}^*{}_i \in \mathbb{R}^{d\times\frac{T}{2^i}}, i \in \{0, 1, 2\}$. Each context-aware feature $\hat{\boldsymbol{H}}_{i-1}^*$ is added to the feature of its adjacent layer $\hat{\boldsymbol{H}}_i^*$.

$$\begin{cases} \hat{\boldsymbol{H}}_0 = CAM(\hat{\boldsymbol{H}}) \\ \hat{\boldsymbol{H}}_i = CAM(DownSampling(\hat{\boldsymbol{H}}_{i-1})) \end{cases} \tag{9}$$

$$\begin{cases} \hat{\boldsymbol{H}}_2^* = Conv_1(\hat{\boldsymbol{H}}_2) \\ \hat{\boldsymbol{H}}_{i-1}^* = \hat{\boldsymbol{H}}_{i-1} + UpSampling(\hat{\boldsymbol{H}}_i^*) \end{cases} \tag{10}$$

Finally, we upsize $\hat{\boldsymbol{H}}_1^*$ and $\hat{\boldsymbol{H}}_2^*$ to the same size as $\hat{\boldsymbol{H}}_0^*$ and concatenate them together and feed into a fully-connected layer. To the end, we get the final context feature $\boldsymbol{Z} \in \mathbb{R}^{d\times T}$.

Temporal Self-attentive Regression

Different from classical self-attention (Vaswani et al. 2017), we propose a temporal self-attention pooling to attend discriminative features for boundary regression, which is formulated as follows:

$$\boldsymbol{a}^z = \mathrm{softmax}(\mathbf{W}^T(\tanh(\mathbf{W}^z\boldsymbol{Z}+\mathbf{b}_1))+\mathbf{b}_2) \in \mathbb{R}^T, \tag{11}$$

where $\mathbf{W}^T \in \mathbb{R}^{1\times\frac{d}{2}}$ and $\mathbf{W}^z \in \mathbb{R}^{\frac{d}{2}\times d}$ are learnable parameters of two fully-connected layers, and $\mathbf{b}_1, \mathbf{b}_2 \in \mathbb{R}^T$ are the biases. Here, $\boldsymbol{a}^z$ is a to-be-learned attention solution. After that, we summarize all the features of $\boldsymbol{Z}$ to an intergraded vector $\boldsymbol{Z}'$:

$$\boldsymbol{Z}' = \sum_{t=0}^{T} \boldsymbol{a}_t^z \boldsymbol{Z}_t \in \mathbb{R}^d, \tag{12}$$

where t denotes the time stamp.

Finally, a two-layer MLP with *sigmoid* activation is further used to predict the starting time t^s and ending time t^e:

$$t^s, t^e = \mathrm{MLP}(\boldsymbol{Z}') \in [0, 1], \tag{13}$$

Loss Optimization

To optimize the proposed model, we adopt a multi-task loss $\mathcal{L}$ including self-attention pooling loss $\mathcal{L}_{cls}$ and temporal localization regression loss $\mathcal{L}_{reg}$. The total objective function is:

$$\mathcal{L} = \mathcal{L}_{cls} + \mathcal{L}_{reg}. \tag{14}$$

where $\mathcal{L}_{cls}$ is utilized to align the self-attention pooling vector $\boldsymbol{a}$ and the location label along the temporal dimension and $\mathcal{L}_{reg}$ is applied to evaluate the regression of starting and ending times (t^s, t^e). To be specific, with the self-attentive weight $\boldsymbol{a}^z$ in Eq. 11, we built $\mathcal{L}_{cls}$ based on the temporal calibration loss proposed in (Yuan et al. 2019):

$$\mathcal{L}_{cls} = -\frac{\sum_{t=0}^{T} \hat{\boldsymbol{a}}_t^z \log(\boldsymbol{a}_t^z)}{\sum_{t=0}^{T} \boldsymbol{a}_t^z} \tag{15}$$

where $\hat{\boldsymbol{a}}_t^z = 1$ when t belongs to the ground-truth (location region), otherwise $\hat{\boldsymbol{a}}_t^z = 0$.

