A Parallel Transformer Framework for Video Moment Retrieval

Thao-Nhu Nguyen*
Dublin City University
Dublin, Ireland
thaonhu.nguyen24@mail.dcu.ie

Zongyao Li*
NEC Corporation
Tokyo, Japan
zongyao-li@nec.com

Satoshi Yamazaki
NEC Corporation
Tokyo, Japan
s-yamazaki31@nec.com

Jianquan Liu
NEC Corporation
Tokyo, Japan
jqliu@nec.com

Cathal Gurrin
Dublin City University
Dublin, Ireland
cathal.gurrin@dcu.ie

ABSTRACT

In the realm of video understanding, Video Moment Retrieval (VMR) is an important yet challenging task that aims to locate the boundary of a moment of interest within a long untrimmed video. Existing VMR methods often focus on the visual content extracted from the video only (or frame sequences), however, the rich semantic information at the object level that describes the image's content has not been explored yet. To overcome those limitations, we propose **PaTF**, an attention-based **Pa**rallel **T**ransformer **F**ramework that enriches the feature representations by exploring both low-level visual cues and high-level relational contexts of video-query pairs. Our framework consists of two parallel transformers: one for the visual-textual stream and the other for the semantic-textual stream. The visual-textual stream extracts the links between global visual features and textual information, while the semantic-textual stream emphasises the relations between objects via scene graph representations. Furthermore, our comprehensive experiment conducted on the Charades-STA dataset demonstrates that the proposed framework outperforms the state-of-the-art methods by a large margin, 5% and 7% at Recall@1 with IoU = 0.5 and IoU = 0.7, respectively.

CCS CONCEPTS

• **Computer systems organization** → **Embedded systems**; *Redundancy*; Robotics; • **Networks** → Network reliability.

KEYWORDS

Video Moment Retrieval; Scene graph; Vision-language Transformer; Video Retrieval; Video Temporal Localisation

ACM Reference Format:
Thao-Nhu Nguyen, Zongyao Li, Satoshi Yamazaki, Jianquan Liu, and Cathal Gurrin. 2024. A Parallel Transformer Framework for Video Moment Retrieval. In *Proceedings of the 2024 International Conference on Multimedia Retrieval (ICMR '24), June 10–14, 2024, Phuket, Thailand*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3652583.3658096

*Equal contribution to this research that is mainly completed during Thao-Nhu Nguyen's internship at NEC Corporation.

ICMR '24, June 10–14, 2024, Phuket, Thailand

ACM ISBN 979-8-4007-0619-6/24/06
https://doi.org/10.1145/3652583.3658096

1 INTRODUCTION

Technology and the Internet, particularly social media platforms, have advanced significantly over the past couple of decades, with an enormous amount of data (images, videos, and text) created and shared every day. Among those forms of media, video is dominant thanks to its ability to both capture the complexity of the human experience and offer an engaging visual interaction to viewers. Consequently, there is a growing demand for understanding and analysing video content for different tasks such as video summarisation [12, 28, 60], video captioning [17, 31, 41], video action recognition [1, 13, 39, 46], and text-to-video retrieval [7, 14, 22]. Beyond the conventional text-to-video retrieval task, where the objective is to search for a single video across a video corpus, people may expect to retrieve more fine-grained moments within videos rather than the entire video sequence. As a result, the Video Moment Retrieval (VMR) task emerged to address the challenge of locating a specific "*moment*" or "*event*" (with a start and an end time) that is semantically relevant to a given query within one long, untrimmed video. The query is a natural language description of a moment captured in a video. For example, given a query like "*the moment when the bride throws the bouquet*", a VMR system is expected to return the exact start and end times of that event in a wedding video.

Most previous work [6, 15, 42, 53] tackled the VMR task with predefined candidates or proposals within generated, then utilising matching techniques to rank them based on the learned representations (proposal-based methods). While effective to some extent, such methods demand significant efforts to annotate moment boundaries, leading to challenges in annotation and scalability. In the meantime, other frameworks [16, 37, 55, 59] learn the cross-modal interactions and attempt to regress the probabilities of all frames, then choose the peaks as the start and end of the event's segments (proposal-free methods). Therefore, our framework will make use of proposal-free techniques to reduce the level of human involvement required.

Despite the undeniable role of encoding low-level visual features to represent visual content, little effort has been made to explore the impact of high-level relational semantic cues for the VMR task. For that reason, we propose to jointly learn from both aforementioned aspects by using a parallel transformer architecture including a global stream (with visual embedding representations) and a relational stream (with scene graph representations). To this

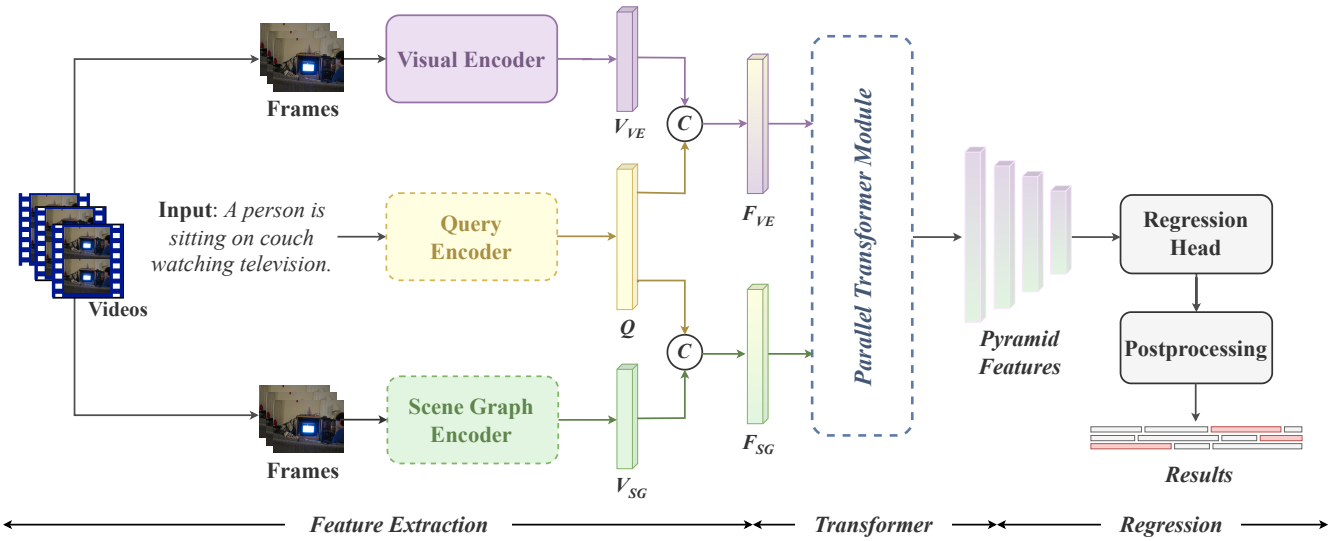


Figure 1: An overview of PaTF, which consists of three main stages, including a feature extraction stage, a transformer stage, and a regression stage.

