Motion Guided Region Message Passing for Video Captioning

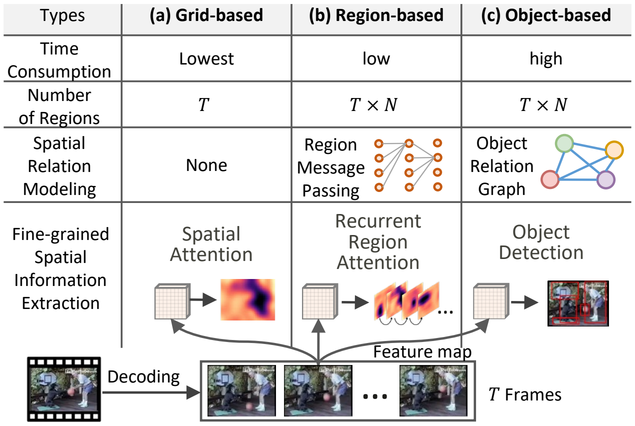
Shaoxiang Chen and Yu-Gang Jiang*
Shanghai Key Lab of Intelligent Information Processing, School of Computer Science, Fudan University
Shanghai Collaborative Innovation Center on Intelligent Visual Computing
{sxchen13, ygj}@fudan.edu.cn

Abstract

Video captioning is an important vision task and has been intensively studied in the computer vision community. Existing methods that utilize the fine-grained spatial information have achieved significant improvements, however, they either rely on costly external object detectors or do not sufficiently model the spatial/temporal relations. In this paper, we aim at designing a spatial information extraction and aggregation method for video captioning without the need of external object detectors. For this purpose, we propose a Recurrent Region Attention module to better extract diverse spatial features, and by employing Motion-Guided Cross-frame Message Passing, our model is aware of the temporal structure and able to establish high-order relations among the diverse regions across frames. They jointly encourage information communication and produce compact and powerful video representations. Furthermore, an Adjusted Temporal Graph Decoder is proposed to flexibly update video features and model high-order temporal relations during decoding. Experimental results on three benchmark datasets: MSVD, MSR-VTT, and VATEX demonstrate that our proposed method can outperform state-of-the-art methods.

1. Introduction

Automatically generating sentences to describe video contents, i.e., video captioning, has been attracting research attention from both the computer vision and natural language processing communities. From the vision perspective, extracting and fully utilizing the information contained in the video is the key of improving video captioning. Recent advancements of video captioning methods [6, 16, 55, 26, 50, 57, 58] can also be mainly attributed to the exploration of more fine-grained spatial information within the video frames. An representative example, ORG-TRL [57] (Fig. 1 (c)), detects spatial bounding boxes of the important objects with an external object detector [32] and then builds an object relation graph to model the relations among all the objects. Objects together with their relations are undoubtedly crucial for video captioning, because the object interactions can be explicitly captured and lead to a better understanding of the video contents.



Types	(a) Grid-based	(b) Region-based	(c) Object-based
Time Consumption	Lowest	low	high
Number of Regions	T	$T \times N$	$T \times N$
Spatial Relation Modeling	None	Region Message Passing	Object Relation Graph
Fine-grained Spatial Information Extraction	Spatial Attention	Recurrent Region Attention	Object Detection

Figure 1. Illustration of video captioning methods with different types of fine-grained spatial information extraction strategies: (a) Grid-based: e.g., MGSA [6], attends to one region per frame. (b) Region-based: our proposed method, extracts multiple regions and performs message passing to conduct relation modeling. (c) Object-based: e.g., ORG-TRL [57], can extract multiple regions on each frame and model their relations, but are computationally less efficient due to the use of object detectors.

However, it is costly to extract localized object features by object detection and pretrained object detectors can not generalize well to animated or video game contents in some datasets [46]. The recent research of Jiang *et al.* [17] compares grid-based and object region-based features, and shows that incorporating object detectors into image visual question answering models can significantly slow down the models by 4.6 to 23.8 times, while using such object features does not bring significant advantages (in terms of accuracy) over using plain CNN feature maps (or grid features). They also conclude that the semantic content that features represent is more critical than the format of features. Besides, incorporating an object detector requires densely annotated external data and also increases the fi-

*Corresponding author.

nal model size of the whole system. Inspired by [17], we revisit grid features for video-and-language models with a focus on video captioning, and preliminarily explore its application for temporal sentence localization in videos.

In fact, utilizing grid features is widely adopted by recent video captioning methods [6, 51, 21, 49, 43, 33]. They usually calculate one spatial attention map for each CNN feature map (Fig. 1 (a)) to capture the most salient object, then spatially aggregate each feature map into a condensed feature vector. The core of these methods is emphasizing one important region[1] in each video frame, and their major differences are in the ways to compute spatial attentions. But the problem with these methods is that only one region is attended in each frame, so some information may be missed, and there is no way to model the interactions among different regions. In this respect, the advantages of object-based methods [50, 16, 57, 55, 26, 58] are 1) the detected objects can comprehensively capture multiple regions of interest, and 2) the relation modeling among objects.

To tackle the issues existed in previous grid feature-based methods, we propose Recurrent Region Attention to extract multiple diverse regions from each video frame and design Motion Guided Cross-Frame Message Passing to encourage the interaction and information communication among regions of consecutive frames. Moreover, we propose Adjusted Temporal Graph Decoder, which updates the high-order temporal relations among video features based on the decoding state to more flexibly form compact video representations. So that our method possesses the two essential factors of object-based methods (Fig. 1 (b)). We also note that the goal of this paper is not to refute the use of object detectors or to reach a compromise between computational cost and accuracy. Instead, we aim at fully utilizing the semantic content in grid features to further explore its potentials and scope of application.

The contributions are summarized as follows: (1) We proposed Recurrent Region Attention and Motion Guided Cross-Frame Message Passing, which jointly are a new method of extracting and encoding fine-grained spatial information for video captioning. (2) We proposed Adjusted Temporal Graph Decoder, which is a more flexible captioning decoder that can adjust video features based on high-order temporal relations. (3) We tested our proposed method on the popular MSVD and MSR-VTT datasets and the newly-released VATEX dataset, and achieved state-of-the-art video captioning performances on all datasets.

2. Related Work

We provide a **brief review of video captioning** methods with the focus on methods that use grid and object features.

Multimodal Fusion. A small number of deep learning-based video captioning methods use only one type of feature [27, 35, 28, 42] from a single modality. Fusing features from multiple modalities (visual, motion, audio, and semantics) is a common strategy adopted by most video captioning methods. Their differences are in 'how to fuse'. Soft-attention is a straightforward way to dynamically assign importance weights to each modality and then combine them [14, 47, 56, 25]. Several methods [44, 31] use memory to organize multimodal features and the language decoder reads from the memory slots via attention when decoding a specific word. If the method is focused on the feature encoding or language decoding, they may adopt simple strategies such as concatenation to combine multimodal features [1, 15, 5, 59]. There are a group of methods that focus on designing complex multimodal fusion strategies. HOCA [18] is a method that can efficiently perform high-order cross-modal attention for up to three modalities. POS-CG [41] designs a cross-gating network to dynamically incorporate multimodal features into the language decoder. SibNet [24] consists of two branches for processing visual content and semantics separately, and both branches are supervised by regularizing objectives. In our method, multimodal fusion is implicitly performed during the MGCMP.