The regression loss $\mathcal{L}_{reg}$ is formulated as follows:

$$\mathcal{L}_{reg} = R(t^s, \hat{t}^s) + R(t^e, \hat{t}^e), \tag{16}$$

where R denotes the smooth $L1$ loss function, $\hat{t}^s$ and $\hat{t}^e$ denote the ground-truth starting and ending times. In a nutshell, in our optimization solution, $\mathcal{L}_{reg}$ optimizes the specific values of (t^s, t^e) to be close to ($\hat{t}^s$, $\hat{t}^e$), and $\mathcal{L}_{cls}$ optimizes the temporal alignment of $\boldsymbol{a}^z$ and $\hat{\boldsymbol{a}}^z$.

Experiments

Experimental Setup

Datasets. In this work, we test three benchmark datasets for video grounding. ***1) Charades-STA*** (Gao et al. 2017) contains 6,672 daily life videos. The duration of the videos is 29.76 seconds on average. Each video has around 2.4 annotated moments, and the average duration of the moment is 8.2 seconds. The dataset involves 16,128 query-clip pairs and is split into training and testing parts with 12,408 pairs and 3,720 pairs, respectively. ***2) ActivityNet-Captions*** (Krishna et al. 2017) consists of 20K videos, and the average duration of the videos is 2 minutes, where the videos and queries are semantic-rich. On average, each video contains 3.65 queries, and each query has an average of 13.48 words. Limited to the unreleased "test" set, in this paper, we adopt the setting of "train" for training, "val_1" for validation, and "val_2" for testing in (Zhang et al. 2019b, 2020b). Thus, the dataset is split into the training/validation/testing sets of 37,421, 17,505, and 17,031 query-clip pairs. ***3) TACoS*** (Regneri et al. 2013) is a cooking activity dataset, which consists of 127 videos with the average length of 4.79 minutes. TACoS has much more temporally annotated video segments with queries per video. Each video has 148 queries on average. There are 10146, 4589, and 4083 query-clip pairs for training, validation, and testing, respectively.

Evaluation Metrics. The metric **"R@N, IoU@θ"** (Gao et al. 2017; Yuan, Mei, and Zhu 2019) is adopted, which records the percentage of test samples having larger IoU than θ in the top-N predicted segments. Since the proposed method is proposal-free, all the results are reported at R@1. We abbreviate it as **"IoU@θ"** in the following tables. Besides, **"mIoU"** is applied to denote the average IoU for all the test queries.

Implementation Details. We use a pre-trained C3D network (Tran et al. 2015) to extract visual features of videos in Charades-STA and ActivityNet-Captions. The C3D features of the TACoS dataset are provided by (Gao et al. 2017). And we also apply the I3D network (Carreira and Zisserman 2017) to extract visual features of Charades-STA. For textual features, we use the GloVe word embeddings (Pennington, Socher, and Manning 2014) with the dimension of 300 as word features. For the convenience of model training, we uniformly sample segments from each video with a fixed $T = 128$. The temporal action boundaries for each sentence are normalized to be in [0, 1]. The transformed dimension d of the feature encoding phase is set to 512. In the implementation of simplified QANet, the kernel size and the layer number of depthwise convolutions are set to 15 and 4, respectively; the head of multi-head self-attention is set to 8. We optimize the network by Adam optimizer (Kingma and Ba 2015) with a batch size of 100 and set the initial learning rate to 1×10^{-4} and gradient clipping of 0.5.

Method	IoU@			mIoU
	0.7	0.5	0.3	
CPNet w/o CP	39.85	59.73	71.24	51.54
CPNet w/o CAM	39.03	59.38	71.61	51.73
CPNet	38.74	**60.27**	**71.94**	**52.00**

Table 1: Ablation study of context-aware feature pyramid on the Charades-STA dataset with I3D features.

Method	Venue	Feature	IoU@		mIoU
			0.7	0.5	
MCN	ICCV'17	C3D	8.01	17.46	–
CTRL	ICCV'17	C3D	8.89	23.63	–
ACRN	SIGIR'18	C3D	7.64	20.26	–
ROLE	MM'18	C3D	–	12.12	–
MAC	WACV'19	C3D	12.20	30.48	–
QSPN	AAAI'19	C3D	15.80	35.60	–
ABLR	AAAI'19	C3D	9.01	24.36	–
SAP	AAAI'19	C3D	13.36	27.42	–
R-W-M	AAAI'19	C3D	–	36.70	–
SM-RL	CVPR'19	C3D	11.17	24.36	32.22
CBP	AAAI'20	C3D	18.87	36.80	35.74
GDP	AAAI'20	C3D	18.49	39.47	36.60
TSP-PRL	AAAI'20	C3D	17.69	37.39	37.22
PMI	ECCV'20	C3D	19.27	39.73	–
CPNet (Ours)	–	C3D	**22.47**	**40.32**	**37.36**
TMLGA	WACV'20	I3D	33.74	52.02	–
DRN	CVPR'20	I3D	31.75	53.09	–
LGI	CVPR'20	I3D	35.48	59.46	51.38
CPNet (Ours)	–	I3D	**38.74**	**60.27**	**52.00**

Table 2: Performance comparison on Charades-STA dataset.

