Fine-grained Iterative Attention Network for Temporal Language Localization in Videos

Xiaoye Qu[†]
School of Electronic Information and Communication, Huazhong University of Science and Technology & Huawei Tech., Hangzhou, China
xiaoye@hust.edu.cn

Pengwei Tang[†]
School of Electronic Information and Communication, Huazhong University of Science and Technology
pengweitang@hust.edu.cn

Zhikang Zou
Department of Computer Vision Technology (VIS), Baidu Inc., China
zouzhikang@baidu.com

Yu Cheng
Microsoft Dynamics 365 AI Research Redmond, WA, United States of America
yu.cheng@microsoft.com

Jianfeng Dong
School of Computer and Information Engineering, Zhejiang Gongshang University
dongjf24@gmail.com

Pan Zhou[*]
The Hubei Engineering Research Center on Big Data Security, School of Cyber Science and Engineering, Huazhong University of Science and Technology
panzhou@hust.edu.cn

Zichuan Xu
School of Software, Dalian University of Technology
z.xu@dlut.edu.cn

ABSTRACT

Temporal language localization in videos aims to ground one video segment in an untrimmed video based on a given sentence query. To tackle this task, designing an effective model to extract grounding information from both visual and textual modalities is crucial. However, most previous attempts in this field only focus on unidirectional interactions from video to query, which emphasizes which words to listen and attends to sentence information via vanilla soft attention, but clues from query-by-video interactions implying where to look are not taken into consideration. In this paper, we propose a Fine-grained Iterative Attention Network (FIAN) that consists of an iterative attention module for bilateral query-video in-formation extraction. Specifically, in the iterative attention module, each word in the query is first enhanced by attending to each frame in the video through fine-grained attention, then video iteratively attends to the integrated query. Finally, both video and query information is utilized to provide robust cross-modal representation for further moment localization. In addition, to better predict the target segment, we propose a content-oriented localization strategy instead of applying recent anchor-based localization. We evaluate the proposed method on three challenging public benchmarks: ActivityNet Captions, TACoS, and Charades-STA. FIAN significantly outperforms the state-of-the-art approaches.

CCS CONCEPTS

• **Information systems** → **Multimedia and multimodal retrieval; Video search**.

KEYWORDS

moment localization with natural language,temporal relationships, cross-modal retrieval

ACM Reference Format:
Xiaoye Qu[†], Pengwei Tang[†], Zhikang Zou, Yu Cheng, Jianfeng Dong, Pan Zhou[*], and Zichuan Xu. 2020. Fine-grained Iterative Attention Network for Temporal Language Localization in Videos. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394171.3414053

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

ACM ISBN 978-1-4503-7988-5/20/10…$15.00
https://doi.org/10.1145/3394171.3414053

1 INTRODUCTION

Localizing activities is a challenging yet pragmatic task for video understanding. In real scenarios, a video may contain multiple activities of interests which are associated with complex language dependencies, and cannot be classified to a pre-defined list of action classes. To solve this problem, temporal language localization [7] is proposed and attracts increasing attention recently. Formally,

[†]Equal Contribution.
[*]Corresponding author: Pan Zhou.

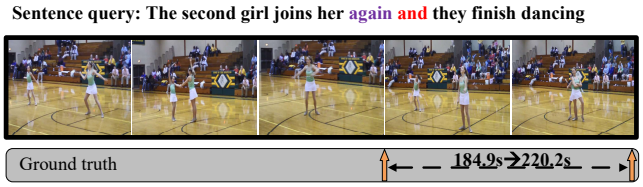

Figure 1: Temporal language localization is designed to localize a video segment with a start point (184.9s) and an end point (220.2s) in an untrimmed video corresponding to the given sentence query.

as shown in Figure 1, given an untrimmed video and a sentence query, this task is to automatically identify the start and end boundaries of the video segment semantically corresponding to the given sentence query. Comparing with other video researches, such as video retrieval and video captioning, this task is more challenging as fine-grained interactions between video and sentence need to be modelled to differentiate video segments in the long video.

Considering the query "The second girl joins her again and they finish dancing" depicted in Figure 1, which emphasizes that "the second girl" appears with a temporal relation "again". A model that only localizes the action "The second girl joins her" is not satisfactory, since this action appears twice in the video. Therefore, designing an effective model to collect grounding information from both modalities is central to task performance. From the video perspective, it is necessary to capture detailed temporal contents and decide which part does the sentence describe. From the sentence aspect, as several words or phrases can give clear cues to identify the target video segment, we should pay more attention to these sentence details. It is conceivable that attending to key words with the video and highlighting critical frames with query both contribute to precise location.

To solve this task, traditional methods [7, 8, 21] for temporal language localization first sample candidate video segments using sliding windows, and then fuse the sentence with each video segment representations separately to calculate the matching relationships. However, they integrate global sentence representations with video segment representations via matrix operations rather than explore the fine-grained interactions across video and sentence. Recently, some work [3, 4, 44] integrate the whole video with sentence query to generate a sentence-aware video representation for further location prediction. Specifically, they aggregate word features in the sentence for each frame with the widely used soft attention to obtain distinguishable frame features. While promising results have been achieved by these works, they fail to adequately exploit the sentence semantic information, since they merely explore the unidirectional interaction from video to the sentence. In this paper, we propose a novel Fine-grained Iterative Attention Network (FIAN), which iteratively performs attention for multi-modal location information gathering. The main idea is as straightforward as to collect grounding clues from both modalities. In specific, we design a cross-modal guided attention (CGA) to capture the fine-grained interactions from different modalities in multiple feature spaces. With CGA, each component from two modalities achieves comprehensively interaction. Furthermore, to integrate the attended information from CGA, we propose a cross-modal encoder to iteratively generate sentence-aware video and video-aware sentence representations. These two representations are then incorporated into the cross-modal feature space with filter controlling. To the end, we are able to obtain a robust cross-modal feature for subsequent temporal localization.