end, we hypothesise that this approach can better exploit not only the visual meaning but also the semantic relations between the query and the video. Particularly, the global stream encodes the overall visual content of each video frame with the pre-trained embedding models such as CLIP [34] or I3D [4]. Meanwhile, the relational stream models the fine-grained details and relationships among different objects in the frame via the scene graph generation model. Additionally, each stream is fed into a transformer module to learn their representations before ensembling the features of both streams for prediction.

The main contributions of our work are summarised as follows:

(1) To the best of our knowledge, we are the first to enrich the feature representations with relational semantic cues in the proposal-free VMR task by extracting the relational information in video frames via scene graph representations.
(2) We propose a novel framework, named PaTF, for the VMR task, which employs an attention-based parallel transformer framework on different feature representations to predict the start and end of the desired moments within the video.
(3) We conduct intuitive experiments and extensive ablation studies on the Charades-STA benchmark and compare with state-of-the-art techniques, highlighting the potential of our proposed framework.

2 RELATED WORKS

2.1 Video Moment Retrieval

The Video Moment Retrieval task aims to identify an event time interval semantically matching a given text query, which requires a deep understanding of semantic relations between visual and textual content. There are two main types of VMR approaches: proposal-based VMR [15, 16, 33, 42, 53, 58] and proposal-free VMR [21, 26, 30, 43, 47, 52, 55]. While the former measures the similarity between the pre-segmented clip proposals and the text query, the latter predicts the probabilities of each frame to be the event boundaries based on their high dimensional representation.

The proposal-based VMR first converts the input long video into small clips as proposal candidates and then sorts them based on their relevance to the given text description. Using a multi-scale temporal sliding window strategy, the existing methods [15, 33, 53] converted input videos into a set of candidate segments. Then, Gao et al. [15] proposed Cross-modal Temporal Regression Localiser (CTRL) to jointly model visual features extracted from pre-trained C3D model and text features, while VLG-Net [33] adopts Syntactic Graph Convolution Networks (SyntacGCN) to encode video and sentence embeddings, followed by a graph matching layer. Zeng et al. [53] captured multi-modal relational graphs of the visual and textual content from the proposals. Meanwhile, FVMR [16] generates moment proposals utilising 2-dimensional maps [57], indicating the moment's start and end times. However, these methods suffer the burden of choosing the candidates since the models rely heavily on the moment candidates' boundary accuracy. Moreover, the annotation process may require much human effort and may introduce subjective bias, as different annotators may have different opinions on the event boundaries.

On the other hand, the proposal-free VMR incorporates a moment generation stage and a moment localisation stage into one single module and directly predicts the start and end times of a moment, without the need to pre-segment the proposals. UVCOM [43] addresses the VMR task with a Comprehensive Integration Module (CIM) designed to achieve intra- and inter-modality interaction across multi-granularity. As a result, the model improves the video's understanding by recognising both local relationships and global knowledge accumulation throughout the entire video. Furthermore, LGI [30] takes extracted video and text embeddings

as input and applies their local-global video-text interaction models in three levels (segment-level fusion, local context modeling, and global context modeling). By doing so, the authors are able to capture an in-depth relationship between video and query and output the moment predictions. Based on the concept proposed by DETR [3] for object detection, recent works [21, 29, 45] resolved the VMR task by leveraging an encoder-decoder transformer architecture that considers the VMR task as a direct set of prediction problems. Concretely, the authors leverage a transformer module together with three different heads (including saliency, fore/background, and moment coordinate head) to predict the moments. UnLoc [47] introduces a unified framework that exploits the large-scale pre-trained model, CLIP [34], together with the feature pyramid. Inspired by those works, our framework is constructed as a proposal-free VMR framework that leverages the power of transformer architecture [38] in capturing long sequences to identify moment boundaries.

2.2 Scene Graph Generation

A scene graph is a data structure, first proposed in [20] for image retrieval tasks. Scene graph plays a crucial role in representing the semantic meaning of an image as it captures the intra-scene object instances and their pair-wise relationships inside an image, represented by nodes and edges, respectively [23, 27, 44, 48, 51, 56]. Prior research has attempted to predict the scene graph for an image, which is employed in various tasks including image captioning [18, 49, 50] and image-text retrieval [20, 40]. Neural Motifs [51] stack LSTMs to create a contextualised representation of each object, while Xu et al. [44] uses standard RNNs to improve graph prediction via message passing. Moreover, recent approaches [8, 9, 11] have integrated transformer and attention mechanisms into the graph-based models to create richer visual relationships and generate scene graphs. There has been little work exploring the effectiveness of the scene graph structure for solving VMR tasks. In addition to global visual embeddings, we aim to enrich the visual representations by emphasising the relational information between objects using scene graphs, which is currently lacking when relying solely on visual embeddings.

3 METHOD

3.1 Problem Formulation

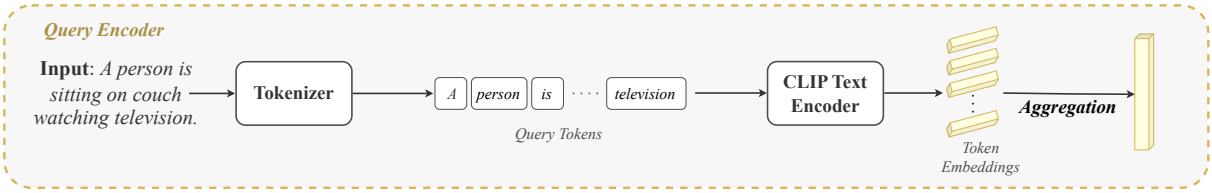
We suppose that $X = \{x_1, x_2, ..., x_T\}$, defined on discretised time steps $t = \{1, 2, ..., T\}$, can be used to represent the given input video X, where x_t is the frame at the time t. The total duration T either is fixed to a specific number or varied across videos. Given an input query string q, the goal of the VMR task is to localise the time frames $Y = (s_i, e_i)$, the start and end times of that moment ($s_i < e_i$), of a moment within the untrimmed video X that matches with the description q.