Ablation Study

Here we mainly verify the effectiveness of the context-aware feature pyramid module on the Charades-STA dataset. There are two variants of the proposed model CPNet: (1) **CPNet w/o CP** removes the whole contextual pyramid module in CPNet. We directly implement the feature encoding and self-attentive regression to predict starting and ending times. (2) **CPNet w/o CAM** removes the temporal context-aware correlation (Eqs. 4 $\sim$ 8) in the context-aware feature pyramid. As shown in Table 1, **CPNet w/o CP** performs a large drop of mIoU compared with **CPNet**; it indicates that the usage of the contextual pyramid benefits to replenish multi-scale temporal clues. This merit refines the predication of the temporal boundary of queried activity. Compare **CPNet w/o CAM** with **CPNet**, the performance drops, especially on "IoU@0.5". It means that CAM further improves activity localization by enhancing discriminative correlation of any two moments in the videos, *i.e.*, the fine-grained 2D correlation modeling.

Method	Venue	IoU@		mIoU
		0.7	0.5	
MCN	ICCV'17	–	9.58	15.83
CTRL	ICCV'17	–	14.00	20.54
ACRN	SIGIR'18	–	16.17	24.16
TGN	SIGIR'18	11.86	27.93	29.17
QSPN	AAAI'19	13.60	27.70	–
ABLR	AAAI'19	–	36.79	36.99
SCDM	NeurIPS'19	19.86	36.75	–
TMLGA	WACV'20	19.26	33.04	–
CBP	AAAI'20	17.80	35.76	36.85
GDP	AAAI'20	–	39.30	39.80
PMI	ECCV'20	17.83	38.28	–
CPNet (Ours)	–	**21.63**	**40.56**	**40.65**

Table 3: Performance comparison on Activity-Captions dataset.

Method	Venue	IoU@		mIoU
		0.5	0.3	
MCN	ICCV'17	5.58	–	–
CTRL	ICCV'17	13.30	19.32	11.98
ACRN	SIGIR'18	14.62	19.52	–
ROLE	MM'18	–	–	–
TGN	SIGIR'18	20.21	25.13	17.93
CMIN	SIGIR'19	18.05	24.64	–
ABLR	AAAI'19	9.40	19.50	–
SAP	AAAI'19	18.24	–	–
SM-RL	AAAI'19	15.95	20.15	–
SCDM	NeurIPS'19	21.17	26.11	–
CBP	AAAI'20	24.79	27.31	21.59
GDP	AAAI'20	13.50	24.14	16.18
2D-TAN	AAAI'20	25.32	37.29	–
DRN	CVPR'20	23.17	–	–
VSLNet	ACL'20	24.03	29.61	24.11
CPNet (Ours)	–	**28.29**	**42.61**	**28.69**

Table 4: Performance comparison on TACoS dataset.

Comparison with Existing State-of-the-art Models

To verify the effectiveness of the proposed method CPNet, we compare it with the existing state-of-the-art methods as follows: 1) *Proposal-based methods*: **CTRL** (Gao et al. 2017), **MCN** (Anne Hendricks et al. 2017), **TGN** (Chen et al. 2018), **ACRN** (Liu et al. 2018a), **ROLE** (Liu et al. 2018b), **MAC** (Ge et al. 2019), **SAP** (Chen and Jiang 2019), **QSPN** (Xu et al. 2019), **MAN** (Zhang et al. 2019a), **CMIN** (Zhang et al. 2019b), **SCDM** (Yuan et al. 2019), **CBP** (Wang, Ma, and Jiang 2020), **2D-TAN** (Zhang et al. 2020b), **GDP** (Chen et al. 2020a), **DRN** (Zeng et al. 2020), **VSLNet** (Zhang et al. 2020a); 2) *Proposal-free methods*: **ABLR** (Yuan, Mei, and Zhu 2019), **TMLGA** (Rodriguez et al. 2020), **Ex-CL** (Ghosh et al. 2019), **LGI** (Mun et al. 2020), **PMI** (Chen et al. 2020b); 3) *Reinforcement Learning based methods*: **R-W-M** (He et al. 2019), **SM-RL** (Wang, Huang, and Wang 2019) and **TSP-PRL** (Wu et al. 2020).