Spatial Attention on Grid Features. The idea of spatially attending to grid features to dynamically select relevant features was originally proposed for image captioning [48]. DMRM [51] is an early attempt at adaptively selecting regions-of-interest for the video frames, and it further incorporates a Dual Memory Recurrent Model (DMRM) to model the temporal dependencies of the aggregated features. MAM-RNN [21] is another contemporary method, and it establishes sequential temporal connections of the attended regions by recurrently passing spatial attention weights. SAM [43] first predicts salient regions in each frame and then spatially aggregate foreground and background grid features with VLAD. Similarly, SeqVLAD [49] directly aggregates grid features of each frame, except that its VLAD assignment parameters are produced by a recurrent CNN so there are temporal connections among frames. VRE [33] first identifies the key frames of a video and then spatially aggregates the grid features of key frames to generate captions. MGSA [6] uses motion information extracted from optical flows to guide the computation of spatial attention on grid features, and the attention maps are temporally connected via a recurrent unit. However, the common problem of these methods is that they attend to each frame only once and the features obtained from each frame lack diversity, and this further leads to their inability to perform interactions among the attended regions. A recent study [22] has shown that densely sampled rectangular areas can form powerful representations for a variety of tasks including image captioning. But for videos, such exhaustive enumera-

[1] We use the term 'region' to refer to an area in a feature map, usually represented by an attention distribution, where some grids are deemed more important than the others.

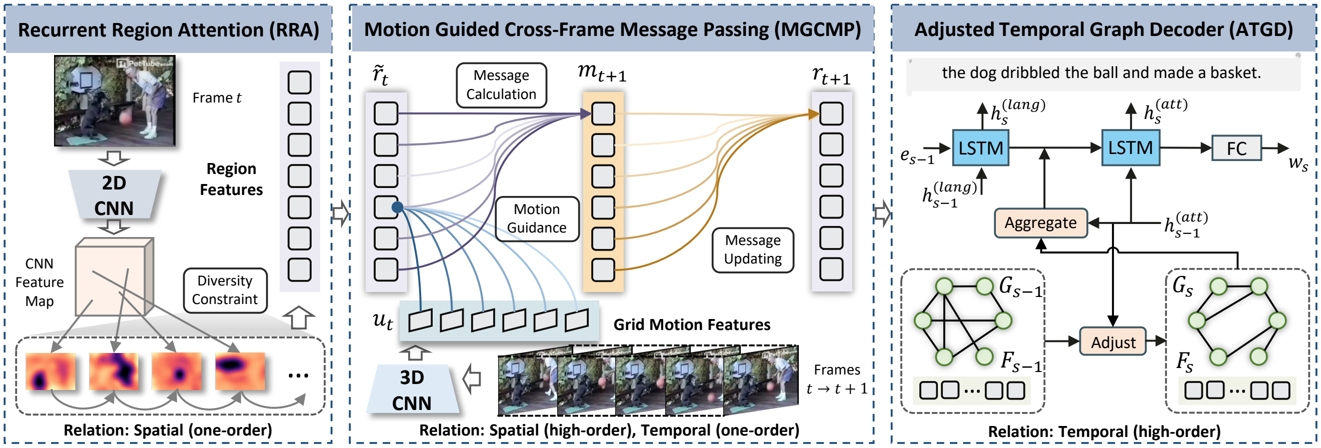


Figure 2. Overview of our proposed method, which has three module: (1) the Recurrent Region Attention (Sec. 3.1, RRA) that extracts diverse visual features from the feature map of each frame; (2) the Motion Guided Cross-Frame Message Passing (Sec. 3.2, MGCMP) to temporally pass information of the region features through motion guided interaction; (3) the Adjusted Temporal Graph Decoder (Sec. 3.3, ATGD) to further encourage high-order interactions of the video features by adjusting the graph structure conditioned on the decoding state, and finally generates caption words. One-order and high-order spatial/temporal relations are modeled properly in these modules.

tion of all possible rectangular areas can lead to unaffordable GPU memory usage and computational cost. Our proposed method aims to solve the lack of region diversity and interaction to obtain better video representations.

Utilizing Detected Objects. Object interaction has been considered in image captioning methods [53, 29, 20], but for videos, the spatial-temporal interactions among objects are more complex and challenging. Most recent methods for video captioning adopt pretrained object detectors to extract object features and model the interactions among objects. HTM [16] and STAT [50] simply extract object region features as a type of local feature and perform soft-attention to aggregate them. OA-BTG [55] builds a bidirectional temporal graph to model the temporal dynamics of the objects and aggregate object features by VLAD. STG [26] builds both a spatial graph among the objects in the same frame and a temporal graph across frames, and updates object features by applying graph convolutions. The updated object features are spatially average-pooled. SAAT [58] incorporates spatial location into objects' regional features and models pairwise object relations by a dot-product attention. ORG-TRL [57] similarly models the pair-wise object relation by a dot-product attention, then the updated object features are temporally connected by similarity and aggregated via temporal attention followed by spatial attention. These methods usually achieve better performances than methods that use grid features, the most distinctive difference is that diverse object regions can be extracted and the pair-wise interactions among objects are explicitly modeled here. Thus in this paper, we mainly focus on the message passing among the extracted regions to encourage their interaction and information communication, and the spatial and temporal high-order relation modeling are decoupled.

3. Methodology

As shown in Figure 2, the proposed method can be divided into three sequential steps: extract, interact, and generate. First, the Recurrent Region Attention extracts region-wise localized features from video frames; second, the Motion Guided Cross-Frame Message Passing allows the features to interact and communicate information across time; finally, the Adjusted Temporal Graph Decoder generates caption words by flexibly updating and aggregating the video features. We will show the details of these modules in the following subsections.

3.1. Recurrent Region Attention

Given a video, we extract spatial feature maps for its uniformly sampled T frames from pre-trained CNNs [34, 13, 36], forming a grid feature sequence $\boldsymbol{V} = \{\boldsymbol{v}_1, ..., \boldsymbol{v}_T\}$, where $\boldsymbol{v}_t \in \mathbb{R}^{L_v \times C_v}$, and $L_v = H_v \times W_v$ and C_v is the number of spatial grids and channels, respectively. Different from previous methods that spatially attend to each frame only once, the goal of recurrent region attention is to extract multiple diverse region features for each frame. For easier implementation, we fix the number of regions in each frame to N, and the computation of the n-th region feature for the t-th frame is formulated as follows.