3.2 Overview

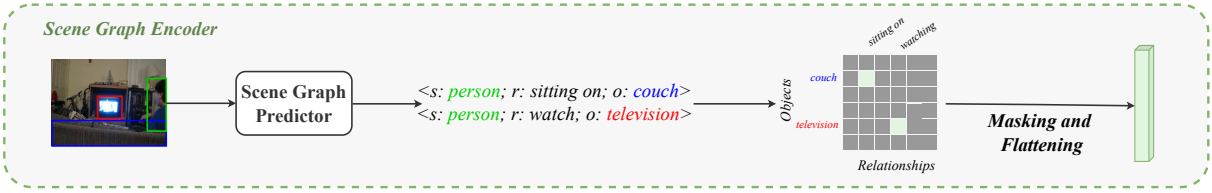
The PaTF framework, illustrated in Figure 1, takes each video-query pair as an input and outputs a ranked list of time intervals, in decreasing order of relevance. Specifically, the model first extracts the representations of video keyframe sequences and input text query tokens individually. While the text features are embedded by the pre-trained text encoder, the two-stream visual features are obtained in a local and global manner using the pre-trained scene graph encoder and image encoder, respectively. The two visual-textual-joint features, constructed by concatenating each of the visual streams with the textual features, are then fed to a Parallel Transformer Module. Ultimately, a Regression head is applied to obtain a ranked list of potential candidates before extracting the target moment onsets and offsets in the postprocessing step.

3.3 Feature Extraction

3.3.1 Query Encoder. For the query encoder, as shown in Figure 2a, the input query q is first tokenised into a list of query tokens $q = \{q_1, q_2, ..., q_K\}$. Then, we adopt a pre-trained CLIP text encoder [34] to encode all tokens into numerical feature vectors $\tilde{q} = \{\tilde{q_1}, \tilde{q_2}, ..., \tilde{q_K}\}, \tilde{q} \in \mathbb{R}^{K \times D}$. Following this, an aggregation layer



(a) Text Encoder block



(b) Scene Graph Encoder block

Figure 2: Text encoder and Scene graph encoder block

(a linear layer) is integrated to unite all-token features with learnable parameters $w \in \mathbb{R}^{D\times 1}$ into one text feature $Q \in \mathbb{R}^{1\times D}$, which represents the overall query. Particularly, this aggregation layer averages $\tilde{q}$ with a series of weights $\alpha \in \mathbb{R}^{K\times 1}$ as below:

$$Q = \alpha^T \cdot \tilde{q}, \text{ where } \alpha = \text{Softmax}(\tilde{q} \cdot w). \quad (1)$$

3.3.2 Visual Encoder. With the aim of capturing the global content of an image or video frame, the visual embedding model transforms the video frame sequence into a high-dimensional feature vector $V_{VE} \in \mathbb{R}^{T\times D}$. Specifically, we use the CLIP image embedding model [34] with a Vision Transformer [10] pre-trained at 336-pixel resolution ($ViT - L/14@336$). Additionally, we also use pre-extracted temporal features I3D [4] provided by MIGCN [58].

3.3.3 Scene Graph Encoder. To capture the interaction between objects in an image, we use scene graphs, graph-based representations of the objects and their relationships, to represent the structural layout of the image. Particularly, a scene graph G of a single frame consists of a set of triplets $\{s_i, r_i, o_i\}_{i=1}^{|G|}$, where s_i, o_i, and r_i are the subject, object, and relationship between s_i and o_i of the i^{th} triplet, respectively.

At the scene graph generation stage in Figure 2b, we utilise the Neural Motifs model [51], which is developed based on the Faster R-CNN [35] object detector, as the scene graph predictor. For the scene graph generation, we use the Action Genome dataset [19], built upon the Charades dataset, providing both action labels and spatio-temporal scene graph labels. As in most videos used in our experiments, only one person appears and interacts with objects, the subject is always the same "*person*" and the scene graph simplifies to $\{r_i, o_i\}_{i=1}^{|G|}$. Once we have the predicted scene graph, we generate a confidence matrix $C \in \mathbb{R}^{|O|\times|R|\times 3}$ in which $|O|$ and $|R|$ are the number of classes of objects and relationships, and the third dimension is about the confidence scores. In the Action Genome dataset, $|O| = 35$ and $|R| = 25$. Then, a mask is applied to the confidence score matrix to filter out all irrelevant information before being flattened to return the corresponding scene graph features $V_{SG} \in \mathbb{R}^{|O||R|}$. Specifically, in the mask $M \in \mathbb{R}^{|O|\times|R|}$, the rows corresponding to classes appearing in the text query are filled with ones, otherwise zeros.

3.3.4 Visual-textual Joint Features. Once the features are extracted as described above, each pair of visual-textual features is concatenated as input to the parallel transformer module. The two joint features F_{VE} (concatenation output from the visual encoder V_{VE} and query encoder Q) and F_{SG} (concatenation output from the graph encoder V_{SG} and query encoder Q) are constructed as follows:

$$F_{VE} = Concat(V_{VE}, Q), \; F_{SG} = Concat(V_{SG}, Q). \quad (2)$$

3.4 Parallel Transformer Module

In addition to the informative visual embeddings, relationship cues between objects are also key factors in enriching the visual content. To exploit the representations from both data streams, we implement a parallel transformer module and then fuse them into a unified representation. Specifically, the parallel transformer module is comprised of two parallel branches: one for processing the visual features in the global context of images using a pre-trained model, and the other one for encoding relationship cues in a structured and explicit way by the scene graph representations that capture the semantic and spatial relationships between objects using a graph neural network.