As shown in Table 2, the proposed method CPNet achieves the best performance with both C3D and I3D visual features on the Charades-STA dataset. Especially, CPNet achieves far superior to all the other methods with "IoU@0.7" of 38.74 and "IoU@0.5" of 60.27 base on I3D features. The results on the ActivityNet Captions dataset are shown in Table 3, our model performs 21.63 and 40.56 on "IoU@0.7" and "IoU@0.5", respectively. The TACoS dataset is challenging since it contains dense queries and various variable-length queried activities in videos. As shown in Table 4, our method still achieves the best performances with the "IoU@0.5" of 28.29 and "IoU@0.3" of 42.61. Compared with the previous best performances of 2D-TAN (Zhang et al. 2020b), our model exhibits 11.73% improvement on "IoU@0.5". These results quantitatively demonstrate the superiority of our method over existing methods.

Figure 3: Different queries about the same video. CPNet locates respective response region with the self-attentive solution a^z.

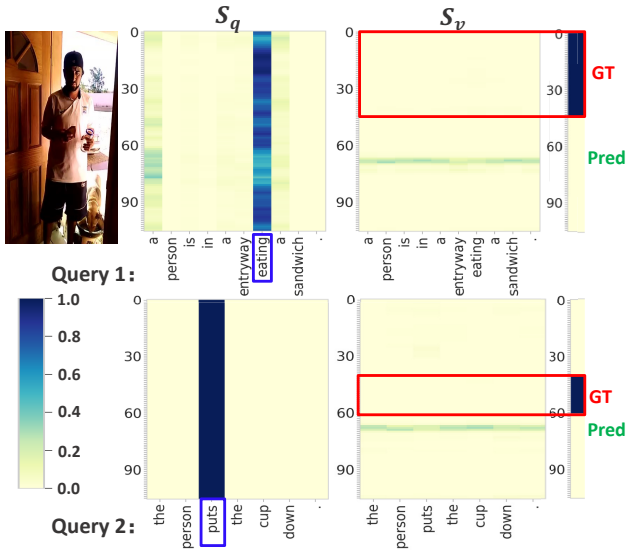

Figure 4: Visualization of attention maps S_q and S_v in preliminary Query-Video co-attention stage. Observing queries 1 and 2, S_q always attends on verbs, whereas S_v fails to attend on related temporal regions in videos but overfits high-frequency proposals in the dataset distribution.

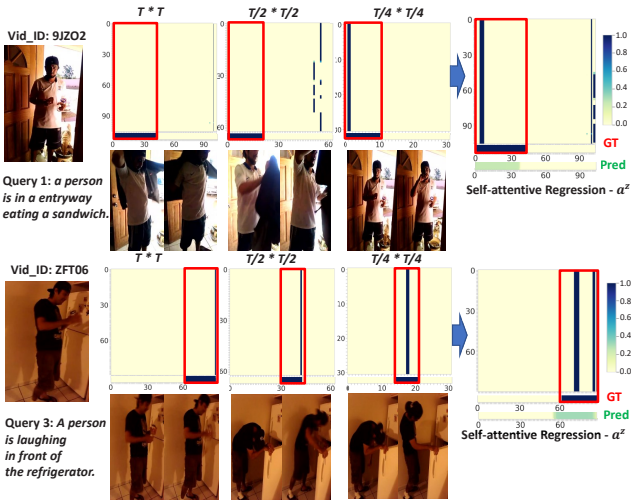

Figure 5: Visualization of hierarchical context-awareness in the pyramid architecture, *i.e.*, the 2D $T * T$ temporal correlation matrix (past $\rightarrow$ current & current $\leftarrow$ future). This module modifies the wrong visual attention in Figure 4. For $Query$ 1, CPNet firstly searches "person in the entryway" with wrong actions ("putting on clothes") at $T*T$ and $\frac{T}{2}*\frac{T}{2}$ scale and finally converts to "eating" at $\frac{T}{4}*\frac{T}{4}$ scale. For $Query$ 3, CPNet progressively supplements the response regions, namely enlarging the temporal receptive fields of the video grounding.

