CAM-RNN: Co-Attention Model Based RNN for Video Captioning

Bin Zhao, Xuelong Li, *Fellow, IEEE*, and Xiaoqiang Lu, *Senior Member, IEEE*

***Abstract*— Video captioning is a technique that bridges vision and language together, for which both visual information and text information are quite important. Typical approaches are based on the recurrent neural network (RNN), where the video caption is generated word by word, and the current word is predicted based on the visual content and previously generated words. However, in the prediction of the current word, there is much uncorrelated visual content, and some of the previously generated words provide little information, which may cause interference in generating a correct caption. Based on this point, we attempt to exploit the visual and text features that are most correlated with the caption. In this paper, a co-attention model based recurrent neural network (CAM-RNN) is proposed, where the CAM is utilized to encode the visual and text features, and the RNN works as the decoder to generate the video caption. Specifically, the CAM is composed of a visual attention module, a text attention module, and a balancing gate. During the generation procedure, the visual attention module is able to adaptively attend to the salient regions in each frame and the frames most correlated with the caption. The text attention module can automatically focus on the most relevant previously generated words or phrases. Moreover, between the two attention modules, a balancing gate is designed to regulate the influence of visual features and text features when generating the caption. In practice, the extensive experiments are conducted on four popular datasets, including MSVD, Charades, MSR-VTT, and MPII-MD, which have demonstrated the effectiveness of the proposed approach.**

***Index Terms*— Attention model, video captioning, recurrent neural network.**

I. Introduction

VIDEO captioning is a task that integrates computer vision and natural language processing [1]. The goal of video captioning is to automatically generate a sentence to describe the activities in the video. Recently, due to the explosive growth of video data, this task has drawn increasing attention, since it provides an efficient way for viewer to browse the video content [2]. What is more, video captioning can also benefit many other applications, such as video summarization [3], [4], scene understanding [5], [6], human-robot interaction [7] and so on.

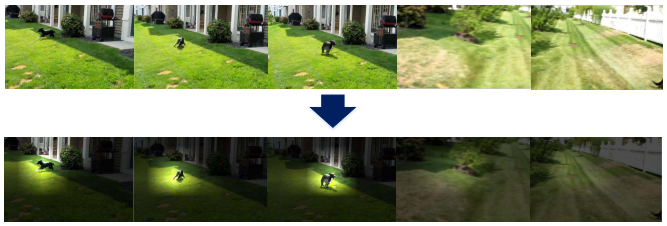


Fig. 1. Our co-attention model adaptively focuses on the most correlated features when generating the caption. Visually, it can not only focus on the most correlated frames but also attend to the most salient regions in each frame. Textually, it selectively focuses on different previous phrases to generate the current word.

Earlier works generated the video caption based mainly on a sentence template [8]–[10]. Concretely, individual classifiers are pretrained to identify the semantic elements (*i.e.*, subject, verb, object, and scene) in the video, and these elements are assigned to the sentence template to generate a fixed-form caption. However, the template-based approaches are inflexible and insufficient to model the rich information of natural language [1], [11].

Recently, video captioning has been tackled as a sequence-to-sequence task, which transfers the data from a frame sequence to a word sequence. Inspired by the great success of the *recurrent neural network* (RNN) in sequence modeling, the RNN has been introduced into video captioning and has achieved inspiring results [12]–[14]. Most of the existing RNN-based approaches follow the encoder-decoder diagram. First, the visual features are encoded into a fixed-size vector and taken as the input to the RNN. Then, the RNN is utilized as the decoder to generate the sentence word by word. Specifically, each word is generated based on the visual feature and the text feature of previously generated word. In this paper, we attempt to improve the performance of RNN-based approaches by exploiting more powerful visual and text features for the video captioning task.

A. Motivation and Overview

On the one hand, the visual feature input to the RNN is quite important for generating correct video captions [15].

Manuscript received September 30, 2017; revised May 1, 2018 and April 17, 2019; accepted May 3, 2019. Date of publication May 20, 2019; date of current version August 28, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61772510 and Grant 61702498, in part by the Key Research Program of Frontier Sciences, CAS, under Grant QYZDY-SSW-JSC044, in part by the Young Top-Notch Talent Program of the Chinese Academy of Sciences under Grant QYZDB-SSW-JSC015, in part by the National Key R&D Program of China under Grant 2017YFB0502900, and in part by the CAS "Light of West China" Program under Grant XAB2017B15. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gang Hua. *(Corresponding author: Xiaoqiang Lu.)*

B. Zhao and X. Li are with the School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an 710072, China (e-mail: binzhao111@gmail.com; xuelong_li@nwpu.edu.cn).

X. Lu is with the Key Laboratory of Spectral Imaging Technology CAS, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China (e-mail: luxq666666@gmail.com).

Digital Object Identifier 10.1109/TIP.2019.2916757

The early approach [16] simply input the average-pooled frame features to the RNN. Recently, researchers have realized that there is considerable redundant and irrelevant content in the video, which may cause interference in generating the correct caption [17], [18]. Based on this point, an attention model is employed to selectively focus on only a few of the video frames that are relevant to the target caption [19]–[21]. However, there is still some irrelevant background information, especially when the described object is small [22]. To address this problem, a more powerful attention model is required to automatically attend to the most salient regions in each frame. Ideally, as depicted in Fig. 1, to generate the caption that *'a dog is running around'*, we seek a visual attention module that can automatically focus on not only the frames containing dogs but also their dog regions.

On the other hand, the text features of previously generated words are indispensable for the determination of the subsequent words [9], [23]. For most of the existing approaches, when generating the current word, the utilized text feature is simply the feature vector of the single previous word. It is insufficient in certain situations even though the historical information of all previous words has been partially recorded by the RNN [24]. As depicted in Fig. 1, when generating the word *'running'*, only taking the previous word *'is'* as input is not enough to guide the correct generation. However, if we input *'dog is'* into the RNN, the possibility of generating the correct word is much improved since the frequency of *'running'* following *'dog is'* is much higher than that just following a single word, *'is'*. Therefore, we prefer to design a phrase-level text attention module to encode text features. Different from existing word-level modules [25], [26], phrase-level text attention module can automatically attend to the most correlated phrases rather than single words to predict the following caption words.

In this paper, we propose a *co-attention model based RNN* (CAM-RNN) for the video captioning task. Actually, it is developed based on the multi-level visual attention module of our MAM-RNN presented in the conference paper [27]. In the developed work, a much more comprehensive approach CAM-RNN is proposed to address the aforementioned two problems, and extensive experiments are conducted to provide more insights for the video captioning task.

Specifically, the CAM-RNN can be divided into four parts, *i.e.*, visual attention module (adopted from [27]), text attention module, balancing gate and the caption generator based on *long short-term memory* (LSTM). The visual attention module is designed to encode the visual features. It has two layers and is able to adaptively focus on the most correlated visual features at both the frame level and region level. When generating the caption, the first layer learns to focus on the most salient regions in each frame, and the second layer tries to attend to the most correlated frames. Then, the video feature is encoded into a fixed-size vector. Furthermore, during generation, the text attention module operates on different phrases of previously generated words and selectively focuses on the most correlated phrase to generate the current word. Finally, both the encoded visual feature and text feature, together with the historical information recorded by the LSTM, are integrated to generate the caption word by word. Furthermore, considering that nonvisual words (*e.g.*, *a, the, and is*) can be easily predicted with natural language information in the text feature and that the visual feature provides little guidance, a balancing gate is developed to regulate the influence of the visual feature and the text feature in the process of caption generation.

B. Contributions

The contributions of this work are threefold and can be summarized as follows:

- A two-layer visual attention module is developed, which can provide more powerful visual features by adaptively focusing on the most correlated frames and the salient regions in each frame. Meanwhile, the structure information in the frame and the smoothness among frames are preserved.
- A phrase-level text attention module is designed. During caption generation, it can selectively attend to the most correlated phrases formed by previously generated words and exploit more accurate text features.
- A balancing gate is introduced to regulate the influence of visual features in the caption generation process. It is helpful for the generation of nonvisual words.

C. Organization

The rest of this paper is organized as follows. The existing approaches related to our work are reviewed in Section II. The detailed process of the proposed approach, the CAM-RNN, is described in Section III. The experimental results and analyses are presented in Section IV. Finally, the conclusion is drawn in Section V.

II. Related Works

In this section, we briefly introduce three closely related tasks, *i.e.*, machine translation, image captioning and video captioning. Actually, much of the processes of video captioning benefit from the first two tasks. Therefore, the introduction of machine translation and image captioning is helpful to understand the development of video captioning.