We first compute a guidance vector [2] $\boldsymbol{g}_{t,n} \in \mathbb{R}^{d_g}$ based on the globally average-pooled feature $\bar{\boldsymbol{v}}_t \in \mathbb{R}^{C_v}$ and the previous region feature $\boldsymbol{r}_{t,n-1} \in \mathbb{R}^{C_v}$:

$$\boldsymbol{g}_{t,n} = \texttt{ReLU}(\boldsymbol{W}_g([\boldsymbol{W}_v \bar{\boldsymbol{v}}_t, \boldsymbol{r}_{t,n-1}])), \qquad (1)$$

where $\boldsymbol{W}_g$ and $\boldsymbol{W}_v$ are learnable weights and [.] denotes tensor concatenation along their last axis. Then an attention

[2] Bold symbols denote multi-dimensional tensors, and we index their axes using comma-separated subscripts or square bracket-enclosed indices following the programming convention of Pytorch.

distribution over each location of the spatial feature map is calculated:

$$\begin{aligned} \boldsymbol{\alpha}_{t,n,l} &= \boldsymbol{W}_{att}(\tanh(\boldsymbol{W}_{g\alpha}\boldsymbol{g}_{t,n} + \boldsymbol{W}_{v\alpha}\boldsymbol{v}_{t,l})), \\ \boldsymbol{\alpha}_{t,n} &= \texttt{Softmax}([\boldsymbol{\alpha}_{t,n,1}, ..., \boldsymbol{\alpha}_{t,n,L_v}]), \end{aligned} \tag{2}$$

where $\boldsymbol{W}_{att}$, $\boldsymbol{W}_{g\alpha}$, and $\boldsymbol{W}_{v\alpha}$ are learnable weights, and $\boldsymbol{\alpha}_{t,n,l}$ is a scalar. Finally, the n-th region feature is obtained via a weighted-sum of each grid feature.

$$\boldsymbol{r}_{t,n} = \textstyle\sum_{l=1}^{L_v} \boldsymbol{\alpha}_{t,n,l}\boldsymbol{v}_{t,l}. \tag{3}$$

The diversity of the regions are enforced by a diversity loss

$$\begin{aligned} &\mathcal{L}_{div} = \textstyle\sum_{t=1}^{T} ||\boldsymbol{A}_t\boldsymbol{A}_t^\top - \lambda \boldsymbol{I}||, \\ &\text{where} \quad \boldsymbol{A}_t \in \mathbb{R}^{N\times N}, \quad \boldsymbol{A}_t[i,j] = \boldsymbol{\alpha}_{t,i} \cdot \boldsymbol{\alpha}_{t,j}, \end{aligned} \tag{4}$$

$\cdot$ denotes dot-product, $\boldsymbol{I}$ is an identity matrix, and λ is a tunable hyperparameter. By constraining the non-diagonal elements in the attention matrices $\boldsymbol{A}_t$ to be small, different attentions can have less overlap. λ controls the 'softness' of the attention distributions, if λ is close to 1, each attention distribution $\boldsymbol{\alpha}_{t,i}$ tends to be one-hot, i.e., becomes hard attention.

3.2. Motion Guided Cross-Frame Message Passing

The RRA have extracted diverse region features from each frame and simultaneously modeled intra-frame relations via the recurrent dependency. Passing information across frames has been proven effective for video captioning, and the goal of MGCMP is to establish temporal connections among the regions and encourage information communication across frames. Inspired by the Message Passing Neural Network framework [11], we iteratively update the N region features from each frame $\boldsymbol{r}_t$, but at each time step, the message passing is performed between two consecutive frames, instead of within the same graph [11].

The message passing runs for T steps, and each step includes message calculation and message updating:

$$\begin{aligned} \boldsymbol{m}_t &= M_t(\tilde{\boldsymbol{r}}_t, \boldsymbol{u}_t, \boldsymbol{A}_t^{(m)}), \\ \tilde{\boldsymbol{r}}_{t+1} &= U_t(\boldsymbol{r}_{t+1}, \boldsymbol{m}_t, \boldsymbol{A}_t^{(u)}). \end{aligned} \tag{5}$$

During the calculation phase $M_t(\cdot)$, we introduce motion guidance $\boldsymbol{U} = \{\boldsymbol{u}_1, ..., \boldsymbol{u}_T\}$, where $\boldsymbol{u}_t \in \mathbb{R}^{L_u \times C_u}$ is a summary of temporal dynamics between frames t and $t+1$, to compensate for the information loss due to frame sampling. For simplicity, the motion features are in grid form, and $\boldsymbol{A}_t^{(m)}$ is a dynamically-computed affinity matrix. During the updating phase $U_t(\cdot)$, the regions of frame $t+1$ absorb calculated messages $\boldsymbol{m}_t$ according to a dynamic affinity matrix $\boldsymbol{A}_t^{(u)}$. $\tilde{\boldsymbol{r}}_{t+1}$ is the updated region features and will be passed to the next step. The process of one message passing step is illustrated in Fig. 3. We then formulate each step in detail.

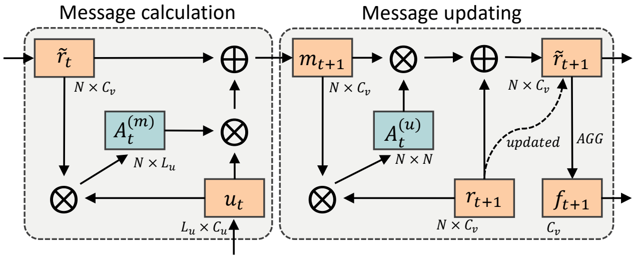


Figure 3. Illustration of the Motion Guided Cross Frame Message Passing. $\bigoplus$ denotes element-wise addition and $\bigotimes$ denotes matrix multiplication.

Message calculation. First, the dynamic affinity matrix between the regions and motion grids is computed as:

$$\begin{aligned} \boldsymbol{A}_t^{(m)}[n,l] &= \texttt{ReLU}(\boldsymbol{W}_{rm}\tilde{\boldsymbol{r}}_{t,n}) \cdot \texttt{ReLU}(\boldsymbol{W}_{um}\boldsymbol{u}_{t,l}), \\ \boldsymbol{A}_t^{(m)}[n,l] &= \texttt{Softmax}_c(\boldsymbol{A}_t^{(m)}[n,l]) \\ &= \frac{\exp(\boldsymbol{A}_t^{(m)}[n,l])}{\sum_{l=1}^{L_u}\exp(\boldsymbol{A}_t^{(m)}[n,l])}, \end{aligned} \tag{6}$$

where $\boldsymbol{W}_{rm}$ and $\boldsymbol{W}_{um}$ are learnable weights, and the function $\texttt{Softmax}_c$ denotes a softmax operation along the column axis. $\boldsymbol{A}_t^{(m)}$ allows each region feature to collect complementary information from motion grids to complete itself, thus carrying richer information to the next sampled frame. Then we compute the message for each region by aggregating information from the motion grids:

$$\boldsymbol{m}_{t,n} = \tilde{\boldsymbol{r}}_{t,n} + \textstyle\sum_{l=1}^{L_u} \boldsymbol{A}_t^{(m)}[n,l]\boldsymbol{W}_{om}\boldsymbol{u}_{t,l}, \tag{7}$$

where $\boldsymbol{W}_{om}$ is a learnable weight matrix used to transform motion features into the same dimensionality as region features. The formulation is similar to self-attention networks [39], and we also find that adding a multi-layer perceptron (MLP) to the message output is helpful.