Besides, in order to fully utilize the cross-modal information for temporal localization, we devise a content-oriented location strategy. Different from traditional anchor-based prediction which simultaneously predicts multi-scale windows by feature at each time step, we utilize the complete features inside each window for prediction. In this way, each candidate window can be comprehensively evaluated, thus leading to precise localization. Overall, the main contributions of this work are:

- We propose a novel Fine-grained Iterative Attention Network (FIAN) for temporal language localization, in which fine-grained sentence and video grounding information is attended. To our best knowledge, we are the first work to explicitly utilize both video-aware sentence and sentence-aware video representations for accurate location.
- We devise a content-oriented localization strategy to better predict the temporal boundary. Based on the cross-modal information, it carefully measures the whole component in each candidate window for candidate moment evaluation.
- We conduct experiments on three public datasets: ActivityNet Captions, TACoS, and Charades-STA and FIAN significantly outperform the state-of-the-art by a large margin.

2 RELATED WORKS

2.1 Temporal Action Localization

Temporal action localization aims to locate action instances in an untrimmed video. Approaches for this task can be classified into three categories: (1) methods performing frame or segment-level classification where the smoothing and merging steps are substantially required to obtain the temporal boundaries [29, 41]. (2) methods adopting a two-stage framework involving proposal generation, classification and boundary refinement [5, 30, 35, 45]. (3) methods developing a end-to-end architecture integrating the proposal generation and classification [19, 33, 37]. Although these works have achieved promising performance, they are limited to a pre-defined list of actions. Thus, temporal language localization is proposed to tackle this issue by introducing the language query.

2.2 Language Localization in images

Language localization in images is also called "locating referring expressions in images", which aims to localize the object instance in an image described by a referring expression phrased in natural language. Traditional works in this field solve this task using a CNN/LSTM framework [13, 24, 25]. The LSTM takes as input a region-level CNN feature and a word vector at each time step, and aims to maximize the likelihood of the expression given the referred region. Another line of work treats referring expression comprehension as a metric learning problem [23, 28, 34], whereby the expression feature and the region feature are embedded into a common feature space to measure the compatibility. The focus of

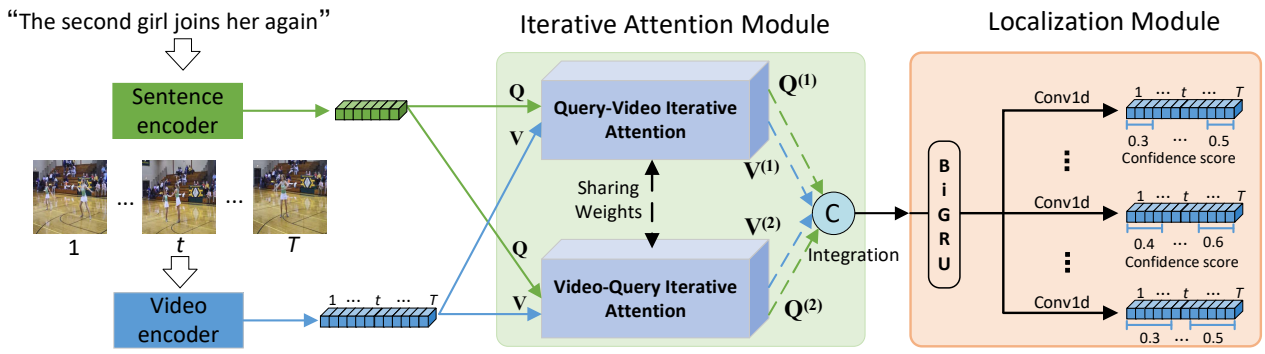

Figure 2: An overview of our Fine-grained Iterative Attention Network (FIAN) for temporal language localization which consists of three parts: (1) Video and sentence query are encoded into feature representations. (2) Based on the two modal information, symmetrical iterative attentions generate both video-aware sentence and sentence-aware video representations for two attention branches. Then all representations are integrated to produce the cross-modal information. (3) The localization module finally locates the boundaries of target moments.

these approaches lies in how to define the matching loss function. These approaches tend to use a single feature vector to represent the expression and the image region. To overcome this limitation of monolithic features, self-attention mechanisms have been used to decompose the expression into sub-components and learn separate features for each of the resulting parts [12, 38, 43].

2.3 Temporal language localization in videos

Temporal language localization in videos requires understandings of both complex video scenes and natural language, which is a new task introduced recently [7, 11]. Several early methods [8, 21] sample video segments through dense sliding windows. After aggregating the two modality information using simple matrix operation, they measure the similarity between these candidate video segments and sentence query in a common embedding space. In this way, the temporal language localization degrades into a multi-modal matching problem. While simple and effective, these methods just consider global representation and fail to exploit the fine-grained interaction between modalities.

Recently, in order to avoid the redundant computation by predefined sliding windows, some works propose to integrate sentence query information with the whole video first, and then predict the temporal boundary by directly regressing the start and end points [4, 22, 40] or designing multi-scale temporal anchors [42, 44] which follow the same spirit of anchor box in object detection. These methods closely integrate the video and sentence representations and obtain improved performance. They usually adopt frame-by-word interactions for language and video feature fusion, which aggregates sentence information for each frame according to the normalized similarity. However, these methods lack fully exploring the sentence semantic information which plays an important role in distinguishing the ambiguous frames and strengthening the integrated video representations for precise localization. Our proposed FIAN captures the more fine-grained interaction between two modalities, thus leading to more distinguished features. Moreover, our work firstly utilizes the video-aware sentence representation for enhancing the sentence-aware video representation, in order to obtain a robust cross-modal information for following moment prediction.

3 MODEL DESCRIPTION

In this section, we first introduce the basic formation of temporal language localization. Then we present the detailed structure of our fine-grained iterative attention network, which consists of feature representation, symmetrical iterative attention module, and moment localization module. The whole structure of our network is shown in Figure 2.

3.1 Problem Formulation

Given an untrimmed video V, and a natural language query Q, the task aims to localize the temporal video segment described by the query. We represent the video frame-by-frame as $V = \{v_i\}_{i=1}^{n_v}$, where v_i is the feature of i-th frame of the video and n_v is the frame number of the video. Similarly, the given natural language query can be denoted as $Q = \{q_i\}_{i=1}^{n_q}$ word-by-word, where q_i is the feature of the corresponding word. Our goal is to predict the start and end temporal coordinates (s, e) in the video.