A. Machine Translation

Machine translation is a task that translates sentences from the source language to the target language [28]–[30]. Similar to video captioning, it is also a sequence-to-sequence task, where the first overwhelming success of sequence modeling took place. Most of the machine translation approaches follow the encoder-decoder paradigm by taking advantage of RNNs [26], [31]–[33]. Specifically, the source sentence is encoded into a fixed-size feature vector, and then, it is taken as the input of the decoder RNN to generate the sentence in the target language. Furthermore, several improvements have been proposed based on the encoder-decoder paradigm. Lin *et al*. [33] and Li *et al*. [32] design a hierarchical RNN to model the high-level semantic information in sentences. In addition, an attention model is developed in [26], so that the decoder can automatically focus on the most important source words when translating into the target language. All these provide inspiration to the development of video captioning.

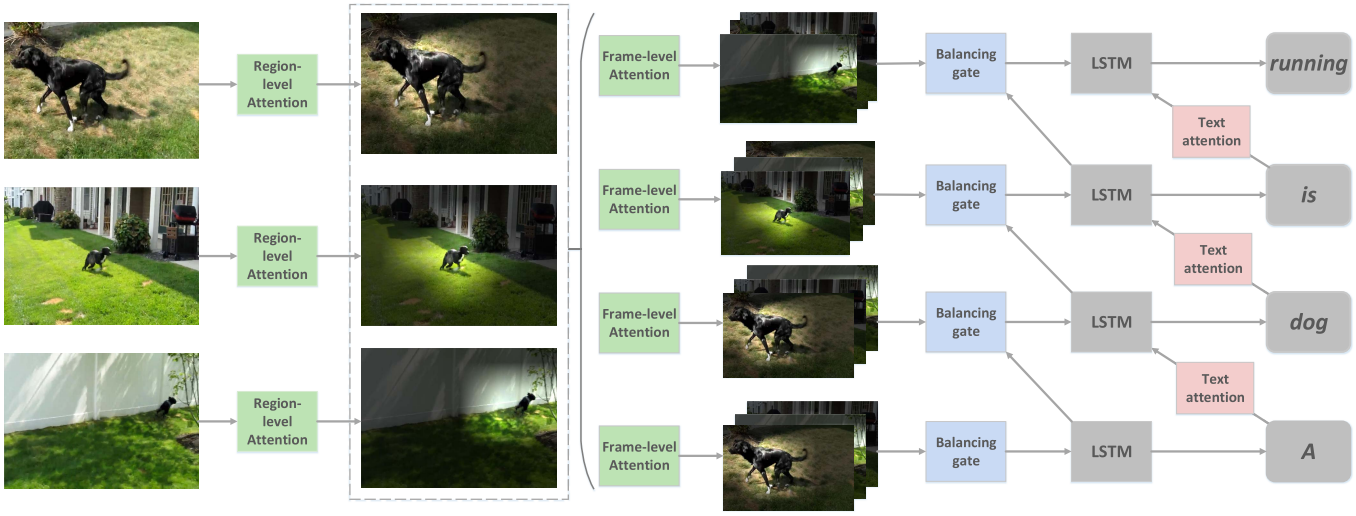


Fig. 2. The architecture of the CAM-RNN (co-attention model based RNN). Specifically, the CAM is composed of three parts as follows. 1) The visual attention module contains the region-level attention layer and frame-level attention layer, which can adaptively focus on the most correlated visual features. 2) The text attention layer operates on previously generated words, which can automatically attend to the most correlated text features. 3) The balancing gate is designed to adaptively regulate the influence of the visual features. Finally, with the encoded visual feature and text feature, LSTM is employed as the caption generator.

B. Image Captioning

Image captioning is the first attempt at visual-to-text translation [34]–[37]. It can be regarded as a simplified case of video captioning when the processed video contains only one frame. Benefiting from the rapid development of deep learning, image captioning has achieved inspiring results. The current approaches usually follow the CNN-RNN architecture, where a *convolutional neural network* (CNN) [38]–[40] is utilized to extract the visual feature of the image, and the RNN is good at modeling the sequence information of the caption. Recently, CNN-RNN-based approaches extended with attention models have achieved state-of-the-arts [41]–[43]. In [42], [44], the visual attention model is designed to make the decoder automatically attend to the most salient regions in the image. In [43], a more powerful semantic attention model is proposed to integrate the semantic concepts and visual attributes of the image.

C. Video Captioning

Earlier video captioning approaches were developed using a sentence template [8]–[10]. They operate in two stages. In the first stage, the classifiers are learned to capture the semantic information in the video, including objects, scenes, actions and so on. Then, the second stage generates the caption by combining the semantic information according to the sentence template.

Recently, inspired by the great success of RNNs in machine translation and image captioning, RNNs have also been introduced in video captioning [11], [45], [46]. Venugopalan *et al.* first proposed the RNN-based video captioning approach [16]. It takes the LSTM [47] as the caption generator and generates the caption step by step. However, the visual feature of the video is simply encoded by mean pooling the frame features. To exploit the temporal information among frames, [1] adopts another LSTM to encode the frame feature sequence. More recently, to exploit the most correlated visual features for video captioning, [19], [20], [48], [49] develop several visual attention modules to selectively attend to a subset of video frames. In addition, [45], [50] try to extract more powerful text information for the video captioning task. Text attention has also been considered in related tasks, such as visual question-answer [51], [52]. The performance improvements verify the necessity to consider the correlated text information.

III. Co-Attention Model Based RNN

The architecture of the proposed approach, the CAM-RNN, is depicted in Fig. 2. It is an RNN-based video captioning approach. Specifically, LSTM is employed as the caption generator. For clarity, we first provide a brief introduction about the RNN, especially LSTM, and then successively present the process of video caption generation, visual feature extraction, text feature extraction and, finally, the balancing gate.

A. Recurrent Neural Network

The *recurrent neural network* (RNN) is extended from the feedforward neural network by adding feedback connections so that it can model sequence information. Specifically, given $(x_1, x_2, \ldots, x_n)$ as the input sequence, the standard RNN can output another sequence $(y_1, y_2, \ldots, y_n)$ iteratively by the following equations:

$$h_t = \phi\left(W_h x_t + U_h h_{t-1} + b_h\right), \tag{1}$$

$$y_t = \phi\left(U_y h_t + b_y\right), \tag{2}$$

where $\phi(\cdot)$ denotes the activation functions, h is the hidden state, and W, U, b are the weights and bias to be learned.

However, due to the gradient vanishing problem, the standard RNN can only contend with short-term temporal information [53]. To address this problem, *long short-term memory* (LSTM) [47] is proposed, which is a variant of the standard RNN. Compared to the standard RNN, LSTM is

equipped with an extra memory cell that can selectively record the previous inputs. There are several variants of LSTM. The one proposed in [54] is employed in our approach. Specifically, it computes the hidden h_t and the memory cell c_t by the following equations:

$$i_t = \sigma\left(W_i x_t + U_i h_{t-1} + b_i\right), \tag{3}$$
$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right), \tag{4}$$
$$o_t = \sigma\left(W_o x_t + U_o h_{t-1} + b_o\right), \tag{5}$$
$$g_t = \phi\left(W_g x_t + U_g h_{t-1} + b_g\right), \tag{6}$$
$$c_t = f_t \odot c_{t-1} + i_t \odot g_t, \tag{7}$$
$$h_t = o_t \odot \phi\left(c_t\right), \tag{8}$$

where σ is the sigmoid function, all Ws, Us and bs are the weights and the bias to be learned, and i_t, f_t and o_t are three gates. Specifically, the input gate i_t and the forget gate f_t decide whether to record the current input x_t or forget the previous memory c_{t-1}. The output gate o_t decides how much information in the memory cell c_t is transferred to the hidden layer h_t. Overall, all of these three gates enable the LSTM to model the long-term sequence information. Therefore, LSTM plays an important role in the development of machine translation, image captioning and video captioning.

B. Video Caption Generation With RNN

In our work, LSTM is adopted as the caption generator. From Eqns. (3)–(8), we can see that the hidden state h_t in LSTM is computed jointly by the current input data x_t and previous hidden state h_{t-1}. For simplicity, the calculation in LSTM is denoted as follows:

$$h_t = LSTM\left(x_t, h_{t-1}\right). \tag{9}$$

Following existing approaches, the dimensionality of the hidden state h_t is fixed as 256. The input data x_t is combined using the visual feature and text feature, i.e.,

$$x_t = \left[W_V V_t, e_{t-1}\right], \tag{10}$$

where $[\cdot,\cdot]$ is the concatenation operation. V_t is the visual feature input to the LSTM in the t-th step. Specifically, the visual features are different at each step of the LSTM. The details are presented in Section III-C. e_{t-1} denotes the feature of the previously generated word. Following existing approaches [50], the embedded one-hot feature is extracted for each word. W_V is the matrix transforming the visual feature into the same space as the word feature.