Message updating. The calculated messages are then passed to the region nodes of the next frame to update the nodes. Similarly, each node of the next frame $\boldsymbol{r}_{t+1,i}$ picks relevant messages via the affinity matrix:

$$\begin{aligned} \boldsymbol{A}_t^{(u)}[i,j] &= \texttt{ReLU}(\boldsymbol{W}_{ru}\boldsymbol{r}_{t+1,i}) \cdot \texttt{ReLU}(\boldsymbol{W}_{uu}\boldsymbol{m}_{t,j}), \\ \boldsymbol{A}_t^{(u)}[i,j] &= \texttt{Softmax}_c(\boldsymbol{A}_t^{(u)}[i,j]) \\ &= \frac{\exp(\boldsymbol{A}_t^{(u)}[i,j])}{\sum_{j=1}^{n}\exp(\boldsymbol{A}_t^{(u)}[i,j])}, \end{aligned} \tag{8}$$

where $\boldsymbol{W}_{ru}$ and $\boldsymbol{W}_{uu}$ are learnable weights. Then we obtain the updated node feature by aggregating its relevant messages:

$$\tilde{\boldsymbol{r}}_{t+1,i} = \boldsymbol{r}_{t+1,i} + \textstyle\sum_{j=1}^{n} \boldsymbol{A}_t^{(u)}[i,j]\boldsymbol{W}_{of}\boldsymbol{m}_{t,j}, \tag{9}$$

where $\boldsymbol{W}_{of}$ is a learnable weight matrix. Compared to ORG-TRL [57], which either isolates each frame or roughly merge all frames into a single graph, our MGCMP respects the inherent temporal structure and models high-order spatial relations among the regions. Since the caption decoder needs a compact video representation for efficient decoding, we aggregate regions of each frame:

$$\boldsymbol{f}_{t+1} = \texttt{AGG}(\{\tilde{\boldsymbol{r}}_{t+1,i}\}_{i=1}^{N}), \tag{10}$$

where $\texttt{AGG}(\cdot)$ is a region feature aggregation function. While there exist multiple choices for $\texttt{AGG}(\cdot)$, in our design, a simple mean-pooling along the region axis can achieve satisfactory performance. The updated region features $\boldsymbol{F} = \{\boldsymbol{f}_1, ..., \boldsymbol{f}_T\} \in \mathbb{R}^{T\times C_v}$ are inputs to the caption decoder.

3.3. Adjusted Temporal Graph Decoder

The goal of the caption generation module is temporally aggregating the video features and forming a more compact representation as the input to each decoding step, but the most widely-adopted aggregation method [52] does not model high-order temporal relations of the video features. In addition to the one-order temporal relation in MGCMP, we introduce Adjusted Temporal Graph Decoder (ATGD), a module which incorporates graph adjustment [54] to establish high-order temporal relations among the video features and also adjust the features based on the decoder states.

The caption decoding runs for S steps. The ATGD consists of a stacked LSTM network and an adjusted graph convolution network. As demonstrated in Figure 2, we show the formulation of a single decoding step here. The high-order temporal relations of the features $\{\boldsymbol{f}_1, ..., \boldsymbol{f}_T\}$ are represented by a dynamic graph, defined by the adjacency matrix $\boldsymbol{G}_s \in \mathbb{R}^{T\times T}$. With the basic form of graph convolution, we iteratively update the video feature representation $\boldsymbol{F}$:

$$\boldsymbol{F}_s = \texttt{ReLU}(\boldsymbol{G}_{s-1}\boldsymbol{F}_{s-1}\boldsymbol{W}_G), \tag{11}$$

where $\boldsymbol{W}_G \in \mathbb{R}^{C_v\times C_v}$ is a learnable weight matrix, $\boldsymbol{F}_0 = \boldsymbol{F}$, and $\boldsymbol{G}_0$ is an identity matrix to emphasize self-relation at the beginning. The updated video features are then aggregated under the guidance of the decoder state:

$$\begin{aligned} \boldsymbol{\beta}_s &= \boldsymbol{W}_{agg}(\tanh(\boldsymbol{W}_{hv}\boldsymbol{h}_{s-1}^{att} + \boldsymbol{W}_F\boldsymbol{F}_s + \boldsymbol{b}_{agg})), \\ \boldsymbol{\beta}_s &= \texttt{Softmax}(\boldsymbol{\beta}_s), \\ \bar{\boldsymbol{F}}_s &= \textstyle\sum_{t=1}^{T}\boldsymbol{\beta}_{s,t}\boldsymbol{F}_{s,t}, \end{aligned} \tag{12}$$

where $\boldsymbol{\beta}_s \in \mathbb{R}^T$ is the aggregation weights, $\bar{\boldsymbol{F}}_s \in \mathbb{R}^{C_v}$ is the aggregated feature input to the decoder, $\boldsymbol{W}_{agg}$, $\boldsymbol{b}_{agg}$, $\boldsymbol{W}_{hv}$, and $\boldsymbol{W}_F$ are learnable weights, and $\boldsymbol{h}_{s-1}^{att}$ is the decoder state described below.

$$\begin{aligned} \boldsymbol{h}_s^{lang} &= \texttt{LSTM}^{lang}(\boldsymbol{e}_{s-1}; \boldsymbol{h}_{s-1}^{lang}), \\ \boldsymbol{h}_s^{att} &= \texttt{LSTM}^{att}([\bar{\boldsymbol{F}}_s, \boldsymbol{h}_s^{lang}]; \boldsymbol{h}_{s-1}^{att}), \end{aligned} \tag{13}$$

where $\texttt{LSTM}(;)$ denotes a LSTM cell that accepts an input and a previous state at each step, $\boldsymbol{e}_{s-1}$ is the word embedding vector of the $s-1$-th word, and $\boldsymbol{h}_s^{lang}$ and $\boldsymbol{h}_s^{att} \in \mathbb{R}^{d_{lstm}}$ are the hidden states of the two LSTMs. The caption is predicted one word at a step by applying a fully-connected layer on the decoder state:

$$\begin{aligned} \boldsymbol{p}_s &= \texttt{Softmax}(\texttt{FC}(\boldsymbol{h}_s^{att})), \\ w_s &= \arg\max \boldsymbol{p}_s, \end{aligned} \tag{14}$$

where $\boldsymbol{p}_s \in \mathbb{R}^{N_w}$ is a distribution over a vocabulary of N_w words and w_s is the index of the predicted word in the vocabulary. Most importantly, we also adjust the graph structure according to the decoder state at each decoding step.

$$\begin{aligned} \boldsymbol{F}_s' &= \texttt{sigmoid}(\boldsymbol{W}_{adj}\boldsymbol{h}_{s-1}^{att}) \odot \boldsymbol{F}_s, \\ \Delta_s &= \texttt{norm}(\boldsymbol{F}_s'\boldsymbol{W}_D\boldsymbol{F}_s'^{\top}), \\ \boldsymbol{G}_s &= \texttt{sigmoid}(\boldsymbol{G}_{s-1} + \Delta_s), \end{aligned} \tag{15}$$

where $\odot$ denotes element-wise multiplication, $\texttt{norm}(\cdot)$ is the ℓ_2 normalization function, and $\boldsymbol{W}_D \in \mathbb{R}^{C_v\times C_v}$ and $\boldsymbol{W}_{adj} \in \mathbb{R}^{C_v\times d_{lstm}}$ are learnable weights. Our adjustment mechanism is inspired by [54]: the video features are gated by transformed decoder state and then used to compute an adjustment to the temporal graph. Combined with MGCMP, we decouple the high-order spatial and temporal relation modeling into two modules. This is a major difference with existing methods.