3.2 Representation For Language and Video

Query representation For query encoding, we obtain its embedding vector by the Glove [26]. To aggregate the contextual information, we then feed the sentence embedding to the bi-directional GRU network [6]. Formally, given word features $Q = (q_1, q_2, ..., q_{n_q})$, we obtain the contextual representation of each word by:

$$h_i^f = \text{GRU}_q^f(q_i, h_{(i-1)}^f) \tag{1}$$

$$h_i^b = \text{GRU}_q^b(q_i, h_{(i-1)}^b) \tag{2}$$

$$h_i^q = [h_i^f; h_i^b] \tag{3}$$

where GRU_q^f and GRU_q^b are the forward and backward GRU network. Thus, we get the contextual query representations h_q =

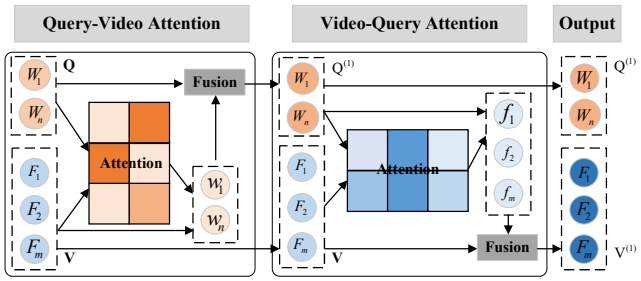

Figure 3: The structure of query-video iterative attention. The query first attends to video to obtain integrated query information and then the video iteratively attends to the integrated query for further enhanced video. During each attention mechanism, each word in the query and each frame in the video achieve fine-grained interactions.

$(h_1^q, h_2^q, ..., h_{n_q}^q)$ by concatenating the forward and backward hidden state at each time step.

Video representation Similar to [44], for video information encoding, we first extract features from the pre-trained 3D network and then employ the self-attention [32] to learn the semantic dependencies in the long video context. Also, in order to learn the contextual information within the video, we feed the video features into the bi-directional GRU network to further incorporate the contextual information. In the similar way, we can get the video representation $h_v = (h_1^v, h_2^v, ..., h_{n_v}^v)$.

3.3 Iterative Attention Module

After obtaining the video and sentence representation, our aim is to achieve fine-grained cross-modal interaction and collect grounding information from both modalities. To this end, as shown in Figure 2, we design an symmetrical iterative attention module which contains two sub-modules sharing same architecture but reverse input, namely query-video iterative attention and video-query iterative attention. As in Figure 3, the query-video iterative attention successively performs query-video attention and video-query attention to obtain integrated query and video information. Next, we will describe the most important components in this architecture including attention mechanism and information fusion.

3.3.1 *Cross-Modal Guided Attention.*

In our localization task, a robust model needs to find out the exact starting and ending point of the video segment, however, a simple soft attention mechanism which captures interaction from one specific attention space may not be enough to solve this problem. Thus, we devise a cross-modal guided attention (CGA) consisting of cross-modal multi-head attention (CMA) and information gate for this task, as shown in Figure 4 (a).

Here we first introduce CMA whose inputs are query $Q \in \mathbb{R}^{n_q \times d_h}$, and value $V \in \mathbb{R}^{n_v \times d_h}$ from two different modalities, where n_q, n_v represent the numbers of queries and values and d_h represents feature dimension. The multi-head attention is composed of n parallel heads and each head performs the scaled dot-product

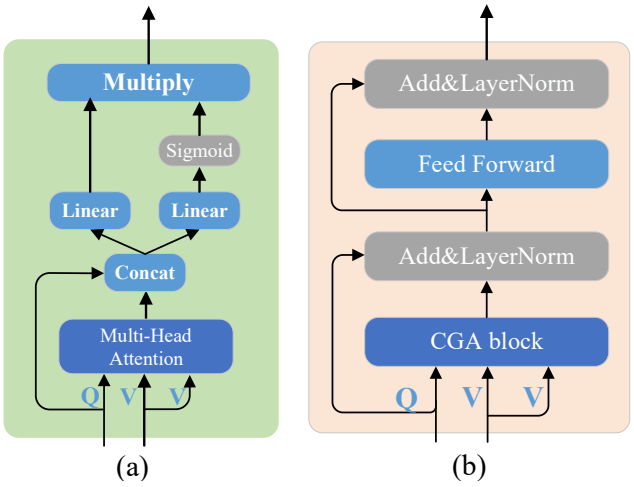

Figure 4: Left: The structure of cross-modal guided attention (CGA). Right: The architecture of Cross-Modal Encoder. We name this encoder as a Q-V encoder which generates V-aware Q representation.

attention as:

$$\text{Att}_i(Q, V) = \text{Softmax}\left(\frac{QW_i^Q \left(VW_i^K\right)^\top}{\sqrt{d_k}}\right) VW_i^V \tag{4}$$

where $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d_h \times d_k}$ are learnable parameter matrices of projections. $d_k = d_h/n$ is the size of the output features for each head. The outputs yielded by each head are concatenated and then projected again to construct the final output:

$$\text{CMA}\ (Q, V) = \ \text{Concat}\ (\text{head}_1, \ldots, \text{head}_n) W^O \tag{5}$$

where $\text{head}_i = \text{Att}_i(Q, V)$, $W^O \in \mathbb{R}^{d_h \times d_h}$ is the linear parameter.

With the help of CMA, we can capture the cross-modal interactions from multiple different aspects corresponding to various attention heads. Furthermore, to obtain more distinguishing features, we employ an information gate [14] on CMA to refine the output information as noisy information could be captured by partial heads. In this process, the outputs from CMA are filtered according to the query semantic meaning, namely the attended information is guided by query information. We name this guided CMA as CGA and the CGA is computed as:

$$\begin{aligned} \text{CGA}(Q, V) =& \sigma(W_q^g Q + W_v^g \text{CMA}(Q, V) + b^g) \\ & \odot (W_q^i Q + W_v^i\ \text{CMA}\ (Q, V) + b^i) \end{aligned} \tag{6}$$

where $W_q^g, W_v^g, W_q^i, W_v^i \in \mathbb{R}^{d_h \times d_h}$, $b^g, b^i \in \mathbb{R}^{d_h}$, and $\odot$ denotes element-wise multiplication. In practice, we keep $W_q^g = W_q^i$ and $W_v^g = W_v^i$ by sharing projection weights at training stage. After the CGA, we can achieve effective interactions between two modalities.