After the hidden state h_t is computed, the prediction probability of each word is calculated by the following:

$$p_t = softmax\left(\tanh\left(W_p\left[\gamma_t V_t, E_t, h_t\right] + b_p\right)\right), \tag{11}$$

where W_p and b_p are the training parameters. p_t has the same dimensionality as the size of the vocabulary, which denotes the probability of each word to be selected as the t-th word of the caption. E_t is the attended text feature of phrases formed by previously generated words. The details are presented in Section III-D. It can be observed from Eqn. (11) that p_t is jointly determined by the current visual feature V_t, text feature E_t and the hidden state h_t. Although most of the current visual information and historical text information have already been encoded in h_t, we still prefer to emphasize the visual features and text features when determining the current word, similar to most existing approaches [15], [27]. Particularly, it is especially important for the text feature since h_t encodes the sequential word-level information, as depicted in Eqns. (9) and (10), and E_t can provide more comprehensive phrase-level information. Thus, they are complementary to one another. In addition, γ_t is the balancing gate that adaptively regulates the influence of the visual feature V_t. It is calculated in Section III-E.

Naturally, in the testing process, each word in the caption is generated by the maximum value in the vector p_t, as formulated in Eqn. (11). In the training process, the log-likelihood loss is adopted for optimization,

$$\Theta = \arg\max_{\Theta} \sum_{t=1}^{T} \log \Pr\left(\psi_t \left| \psi_{t-1}, V_t, E_t; \Theta\right.\right), \tag{12}$$

where ψ_t denotes the t-th word of the reference caption, and Θ stands for the training parameters in the CAM-RNN. T is the fixed steps of LSTM. It should be noted that the reference caption is padded by zeros if it is shorter than T. Overall, Eqn. (12) illustrates that the proposed approach is optimized by maximizing the probability of the reference caption.

C. Attention Extended Visual Feature Extraction

The task of the visual attention module is to extract the visual feature. As previously mentioned, the visual attention module is composed of two layers, including region-level attention and frame-level attention. They are introduced in the following subsections, respectively.

1) Region-Level Attention: The target of region-level attention is to extract the frame feature by adaptively focusing on the salient regions in each frame.

First, inspired by [42], [48], the convolutional layer of the CNN is employed to extract the region features for each frame. In this case, a set of feature vectors are obtained for a certain frame i, denoted as $\left\{r_{i,1}, r_{i,2}, \ldots, r_{i,m}\right\}$. m stands for the number of regions.

Then, the frame feature ξ is calculated by the weighted sum of region features,

$$\xi_i = \sum_{j=1}^{m} \alpha_{i,j} r_{i,j}, \tag{13}$$

where $\alpha_{i,j}$ is the attention weight of region j in frame i. It is computed by the following:

$$q_{i,j} = w_r \tanh\left(W_r r_{i,j} + U_r \alpha_{i-1,j} + b_r\right), \tag{14}$$
$$\alpha_{i,j} = \exp\left\{q_{i,j}\right\} / \sum_{j=1}^{m} \exp\left\{q_{i,j}\right\}, \tag{15}$$

where W_r, U_r, w_r and b_r are the parameters to be learned. With the help of α, the frame features are generated by selectively attending to the salient regions in the frame. In this case, the interference caused by irrelevant and meaningless region features is reduced. Specifically, we can see from Eqn. (14) that the attention weight of a certain region in frame

i is jointly determined by its region feature and the attention weight of the corresponding region in the previous frame. It is reasonable since consecutive frames are quite similar to one another. Correspondingly, the attention weights ought to vary smoothly according to time. Note that to avoid bias in the attention weight calculation, $\{\alpha_{0,j}\}_{j=1}^{m}$ is initialized with $1/m$, which means that each region is equally emphasized.

2) Frame-Level Attention: Frame-level attention encodes the visual feature of the video by adaptively attending to a subset of frame features that are most correlated to the video caption.

At each step, the visual feature input to the summary generator is calculated by the weighted sum of frame features, which is formulated as

$$V_t = \sum_{i=1}^{n} \beta_{i,t} \xi_i, \tag{16}$$

where $\beta_{i,t}$ denotes the attention weight of frame i at the t-th time. n is the number of frame features. Ideally, for a certain frame i, if it is highly correlated with the t-th word of the caption, it should be emphasized at the t-th time. In this case, the value of $\beta_{i,t}$ should be relatively high.

In practice, to exploit the temporal information, the previously encoded features should be considered in the determination of $\beta_{i,t}$, including both the visual feature and the word feature. Fortunately, they are already captured by the previous hidden state h_{t-1} of the caption generator. Following existing approaches [19], the attention weight $\beta_{i,t}$ represents the relevance of ξ_i to the video caption. Based on this assumption, the relevance score is obtained by

$$l_{i,t} = w_f \tanh\left(W_f \xi_i + U_f h_{t-1} + b_f\right), \tag{17}$$

where W_f, U_f, b_f are parameters to be learned. Then, $\beta_{i,t}$ is obtained by normalizing $l_{i,t}$, i.e.,

$$\beta_{i,t} = \exp\{l_{i,t}\} / \sum_{i=1}^{n} \exp\{l_{i,t}\}. \tag{18}$$

With the help of $\beta_{i,t}$, the visual feature at time t, *i.e.*, V_t, can be extracted by automatically attending to the most correlated frames. It can further improve the performance in video caption generation.

D. Attention Extended Text Feature Extraction

The target of the text attention module is to calculate the text feature E_t at each time step. Specifically, it operates on phrases formed by previously generated words, so that our approach can attend to the most correlated text information when generating each word.

Concretely, given the previously generated words $\{e_1, e_2, \ldots, e_{t-1}\}$, the phrase features are computed by

$$\rho_{s,t} = \tanh\left(W_s e_{t-s:t-1}\right), \quad s \in \{1, 2, ..., S\}, \tag{19}$$

where W_s is the training weight, $\rho_{s,t}$ is the feature of the s-gram phrase at time t, and S means that previously generated S words are considered in the text attention. In this paper, S is set as 3 (the influence of S to the performance is discussed in the experimental part). This means that at each step of the generation procedure, three phrases are taken into consideration, *i.e.*, unigram, bigram and trigram, which has been proven to be effective in most situations in relevant tasks [24]. Note that, at the first two steps, the previously generated word set is zero-padded by adding two virtual words e_0 and e_{-1}.

In practice, the text feature E_t is calculated by the weighted-sum of the phrase features, *i.e.*,

$$E_t = \sum_{s} \eta_{s,t} \rho_{s,t}, \tag{20}$$

where $\eta_{s,t}$ is the attention weight of the s-gram phrase at time t. It is determined by both the feature vector of the corresponding phrase and the historical information recorded in the hidden state of the caption generator. Specifically, it is calculated by the following two functions:

$$z_{s,t} = w_z \tanh\left(W_z \rho_{s,t} + U_z h_{t-1} + b_z\right), \tag{21}$$

$$\eta_{s,t} = \exp\{z_{s,t}\} / \sum_{s=1}^{S} \exp\{z_{s,t}\}. \tag{22}$$

where W_z, U_z, and b_z are the training parameters.

E. Balancing Gate

The balancing gate is designed to balance the influence of the visual feature and the text feature. Essentially, it is utilized to reduce the impact of the visual feature when generating nonvisual words (*e.g.*, a and the) since these words can be easily predicted with natural language information encoded in the text feature and the hidden state of LSTM, while the visual feature provides little information to them.

In this paper, considering that the hidden state of the decoder LSTM records most of the information of the caption generator, the balancing gate is calculated by

$$\gamma_t = sigmoid\left(W_\gamma h_t\right), \tag{23}$$

where the sigmoid function is utilized to project γ_t into the range of [0,1], and W_γ is the training parameter. It can be observed form Eqn. (11) in Section III-B that γ_t is the coefficient of the visual feature V_t. When generating the nonvisual words, a smaller γ_t is expected, *i.e.*, the impact of the visual feature is reduced, whereas for visual words, the value of γ_t is relatively high.

IV. Experiments

In this section, we evaluate the proposed approach on four popular video captioning datasets, *i.e.*, MSVD [8], Charades [55] MSR-VTT [56] and MPII-MD [57]. First, the proposed approach is compared with several baselines to verify the effectiveness of the visual attention module, text attention module and balancing gate, respectively. Then, the proposed approach is compared with several state-of-the-art approaches on the four datasets.

A. Experimental Details

1) Datasets: The MSVD dataset [8] is comprised of 1970 video clips crawled from YouTube. In general, each clip displays a certain activity, including riding, cooking, and running. They are annotated with multiple captions in different languages. In our work, only the captions in English are considered, including 80839 sentences in total and approximately 41 sentences for each clip. Following existing protocols, the MSVD dataset is split into three parts, i.e., 1200 clips for training, 100 clips for validation, and the remaining 670 clips for testing.