The optimization objective for captioning is to maximize the probabilities of the ground-truth sentence:

$$\mathcal{L}_{cap} = \textstyle\sum_{s=1}^{S} -\log \boldsymbol{p}_s[\hat{w}_s], \tag{16}$$

where $\hat{w}_s$ is the index of the s-th ground-truth word in the vocabulary. $\mathcal{L}_{cap}$ and $\mathcal{L}_{div}$ are combined as the final optimization objective with a tunable hyperparameter γ:

$$\mathcal{L}_{all} = \mathcal{L}_{cap} + \gamma\mathcal{L}_{div}. \tag{17}$$

4. Experiments

4.1. Experimental Settings

Datasets and Evaluation Metrics.

1) **MSVD** [4] contains 1,970 video clips and each clip is 9.6 seconds long on average. Each clip is originally annotated with multilingual sentences and we only keep English sentences, and this results in 40 sentences per clip. As in previous works, the dataset is split into 1,200/100/670 videos for training, validation, and testing, respectively. The vocabulary of the training set has 9,562 words.

2) **MSR-VTT** [46] contains 10,000 video clips collected from 20 manually defined categories and each clip is 14.9 seconds long on average. Each clip is annotated with 20 English sentences. We follow the official setting [46] to split the videos into 6,513/497/2,990 videos for training, validation, and testing, respectively. The vocabulary of the training set has 23,525 words.

Method	Year	Features			MSVD				MSR-VTT			
		Appearance	Motion	Obj./Grid	B	M	R	C	B	M	R	C
TDDF [56]	2017	VGG	C3D	-	45.8	33.3	69.7	73.0	37.3	27.8	59.2	43.8
M^3 [44]	2018	VGG	C3D	-	51.8	32.5	-	-	38.1	26.6	-	-
DenseLSTM [59]	2019	VGG	C3D	-	50.4	32.9	-	72.6	38.1	26.6	-	42.8
MARN [31]	2019	Res-101	3D ResNeXt-101	-	48.6	35.1	71.9	92.2	40.4	28.1	60.7	47.1
GRU-EVE [1]	2019	IRv2	C3D	-	47.9	35.0	71.5	78.1	38.3	28.4	60.7	48.1
POS-CG [41]	2019	IRv2	C3D	-	52.5	34.1	71.3	88.7	42.3	28.1	61.3	48.6
DMRM [51]	2017	INv1	-	Grid	51.1	33.6	-	74.8	-	-	-	-
MAM-RNN [21]	2017	INv1	-	Grid	41.3	32.2	68.8	53.9	-	-	-	-
SeqVLAD [49]	2018	Res-200	-	Grid	51.0	35.2	-	86.0	-	-	-	-
SAM [43]	2018	Res-200	-	Grid	54.0	35.3	-	87.4	-	-	-	-
VRE [33]	2019	Res-152	-	Grid	51.7	34.3	71.9	86.7	43.2	28.0	62.0	48.3
MGSA [6]	2019	IRv2	C3D	Grid	53.4	35.0	-	86.7	42.4	27.6	-	47.5
HTM [16]	2019	Res-152	-	Obj.	54.7	35.2	72.5	91.3	-	-	-	-
OA-BTG [55]	2019	Res-200	-	Obj.	**56.9**	36.2	-	90.6	41.4	28.2	-	46.9
STG [26]	2020	Res-101	I3D	Obj.	52.2	**36.9**	73.9	93.0	40.5	28.3	60.9	47.1
STAT [50]	2020	Res-152	C3D	Obj.	52.0	33.3	-	73.8	39.3	27.1	-	43.9
SAAT [58]	2020	IRv2	C3D	Obj.	46.5	33.5	69.4	81.0	39.9	27.7	61.2	51.0
ORG-TRL [57]	2020	IRv2	3D ResNeXt-101	Obj.	54.3	36.4	73.9	95.2	**43.6**	28.8	**62.1**	50.9
Ours	2021	IRv2	C3D	Grid	53.2	35.4	73.5	90.7	42.1	28.8	61.4	50.1
Ours	2021	IRv2	3D ResNeXt-101	Grid	55.8	**36.9**	**74.5**	**98.5**	41.7	**28.9**	**62.1**	**51.4**

Table 1. Performance comparison on the MSVD and MSR-VTT datasets. Features used by each method are listed: VGG [34], ResNet [13], Inception-v1 [37], InceptionResNet-v2 [36], C3D [38], I3D [3], and 3D ResNeXt [12]. Obj. and Grid indicate the appearance feature is object-level and grid-level, respectively.

3) **VATEX** [45] is a recently released large-scale video captioning dataset and its videos are from a subset of the Kinetics-600 [19] dataset. Each video clip is 10 seconds long and annotated with 10 English sentences and 10 Chinese sentences (not used in this paper). We follow the official setting [45] to split the videos into 25,991/3,000/6,000 videos for training, validation, and testing, respectively. The vocabulary of the training set has 26,759 words.

Following previous works, we evaluate the caption quality using the Microsoft COCO Caption Evaluation codes[3] and report four popular metrics: BLEU-4 [30], METEOR [8], CIDEr [40] and ROUGE-L [23].

Implementation Details. For all three datasets used in our experiments, we preprocess the videos by uniformly sampling 32 frames to extract 2D CNN feature maps, and motion feature maps are extracted from 16 (or 64 depending on the type of 3D CNN) consecutive frames centered on the middle of two sampled frames. The appearance feature is extracted with a 2D CNN, InceptionResNet-v2 [36], and the motion feature is from a 3D CNN, C3D [38] or 3D ResNeXt [12]. The feature maps are bilinearly resized to 8×8 grids. The sentences are lower-cased and clipped to 20 words after punctuations are removed. Each word is embedded to a randomly initialized 512D embedding vector and jointly learned with the whole model. The hidden dimension of the LSTM cells is set to 1024, and a dropout layer with ratio 0.5 is applied after each LSTM cell. For all datasets, the number of regions is set to 8, λ is set to 0.1, and γ is set to 1.0. We train the model using a batch size of 32 and learning rate 0.0001 with Adam optimizer.