3.3.2 *Information Fusion.*

After CGA, we attempt to fuse query Q with attended information $\text{CGA}(Q, V)$ to enhance the the query representation. Inspired by Transformer's encoder, we apply a cross-modal encoder structure

in Figure 4 (b) to integrate these two branches. Here we just present the feed-forward procedure by:

$$\mathrm{FFN}(X) = \max(0, XW^{(1)} + b^{(1)})W^{(2)} + b^{(2)} \tag{7}$$

but omit the internal shortcut connection [10], and layer normalization [1]. $W^{(1)}$ and $W^{(2)}$ are linear transformation parameters, $b^{(1)}$ and $b^{(2)}$ are bias parameters. Thus, the cross-modal encoder can be denoted as:

$$\mathrm{Encoder}(Q, V) = \mathrm{FFN}(\mathrm{CGA}(Q, V)) \tag{8}$$

This encoder intrinsically captures the fine-grained interaction from query to value as it computes the similarity for each query element to all value elements, and generates a **V-aware Q representation**. We call this encoder as **Q-V encoder**. Similarly, the V-Q encoder can yield a Q-aware V representation.

Based on above cross-modal encoder, the query-video iterative attention in Figure 3 can be denoted as:

$$Q^{(1)} = \mathrm{Encoder}(Q, V), \;\; V^{(1)} = \mathrm{Encoder}(V, Q^{(1)}) \tag{9}$$

Analogously, the video-query iterative attention in Figure 2 can be presented as:

$$V^{(2)} = \mathrm{Encoder}(V, Q), \;\; Q^{(2)} = \mathrm{Encoder}(Q, V^{(2)}) \tag{10}$$

These two iterative attention branch compensate each other and contributes to generating robust cross-modal information.

3.3.3 *Video-enhanced Integration.*

With the symmetrical iterative attention module, we obtain two sentence-aware video representations $V^{(1)}, V^{(2)} \in \mathbb{R}^{n_v \times d_h}$ and two video-aware query representations $Q^{(1)}, Q^{(2)} \in \mathbb{R}^{n_q \times d_h}$. To complement the sentence-aware video representation for further enhanced video information, we first project each query representation to same length as video by:

$$\hat{Q}^{(1)} = W_1 Q^{(1)} + b_1, \;\; \hat{Q}^{(2)} = W_2 Q^{(2)} + b_2 \tag{11}$$

where $W_1 \in \mathbb{R}^{n_v \times n_q}$ and $W_2 \in \mathbb{R}^{n_v \times n_q}$ are matrices for linear transformation, $b_1 \in \mathbb{R}^{n_q}$ and $b_2 \in \mathbb{R}^{n_q}$ are the bias terms. We then combine the corresponding modality information by column as:

$$\hat{V} = \mathrm{Concat}[V^{(1)}, V^{(2)}], \;\; \hat{Q} = \mathrm{Concat}[\hat{Q}^{(1)}, \hat{Q}^{(2)}] \tag{12}$$

Finally, we devise a filter to control the ratio of query information to incorporate with the video features.

$$\boldsymbol{r} = \sigma(\hat{Q}W^r + b^r) \tag{13}$$

$$M = \mathrm{LayerNorm}(\hat{V} + \boldsymbol{r} \odot \hat{Q}) \tag{14}$$

where $W^r \in \mathbb{R}^{2d_h \times 2d_h}$, $b^r \in \mathbb{R}^{2d_h}$ are learnable parameters and $\odot$ denotes element-wise multiplication.

3.4 Localization Module

In this section, we introduce the localization module. Previous anchor-based predictions [3, 44] simultaneously score a set of candidate moments with multi-scale windows at each time step, however, the confidence scores of different window scales are predicted based on the same point feature. In this paper, we devise a content-oriented strategy which differs from traditional anchor-based localization. During our localization process, the confidence scores of candidate moments with multi-scale windows can be calculated with the entire features in the corresponding time duration.

With the integrated cross-modal representation M, we first apply a bi-directional GRU network to aggregate contextual information, resulting in final representation sequence $\hat{M} = (f_1, f_2, \cdots, f_{n_v})$. To predict the target video segment, we pre-define several candidate moments for grounding by dividing representation sequence into overlapped windows. In practice, we have multi-size windows. Taking one window size for example, as shown in Figure 5, j-th candidate moment can be denoted as $C_j = (\hat{s}_j, \hat{e}_j)$, where $\hat{s}_j, \hat{e}_j$ are the starting and ending coordinates between 1 to n_v. Subsequently, our goal is to score these candidate moments and adjust the temporal boundary for them. Here we adopt temporal 1D convolution to process features of each candidate moment to produce the corresponding confidence score cs_j and temporal offsets $\hat{\delta}_j = (\hat{\delta^s}, \hat{\delta^e})$. The temporal 1D convolution can be simply denoted as $\mathrm{Conv1d}(C_f, \theta_k, \theta_s)$, where C_f, θ_k, θ_s are filter numbers, kernel size and stride size, respectively. Given a representation sequence corresponding to candidate moment C_j, we apply two distinct convolution $\mathrm{Conv1d}(1, \theta_k, \theta_s)$ and $\mathrm{Conv1d}(2, \theta_k, \theta_s)$ for producing confidence score and temporal offsets separately. Then the confidence scores will be normalized by sigmoid function. θ_k is same as the window size and θ_s is equal to window size minus overlap length. In this way, temporal 1D convolution can properly process all candidate moments.

3.5 Training Loss and Inference

We first compute the IoU (Intersection over Union) o_j between each candidate moment $(\hat{s}_j, \hat{e}_j)$ with ground truth (s, e). If o_j is larger than a threshold value τ, this candidate moment is viewed as positive sample, reverse as the negative sample. Thus we can obtain N_{pos} positive samples and N_{neg} negative samples in total.