The Charades dataset [55] is a challenging dataset. It contains 9848 videos with a mean duration of 30 seconds. The videos in the Charades dataset typically record the indoor activities of human beings, such as reading, cooking, and eating. Statistically, the dataset is annotated with 27847 captions, and each caption has 23 words on average. In the experiment, 7585 videos are adopted for training, 400 videos for validation and 1863 videos for testing.

The MSR-VTT dataset [56] is similar to MSVD but much larger. It contains 10000 clips from 7180 web videos. These videos are quite diverse and include 20 categories, such as music, people, gaming, and sports. Specifically, each clip is annotated with approximately 20 captions, and 29316 different words in total are utilized in the dataset. Following the setting of the original paper [56], MSR-VTT is split into 6513 clips for training, 497 for validation and 2990 for testing.

The MPII-MD dataset [57] is quite different from the above three datasets. It is composed of more than 68000 video clips that are segmented from 94 movies. Each video clip is accompanied by 1 sentence sourced from movie scripts or an audio description. On average, each sentence has approximately 7 words. Due to the high diversity of the visual and text content, as well as having only a single reference caption provided for each clip, this dataset is much more challenging than the previous three. According to [57], the video clips in the MPII-MD dataset are separated into three subsets: 4015 clips (from 8 movies) for testing, 1834 clips (from 3 movies) for validation, and the rest of the clips for training.

2) Feature Extraction: Following previous works [19], [58], the convolutional layer of the CNN is employed to extract the region features. In our work, to analyze the effects of features in terms of the performance, three popular CNNs are adopted, including VggNet 16 [59], GoogLeNet [40] and C3D [60], where the first two are pretrained on ImageNet, and the last is pretrained on Sports-1M. Correspondingly, the pool5 layer of VGGnet 16, the inception5b layer of GoogLeNet and the conv5b layer of C3D are used for region feature extraction. Furthermore, it should be noted that just 160 frames are considered in the video captioning process. For longer videos, the frames are sampled uniformly to meet this constraint. For shorter videos, the frame sequence is padded with zeros.

The word vocabulary for each dataset is constructed by tokenizing the captions, converting words into lower case and removing rare words. After that, the size of the vocabulary is 2743 for MSVD, 1525 for Charades, 3871 for MSR-VTT and 3370 for MPII-MD. Finally, the word feature is extracted by embedding the one-hot feature into a 300-dimensional GloVe vector [61], which is widely used in text analysis tasks.

3) Evaluation Metrics: Similar to machine translation tasks, the quality of the predicted caption is measured by the consistency with the reference caption. In our work, various metrics are adopted for evaluation, including BLEU [62], ROUGE–L [63], METEOR [64] and CIDEr [65]. More specifically, BLEU has four versions, *i.e.*, BLEU 1–4. In general, for all the adopted metrics, the higher scores indicate the better quality of the generated caption.

4) Training Details: The proposed CAM-RNN is composed of several components, *i.e.*, visual attention module (frame-level attention and region-level attention), text attention module, balancing gate, and a plain caption generator with RNN (*i.e.*, LSTM). In this paper, we develop several baselines of our approach.

1) None-attention with RNN (None-RNN). The frame features are extracted by the final layer of the CNN and average pooled to a fixed-size vector and finally input into the LSTM to generate the video caption.

2) Frame-level attention with RNN (Frame-RNN). The frame features are extracted by the final layer of the CNN, weight-summed by the frame-level attention module and finally input into the LSTM to generate the video caption.

3) Visual attention with RNN (Visual-RNN). The multilayered visual attention module is equipped to the decoder LSTM, so that the visual feature can automatically attend to the most correlated features both in the region level and the frame level.

4) Visual and text attention with RNN (VT-RNN). The visual attention module and text attention module are both equipped.

5) Co-attention model with RNN (CAM-RNN). The completed version of our approach such that the visual attention module, text attention module and balancing gate are all equipped.

In this paper, to provide a fair comparison and better analyze the effectiveness of each part in our approach, the above five baselines are carried out with the same experimental settings. The experimental results are discussed in the following subsection.

B. Results of Baselines

In this part, we try to verify the effectiveness of the three parts, visual attention module, text attention module and balancing gate, by comparing the baselines of the proposed approach. The statistical results are displayed in Tables I and II. Note that for a fair judgment, the frame feature and region feature for all baselines are extracted by VggNet.

1) The Analysis on the Visual Attention Module: To verify the effectiveness of our visual attention module, we compare three baselines, None-RNN, Frame-RNN, and Visual-RNN, where Frame-RNN is constructed by adding the frame-level attention layer on None-RNN, and Visual-RNN is built on Frame-RNN by adding the region-level attention layer. From the first three rows in Tables I and II, we can clearly see

TABLE I
THE RESULTS OF BASELINES ON THE MSVD DATASET

Metrics	CIDEr	BLEU–1	BLEU–2	BLEU–3	BLEU–4	ROUGE–L	METEOR
None-RNN	0.481	0.732	0.568	0.479	0.362	0.613	0.283
Frame-RNN	0.485	0.750	0.591	0.501	0.376	0.645	0.298
Visual-RNN	0.535	0.803	0.655	0.544	0.411	0.690	0.319
VT-RNN	0.532	0.791	0.671	0.555	0.419	0.690	0.325
CAM-RNN	0.543	0.803	0.676	0.560	0.424	0.694	0.334

TABLE II
THE RESULTS OF BASELINES ON THE CHARADES DATASET

Metrics	CIDEr	BLEU–1	BLEU–2	BLEU–3	BLEU–4	ROUGE-L	METEOR
None-RNN	0.168	0.427	0.288	0.165	0.124	0.312	0.143
Frame-RNN	0.175	0.441	0.296	0.164	0.118	0.313	0.149
Visual-RNN	0.178	0.519	0.303	0.185	0.132	0.331	0.177
VT-RNN	0.182	0.530	0.311	0.183	0.133	0.338	0.186
CAM-RNN	0.186	0.534	0.318	0.191	0.133	0.347	0.189

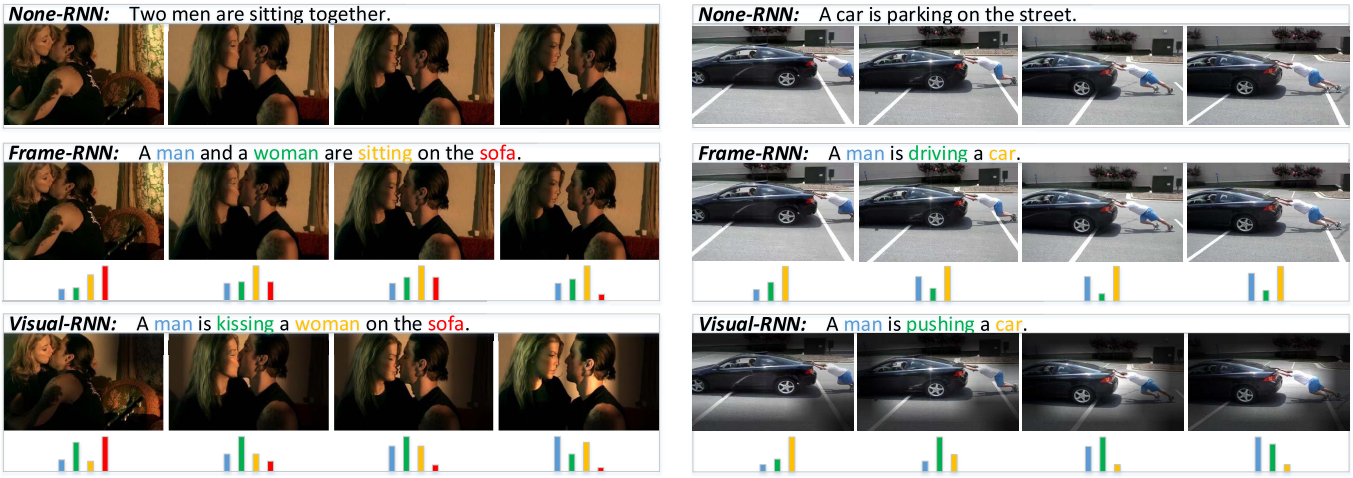


Fig. 3. Example results of three baselines, *i.e.*, None-RNN, Frame-RNN and Visual-RNN. They are from the MSVD dataset. The sentences above the frames denote the generated captions. The histograms below the frames represent the frame-level attention when generating each word (distinguished by color). Here, we only show the attention weights of some key words since the visual feature is quite important to them. The brightness distribution in the third row frames reflects the region-level attention, where the brighter regions are more emphasized. The best view is in color.

that the performance improves by applying the frame-level attention module and region-level attention module, which quantitatively verifies the effectiveness of each layer of the visual attention module.