4.2. Comparison to State-of-the-art

Compared Methods. We roughly divide the state-of-the-art methods that we compare with into three categories:

- Multimodal Fusion: These methods mainly focus on extracting better features from multiple modalities, organizing and encoding features (M^3 [44], MARN [31], DenseLSTM [59], and GRU-EVE [1]), and feature fusion techniques (TDDF [56] and POS-CG [41]).
- Grid Feature-based: DMRM [51], MAM-RNN [21], SeqVLAD [49], SAM [43], VRE [33], and MGSA [6]. These methods all extract grid features and perform spatial attention for feature aggregation.
- Object Feature-based: HTM [16], OA-BTG [55], STG [26], STAT [50], SAAT [58], and ORG-TRL [57]. These are more recent methods and all use object detectors to extract features from accurately localized object regions, and some of them model spatial/temporal interactions among objects.

Note that besides the listed features, some methods use audio [33] or the category information [6, 31, 58] of MSR-VTT, so a completely fair comparison is not possible.

Performance Comparison. Since most recent methods use the feature combination of InceptionResNet-v2 [36] and C3D (original [38] or with the ResNeXt-101 backbone [12]), we also adopt these features. As shown in Table 1, on the MSVD dataset, when using the IRv2+C3D features, our proposed method can outperform state-of-the-art methods from all three categories: GRU-EVE [1], POS-CG [41], MGSA [6], and SAAT [58], despite that they have used additional features. Among existing methods, ORG-TRL [57] (assisted by external BERT [9] models) currently

[3] https://github.com/tylin/coco-caption

Method	B	M	R	C
Shared Enc [45]	28.9	21.9	47.4	46.8
Shared Enc-Dec [45]	28.7	21.9	47.2	45.6
ORG-TRL [57]	32.1	22.2	48.9	49.7
Ours	**34.2**	**23.5**	**50.3**	**57.6**

Table 2. Performance comparison on the VATEX public testing set. ORG-TRL and Ours both use IRv2+3D ResNeXt-101.

Decomposed step	MGSA	ORG-TRL	Ours
Video Decoding	0.66s	0.66s	0.66s
Object Detection	-	24.62s	-
Feature Extraction	6.59s	4.45s	4.45s
Caption Generation	0.35s	0.25s	0.41s
Total	7.60s	29.98s	5.52s
FPS	47.37	12.01	65.22

Table 3. End-to-end running times for a video of 12 seconds (30FPS, 640x480 pixels). Measured on one RTX 2080 Ti.

achieves the highest performance on MSVD, which proves the effectiveness of object features and relation modeling. Our method can also outperform ORG-TRL on MSVD if the same set of features are used. This validates that our 'region attention' + 'message passing' have similar effects as 'object detection' + 'relation modeling', while being more efficient and can achieve better captioning performance. Similar conclusion can also be drawn on the MSR-VTT dataset, but the performance advantage of our method is not as significant as on MSVD. This is because MSR-VTT has a larger scale in terms of both number of videos and size of vocabulary, which makes it more challenging. Likewise, the most challenging video captioning dataset is currently VATEX. Since it is newly released, there are only a few methods [45, 57] tested on it. To fairly compare with ORG-TRL, we use the IRv2+3D ResNeXt-101 features. As shown in Table 2, our method outperforms ORG-TRL with a significant margin, especially on the CIDEr score.

Speed Comparison. We also compare the end-to-end running time of our proposed method with ORG-TRL and MGSA. For training, video decoding and feature extraction can be preprocessed, but they should be processed in real time during inference if the captioning system is deployed. As shown in Table 3, incorporating object detectors can significantly slow down the whole model's inference speed. Our method is 5.43 times faster than ORG-TRL.

4.3. Ablation Study

4.3.1 Regions vs Objects

We first study the difference of region features and object (detected bounding box) features by replacing the Recurrent Region Attention with an external object detector as in [57]. For the 'Ours w/ Obj.' variant, the number of objects per frame is set to the same as the number of regions in RRA. As shown in Table 4, the performances are fairly close, which indicates that the format of fine-grained spatial information is not critical for video captioning, and this also agrees with [17]. But using RRA can significantly simplify and speed up the whole pipeline, and may empower real end-to-end training, where the feature representation from CNN will be fine-tuned like in image captioning [48].

Method	B	M	R	C
Ours w/ Obj.	41.4	29.0	61.9	51.3
Ours	41.7	28.9	62.1	51.4

Table 4. Performances of our model variants using object and region features on the MSR-VTT dataset.

#	RRA	MGCMP	ATGD	B	M	R	C
0	✗	✗	✗	37.4	27.0	58.8	42.3
1	✓	✗	✗	37.5	26.9	58.9	43.1
2	✓	✓	✗	40.9	28.4	61.2	49.9
3	✓	✓	✓	41.7	28.9	62.1	51.4

Table 5. Results of main ablation studies on MSR-VTT. Due to space limit, some are placed in the Supplementary Materials.

4.3.2 Effects of Each Module

We design 4 model variants to study the effectiveness of each module in our proposed method. The results are shown in Table 5. The #0 variant is the baseline model, where all proposed modules are removed: the CNN feature maps are spatially mean-pooled into vectors and fed to a two-layer LSTM network with temporal attention. The #1 variant adds RRA, but without message passing, the mean-pooled region features are effectively the same as grid features and do not improve captioning performance. With cross-frame message passing and motion guidance, #2 variant has a significant improvement, and this validates the effectiveness of MGCMP and also indicates that modeling high-order spatial relations is important for video captioning. Finally, #3 variant obtains clear improvements over #2, demonstrating the effectiveness of high-order temporal relation modeling in ATGD, and also shows that the combination of these modules can jointly enhance video captioning performance.

We also further investigate the specific designs of each module by several control experiments shown in Table 6. For the MGCMP, if we remove motion features, the performance drops due to lack of motion guidance and multimodal complementary information. If we do not propagate messages across frames, i.e., the previously updated region features are not used to calculate messages, the performance also drops. This indicates that besides spatial relation, the temporal information communication is also critical for video captioning. For RRA, the region diversity constraint $\mathcal{L}_{div}$ ensures that different parts of a frame can be simultaneously captured, as the results demonstrate, has a positive effect on the captioning performance.

4.4. Hyperparameter Analysis

There are two important hyperparameters in our method: the number of regions in RRA and the λ in Eq. (4).

Number of regions. As shown in Fig 4 (left), this can dramatically affect captioning performance, which is reasonable, since a sufficient number of regions are needed to fully cover various details in video frames. But too many

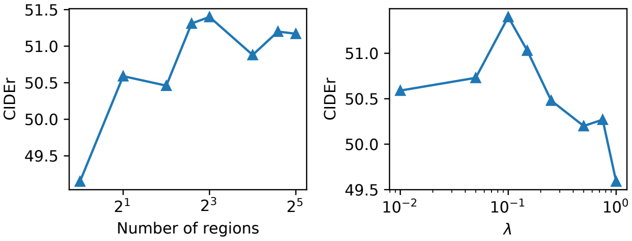


Figure 4. Performance (CIDEr score) comparison of our model variants with different hyperparameter settings.