Alignment Loss: We adopt an alignment loss to align the predicted confidence scores cs and IoU o, which promotes candidate moments with higher IoUs achieve higher confidence scores, the alignment loss is calculated as:

$$\mathcal{L}_j = o_j \log\left(cs_j\right) + \left(1 - o_j\right) \log\left(1 - cs_j\right) \tag{15}$$

$$\mathcal{L}_{align} = \sum_{z \in \{pos, neg\}} -\frac{1}{N_z} \sum_{j}^{N_z} \mathcal{L}_j \tag{16}$$

Boundary Loss: As the boundaries of pre-defined candidate moments are relatively coarse, we devise a boundary loss for N_{pos} positive samples to promote exploring the precise start and end points. The boundary loss is:

$$\mathcal{L}_b = \frac{1}{N_{pos}} \sum_{j}^{N_{pos}} \mathcal{R}_1\left(\hat{\delta}_j^s - \delta_j^s\right) + \mathcal{R}_1\left(\hat{\delta}_j^e - \delta_j^e\right) \tag{17}$$

where $\mathcal{R}_1$ represents the smooth L1 function, δ_j^s and δ_j^e are the starting and ending offsets of the ground-truth coordinates compared to the pre-defined candidate coordinates, and $\hat{\delta}_j^s$ and $\hat{\delta}_j^e$ are the predicted offsets.

Joint Loss: We adopt α to control the balance of the alignment loss and boundary loss:

$$\mathcal{L} = \mathcal{L}_{align} + \alpha \mathcal{L}_b \tag{18}$$

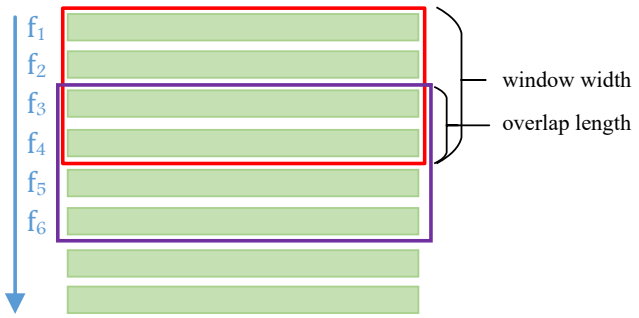

Figure 5: Red and pink rectangles denote representation sequence corresponding to candidate moments which can be marked as $C_1 = (1, d), C_2 = (1 + d - p, 2d - p)$, where d and p represent window width and overlap length. More candidate moments can be sampled by overlapped sliding windows in the same way.

Inference: We rank all candidate moments according to their predicted confidence scores, and then "Top-n (Rank@n)" candidates will be selected with non maximum suppression.

4 EXPERIMENTS

4.1 Datasets

We validate our proposed approach on three datasets.

ActivityNet Captions [16]: It is a large dataset which contains 20k videos with 100k language descriptions. The video contents of this dataset are diverse and open. This dataset pays attention to complicated human activities in daily life. Following public split, we use 37,417, 17,505, and 17,031 sentence-video pairs for training, validation, and testing respectively.

TACoS [27]: It collects 127 long videos, which are mainly about cooking scenarios. On this dataset, we use the same split as [7], which has 10146, 4589 and 4083 sentence-video pairs for training, validation, and testing.

Charades-STA [7]: Gao et al.[7] first label the start and end time of moments) of this dataset with language descriptions. It consists of 9,848 videos of daily life indoors activities. There are 12,408 sentence-video pairs for training and 3,720 pairs for testing.

4.2 Evaluation Metrics

Following previous works [39, 44], we adopt "Rank @n, IoU @m" as evaluation metrics. "Rank @n, IoU @m" is defined as the percentage of the language queries having at least one matched retrieval (IoU with ground-truth moment is larger than m) in the top-n retrieved moments.

4.3 Implementation Details

Following [39], we adopt C3D [31] for ActivityNet Captions and TACoS, and I3D [2] for Charades-STA to encode videos. Next, a fully-connected layer is used to reduce video feature dimension to 512. We set the length of video feature sequences to 200 for ActivityNet Captions and TACoS, and 64 for Charades-STA. For too long videos, we downsample them uniformly. During prediction, we use convolution kernel size of [16, 32, 64, 96, 128, 160, 192] for ActivityNet Captions, [8, 16, 32, 64] for TACoS, and [16, 24, 32, 40] for Charades-STA. We then set stride size as 0.25, 0.125, 0.125 of kernel size for ActivityNet Captions, TACoS and Charades-STA, respectively. The trade-off parameter α is set 0.001 for ActivityNet Captions, 0.005 for TACoS and Charades-STA. In symmetrical mutual attention, we set heads as 8 for ActivityNet Captions and TACoS, and 4 for Charades-STA. The positive threshold value τ is set to 0.55. We train our model using Adam optimizer [15] with learning rate of 8×10^{-4}, 4×10^{-4}, and 4×10^{-4} for ActivityNet Captions, TACoS, and Charades-STA, respectively. The batch size is set to 128, 64, and 64, respectively. Hidden dimension of all bi-directional GRUs is set as 512 in our model.

Table 1: Performance compared with previous methods on the Activity Captions dataset.

Method	R@1 IoU=0.3	R@1 IoU=0.5	R@1 IoU=0.7	R@5 IoU=0.3	R@5 IoU=0.5	R@5 IoU=0.7
MCN [11]	39.35	21.36	6.43	68.12	53.23	29.70
CTRL [7]	47.43	29.01	10.34	75.32	59.17	37.54
QSPN [36]	45.30	27.70	13.60	75.70	59.20	38.30
TripNet [9]	48.42	32.19	13.93	-	-	-
ACRN [21]	49.70	31.67	11.25	76.50	60.34	38.57
ABLR [40]	55.67	36.79	-	-	-	-
CMIN [44]	63.61	43.40	23.88	80.54	67.95	50.73
SCDM [39]	54.80	36.75	19.86	77.29	64.99	41.53
FIAN	**64.10**	**47.90**	**29.81**	**87.59**	**77.64**	**59.66**

Table 2: Performance compared with previous methods on the TACoS dataset.