In Fig. 3, we show several examples of the results of the three baselines. We can clearly see that the Visual-RNN has shown its superiority in extracting the visual feature. Specifically, the region-level attention layer can adaptively attend to the salient regions, and the frame-level attention layer selectively focuses on the correlated frames. In this case, our visual attention module can reduce the interference caused by the uncorrelated visual information while generating the caption. The effectiveness of our visual attention module is also reflected in the generated captions in Fig. 3.

2) The Analysis on the Text Attention Module: To analyze the effectiveness of the text attention module, we compare the results of Visual-RNN and VT-RNN since the only difference between them is whether the text attention module is contained or not. From the third and fourth rows in Tables I and II, it can be seen that VT-RNN performs better than Visual-RNN. Thus, we can draw the conclusion that by exploiting the most correlated text information, the text attention module is helpful to improve the quality of video caption.

In Fig. 4, we show some examples of the generated captions. By comparing the results of None-RNN, Visual-RNN and VT-RNN, it can be seen that the captions generated by VT-RNN have less grammatical errors than those generated by the previous two baselines. This phenomenon indicates that the text attention module has superiority in exploiting more powerful text features to reduce the grammatical errors in sentences.

Furthermore, in Fig. 6, we have plotted the performance variance of our approach on the validation dataset according to parameter S (i.e., how many previously generated words are

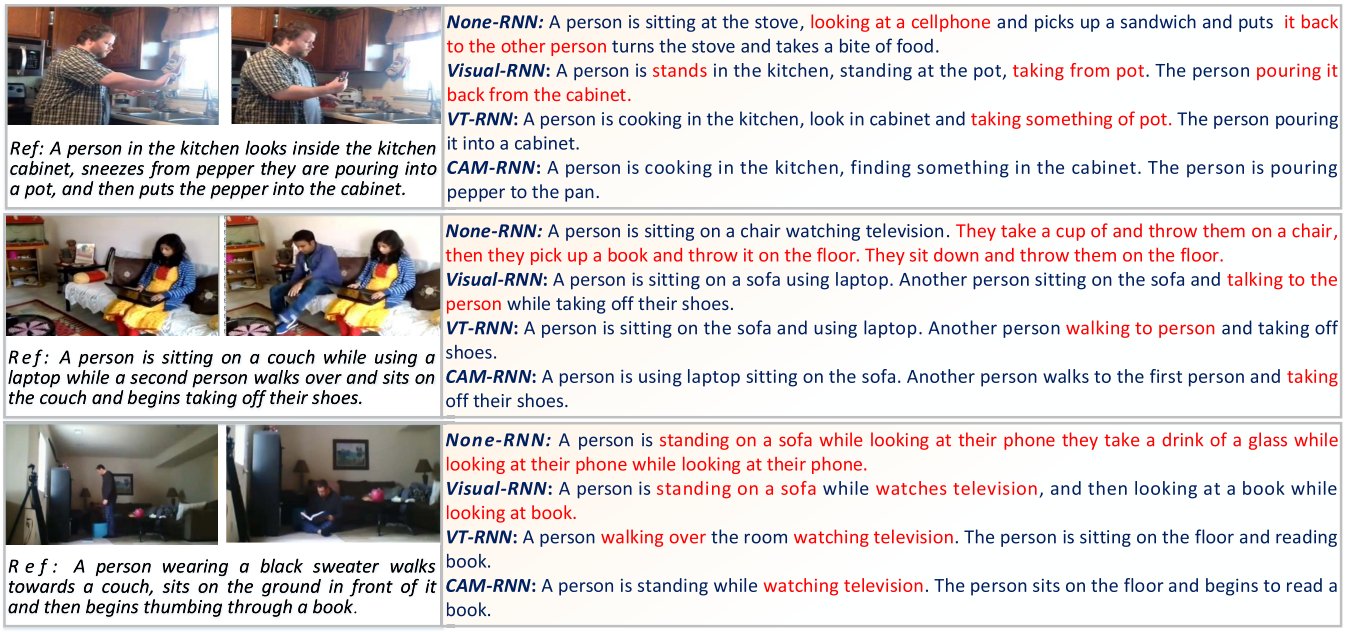


Fig. 4. Example results of different baselines of our approach. These videos are from the Charades dataset. The frames are extracted from each video, and the black sentences below the frames are the reference captions annotated by humans. The blue sentences on the right are the captions generated by different approaches. The obvious errors in generated captions are highlighted in red color. Note that there is no punctuation in raw generated captions. Here, the punctuation is added by humans to make the captions more readable. However, the punctuation is not considered in the evaluation. The best view is in color.

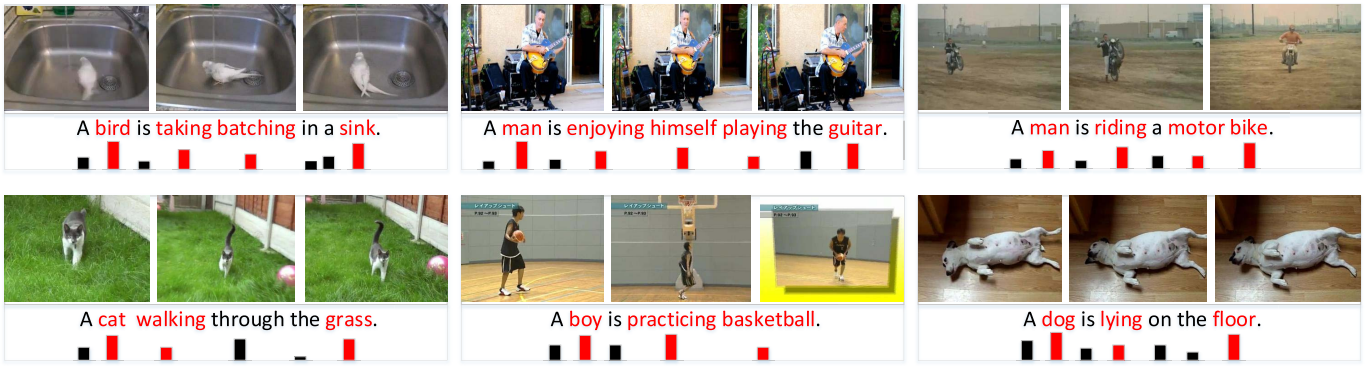
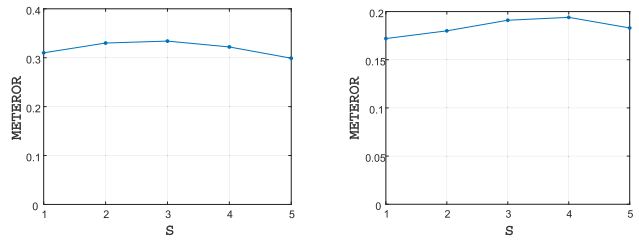


Fig. 5. Visualization of the balancing gate γ. The frames are extracted to represent the video content. The sentence below the frames is the caption generated by the CAM-RNN, and the histograms below each sentence denote the values of the balancing gate. Specifically, the higher the histogram means the larger the value, i.e., the visual feature is more emphasized, and vice versa. Note that the visual words and nonvisual words are distinguished by red color and black color, respectively. The best view is in color.



(a) (b)

Fig. 6. The performance distribution of the CAM-RNN according to the variance of S. (a) METEOR on MSVD. (b) METEOR on charades.

considered in the text attention module). It can be observed that on the MSVD dataset, the performance rises with the increase in S and begins to decline when $S > 3$, while on the Charades dataset, there is little improvement when S changes from 3 to 4. In this case, it is proper for us to set $S = 3$.