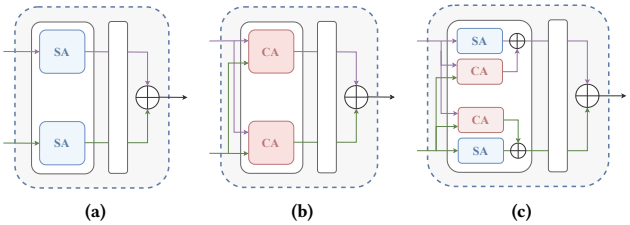


Figure 3: Parallel Transformer module: (a) Dual Self-Attention (SA), (b) Dual Cross-Attention (CA), and (c) Combined Attention (SA & CA).

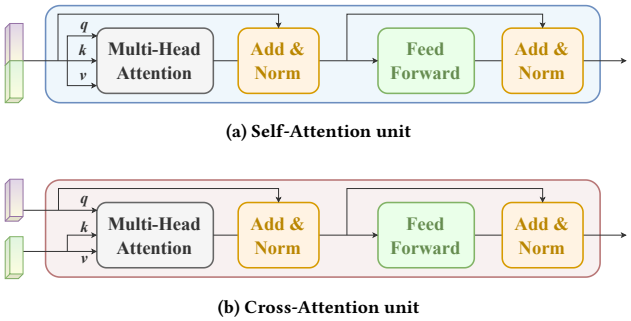


Figure 4: Attention mechanisms

Transformer [38] is a powerful model that can model long data sequences, such as sentences or video frames, without losing their context with the use of attention mechanisms. By concentrating on different parts of the input sequences, these mechanisms enable the model to capture long-range dependencies and contextual information. With the advance of transformer mechanisms, researchers apply transformer-based models to resolve the VMR task [21, 59] and achieve promising results. Inspired by these works, we constructed a parallel transformer structure as our backbone in the framework. By doing so, the framework is able to maximise each stream's advantages to achieve better results.

As depicted in Figure 3, we explore three implementation variants for parallel transformers: Dual Self-Attention block (SA), Dual Cross-Attention block (CA), and Combined-Attention block (SA & CA). These blocks are inherited from the standard transformer architecture [38]. In particular, both Self-Attention and Cross-Attention units consist of a multi-head attention layer and a feed-forward layer, illustrated in Figure 4. The key difference between them is the input modality. While the self-attention unit takes only one modality as input and computes the attention weights for itself to model the intra-modal refinement, the cross-attention unit considers interactions between different input modalities.

3.5 Regression stage

3.5.1 Feature Pyramid Network (FPN). Leveraging the pyramidal structure of the ConvNet features, the Feature Pyramid Network (FPN) [24] creates a feature pyramid with strong semantics from various levels. Concretely, the FPN's construction involves a bottom-up pathway, a top-down pathway, and lateral connections. While the former, with low-resolution levels, captures more global content and represents richer semantic meanings of the data, the latter, with high-resolution levels, highlights local information and more accurate spatial information. With the variety of applications, it has been widely used not only in object detection tasks but also in other tasks such as semantic segmentation and image classification. In this paper, FPN is implemented to generate segment candidates of the predicted moments by combining the features from different levels and estimating the start and end times of the relevant segment. Particularly, we used six layers of feature pyramids to obtain features from various levels.

3.5.2 Moment Boundary Regression head. The objective of the regression head is to examine all levels of features given in the feature pyramid and predict the distance to the moment onsets and offsets. The regression head, guided by relevance scores, provides valuable insights into how the model weighs different cues to determine the temporal boundaries of moments in the retrieval task. To that end, the regression head is implemented with a 1D convolutional network, including three 1D convolutional layers and two normalisation layers.

3.5.3 Postprocessing. Once the distances are obtained, we need to align all frame outputs into the start and end frame orders. The moment segments are obtained as follows:

$$(s, e)_i = (\max(i - ds_i, 0), \min(i + de_i, n)) \tag{3}$$

where n is the number of frames from that video and $(s, e)_i$ is the predicted start and end frame of the moment, ds_i and de_i are the distances from the current frame position i to the start and end of the predicted moment, respectively. By doing so, a list of (s, e) candidates is ranked based on their relevance score.

To obtain multiple potential candidate moments (top k) and avoid the overlapped predictions, we adopt Soft Non-Maximum Suppression (SoftNMS) [2] as a postprocessing step to select moments with the highest relevance score.

3.6 Loss functions

For the VMR task, the loss functions are considered with two terms: (1) Focal loss $\mathcal{L}_{rel}$ for relevance score and (2) $L1$ loss $\mathcal{L}_{reg}$ for regression head. The former is used to handle the problem of class imbalance in object detection, while the latter computes the loss value based on the absolute distance between the predicted output and the ground truth moment boundaries. The final loss function for each input is defined below:

$$\mathcal{L}_{final} = \mathcal{L}_{rel} + \lambda \times \mathcal{L}_{reg} \tag{4}$$

where λ is a weight constraint to balance between the two aforementioned losses. λ is practically suggested as 1.

4 EXPERIMENTS

4.1 Experimental Settings

4.1.1 Charades-STA dataset. We evaluated our framework on the Charades-STA dataset [15], an extension of the original Charades dataset [36], designed especially for VMR tasks by collecting the sentence temporal annotations for video segments. This dataset contains **6,672** videos (**5,338** and **1,334** videos for training and testing) and **16,128** pairs of textual description and action segments (**12,408** and **3,720** pairs for training and testing). More precisely, in the Charades-STA dataset, each video has a duration of approximately 30 seconds and an average of 2.4 moments (each lasting approximately 8.2 seconds). We evaluated our framework on this dataset since it is the only VMR dataset having scene graph annotations.

4.1.2 Evaluation Metrics. The ("$R@k - IoU = v$") [15] metric is adopted as an evaluation criterion in order to measure the performance of our framework for the VMR task. Particularly, it is defined as the percentage of at least one of top-k selected moments whose IoU is larger than v [15], where IoU is the intersection over

Table 1: Performance comparison between PaTF and SOTA methods on Charades-STA dataset. There are three types of visual features: SlowFast (SF), CLIP, and I3D. The best and suboptimal values are highlighted in bold and underlined, respectively.