Method	R@1 IoU=0.1	R@1 IoU=0.3	R@1 IoU=0.5	R@5 IoU=0.1	R@5 IoU=0.3	R@5 IoU=0.5
MCN [11]	14.42	-	5.58	37.35	-	10.33
CTRL [7]	24.32	18.32	13.30	48.73	36.69	25.42
QSPN [36]	25.31	20.15	15.23	53.21	36.72	25.30
ABLR [40]	34.70	19.50	9.40	-	-	-
TripNet [9]	-	23.95	19.17	-	-	-
ACRN [21]	24.22	19.52	14.62	47.42	34.97	24.8
CMIN [44]	32.48	24.64	18.05	**62.13**	38.46	27.02
SCDM [39]	-	26.11	21.17	-	40.16	32.18
FIAN	**39.55**	**33.87**	**28.58**	**56.14**	**47.76**	**39.16**

4.4 Performance Comparison

We compare our FIAN with existing state-of-the-art methods, which can be classified into: (1) Sliding window-based models: MCN [11], CTRL [7], ACRN [21], QSPN [36], TripNet [9]. (2) Recent works generate sentence-aware video representations: ABLR [40], MAN [42], CMIN [44], SCDM [39]. The performance comparisons of previous methods on three public benchmarks are shown in Table 1-3. We can observe that the FIAN achieves a new state-of-the-art performance under nearly all evaluation metrics and benchmarks.

Table 3: Performance compared with previous methods on the Charades-STA dataset.

Method	R@1 IoU=0.5	R@1 IoU=0.7	R@5 IoU=0.5	R@5 IoU=0.7
MCN [11]	17.46	8.01	48.22	26.73
CTRL [7]	23.63	8.89	58.92	29.52
QSPN [36]	35.60	15.80	79.40	45.40
TripNet [9]	36.61	14.50	-	-
ACRN [21]	20.26	7.64	71.99	27.79
MAN [42]	46.53	22.72	86.23	53.72
SCDM [39]	54.44	33.43	74.43	58.08
FIAN	**58.55**	**37.72**	**87.80**	**63.52**

Table 4: Ablation study on the TACoS dataset.

Method	R@1 IoU=0.1	R@1 IoU=0.3	R@1 IoU=0.5	R@5 IoU=0.1	R@5 IoU=0.3	R@5 IoU=0.5
FIAN-Soft	34.97	27.53	21.64	52.27	43.99	32.52
FIAN-CMA	36.82	29.27	23.05	53.86	44.40	34.61
FIAN-VQ	36.32	29.32	24.03	53.56	44.72	37.08
FIAN-VQMA	36.91	30.59	25.54	54.37	45.28	37.30
FIAN-QVMA	36.59	30.10	24.81	53.15	43.84	35.42
FIAN-Concat	38.03	31.59	26.01	54.86	46.14	36.71
FIAN-Matrix	38.31	32.21	25.96	55.45	45.43	37.50
FIAN	**39.55**	**33.87**	**28.58**	**56.14**	**47.76**	**39.16**

ActivityNet Captions. As we can see from Table 1, FIAN brings 5.93% improvement in the strict "R@1, IoU=0.7" metric, and outperforms around 10% in the all R@5 metrics than the previous state-of-the-art method in absolute values.

TACoS. Table 2 shows that our proposed FIAN achieves around 7% higher improvements than previous methods in all metrics except the "Rank@5, IoU=0.1". Compared to other datasets, the performances of TACoS are worst, which may come from similar background and objects during the whole video. However, it is worth noting that FIAN still achieves significant improvements, which demonstrates that our proposed model can effectively differentiate the visual similarity frames.

Charades-STA. In Table 3, we can observe that FIAN surpasses much over SCDM, especially 13.37% in "Rank5, IoU=0.5" and 5.44% in "Rank 5, IoU=0.7", demonstrating that FIAN has powerful capacity of generating robust cross-modal information for prediction.

Overall Analysis. Compared to state-of-the-art methods, FIAN gains significant improvements, especially in the stiff "Rank1, IoU=0.5" or "Rank1, IoU=0.7" metrics. The results of sliding window-based methods are obviously inferior to recent methods. ABLR directly predicts start and end points based on the features rather than design windows for prediction, thus leading to inaccurate results. MAN and CMIN only exploit sentence-aware video representation, while SCDM further devises a mechanism to dynamically modulate the sentence-aware video representation. Although these methods achieve promising performance, they do not explore the video-aware sentence representation to enhance the cross-modal information. From the results, we can see that FIAN achieves better results by utilizing both sentence-aware video and video-aware sentence representations.

Table 5: Ablation study on the TACoS dataset. We apply different localization strategy to substitute the localization module in our network. FIAN-Full is our method with content-oriented location strategy.

Method	R@1 IoU=0.1	R@1 IoU=0.3	R@1 IoU=0.5	R@5 IoU=0.1	R@5 IoU=0.3	R@5 IoU=0.5
FIAN-Full	39.55	33.87	28.58	56.14	47.76	39.16
FIAN-TGN	34.82	23.37	20.05	48.76	38.40	31.64
FIAN-CMIN	35.56	27.46	22.39	51.41	39.92	33.88

Table 6: Ablation study on the TACoS dataset. We evaluate the influence of different stride sizes in the localization module. θ_k is the kernel size.

Stride Size	R@1 IoU=0.1	R@1 IoU=0.3	R@1 IoU=0.5	R@5 IoU=0.1	R@5 IoU=0.3	R@5 IoU=0.5
1	37.45	29.83	21.09	51.68	40.73	29.90
$1/8\theta_k$	**39.55**	**33.87**	**28.58**	56.14	47.76	39.16
$1/4\theta_k$	37.01	30.32	25.84	57.09	45.09	37.89
$1/2\theta_k$	38.43	31.40	25.37	**60.72**	**51.14**	**41.81**

5 ABLATION STUDY

5.1 Influence of fine-grained iterative attention

In this section, we present ablation studies to understand the effects of CGA block, video-aware sentence representation, and symmetrical attention. We re-train our approach with following settings:

- **FIAN-Soft**: Instead of symmetrical iterative attention, we just use one video-query encoder and substitute the CGA in the encoder with soft attention.
- **FIAN-CMA**: We adopt one video-query encoder as above while substituting the CGA with CMA.
- **FIAN-VQ**: Compared with above, we apply one complete video-query encoder to generate sentence-aware video representation $V^{(2)}$ as shown in Figure 2.
- **FIAN-VQMA**: We utilize one iterative attention branch to generate sentence-aware video representation $V^{(2)}$ and video-aware sentence representation $Q^{(2)}$.
- **FIAN-QVMA**: We apply iterative attention to generate sentence-aware video representation $V^{(1)}$ and corresponding sentence representation $Q^{(1)}$.
- **FIAN-Concat**: We directly concatenate two video representations $[V^{(1)}, V^{(2)}]$ from two iterative attention branches instead of applying video-enhanced integration.
- **FIAN-Matrix**: We use $[V^{(1)}, V^{(2)}, V^{(1)} - V^{(2)}, V^{(1)} \odot V^{(2)}]$ to fuse the video representations from two branches instead of fusion gate.
- **FIAN**: Our full of FIAN model.