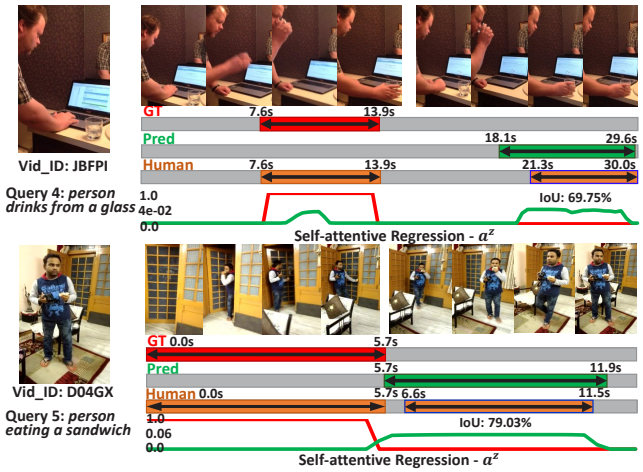

Figure 6: Some failure cases but correct grounding on videos (on the Charades-STA dataset). Due to coarse annotations, these correct results are deemed negative grounding, while CPNet properly identifies the activity boundaries.

Qualitative Visualization and Analysis

To discuss intuitive explanations on how does the contextual pyramid network takes effect for video grounding, we visualize the several exemplars in the Charades-STA dataset. As shown in Figure 3, the video contains multiple actions, *"eating a sandwich"*, *"puts the cup down"*, *etc*. Our method correctly locates different actions for both $Queries$ 1 and 2. The proposed method achieves high temporal IoU values of 92.05% and 88.57%. Taking $Queries$ 1 and 2 as examples, we respectively display the attention maps $\boldsymbol{S}_q$ of query and $\boldsymbol{S}_v$ of video on the preliminary feature encoding stage. As shown in Figure 4, $\boldsymbol{S}_q$ performs well, whereas $\boldsymbol{S}_v$ not. $\boldsymbol{S}_v$ is significantly influenced by the distribution of location labels in the datasets. Thus, merely implementing a query-to-video co-attention mechanism is insufficient for video grounding.

We further visualize the effect of the contextual pyramid module. As shown in Figure 5, we display the hierarchical context-awareness, *i.e.*, multi-scale 2D temporal correlation maps $(T*T)$, $(\frac{T}{2}*\frac{T}{2})$, $(\frac{T}{4}*\frac{T}{4})$. Even though there is wrong visual attention in $\boldsymbol{S}_v$, CPNet attempts to modify it. For $Query$ 1 in Figure 4, CPNet firstly searches *"person is in a entryway"* with a wrong action (*i.e.*, *"putting on clothes"*) at $(T*T)$ and $(\frac{T}{2}*\frac{T}{2})$ scales. Finally, at $(\frac{T}{4}*\frac{T}{4})$ scale, CPNet eventually attends on the queried activity *"eating"*. In other words, the pyramid enlarges the temporal fields to discover more useful or new clues. $Query$ 3 is a completely positive example of pyramid architecture. CPNet progressively replenishes the response region at each scale. The contextual pyramid network refines the coverage of highly responsive locations.

Besides, there are some interesting results. Examples in Figure 6 are evaluated as failure cases under the ground-truth labels. However, they are judged as the correct answer by human evaluation. For $Query$ 4, the person in the video indeed drank water twice. Consistently, the temporal attention weight a^z in self-attentive regression has high responses to both two actions. The first *"drink"* happens in few frames, whereas the second *"drink"* occurs in a larger temporal location that achieves the temporal IoU of 69.75%. However, limited by the nature of regression loss in this proposal-free method, our model only predicts one activity location. $Query$ 5 has a similar annotation too. From the cropped frames shown in Figure 6, we obviously observe that the second one is more discriminative. In summary, CPNet discovers more related visual contents and can correctly locate the activity, even with coarse annotations. The qualitative results in Figures 3-6 provide sufficient evidence that the proposed CPNet is capable of recalibrating the temporal range of activity by leverage multi-scale 2D correlation maps.