Method	Visual Features	Scene Graphs	$R@1$		$R@5$	
			$IoU = 0.5$	$IoU = 0.7$	$IoU = 0.5$	$IoU = 0.7$
VSLNet [55]	I3D	–	54.2	35.2	–	–
Moment-DETR [21]	CLIP	–	55.7	34.2	–	–
QD-DETR [29]	SF+CLIP	–	57.3	32.6	–	–
MH-DETR [45]	I3D	–	56.4	35.8	–	–
UVCOM [43]	SF+CLIP	–	59.3	36.6	–	–
LGI [30]	I3D	–	59.5	35.5	–	–
UnLoc-L [47]	CLIP	–	60.8	38.4	88.2	61.1
PaTF (Ours)	CLIP	✓	63.6	40.8	<u>90.7</u>	65.3
PaTF (Ours)	I3D	✓	<u>64.0</u>	<u>43.4</u>	**90.9**	<u>66.5</u>
PaTF (Ours)	I3D+CLIP	✓	**65.8**	**45.1**	90.6	**68.2**

the union between the ground truths and the predictions. As this metric is obtained on a query level, the overall performance will be the average value across all queries, denoted as follows:

$$R(k, v) = \frac{1}{N_q} \sum_{i=1}^{N_q} r(k, v, q_i), \tag{5}$$

where the $r(k, v, q_i)$ represents whether one of the top-k selected moments of the query q_i has $IoU > v$, and N_q is the total number of testing queries. In this work, we use the recall with k of 1 and 5, and IoU thresholds 0.5 and 0.7.

4.1.3 Implementation Details. We build our model upon ActionFormer [54] with PyTorch [32] for the VMR task. Our model is trained on one NVIDIA GTX 1080ti GPU with a total batch size of 16 for 10 epochs. During the training process, we use the Adam optimiser [25] to minimise the final loss (calculated in Equation 4) with the initial learning rate of 0.001. The visual features are extracted by the pre-trained CLIP model [34] and I3D model [5], and the text features are extracted by the pre-trained CLIP model. Motifs [51] is used as the model for scene graph generation (SGG) and trained on the Action Genome dataset [19]. Within this research, both training losses are equally weighted, which means the value of λ is 1.

4.2 Comparison with SOTA methods

To investigate the effectiveness of our framework, we make a comparison between the performance of PaTF and other state-of-the-art approaches conducted on the Charades-STA dataset. As indicated in Table 1, the best results are in bold, while the suboptimal values are underlined. In general, our framework outperformed all the existing methods in terms of $R@1$ and $R@5$ with both $IoU = 0.5$ and $IoU = 0.7$. Delving deeper into the comparative analysis, our model achieves a substantial improvement of roughly 3% for $R@1 - IoU = 0.5$ by incorporating scene graph representation in one additional branch, alongside similar visual features and backbone with the latest approach, UnLoc. Note that when using the combination of I3D and CLIP features as visual features, we obtain the best results ($R@1 - IoU = 0.5$ of 65.8 and $R@1 - IoU = 0.7$ of 45.1). Notably, this version surpasses UnLoc, with an increase of 5% for $R@1 - IoU = 0.5$, and approximately 7% for $R@1 - IoU = 0.7$. These large performance gaps prove our stronger localisation ability as compared to the SOTA methods.

To the best of our knowledge, our framework is the first to use two parallel branches with the standard transformer backbone to achieve competitive performance with other techniques. This indicates the fact that beyond the advance of the transformer model, which has already been proven in other work, the semantics represented via scene graphs is a key factor that benefits the process of localising the target video moments in our framework.

4.3 Ablation Studies

To evaluate the contribution of each module in our method and make the best choices of the modules, we conduct a series of ablation studies. All the experiments of ablation studies are conducted with I3D features on the test split.

4.3.1 Contributions of visual features and scene graphs. To give deeper insight into the contributions of the visual features and scene graphs, we conducted experiments with different combinations of visual features and scene graphs as well as each of them individually, resulting in Table 2. We attempt to use the ground truth scene graphs during both training and test to demonstrate the potential for further performance improvement using a more powerful SGG model. Note that the Action Genome dataset annotates the scene graphs for only some keyframes of the videos, and the missing frames' features will be filled with zeros. It is possible to estimate that the performance will be further improved by accurate scene graphs for all frames.

As indicated in Table 2, introducing the predicted scene graphs increases the recall score by roughly $2-3\%$ except for $R@1 - IoU = 0.7$, when compared to using only I3D or CLIP features. The reason behind this increase in recall when using CLIP features could be the loss of temporal information during the feature extraction. To reach a high temporal IoU, the temporal information needs to be preserved well in the extracted features. Compared to the I3D features each of which is extracted from a clip of continuous frames, the CLIP features are extracted from single sampled frames, leading to more temporal information loss. Moreover, when using I3D and CLIP features together via the parallel transformer (dual-stream or triple-stream if using scene graphs), the performance is better than that with either of the two kinds of visual features. This indicates the efficacy of ensembling different kinds of features. The results using only scene graph features, shown in the last two rows in Table 2, degrade compared to those of the combinations with visual features. For the predicted scene graphs, the drop in performance without visual features implies that the performance of our current SGG model is unsatisfactory for the VMR task. As to the ground truth, the performance degradation is due to the information loss of the missing frames.

4.3.2 Scene graph features. Table 3 shows the results of the ablation study for the scene graph features. Our baseline is the PaTF framework without the scene graph branch. As mentioned in Section 3.3.3, the confidence score matrix is masked to filter out irrelevant

Table 2: Ablation study for evaluating the contributions of the visual features (I3D, CLIP) and scene graphs (Pred: generated by SGG model, GT: ground truth).

Visual features		Scene graphs		$R@1$	
I3D	CLIP	Pred	GT	$IoU = 0.5$	$IoU = 0.7$
✓				60.2	41.1
✓		✓		64.0	43.4
✓			✓	73.9	55.2
	✓			61.2	40.5
	✓	✓		63.6	40.8
	✓		✓	75.0	56.6
✓	✓			62.7	42.8
✓	✓	✓		65.8	45.1
✓	✓		✓	75.6	56.3
		✓		53.9	32.8
			✓	71.0	51.0

Table 3: Ablation study for evaluating the importance of relational information and query-irrelevant masking for the scene graph features.