Table 4 shows the performance comparisons of our FIAN and these ablations on the most difficult TACoS dataset.

Effect of CGA block. Comparing FIAN-CMA with FIAN-Soft, it is significant that multi-head attention obtains a more fine-grained interaction than plain soft attention. With information gate on FIAN-CMA, FIAN-VQ effectively performs filtering information for

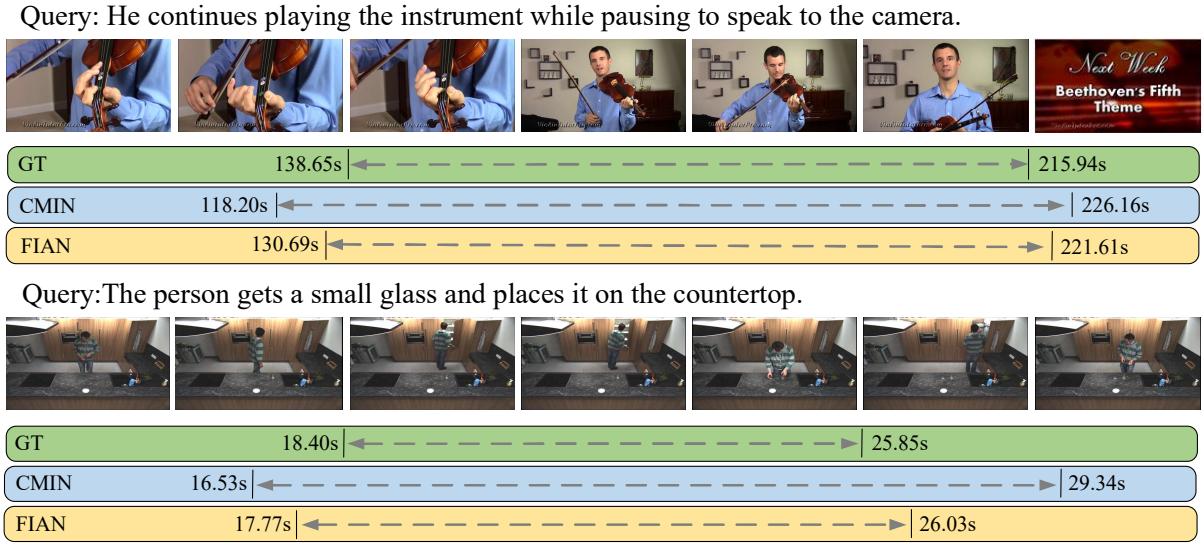

Figure 6: Qualitative visualization of temporal language localization results (Rank@1) by CMIN, and FIAN. First example comes from Activity Captions and second example is from TACoS dataset. Ground truth (GT) is also provided for both samples.

all candidates, which is demonstrated in "Rank@5, IoU=0.5". Thus, our CGA block captures more detailed cross-modal interactions than widely used soft attention.

Effect of video-aware sentence representation. (1) Comparing FIAN-VQMA with FIAN-VQ, it can be observed that "Rank@1, IoU=0.5" improves 1.51% and performance gains in all metrics. It indicates that video-aware sentence representation can also benefit cross-modal information integration for prediction. (2) Comparing FIAN-Concat and FIAN-Matrix with FIAN, direct concatenation or performing matrix operations between video representations show weaker results. It denotes that video-aware sentence information from two branches also effectively enhances the cross-modal integration.

Effect of symmetrical iterative attention. Comparing FIAN-QVMA and FIAN-VQMA with FIAN, single branch performs worse than full FIAN, which demonstrates that two branches from symmetrical mutual attention can compensate each other to generate more robust cross-modal information.

5.2 Influence of location module

In this section, we first compare our localization module with existing techniques by replacement in order to further verify the effectiveness of the proposal. Here we replace our localization module with corresponding component in TGN [3], and CMIN [44], and rename these variants as FIAN-TGN and FIAN-CMIN. The performance degeneration observed in Table 5 verifies the superiority of our proposed modules over their competitors. Moreover, we adjust different stride sizes θ_s during candidate moments sampling. The results are shown in Table 6. We can observe that too many candidate moments (stride=1) leads to performance drop. This indicates that dense candidate moments confuse the learning process of regression. Meanwhile, too few candidate moments (stride=$1/2\theta_k$) brings the highest R@5 metrics. For more precise R@1 retrieving results, we choose $1/8\theta_k$ as our experiments setting.

5.3 Qualitative Results

To qualitatively validate the locating effectiveness of our FIAN, we show two examples on ActivityNet Captions and TACoS dataset. As shown in Figure 6, in both samples, FIAN is capable of locating an event which consists of two diverse activities, although the target moment in ActivityNet Captions is obviously longer than the one in TACoS dataset. By intuitive comparison, our FIAN localizes more accurate boundaries than CMIN [44] and explicitly decreases the location error brought by ambiguous frames.

6 CONCLUSIONS

In this paper, we propose a novel Fine-grained Iterative Attention Network (FIAN) to adequately extract bilateral video-query interaction information for temporal language localization in videos. Besides proposing a refined cross-modal guided attention (CGA) block to capture the detailed cross-modal interactions, FIAN further adopts a symmetrical iterative attention to generate both sentence-aware video and video-aware sentence representations, where the latter is explicitly facilitated to enhance the former and finally both parts contribute to a robust cross-modal feature. In addition, we devise a content-oriented localization strategy to better predict the temporal boundary. Extensive experiments on three real-world datasets validate the effectiveness of our method.