3) The Analysis on the Balancing Gate: To verify the effectiveness of the balancing gate, we compare our full approach CAM-RNN with VT-RNN since the only difference is that VT-RNN does not contain the balancing gate. We can clearly see that the CAM-RNN exceeds VT-RNN in Table I and II, which indicates the effectiveness of the balancing gate. Moreover, in Fig. 4, by comparing the results of VT-RNN and the CAM-RNN, it can be observed that the captions generated by the CAM-RNN have less errors in nonvisual words, such as the wrong use of prepositions in the first video, the absence of link verbs (*is*) and articles (*the, a*) in the second and third video, *etc.*

TABLE III
THE RESULTS OF VARIOUS APPROACHES ON MSVD DATASET (THE SCORES IN BOLD INDICATE THE BEST VALUE)

Metrics	CIDEr	BLEU–1	BLEU–2	BLEU–3	BLEU–4	ROUGE–L	METEOR
FGM [9]	–	–	–	–	–	–	0.239
Mean Pool [16]	–	–	–	–	0.372	–	0.281
S2VT [1]	0.486	0.735	0.593	0.482	0.369	0.652	0.289
SA [19]	0.481	0.741	0.589	0.482	0.366	0.647	0.294
LSTM-E [23]	–	0.749	0.609	0.506	0.402	–	0.295
p-RNN [48]	**0.621**	0.773	0.645	0.546	**0.443**	–	0.311
HRNE [49]	–	0.784	0.661	0.551	0.436	–	0.321
CAM-RNN	0.543	**0.803**	**0.676**	**0.560**	0.424	**0.694**	**0.334**

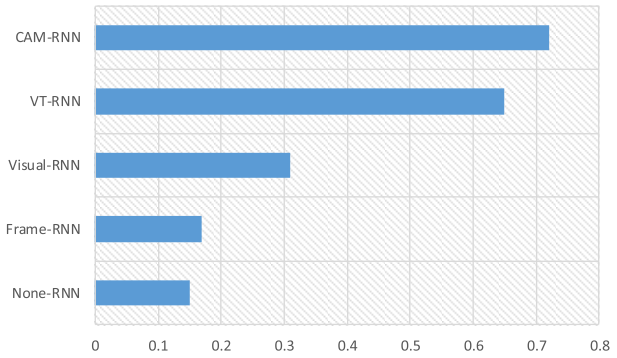


Fig. 7. The human evaluation results on the MSVD dataset. The histograms denote the ratio of captions generated by each baseline to be selected as the best.

TABLE IV
THE RUNNING TIME (SECONDS) FOR GENERATING PER CAPTION

None-RNN	Frame-RNN	Visual-RNN	VT-RNN	CAM-RNN
0.145 s	0.145 s	0.146 s	0.148 s	0.149 s

To better understand the mechanism of the balancing gate, its values when generating each word are visualized in Fig. 5. We can clearly see that the balancing gate γ is larger when generating the visual words (the visual feature is emphasized) and smaller when generating the nonvisual words, which is consistent with our intuition. Therefore, the balancing gate is effective in the regulation of visual features when generating words, regardless of whether they are visual words or nonvisual words. That is why the CAM-RNN (equipped with the balancing gate) performs better than VT-RNN, as described in Tables I and II and in Fig. 4.

4) The Human Evaluation: To further verify the effectiveness of each part of the CAM-RNN, a human evaluation is performed on MSVD to compare the quality of the captions generated by our five baselines. In practice, 30 subjects are employed in this part. Each subject is provided with the video clip and the five generated captions. They are asked to select the one that can best describe the video content. Particularly, the selection of two or three best captions is permitted if the subject believes that those sentences are equally good. Note that to reduce the subjectivity in human evaluation, each video clip is evaluated by five subjects, and those sentences selected more than 3 times are taken as the final best sentences. The results are depicted in Fig. 7. It can be observed that, in most cases (72%), the captions generated by our full model, the CAM-RNN, can best describe the video content.

Moreover, to discuss the efficiency of the proposed approach, the running time of different baselines for generating per caption is presented in Table IV. Specifically, the approaches are operated on Tesla K80. To ensure the results are more convincing, the running time is obtained by averaging the testing process of 100 captions. We can clearly see that the efficiency of different baselines are comparable with each other. In other words, the efficiency of the proposed approach is not much degraded by the increasing complexity from None-RNN to CAM-RNN.

C. Results on the MSVD Dataset

In Table III, the results of different approaches on the MSVD dataset are presented. For fairness, the influence of different features are removed by constraining all the approaches to extract the video feature with VggNet.

In Table III, all the listed approaches generate captions based on the RNN, except for the FGM. Specifically, the FGM develops a probability graph model to predict the video caption based on visual detections. It achieves state-of-the-art in traditional approaches. RNN-based approaches outperform the FGM significantly when comparing their METEOR scores. The FGM mainly benefits from the great ability of the RNN in sequence modeling.

RNN-based approaches mainly differ in their visual and text feature extraction. Specifically, Mean Pool and LSTM-E extract visual features by operating mean pooling on the frame features. S2VT employs LSTM to encode the frame features sequentially. HRNE develops a two-layer structure of the LSTM to extract the visual features hierarchically. Our CAM-RNN performs better than the above methods by taking advantage of the visual attention module that can attend to the most correlated features in the video. Actually, SA and p-RNN also utilize attention modules for visual feature extraction, where frame-level attention and region-level attention are adopted, respectively. The better results of CAM-RNN indicates the necessity to emphasize correlated visual features in different levels jointly.

For the text feature, most of the proposed approaches only utilize the feature of the single previous word in the generation of the current word. By extracting more correlated text features

Fig. 8. Example results of our CAM-RNN on the MSVD dataset. Specifically, the blue sentence is generated by the CAM-RNN, while the red sentence denotes the reference caption generated by humans.

TABLE V
THE RESULTS WITH DIFFERENT VIDEO FEATURES ON THE MSVD DATASET

Metrics	METEOR
Mean pool (GoogLeNet) [16]	0.287
S2VT(RGB+FLOW VggNet) [1]	0.297
SA(GoogLeNet+C3D) [19]	0.296
LSTM-E(C3D) [23]	0.299
LSTM-E(VggNet+C3D) [23]	0.310
p-RNN (C3D) [48]	0.303
p-RNN (VggNet+C3D) [48]	0.326
HRNE(C3D) [49]	0.310
Multimodal Attention (VggNet+C3D) [20]	0.317
LSTM-TSA (VggNet+C3D) [45]	0.335
TDDF (VggNet+C3D) [66]	0.333
MA-LSTM (GoogLeNet+C3D) [21]	0.336
CAM-RNN (VggNet)	0.334
CAM-RNN (C3D)	0.339
CAM-RNN (GoogLeNet)	0.342
CAM-RNN (VggNet+C3D)	0.340
CAM-RNN (GoogLeNet+C3D)	**0.345**

with the text attention module, the proposed CAM-RNN achieves better performance. Moreover, it is noteworthy that in Table III, the CAM-RNN does not perform well on CIDEr. This is mainly because CIDEr weakens the weight of non-visual words in the evaluation [65], while our text attention module and balancing gate work on improving the accuracy of generated nonvisual words. As a result, the improvements are not well reflected in the CIDEr scores, especially on the MSVD dataset, since the generated captions are short, and the nonvisual words form a majority. Considering that nonvisual words are also very important for the caption quality, BLEU, ROUGE-L and METEOR are more reliable in this part.

In Table V, the METEOR values of various approaches with different features are depicted. Specifically, three popular CNNs are considered, *i.e.*, VggNet, GoogLeNet and C3D. From the results in Table V, we can clearly see that 1) our approach with C3D and GoogLeNet as the feature extractor performs better than that with VggNet. This is mainly because the 3D convolution in C3D can capture the dynamic information in the video, and the deeper GoogLeNet indicates a more powerful capability in visual feature exploitation. 2) our approach with the five kinds of features achieves better performance than most of the compared approaches, and the first three versions with a single feature even perform better than the compared approaches with combined features. Specifically, we have also tested two combined features for our approach, including VggNet+C3D and GoogLeNet+C3D. The rationale is to combine the ability of VggNet and GoogLeNet in the appearance information extraction and the ability of C3D in the dynamic information exploitation. In the architecture of the combined features, the probability p for predicting the current word (defined in Eqn. (11)) is determined jointly by the probability values generated by the two features, *i.e.*,

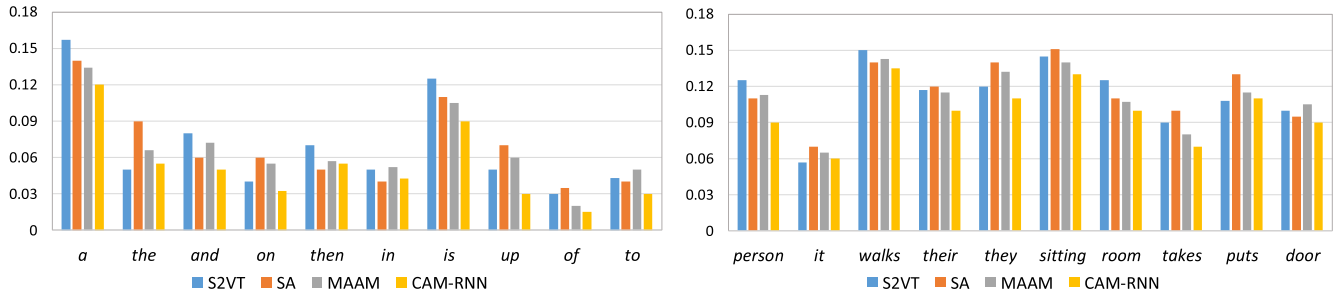
$$p = \frac{p_{vggnet} + p_{c3d}}{2} \quad or \quad p = \frac{p_{googlenet} + p_{c3d}}{2}.$$

We can see in Table V that the results of our approach are improved with combined features. Overall, based on the difference among the results of our approaches with different features, it can be ascertained that an even better performance can be achieved by our approach once a more powerful feature extractor is provided.