Relationship	Masking	R@1	
		$IoU = 0.5$	$IoU = 0.7$
✓	✗	63.1	41.8
✓	✓	64.0	43.4
✗	✗	60.7	41.2
✗	✓	62.6	42.2
Baseline w/o scene graphs		60.2	41.1

noise. Comparing the first and second rows in Table 3, the query-irrelevant masking clearly enhances the performance of using scene graphs. Moreover, to prove the importance of the relational information, we remove the relationship dimension of the matrix and replace the scene graph feature with a $|O|$-length vector of object scores only indicating the presence of objects, with the results displayed in the third and fourth rows. Additionally, the object-score vector slightly improves the baseline performance. Although the improvement becomes clearer with the query-irrelevant masking, the recalls of using the scene graph features are higher by 1.4% and 1.2%, highlighting the importance of the relational information.

Table 4: Comparison on transformer module's architecture.

Transformer module	Block	R@1	
		$IoU = 0.5$	$IoU = 0.7$
Vanilla	Self-Attention	63.3	42.1
Dual	Self-Attention	64.0	43.4
Dual	Cross-Attention	63.1	44.1
Dual	Combination	63.7	43.6

4.3.3 Parallel transformer module. We further make a comparison between the vanilla and parallel transformer modules, shown in Table 4. The vanilla one takes concatenation of the visual and scene graph features as input while the parallel one processes the two feature streams separately. We also compare three different kinds of attention blocks for the parallel module, a self-attention block, a cross-attention block, and a combination of the two blocks, as depicted in Figure 3. The results show that the three parallel modules perform similarly, but all outperform the vanilla module with concatenated features, which indicates: 1) ensembling the features processed by two separate branches can benefit the integration of the visual cues and relational contexts as compared to a single branch with feature concatenation; 2) interaction between the two branches might be not essential in this specific case.

Table 5: Comparison on approaches of vision-text fusion.

Module	Tokens	Concat dim	R@1	
			$IoU = 0.5$	$IoU = 0.7$
Aggr.	All	Channel	60.2	41.1
AvgPool	All	Channel	60.2	40.1
–	EOT	Channel	59.2	39.8
–	All	Temporal	59.0	38.9
Cross-att	All	–	59.3	38.9

4.3.4 Vision-text fusion. In Table 5, we compare several approaches for the fusion between the textual features and the visual features. The results are obtained using only the I3D features. The first one in Table 5, "*Aggr.*" used in the proposed method, aggregates all the tokens output by CLIP's text encoder with a trainable aggregation layer, while the second one aggregates the tokens with the average pooling. The third one uses only the EOT token (the last token in the text embedding sequence) of CLIP without an additional module. The first three approaches concatenate the visual and text features along the dimension of the feature channel. The fourth one, used in UnLoc [47], concatenates all the tokens with the visual features along the temporal dimension. The last one, used in QD-DETR [29], fuses the features with a cross-modal attention module. Among the approaches, our aggregation module performs the best. While the previous work [29] reported performance improvement by the cross-modal attention, this module fails to demonstrate any improvement for our method. Possible explanations could be the difference in framework, prediction based on temporal-sequence features (ours), or learned queries (QD-DETR).

4.4 Qualitative Analysis

To get an intuitive perception of the impact of the scene graph representations, we show two visualisations of the prediction from the backbone and PaTF framework, which can be seen in Figure 5. These qualitative examples show the failure cases of the backbone without the scene graphs. One potential reason for this could be it only identifies the objects in the frame, but not the relations or interactions among them. In the first example, "*the person*" and "*the sandwich*" are visible in almost every frame throughout the video, but the action of putting the sandwich only occurs at the beginning of the video. The backbone model, however, fails to do so, as it only recognises the objects without considering when the person actually "*put*" the sandwich on the table. On the other hand, the PaTF with the semantic cues successfully highlights the relational information and predicts the more accurate boundaries. A similar situation happens in the second example, where the backbone cannot identify the end of action "*open the bag*" resulting in the wrong offset predictions.

5 CONCLUSION AND FUTURE WORK

We have presented an attention-based parallel transformer framework PaTF, which combines visual and semantic cues in order to enhance the accuracy of seeking the desired video moment for the VMR task. Concretely, our framework leverages scene graph representations as a separate stream of visual content to model the high-level semantic information that has not been used in existing methods. We argued that relational semantic cues play a crucial role in representing visual content, along with low-level visual features. Our comprehensive experiments and ablation studies demonstrate that the adoption of scene graph representation results in performance improvement as compared to the latest methods.

Figure 5: Qualitative comparison of top-1 examples on Charades-STA dataset (best viewed in colors). The three colored boxes are the moment boundaries corresponding to the input query. The ground truth is in blue, while the predictions from the baseline and PaTF are in red and green, respectively.

Despite showing competitive performance, some potential limitations can be recognised. First, our method remains to be evaluated on other datasets, which require scene graph annotations. Second, the scene graph representation is simplified in the current method, and its generalisation is necessary for a wider range of applications. Finally, query-irrelevant masking currently relies on keyword matching which is also not a generalisable technique.

ACKNOWLEDGEMENT

This research was conducted according to the agreement on internship program of NEC Corporation. The authors affiliated with Dublin City University were supported by Science Foundation Ireland under Grant Agreement Nos. 18/CRT/6223, and 13/RC/2106_P2 at the ADAPT SFI Research Centre at DCU. ADAPT, the SFI Research Centre for AI-Driven Digital Content Technology is funded by Science Foundation Ireland through the SFI Research Centres Programme.

REFERENCES

[1] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. 2021. Is Space-Time Attention All You Need for Video Understanding? *CoRR* abs/2102.05095 (2021). arXiv:2102.05095 https://arxiv.org/abs/2102.05095

[2] Navaneeth Bodla, Bharat Singh, Rama Chellappa, and Larry S. Davis. 2017. Soft-NMS – Improving Object Detection With One Line of Code. (2017).

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-End Object Detection with Transformers. *CoRR* abs/2005.12872 (2020). arXiv:2005.12872 https://arxiv.org/abs/2005.12872

[4] João Carreira and Andrew Zisserman. 2017. Quo Vadis, Action Recognition? A New Model and the Kinetics Dataset. *CoRR* abs/1705.07750 (2017). arXiv:1705.07750 http://arxiv.org/abs/1705.07750

[5] Joao 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*. 6299–6308.