Conclusion

In this paper, we propose a novel proposal-free contextual pyramid network for video grounding. The contextual pyramid network explores a hierarchical architecture based on multi-scale 2D correlation maps with different temporal scales $T*T$, $\frac{T}{2}*\frac{T}{2}$, and $\frac{T}{4}*\frac{T}{4}$. Compared with works addressing highly responsive but inexact location regions, the proposed CPNet progressively recalibrates the queried activity's temporal boundary by aggregating these multi-scale 2D correlation maps. Experimental results on three benchmark datasets show its effectiveness.

Acknowledgements

This work is partly supported by the National Key Research and Development Program of China under Grant 2018YFB0804202, and partly supported by the National Natural Science Foundation of China (NSFC) under Grants 62020106007, 61876058, 61725203, and partly supported by the Fundamental Research Funds for the Central Universities under Grant JZ2020HGTB0020.

References

Anne Hendricks, L.; Wang, O.; Shechtman, E.; Sivic, J.; Darrell, T.; and Russell, B. 2017. Localizing moments in video with natural language. In *ICCV*, 5803–5812.

Buch, S.; Escorcia, V.; Ghanem, B.; Fei-Fei, L.; and Niebles, J. C. 2017a. End-to-End, Single-Stream Temporal Action Detection in Untrimmed Videos. In *BMVC*.

Buch, S.; Escorcia, V.; Shen, C.; Ghanem, B.; and Carlos Niebles, J. 2017b. Sst: Single-stream temporal action proposals. In *CVPR*, 2911–2920.

Carreira, J.; and Zisserman, A. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *CVPR*, 6299–6308.

Chen, J.; Chen, X.; Ma, L.; Jie, Z.; and Chua, T.-S. 2018. Temporally grounding natural sentence in video. In *EMNLP*, 162–171.

Chen, L.; Lu, C.; Tang, S.; Xiao, J.; Zhang, D.; Tan, C.; and Li, X. 2020a. Rethinking the Bottom-Up Framework for Query-based Video Localization. In *AAAI*, 10551–10558.

Chen, S.; Jiang, W.; Liu, W.; and Jiang, Y.-G. 2020b. Learning Modality Interaction for Temporal Sentence Localization and Event Captioning in Videos. In *ECCV*, 333–351.

Chen, S.; and Jiang, Y.-G. 2019. Semantic Proposal for Activity Localization in Videos via Sentence Query. In *AAAI*, 8199–8206.

Chollet, F. 2017. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, 1251–1258.

Gao, J.; Sun, C.; Yang, Z.; and Nevatia, R. 2017. Tall: Temporal activity localization via language query. In *ICCV*, 5267–5275.

Ge, R.; Gao, J.; Chen, K.; and Nevatia, R. 2019. MAC: Mining Activity Concepts for Language-based Temporal Localization. In *WACV*, 245–253.

Ghosh, S.; Agarwal, A.; Parekh, Z.; and Hauptmann, A. G. 2019. ExCL: Extractive Clip Localization Using Natural Language Descriptions. In *NAACL*.

He, D.; Zhao, X.; Huang, J.; Li, F.; Liu, X.; and Wen, S. 2019. Read, Watch, and Move: Reinforcement Learning for Temporally Grounding Natural Language Descriptions in Videos. In *AAAI*, 8393–8400.

Kim, J.; Ma, M.; Pham, T.; Kim, K.; and Yoo, C. D. 2020. Modality Shifting Attention Network for Multi-Modal Video Question Answering. In *CVPR*, 10106–10115.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

Krishna, R.; Hata, K.; Ren, F.; Fei-Fei, L.; and Carlos Niebles, J. 2017. Dense-captioning events in videos. In *ICCV*, 706–715.

Lin, T.; Zhao, X.; and Shou, Z. 2017. Single shot temporal action detection. In *ACM MM*, 988–996.

Liu, M.; Wang, X.; Nie, L.; He, X.; Chen, B.; and Chua, T.-S. 2018a. Attentive moment retrieval in videos. In *SIGIR*, 15–24.

Liu, M.; Wang, X.; Nie, L.; Tian, Q.; Chen, B.; and Chua, T.-S. 2018b. Cross-modal moment localization in videos. In *ACM MM*, 843–851.

Mun, J.; Cho, M.; ; and Han, B. 2020. Local-Global Video-Text Interactions for Temporal Grounding. In *CVPR*, 10810–10819.