Finally, to provide a better understanding of the results, in Fig. 8, we present several examples generated by the proposed approach. With the frames representing the video content and the reference captions, it can be observed that most of the generated captions can basically describe the activities in the video, which has shown the effectiveness of the CAM-RNN in video captioning.

TABLE VI
THE RESULTS OF VARIOUS APPROACHES ON THE CHARADES DATASET (THE SCORES IN BOLD INDICATE THE BEST VALUE)

Metrics	CIDEr	BLEU–1	BLEU–2	BLEU–3	BLEU–4	METEOR
S2VT (VggNet) [1]	0.140	0.490	0.300	0.180	0.110	0.160
SA (VggNet) [19]	0.181	0.403	0.247	0.155	0.076	0.143
MAAM (VggNet) [58]	0.167	0.500	0.311	0.188	0.115	0.176
CAM-RNN (VggNet)	0.186	**0.534**	0.318	0.191	0.133	0.189
CAM-RNN (C3D)	0.181	0.512	0.318	0.210	**0.140**	0.192
CAM-RNN (GoogLeNet)	**0.188**	0.513	**0.321**	**0.217**	0.129	**0.197**



(a) (b)

Fig. 9. The error rates of nonvisual words and visual words on the Charades dataset. Note that the captions evaluated here are generated by the CAM-RNN (VggNet). The best view is in color. (a) Error rates of nonvisual words. (b) Error rates of visual words.

D. Results on the Charades Dataset

In Table VI, the performance of different approaches on the Charades dataset is presented. S2VT and SA have been introduced in the previous subsection. The MAAM is an attention model augmented by the memory unit, where the attention memory in the past time is utilized to determine the current attention weight. Similarly, our region-level attention layer also adopts this strategy. It is effective in the task of video captioning, which has been verified by the better performance of the MAAM than SA (a simple attention model). Furthermore, our approach achieves even better performance. This result is mainly because our visual attention module can selectively attend to the most correlated features in both the frame level and the region level, while the MAAM just encodes the visual feature in the frame level. In addition, the results of the CAM-RNN with three different features are also provided in Table VI. The variance among these results demonstrates that the performance of the CAM-RNN is influenced by the feature extractors and indicates that better performance can be achieved with more powerful feature extractors.

To verify the improvements of the CAM-RNN on the accuracy of generated captions, we perform user studies in this part. First, we compute 10 nonvisual words (*i.e.*, a, the, and, on, then, in, is, up, of, and to) and 10 visual words (person, it, walks, their, they, sitting, room, takes, puts, and door) that occur most frequently in the test set of the Charades dataset. Then, we employ 30 subjects to judge the omissions and misuses of these words in the generated captions. In this part, each video is in the charge of 3 subjects, and each subject is responsible for approximately 200 videos. Note that, to reduce bias, the average value of the 3 subjects in charge is taken as the result of that video.

The results of user studies are displayed in Fig. 9. Fig. 9(a) and 9(b) show the error rates of nonvisual words and visual words, respectively. It can be observed that, compared to S2VT, SA and MAAM, the error rates of the captions generated by the CAM-RNN are much lower for both nonvisual words and visual words. In this case, this result illustrates the improvements of CAM-RNN on the accuracy of the generated captions.

Furthermore, to better understand the results, the examples of generated captions are displayed in Fig. 10. We can see that the captions generated by the CAM-RNN have little grammatical error, and most of them can basically describe the content of the video according to the displayed video frames. As aforementioned, the Charades dataset is more challenging since each caption describes a series of activities in the video. By comparing the generated results with the reference captions, it can be observed that the generated captions capture most of the activities in the video. It mainly benefits from our visual and text attention modules together with the balancing gate. However, in some generated captions, there are still several omissions and errors, mainly because those objects are very small, and the activities are quite short. It is difficult to extract their visual features.

E. Results on the MSR-VTT Dataset

Table VII presents the results of various approaches on the MSR-VTT dataset. Specifically, all the compared approaches are equipped with combined features, GoogLeNet+C3D or VGGnet+C3D. Moreover, some of these approaches are even equipped with audio information, which is a good practice in video captioning since audio is an important modality in understanding video content. From the results of Mean Pool and MA-LSTM, we can see that the audio information

Fig. 10. Example results of our CAM-RNN on the Charades dataset. Specifically, the blue sentence is generated by the CAM-RNN, while the red sentence denotes the reference caption generated by humans.

TABLE VII
THE RESULTS OF VARIOUS APPROACHES ON THE MSR-VTT DATASET (THE SCORES IN BOLD INDICATE THE BEST VALUE)

Metrics	CIDEr	BLEU–4	ROUGE-L	METEOR
Mean pool (GoogLeNet+C3D) [16]	0.355	0.341	0.548	0.248
Mean pool (GoogLeNet+C3D+Audio) [16]	0.381	0.357	0.582	0.256
S2VT (GoogLeNet+C3D+Audio) [1]	0.391	0.360	0.584	0.260
SA (GoogLeNet+C3D+Audio) [19]	0.367	0.348	0.571	0.251
LSTM-E (GoogLeNet+C3D+Audio) [23]	0.385	0.361	0.586	0.258
MA-LSTM (GoogLeNet+C3D) [21]	0.381	0.354	0.582	0.258
MA-LSTM (GoogLeNet+C3D+Audio) [21]	0.401	0.363	**0.591**	0.263
TDDF (VGGnet+C3D) [66]	**0.441**	0.372	0.586	0.277
TDDF (GoogLeNet+C3D) [66]	0.438	0.373	0.592	0.278
CAM-RNN (VggNet+C3D)	0.383	**0.377**	0.585	0.267
CAM-RNN (GoogLeNet+C3D)	0.388	0.362	0.588	**0.279**

can indeed improve the performance. Fortunately, benefiting from our co-attention model, our approach can still attain comparable results even without the audio features. In detail, in Table VII, the CAM-RNN still achieves the best performance on BLEU-4 and METEOR and obtains comparable results on ROUGE-L. Similar to MSVD, the poorer performance on CIDEr is because CIDEr weakens the weight of nonvisual words in the evaluation of the caption quality.

F. Results on the MPII-MD Dataset

As previously mentioned, MPII-MD is a very challenging dataset, so the values of results reflected in the evaluation metrics are very low. Following existing approaches, such as [1], [23], we just provide the METEOR metric in Table VIII.

Specifically, Mean Pool, S2VT and LSTM-E have been already introduced in previous subsections. Visual-Labels is based on both CNN classifiers and LSTM, where classifiers are utilized to predict the semantic elements (*i.e.*, verbs, objects and places), and LSTM is employed to generate the caption with these elements as input. Visual-Labels is a combination of traditional approaches and RNN-based approaches. Compared to Mean Pool, Visual-Labels shows the superiority of borrowing the classification results of CNN rather than just employing the CNN feature. Nevertheless, our approach performs better because the visual attention module can partially compensate

TABLE VIII
THE RESULTS OF VARIOUS APPROACHES ON THE MPII-MD DATASET

Metrics	METEOR
Mean pool [16]	0.058
Visual-Labels [67]	0.063
S2VT [1]	0.063
S2VT-LM [50]	0.068
LSTM-E [23]	0.073
LSTM-TSA [45]	0.076
CAM-RNN (VggNet)	0.076
CAM-RNN (C3D)	0.075
CAM-RNN (GoogLeNet)	**0.078**

for the drawback of CNN features. S2VT-LM is extended from S2VT by adding extra language modules so that it can mine more linguistic knowledge from text. The better performance of S2VT-LM than S2VT has verified the necessity of exploiting more powerful text features. In addition, LSTM-TSA generates the caption with extra video attributes (*i.e.*, objects and activities). However, our approach achieves comparable results, even without this information.

V. Conclusion

In this paper, to extract the most correlated visual feature and text feature for the task of video captioning, we propose a new approach, the CAM-RNN. Specifically, the CAM (co-attention model) is composed of three parts, *i.e.*, visual attention module, text attention module and balancing gate. In particular, the visual attention module is utilized to encode the visual feature in both the frame level and region level, so that it can selectively focus on the most correlated frames and the salient regions in each frame. Moreover, it can effectively reduce the interference caused by the irrelevant background information. The text attention module is utilized to encode the text feature by adaptively focusing on the most correlated phrases formed by previously generated words, so that more accurate text features are exploited. The balancing gate is employed to regulate the influence of the visual feature, as such, it is considerably helpful when generating nonvisual words. All three parts construct the visual feature and text feature encoder, and LSTM is utilized as the decoder to generate the caption word by word. The results on four popular datasets have verified the effectiveness of the proposed approach.