Variant	B	M	R	C
Complete model	41.7	28.9	62.1	51.4
MGCMP-motion guidance	39.2	27.5	60.2	46.7
MGCMP-cross frame	41.5	28.5	61.4	50.6
RRA-$\mathcal{L}_{div}$	40.8	28.2	61.2	49.7

Table 6. Results of more ablation studies on MSR-VTT.

Method	IoU=0.3	IoU=0.5	IoU=0.7
HVTG	61.37	47.27	23.30
HVTG w/ RRA	57.58	42.31	20.23

Table 7. Performance (Top-1 recall for temporal sentence localization) comparison of the HVTG [7] with and without our RRA on the Charades-STA [10] dataset.

regions can bring redundant information and slightly hurt the performance and slow down the model.

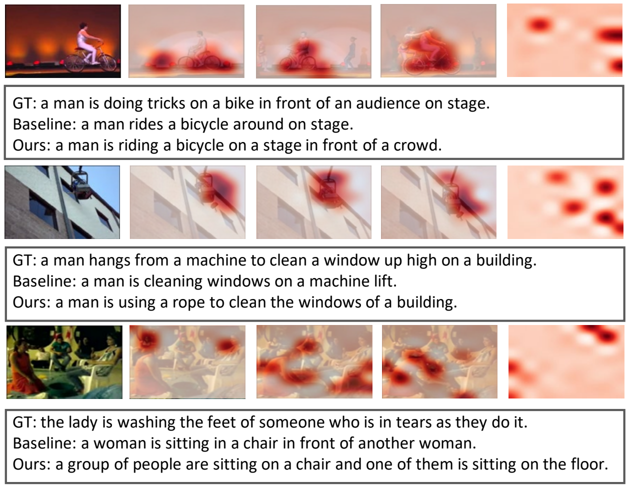
λ **in Eq.** (4)**.** λ in fact controls the 'softness' of the attention distributions, as λ increases from 0 to 1, the attention distributions tend to be closer to one-hot and there is less overlap among different region attention maps. In that case, spatial information is not sufficiently captured. As shown in Fig 4 (right), λ=0.1 achieves a nice balance.

4.5. Application to Temporal Sentence Localization

To preliminarily explore the generalization of region features in video-and-language models, we also apply RRA to a state-of-the-art model, HVTG [7], for the task of temporal sentence localization (TSL) in videos (Please refer to [10] for details about the task). Originally, HVTG extracts object features [2] just like ORG-TRL. If we replace its object detector with RRA, as Table 7 shows, the localization performance drops. We conjecture the reason may be the intrinsic difference between TSL and video captioning. TSL relies on accurate localization of the entities, while video captioning only needs compact and global summarizations of the video contents and is less sensitive to inaccuracies of localized regions. Thus region features are more suitable for tasks that will ultimately summarize the video into a compact representation (e.g., captioning, question answering), and incorporating external object detection has its own advantages (e.g., explainability, spatial/temporal localization).

4.6. Qualitative Results

We show some qualitative examples in Fig. 5. As can be seen, the captions generated by our method contain richer and more accurate content than the baseline model without fine-grained spatial information extracted by RRA. We also show the attended regions of RRA, and note that the regions of key objects can mostly be captured by at least one attention map (Row #1 and Row #2), which proves the RRA has a reasonable ability of accurately extracting fine-grained information even without object detection. Note that region attention is also able to capture multiple areas in one attention map as a result of the diversity constraint (Row #3). There is also a problem with RRA. Since we fixed the number of regions, the captured regions can contain some redundancies (examples shown in the last column). However, it is also unavoidable for object detector-based methods to have redundant object boxes.



GT: a man is doing tricks on a bike in front of an audience on stage.
Baseline: a man rides a bicycle around on stage.
Ours: a man is riding a bicycle on a stage in front of a crowd.

GT: a man hangs from a machine to clean a window up high on a building.
Baseline: a man is cleaning windows on a machine lift.
Ours: a man is using a rope to clean the windows of a building.

GT: the lady is washing the feet of someone who is in tears as they do it.
Baseline: a woman is sitting in a chair in front of another woman.
Ours: a group of people are sitting on a chair and one of them is sitting on the floor.

Figure 5. Examples of generated captions on VATEX from a baseline model and our proposed method. The upsampled region attention maps are shown on top of corresponding images.

5. Conclusion

In this paper, we proposed a new method for video captioning, which does not rely on external object detectors to extract fine-grained spatial information. By recurrently attending to the important regions and a diversity constraint, multiple regions are extracted and their information is communicated across frames via message passing. Furthermore, the ATGD is able to flexibly update and aggregate video features according to decoding state. The three modules jointly model spatial and temporal relations in a decoupled manner. State-of-the-art performances on MSVD, MSR-VTT, and VATEX datasets are achieved. Our findings suggest that region features can be sufficient to generate a compact video representation for captioning, given that the interactions among regions are sufficiently modeled, however, object features still have their advantages in tasks that require accurate localization of visual entities.

Acknowledgement. This work was supported in part by National Natural Science Foundation of China (#62032006). The authors are also grateful for the support from Huawei Technologies Co., Ltd.

References

[1] Nayyer Aafaq, Naveed Akhtar, Wei Liu, Syed Zulqarnain Gilani, and Ajmal Mian. Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning. In *CVPR*, 2019. 2, 6

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, pages 6077–6086, 2018. 8

[3] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *CVPR*, pages 4724–4733, 2017. 6

[4] David L. Chen and William B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, pages 190–200, 2011. 5

[5] Jingwen Chen, Yingwei Pan, Yehao Li, Ting Yao, Hongyang Chao, and Tao Mei. Temporal deformable convolutional encoder-decoder networks for video captioning. In *AAAI*, 2019. 2

[6] Shaoxiang Chen and Yu-Gang Jiang. Motion guided spatial attention for video captioning. In *AAAI*, 2019. 1, 2, 6

[7] Shaoxiang Chen and Yu-Gang Jiang. Hierarchical visual-textual graph for temporal activity localization via language. In *ECCV*, pages 601–618, 2020. 8

[8] Michael J. Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *WMT@ACL*, pages 376–380, 2014. 6

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186, 2019. 6

[10] Jiyang Gao, Chen Sun, Zhenheng Yang, and Ram Nevatia. TALL: temporal activity localization via language query. In *ICCV*, pages 5277–5285, 2017. 8

[11] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *ICML*, pages 1263–1272, 2017. 4

[12] Kensho Hara, Hirokatsu Kataoka, and Yutaka Satoh. Can spatiotemporal 3d cnns retrace the history of 2d cnns and imagenet? In *CVPR*, pages 6546–6555, 2018. 6

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 3, 6

[14] Chiori Hori, Takaaki Hori, Teng-Yok Lee, Ziming Zhang, Bret Harsham, John R. Hershey, Tim K. Marks, and Kazuhiko Sumi. Attention-based multimodal fusion for video description. In *ICCV*, 2017. 2

[15] Jingyi Hou, Xinxiao Wu, Wentian Zhao, Jiebo Luo, and Yunde Jia. Joint syntax representation learning and visual cue translation for video captioning. In *ICCV*, 2019. 2