[6] Jingyuan Chen, Xinpeng Chen, Lin Ma, Zequn Jie, and Tat-Seng Chua. 2018. Temporally Grounding Natural Sentence in Video. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii (Eds.). Association for Computational Linguistics, Brussels, Belgium, 162–171. https://doi.org/10.18653/v1/D18-1015

[7] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. 2020. Fine-grained Video-Text Retrieval with Hierarchical Graph Reasoning. *CoRR* abs/2003.00392 (2020). arXiv:2003.00392 https://arxiv.org/abs/2003.00392

[8] Yuren Cong, Michael Ying Yang, and Bodo Rosenhahn. 2023. RelTR: Relation Transformer for Scene Graph Generation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45, 9 (2023), 11169–11183. https://doi.org/10.1109/TPAMI.2023.3268066

[9] Naina Dhingra, Florian Ritter, and Andreas M. Kunz. 2021. BGT-Net: Bidirectional GRU Transformer Network for Scene Graph Generation. *CoRR* abs/2109.05346 (2021). arXiv:2109.05346 https://arxiv.org/abs/2109.05346

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

[11] Vijay Prakash Dwivedi and Xavier Bresson. 2020. A Generalization of Transformer Networks to Graphs. *CoRR* abs/2012.09699 (2020). arXiv:2012.09699 https://arxiv.org/abs/2012.09699

[12] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. 2018. Summarizing Videos with Attention. *CoRR* abs/1812.01969 (2018). arXiv:1812.01969 http://arxiv.org/abs/1812.01969

[13] Christoph Feichtenhofer. 2020. X3D: Expanding Architectures for Efficient Video Recognition. *CoRR* abs/2004.04730 (2020). arXiv:2004.04730 https://arxiv.org/abs/2004.04730

[14] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. 2020. Multi-modal Transformer for Video Retrieval. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 214–229.

[15] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. 2017. Tall: Temporal activity localization via language query. In *Proceedings of the IEEE international conference on computer vision*. 5267–5275.

[16] J. Gao and C. Xu. 2021. Fast Video Moment Retrieval. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE Computer Society, Los Alamitos, CA, USA, 1503–1512. https://doi.org/10.1109/ICCV48922.2021.00155

[17] Lianli Gao, Zhao Guo, Hanwang Zhang, Xing Xu, and Heng Tao Shen. 2017. Video Captioning With Attention-Based LSTM and Semantic Consistency. *IEEE Transactions on Multimedia* 19, 9 (2017), 2045–2055. https://doi.org/10.1109/TMM.2017.2729019

[18] Jiuxiang Gu, Shafiq R. Joty, Jianfei Cai, Handong Zhao, Xu Yang, and Gang Wang. 2019. Unpaired Image Captioning via Scene Graph Alignments. *CoRR* abs/1903.10658 (2019). arXiv:1903.10658 http://arxiv.org/abs/1903.10658

[19] Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. 2019. Action Genome: Actions as Composition of Spatio-temporal Scene Graphs. *CoRR* abs/1912.06992 (2019). arXiv:1912.06992 http://arxiv.org/abs/1912.06992

[20] Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2015. Image retrieval using scene graphs.

In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3668–3678. https://doi.org/10.1109/CVPR.2015.7298990

[21] Jie Lei, Tamara L. Berg, and Mohit Bansal. 2021. QVHighlights: Detecting Moments and Highlights in Videos via Natural Language Queries. *CoRR* abs/2107.09609 (2021). arXiv:2107.09609 https://arxiv.org/abs/2107.09609

[22] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. 2021. Less is More: ClipBERT for Video-and-Language Learning via Sparse Sampling. *CoRR* abs/2102.06183 (2021). arXiv:2102.06183 https://arxiv.org/abs/2102.06183

[23] Yikang Li, Wanli Ouyang, Bolei Zhou, Kun Wang, and Xiaogang Wang. 2017. Scene Graph Generation from Objects, Phrases and Caption Regions. *CoRR* abs/1707.09700 (2017). arXiv:1707.09700 http://arxiv.org/abs/1707.09700

[24] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. 2016. Feature Pyramid Networks for Object Detection. *CoRR* abs/1612.03144 (2016). arXiv:1612.03144 http://arxiv.org/abs/1612.03144

[25] Ilya Loshchilov and Frank Hutter. 2017. Fixing Weight Decay Regularization in Adam. *CoRR* abs/1711.05101 (2017). arXiv:1711.05101 http://arxiv.org/abs/1711.05101

[26] 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 *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 5144–5153. https://doi.org/10.18653/v1/D19-1518

[27] Cewu Lu, Ranjay Krishna, Michael S. Bernstein, and Li Fei-Fei. 2016. Visual Relationship Detection with Language Priors. *CoRR* abs/1608.00187 (2016). arXiv:1608.00187 http://arxiv.org/abs/1608.00187

[28] Shaohui Mei, Genliang Guan, Zhiyong Wang, Shuai Wan, Mingyi He, and David Dagan Feng. 2015. Video summarization via minimum sparse reconstruction. *Pattern Recognition* 48, 2 (2015), 522–533. https://doi.org/10.1016/j.patcog.2014.08.002

[29] WonJun Moon, Sangeek Hyun, SangUk Park, Dongchan Park, and Jae-Pil Heo. 2023. Query-dependent video representation for moment retrieval and highlight detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 23023–23033.

[30] Jonghwan Mun, Minsu Cho, and Bohyung Han. 2020. Local-Global Video-Text Interactions for Temporal Grounding. *CoRR* abs/2004.07514 (2020). arXiv:2004.07514 https://arxiv.org/abs/2004.07514

[31] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. 2020. Spatio-Temporal Graph for Video Captioning With Knowledge Distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[32] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *CoRR* abs/1912.01703 (2019). arXiv:1912.01703 http://arxiv.org/abs/1912.01703

[33] Sally Sisi Qu, Mattia Soldan, Mengmeng Xu, Jesper Tegnér, and Bernard Ghanem. 2020. VLG-Net: Video-Language Graph Matching Network for Video Grounding. *CoRR* abs/2011.10132 (2020). arXiv:2011.10132 https://arxiv.org/abs/2011.10132

[34] Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. In *ICML*.