Pennington, J.; Socher, R.; and Manning, C. D. 2014. Glove: Global vectors for word representation. In *EMNLP*, 1532–1543.

Regneri, M.; Rohrbach, M.; Wetzel, D.; Thater, S.; Schiele, B.; and Pinkal, M. 2013. Grounding action descriptions in videos. *TACL* 25–36.

Rodriguez, C.; Marrese-Taylor, E.; Saleh, F. S.; Li, H.; and Gould, S. 2020. Proposal-free temporal moment localization of a natural-language query in video using guided attention. In *WACV*, 2464–2473.

Seo, M.; Kembhavi, A.; Farhadi, A.; and Hajishirzi, H. 2017. Bidirectional attention flow for machine comprehension. *ICLR* .

Shou, Z.; Wang, D.; and Chang, S.-F. 2016. Temporal action localization in untrimmed videos via multi-stage cnns. In *CVPR*, 1049–1058.

Simonyan, K.; and Zisserman, A. 2014. Two-stream convolutional networks for action recognition in videos. In *NeurIPS*, 568–576.

Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; and Paluri, M. 2015. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 4489–4497.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *NeurIPS*, 5998–6008.

Wang, H.; Kläser, A.; Schmid, C.; and Liu, C.-L. 2011. Action recognition by dense trajectories. In *CVPR*, 3169–3176.

Wang, H.; and Schmid, C. 2013. Action recognition with improved trajectories. In *ICCV*, 3551–3558.

Wang, J.; Ma, L.; and Jiang, W. 2020. Temporally Grounding Language Queries in Videos by Contextual Boundary-aware Prediction. In *AAAI*, 12168–12175.

Wang, L.; Xiong, Y.; Wang, Z.; Qiao, Y.; Lin, D.; Tang, X.; and Van Gool, L. 2016. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 20–36.

Wang, W.; Huang, D.; and Wang, L. 2019. Language-Driven Temporal Activity Localization: A Semantic Matching Reinforcement Learning Model. In *CVPR*, 334–343.

Wang, X.; Girshick, R. B.; Gupta, A.; and He, K. 2018. Non-local Neural Networks. In *CVPR*, 7794–7803.

Wu, J.; Li, G.; Liu, S.; and Lin, L. 2020. Tree-Structured Policy based Progressive Reinforcement Learning for Temporally Language Grounding in Video. In *AAAI*, 12386–12393.

Xu, H.; He, K.; Plummer, B. A.; Sigal, L.; Sclaroff, S.; and Saenko, K. 2019. Multilevel language and vision integration for text-to-clip retrieval. In *AAAI*, 9062–9069.

Yuan, Y.; Ma, L.; Wang, J.; Liu, W.; and Zhu, W. 2019. Semantic Conditioned Dynamic Modulation for Temporal Sentence Grounding in Videos. In *NeurIPS*, 536–546.

Yuan, Y.; Mei, T.; and Zhu, W. 2019. To find where you talk: Temporal sentence localization in video with attention based location regression. In *AAAI*, 9159–9166.

Zeng, R.; Xu, H.; Huang, W.; Chen, P.; Tan, M.; and Gan, C. 2020. Dense Regression Network for Video Grounding. In *CVPR*, 10287–10296.

Zhang, D.; Dai, X.; Wang, X.; Wang, Y.-F.; and Davis, L. S. 2019a. Man: Moment alignment network for natural language moment retrieval via iterative graph adjustment. In *CVPR*, 1247–1257.

Zhang, H.; Sun, A.; Jing, W.; and Zhou, J. T. 2020a. Span-based Localizing Network for Natural Language Video Localization. In *ACL*, 6543–6554.

Zhang, S.; Peng, H.; Fu, J.; and Luo, J. 2020b. Learning 2D Temporal Adjacent Networks forMoment Localization with Natural Language. In *AAAI*, 12870–12877.

Zhang, Z.; Lin, Z.; Zhao, Z.; and Xiao, Z. 2019b. Cross-Modal Interaction Networks for Query-Based Moment Retrieval in Videos. In *SIGIR*, 655–664.

Zhang, Z.; Shi, Y.; Yuan, C.; Li, B.; Wang, P.; Hu, W.; and Zha, Z.-J. 2020c. Object Relational Graph With Teacher-Recommended Learning for Video Captioning. In *CVPR*, 13278–13288.