[16] Yaosi Hu, Zhenzhong Chen, Zheng-Jun Zha, and Feng Wu. Hierarchical global-local temporal modeling for video captioning. In *ACM MM*, 2019. 1, 2, 3, 6

[17] Huaizu Jiang, Ishan Misra, Marcus Rohrbach, Erik G. Learned-Miller, and Xinlei Chen. In defense of grid features for visual question answering. In *CVPR*, pages 10264–10273, 2020. 1, 2, 7

[18] Tao Jin, Siyu Huang, Yingming Li, and Zhongfei Zhang. Low-rank HOCA: efficient high-order cross-modal attention for video captioning. In *EMNLP*, 2019. 2

[19] Will Kay, João Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, Mustafa Suleyman, and Andrew Zisserman. The kinetics human action video dataset. *CoRR*, abs/1705.06950, 2017. 6

[20] Guang Li, Linchao Zhu, Ping Liu, and Yi Yang. Entangled transformer for image captioning. In *ICCV*, pages 8927–8936, 2019. 3

[21] Xuelong Li, Bin Zhao, and Xiaoqiang Lu. MAM-RNN: multi-level attention model based RNN for video captioning. In *IJCAI*, 2017. 2, 6

[22] Yang Li, Lukasz Kaiser, Samy Bengio, and Si Si. Area attention. In *ICML*, pages 3846–3855, 2019. 2

[23] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 6

[24] Sheng Liu, Zhou Ren, and Junsong Yuan. Sibnet: Sibling convolutional encoder for video captioning. In *ACM MM*, 2018. 2

[25] Xiang Long, Chuang Gan, and Gerard de Melo. Video captioning with multi-faceted attention. *TACL*, 6:173–184, 2018. 2

[26] Boxiao Pan, Haoye Cai, De-An Huang, Kuan-Hui Lee, Adrien Gaidon, Ehsan Adeli, and Juan Carlos Niebles. Spatio-temporal graph for video captioning with knowledge distillation. In *CVPR*, 2020. 1, 2, 3, 6

[27] Yingwei Pan, Tao Mei, Ting Yao, Houqiang Li, and Yong Rui. Jointly modeling embedding and translation to bridge video and language. In *CVPR*, pages 4594–4602, 2016. 2

[28] Yingwei Pan, Ting Yao, Houqiang Li, and Tao Mei. Video captioning with transferred semantic attributes. In *CVPR*, pages 984–992, 2017. 2

[29] Yingwei Pan, Ting Yao, Yehao Li, and Tao Mei. X-linear attention networks for image captioning. In *CVPR*, pages 10968–10977, 2020. 3

[30] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002. 6

[31] Wenjie Pei, Jiyuan Zhang, Xiangrong Wang, Lei Ke, Xiaoyong Shen, and Yu-Wing Tai. Memory-attended recurrent network for video captioning. In *CVPR*, 2019. 2, 6

[32] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015. 1

[33] Xiangxi Shi, Jianfei Cai, Shafiq R. Joty, and Jiuxiang Gu. Watch it twice: Video captioning with a refocused video encoder. In *ACM MM*, 2019. 2, 6

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 3, 6

[35] Jingkuan Song, Lianli Gao, Zhao Guo, Wu Liu, Dongxiang Zhang, and Heng Tao Shen. Hierarchical LSTM with ad-

justed temporal attention for video captioning. In *IJCAI*, 2017. 2

[36] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alexander A. Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. In *AAAI*, pages 4278–4284, 2017. 3, 6

[37] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015. 6

[38] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015. 6

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, pages 5998–6008, 2017. 4

[40] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *CVPR*, pages 4566–4575, 2015. 6

[41] Bairui Wang, Lin Ma, Wei Zhang, Wenhao Jiang, Jingwen Wang, and Wei Liu. Controllable video captioning with POS sequence guidance based on gated fusion network. In *ICCV*, 2019. 2, 6

[42] Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. Reconstruction network for video captioning. In *CVPR*, 2018. 2

[43] Huiyun Wang, Youjiang Xu, and Yahong Han. Spotting and aggregating salient regions for video captioning. In *ACM MM*, 2018. 2, 6

[44] Junbo Wang, Wei Wang, Yan Huang, Liang Wang, and Tieniu Tan. M3: multimodal memory modelling for video captioning. In *CVPR*, 2018. 2, 6

[45] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *ICCV*, pages 4580–4590, 2019. 6, 7

[46] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *CVPR*, pages 5288–5296, 2016. 1, 5

[47] Jun Xu, Ting Yao, Yongdong Zhang, and Tao Mei. Learning multimodal attention LSTM networks for video captioning. In *ACM MM*, 2017. 2

[48] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, pages 2048–2057, 2015. 2, 7

[49] Youjiang Xu, Yahong Han, Richang Hong, and Qi Tian. Sequential video VLAD: training the aggregation locally and temporally. *IEEE TIP*, 27(10):4933–4944, 2018. 2, 6

[50] Chenggang Yan, Yunbin Tu, Xingzheng Wang, Yongbing Zhang, Xinhong Hao, Yongdong Zhang, and Qionghai Dai. STAT: spatial-temporal attention mechanism for video captioning. *IEEE TMM*, 22(1):229–241, 2020. 1, 2, 3, 6

[51] Ziwei Yang, Yahong Han, and Zheng Wang. Catching the temporal regions-of-interest for video captioning. In *ACM MM*, 2017. 2, 6

[52] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher J. Pal, Hugo Larochelle, and Aaron C. Courville. Describing videos by exploiting temporal structure. In *ICCV*, pages 4507–4515, 2015. 5

[53] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. Exploring visual relationship for image captioning. In *ECCV*, pages 711–727, 2018. 3

[54] Da Zhang, Xiyang Dai, Xin Wang, Yuan-Fang Wang, and Larry S. Davis. MAN: moment alignment network for natural language moment retrieval via iterative graph adjustment. In *CVPR*, pages 1247–1257. Computer Vision Foundation / IEEE, 2019. 5

[55] Junchao Zhang and Yuxin Peng. Object-aware aggregation with bidirectional temporal graph for video captioning. In *CVPR*, 2019. 1, 2, 3, 6

[56] Xishan Zhang, Ke Gao, Yongdong Zhang, Dongming Zhang, Jintao Li, and Qi Tian. Task-driven dynamic fusion: Reducing ambiguity in video description. In *CVPR*, 2017. 2, 6

[57] Ziqi Zhang, Yaya Shi, Chunfeng Yuan, Bing Li, Peijin Wang, Weiming Hu, and Zheng-Jun Zha. Object relational graph with teacher-recommended learning for video captioning. In *CVPR*, 2020. 1, 2, 3, 4, 6, 7

[58] Qi Zheng, Chaoyue Wang, and Dacheng Tao. Syntax-aware action targeting for video captioning. In *CVPR*, 2020. 1, 2, 3, 6

[59] Yongqing Zhu and Shuqiang Jiang. Attention-based densely connected LSTM for video captioning. In *ACM MM*, 2019. 2, 6