[35] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. 2015. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *CoRR* abs/1506.01497 (2015). arXiv:1506.01497 http://arxiv.org/abs/1506.01497

[36] Gunnar A. Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. 2016. Hollywood in Homes: Crowdsourcing Data Collection for Activity Understanding. *CoRR* abs/1604.01753 (2016). arXiv:1604.01753 http://arxiv.org/abs/1604.01753

[37] Haoyu Tang, Jihua Zhu, Lin Wang, Qinghai Zheng, and Tianwei Zhang. 2022. Multi-Level Query Interaction for Temporal Language Grounding. *IEEE Transactions on Intelligent Transportation Systems* 23, 12 (2022), 25479–25488. https://doi.org/10.1109/TITS.2021.3110713

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

[39] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. 2016. Temporal Segment Networks: Towards Good Practices for Deep Action Recognition. *CoRR* abs/1608.00859 (2016). arXiv:1608.00859 http://arxiv.org/abs/1608.00859

[40] Sijin Wang, Ruiping Wang, Ziwei Yao, Shiguang Shan, and Xilin Chen. 2019. Cross-modal Scene Graph Matching for Relationship-aware Image-Text Retrieval. *CoRR* abs/1910.05134 (2019). arXiv:1910.05134 http://arxiv.org/abs/1910.05134

[41] Zhaoqing Wang, Yu Lu, Qiang Li, Xunqiang Tao, Yandong Guo, Mingming Gong, and Tongliang Liu. 2022. CRIS: CLIP-Driven Referring Image Segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 11686–11695.

[42] Shaoning Xiao, Long Chen, Songyang Zhang, Wei Ji, Jian Shao, Lu Ye, and Jun Xiao. 2021. Boundary Proposal Network for Two-Stage Natural Language Video Localization. *CoRR* abs/2103.08109 (2021). arXiv:2103.08109 https://arxiv.org/abs/2103.08109

[43] Yicheng Xiao, Zhuoyan Luo, Yong Liu, Yue Ma, Hengwei Bian, Yatai Ji, Yujiu Yang, and Xiu Li. 2023. Bridging the Gap: A Unified Video Comprehension Framework for Moment Retrieval and Highlight Detection. arXiv:2311.16464 [cs.CV]

[44] Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. 2017. Scene Graph Generation by Iterative Message Passing. *CoRR* abs/1701.02426 (2017). arXiv:1701.02426 http://arxiv.org/abs/1701.02426

[45] Yifang Xu, Yunzhuo Sun, Yang Li, Yilei Shi, Xiaoxiang Zhu, and Sidan Du. 2023. MH-DETR: Video Moment and Highlight Detection with Cross-modal Transformer. *arXiv preprint arXiv:2305.00355* (2023).

[46] Shen Yan, Xuehan Xiong, Anurag Arnab, Zhichao Lu, Mi Zhang, Chen Sun, and Cordelia Schmid. 2022. Multiview Transformers for Video Recognition. *CoRR* abs/2201.04288 (2022). arXiv:2201.04288 https://arxiv.org/abs/2201.04288

[47] Shen Yan, Xuehan Xiong, Arsha Nagrani, Anurag Arnab, Zhonghao Wang, Weina Ge, David Ross, and Cordelia Schmid. 2023. UnLoc: A Unified Framework for Video Localization Tasks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 13623–13633.

[48] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. 2018. Graph R-CNN for Scene Graph Generation. *CoRR* abs/1808.00191 (2018). arXiv:1808.00191 http://arxiv.org/abs/1808.00191

[49] Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019. Auto-Encoding Scene Graphs for Image Captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

[50] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring Visual Relationship for Image Captioning. *CoRR* abs/1809.07041 (2018). arXiv:1809.07041 http://arxiv.org/abs/1809.07041

[51] Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2017. Neural Motifs: Scene Graph Parsing with Global Context. *CoRR* abs/1711.06640 (2017). arXiv:1711.06640 http://arxiv.org/abs/1711.06640

[52] Runhao Zeng, Haoming Xu, Wenbing Huang, Peihao Chen, Mingkui Tan, and Chuang Gan. 2020. Dense Regression Network for Video Grounding. *CoRR* abs/2004.03545 (2020). arXiv:2004.03545 https://arxiv.org/abs/2004.03545

[53] Yawen Zeng, Da Cao, Xiaochi Wei, Meng Liu, Zhou Zhao, and Zheng Qin. 2021. Multi-Modal Relational Graph for Cross-Modal Video Moment Retrieval. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2215–2224. https://doi.org/10.1109/CVPR46437.2021.00225

[54] Chen-Lin Zhang, Jianxin Wu, and Yin Li. 2022. ActionFormer: Localizing Moments of Actions with Transformers. In *European Conference on Computer Vision (LNCS, Vol. 13664)*. 492–510.

[55] Hao Zhang, Aixin Sun, Wei Jing, and Joey Tianyi Zhou. 2020. Span-based Localizing Network for Natural Language Video Localization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Online, 6543–6554. https://www.aclweb.org/anthology/2020.acl-main.585

[56] Ji Zhang, Kevin J. Shih, Ahmed Elgammal, Andrew Tao, and Bryan Catanzaro. 2019. Graphical Contrastive Losses for Scene Graph Generation. *CoRR* abs/1903.02728 (2019). arXiv:1903.02728 http://arxiv.org/abs/1903.02728

[57] Songyang Zhang, Houwen Peng, Jianlong Fu, and Jiebo Luo. 2019. Learning 2D Temporal Adjacent Networks for Moment Localization with Natural Language. *CoRR* abs/1912.03590 (2019). arXiv:1912.03590 http://arxiv.org/abs/1912.03590

[58] Zongmeng Zhang, Xianjing Han, Xuemeng Song, Yan Yan, and Liqiang Nie. 2021. Multi-Modal Interaction Graph Convolutional Network for Temporal Language Localization in Videos. *CoRR* abs/2110.06058 (2021). arXiv:2110.06058 https://arxiv.org/abs/2110.06058

[59] Zhu Zhang, Zhijie Lin, Zhou Zhao, and Zhenxin Xiao. 2019. Cross-Modal Interaction Networks for Query-Based Moment Retrieval in Videos. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM. https://doi.org/10.1145/3331184.3331235

[60] Kaiyang Zhou and Yu Qiao. 2018. Deep Reinforcement Learning for Unsupervised Video Summarization with Diversity-Representativeness Reward. *CoRR* (2018). http://arxiv.org/abs/1801.00054