The future work includes apply FIAN to more complicated benchmarks [17, 20]. Combining our method with some pre-training model [18] could also further boost the performance.

7 ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China (No. 61972448), National Natural Science Foundation of China (No. 61902347), and Zhejiang Provincial Natural Science Foundation (No. LQ19F020002).

REFERENCES

[1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450* (2016).

[2] João Carreira and Andrew Zisserman. 2017. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4724–4733.

[3] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. 2018. Temporally Grounding Natural Sentence in Video. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[4] Jingyuan Chen, Lin Ma, Xinpeng Chen, Zequn Jie, and Jiebo Luo. 2019. Localizing Natural Language in Videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

[5] Yu Cheng, Quanfu Fan, Sharath Pankanti, and Alok Choudhary. 2014. Temporal Sequence Modeling for Video Event Detection. In *CVPR*.

[6] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. 2014. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* (2014).

[7] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ramakant Nevatia. 2017. TALL: Temporal Activity Localization via Language Query. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 5277–5285.

[8] Runzhou Ge, Jiyang Gao, Kan Chen, and Ram Nevatia. 2019. MAC: Mining Activity Concepts for Language-based Temporal Localization. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 245–253.

[9] Meera Hahn, Asim Kadav, James M. Rehg, and Hans Peter Graf. 2019. Tripping through time: Efficient Localization of Activities in Videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,Workshop*.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[11] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan C. Russell. 2017. Localizing Moments in Video with Natural Language. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 5804–5813.

[12] Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. 2017. Modeling relationships in referential expressions with compositional modular networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1115–1124.

[13] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. 2016. Natural language object retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4555–4564.

[14] Lun Huang, Wenmin Wang, Jie Chen, and Xiao-Yong Wei. 2019. Attention on Attention for Image Captioning. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

[15] Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *CoRR* abs/1412.6980 (2014).

[16] Ranjay Krishna, Kenji Hata, Frederic Ren, Fei-Fei Li, and Juan Carlos Niebles. 2017. Dense-Captioning Events in Videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 706–715.

[17] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L Berg. 2018. TVQA: Localized, Compositional Video Question Answering. In *EMNLP*.

[18] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. 2020. HERO: Hierarchical Encoder for Video+ Language Omni-representation Pre-training. *arXiv preprint arXiv:2005.00200* (2020).

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

[20] Jingzhou Liu, Wenhu Chen, Yu Cheng, Zhe Gan, Licheng Yu, Yiming Yang, and Jingjing Liu. 2020. VIOLIN: A Large-Scale Dataset for Video-and-Language Inference. In *CVPR 2020*.

[21] Meng Liu, Xiang Wang, Liqiang Nie, Xiangnan He, Baoquan Chen, and Tat-Seng Chua. 2018. Attentive moment retrieval in videos. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. ACM, 15–24.

[22] Chujie Lu, Long Chen, Chilie Tan, Xiaolin Li, and Jun Xiao. 2019. DEBUG: A Dense Bottom-Up Grounding Approach for Natural Language Video Localization. In *EMNLP/IJCNLP*.

[23] Ruotian Luo and Gregory Shakhnarovich. 2017. Comprehension-guided referring expressions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7102–7111.

[24] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 11–20.

[25] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. 2016. Modeling context between objects for referring expression understanding. In *European Conference on Computer Vision*. Springer, 792–807.

[26] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

[27] Michaela Regneri, Marcus Rohrbach, Dominikus Wetzel, Stefan Thater, Bernt Schiele, and Manfred Pinkal. 2013. Grounding Action Descriptions in Videos. *Transactions of the Association for Computational Linguistics* 1 (2013), 25–36.

[28] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. 2016. Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision*. Springer, 817–834.

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

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

[31] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. 2014. Learning Spatiotemporal Features with 3D Convolutional Networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. 4489–4497.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Proceedings of the Thirty-third Conference on Neural Information Processing Systems (NeurIPS)*.

[33] Jing Wang, Yu Cheng, and Rogerio Schmidt Feris. 2016. Walk and Learn: Facial Attribute Representation Learning From Egocentric Video and Contextual Data. In *CVPR 2016*.

[34] Liwei Wang, Yin Li, and Svetlana Lazebnik. 2016. Learning deep structure-preserving image-text embeddings. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 5005–5013.

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

[36] Huijuan Xu, Kun He, Bryan A Plummer, Leonid Sigal, Stan Sclaroff, and Kate Saenko. 2019. Multilevel language and vision integration for text-to-clip retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9062–9069.

[37] Serena Yeung, Olga Russakovsky, Greg Mori, and Li Fei-Fei. 2016. End-to-end learning of action detection from frame glimpses in videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2678–2687.

[38] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. 2018. Mattnet: Modular attention network for referring expression comprehension. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1307–1315.

[39] Yitian Yuan, Lin Ma, Jingwen Wang, Wei Liu, and Wenwu Zhu. 2019. Semantic Conditioned Dynamic Modulation for Temporal Sentence Grounding in Videos. In *Proceedings of the Thirty-third Conference on Neural Information Processing Systems (NeurIPS)*.

[40] Yitian Yuan, Tao Mei, and Wenwu Zhu. 2019. To find where you talk: Temporal sentence localization in video with attention based location regression. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 9159–9166.

[41] Runhao Zeng, Chuang Gan, Peihao Chen, Wenbing Huang, Qingyao Wu, and Mingkui Tan. 2019. Breaking winner-takes-all: Iterative-winners-out networks for weakly supervised temporal action localization. *IEEE Transactions on Image Processing* 28, 12 (2019), 5797–5808.

[42] Da Zhang, Xiyang Dai, Xin Wang, Yuan-Fang Wang, and Larry S. Davis. 2019. MAN: Moment Alignment Network for Natural Language Moment Retrieval via Iterative Graph Adjustment. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

[43] Hanwang Zhang, Yulei Niu, and Shih-Fu Chang. 2018. Grounding referring expressions in images by variational context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4158–4166.

[44] Zhu Zhang, Zhijie Lin, Zhou Zhao, and Zhenxin Xiao. 2019. Cross-Modal Interaction Networks for Query-Based Moment Retrieval in Videos. In *The 42nd International ACM SIGIR Conference on Research and Development in Information*. ACM.

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