Furthermore, although this approach is proposed for the video captioning task, it can be transferred to many video and text analysis tasks, such as image captioning and video question-answer.

References

[1] S. Venugopalan, M. Rohrbach, J. Donahue, R. Mooney, T. Darrell, and K. Saenko, "Sequence to sequence—Video to text," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 2015, pp. 4534–4542.

[2] X. Tan, Y. Guo, Y. Chen, and W. Zhu, "Accurate inference of user popularity preference in a large-scale online video streaming system," *Sci. China Inf. Sci.*, vol. 61, no. 1, 2017, Art. no. 018101.

[3] X. Li, B. Zhao, and X. Lu, "A general framework for edited video and raw video summarization," *IEEE Trans. Image Process.*, vol. 26, no. 8, pp. 3652–3664, Aug. 2017.

[4] X. Li, B. Zhao, and X. Lu, "Key frame extraction in the summary space," *IEEE Trans. Cybern.*, vol. 48, no. 6, pp. 1923–1934, Jun. 2018.

[5] B. Zhao, X. Li, X. Lu, and Z. Wang, "A CNN–RNN architecture for multi-label weather recognition," *Neurocomputing*, vol. 322, pp. 47–57, Dec. 2018.

[6] Q. Wang, J. Gao, W. Lin, and Y. Yuan, "Learning from synthetic data for crowd counting in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2019, pp. 8198–8207.

[7] H. Fang, C. Shang, and J. Chen, "An optimization-based shared control framework with applications in multi-robot systems," *Sci. China Inf. Sci.*, vol. 61, no. 1, 2018, Art. no. 014201.

[8] S. Guadarrama *et al.*, "YouTube2Text: Recognizing and describing arbitrary activities using semantic hierarchies and zero-shot recognition," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2712–2719.

[9] J. Thomason, S. Venugopalan, S. Guadarrama, K. Saenko, and R. Mooney, "Integrating language and vision to generate natural language descriptions of videos in the wild," in *Proc. 25th Int. Conf. Comput. Linguistics, Tech. Papers (COLING)*, Dublin, Ireland, Aug. 2014, pp. 1218–1227.

[10] N. Krishnamoorthy, G. Malkarnenkar, R. J. Mooney, K. Saenko, and S. Guadarrama, "Generating natural-language video descriptions using text-mined knowledge," in *Proc. 27th AAAI Conf. Artif. Intell.*, 2013, pp. 541–547.

[11] J. Donahue *et al.*, "Long-term recurrent convolutional networks for visual recognition and description," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 677–691, Apr. 2017.

[12] K.-H. Zeng, T.-H. Chen, J. C. Niebles, and M. Sun, "Title generation for user generated videos," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 609–625.

[13] R. Shetty and J. Laaksonen, "Video captioning with recurrent networks based on frame- and video-level features and visual content classification," *CoRR*, vol. abs/1512.02949, Dec. 2015. [Online]. Available: http://arxiv.org/abs/1512.02949

[14] X. Long, C. Gan, and G. de Melo, "Video captioning with multi-faceted attention," *CoRR*, vol. abs/1612.00234, Dec. 2016.

[15] B. Zhao, X. Li, and X. Lu, "Video captioning with tube features," in *Proc. 27th Int. Joint Conf. Artif. Intell. (IJCAI)*, Stockholm, Sweden, Jul. 2018, pp. 1177–1183.

[16] S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, and K. Saenko, "Translating videos to natural language using deep recurrent neural networks," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2015, pp. 1494–1504.

[17] X. Long, C. Gan, and G. de Melo. (2016). "Video captioning with multi-faceted attention." [Online]. Available: https://arxiv.org/abs/1612.00234

[18] Y. Yu, H. Ko, J. Choi, and G. Kim. (2016). "Video captioning and retrieval models with semantic attention." [Online]. Available: https://arxiv.org/abs/1610.02947

[19] L. Yao *et al.*, "Describing videos by exploiting temporal structure," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 4507–4515.

[20] C. Hori, T. Hori, T.-Y. Lee, K. Sumi, J. R. Hershey, and T. K. Marks, "Attention-based multimodal fusion for video description," *CoRR*, vol. abs/1701.03126, Jan. 2017.

[21] J. Xu, T. Yao, Y. Zhang, and T. Mei, "Learning multimodal attention LSTM networks for video captioning," in *Proc. 25th ACM Int. Conf. Multimedia*, Mountain View, CA, USA, Oct. 2017, pp. 537–545.

[22] Y. Jang, Y. Song, Y. Yu, Y. Kim, and G. Kim, "TGIF-QA: Toward spatio-temporal reasoning in visual question answering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1359–1367.

[23] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui, "Jointly modeling embedding and translation to bridge video and language," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 4594–4602.

[24] J. Lu, J. Yang, D. Batra, and D. Parikh, "Hierarchical question-image co-attention for visual question answering," in *Proc. Adv. Neural Inf. Process. Syst.*, Barcelona, Spain, Dec. 2016, pp. 289–297.

[25] H. Nam, J.-W. Ha, and J. Kim, "Dual attention networks for multimodal reasoning and matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 2156–2164.

[26] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Lisbon, Portugal, Sep. 2015, pp. 1412–1421.

[27] X. Li *et al.*, "MAM-RNN: Multi-level attention model based RNN for video captioning," in *Proc. IJCAI*, 2017, pp. 2208–2214.

[28] N. Kalchbrenner and P. Blunsom, "Recurrent continuous translation models," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Washington, DC, USA, Oct. 2013, pp. 1700–1709.

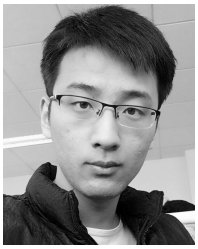
[29] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *CoRR*, vol. abs/1409.0473, Sep. 2014.

[30] A. Eriguchi, K. Hashimoto, and Y. Tsuruoka, "Tree-to-sequence attentional neural machine translation," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Berlin, Germany, vol. 1, Aug. 2016, pp. 823–833.

[31] I. V. Serban *et al.*, "A hierarchical latent variable encoder-decoder model for generating dialogues," in *Proc. 31st AAAI Conf. Artif. Intell.*, San Francisco, CA, USA, Feb. 2017, pp. 3295–3301.

[32] J. Li, M.-T. Luong, and D. Jurafsky, "A hierarchical neural autoencoder for paragraphs and documents," in *Proc. 53rd Annu. Meeting Assoc. Comput. Linguistics, 7th Int. Joint Conf. Natural Lang. Process. Asian Fed. Natural Lang. Process. (ACL)*, Beijing, China, vol. 1, Jul. 2015, pp. 1106–1115.

[33] R. Lin, S. Liu, M. Yang, M. Li, M. Zhou, and S. Li, "Hierarchical recurrent neural network for document modeling," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Lisbon, Portugal, Sep. 2015, pp. 899–907.
[34] A. Farhadi *et al.*, "Every picture tells a story: Generating sentences from images," in *Proc. 11th Eur. Conf. Comput. Vis.*, Heraklion, Greece, Sep. 2010, pp. 15–29.
[35] P. Kuznetsova, V. Ordonez, A. Berg, T. Berg, and Y. Choi, "Generalizing image captions for image-text parallel corpus," in *Proc. 51st Annu. Meeting Assoc. Comput. Linguistics (ACL)*, Sofia, Bulgaria, vol. 2, Aug. 2013, pp. 790–796.
[36] Y. Jia, M. Salzmann, and T. Darrell, "Learning cross-modality similarity for multinomial data," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Barcelona, Spain, Nov. 2011, pp. 2407–2414.
[37] G. Kulkarni *et al.*, "Babytalk: Understanding and generating simple image descriptions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 12, pp. 2891–2903, Dec. 2013.
[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[39] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556
[40] C. Szegedy *et al.*, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
[41] K. Cho, A. Courville, and Y. Bengio, "Describing multimedia content using attention-based encoder-decoder networks," *IEEE Trans. Multimedia*, vol. 17, no. 11, pp. 1875–1886, Nov. 2015.
[42] K. Xu *et al.*, "Show, attend and tell: Neural image caption generation with visual attention," in *Proc. 32nd Int. Conf. Mach. Learn. (ICML)*, Lille, France, Jul. 2015, pp. 2048–2057.
[43] Q. You, H. Jin, Z. Wang, C. Fang, and J. Luo, "Image captioning with semantic attention," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 4651–4659.
[44] J. Lu, C. Xiong, D. Parikh, and R. Socher, "Knowing when to look: Adaptive attention via a visual sentinel for image captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 3242–3250.
[45] Y. Pan, T. Yao, H. Li, and T. Mei, "Video captioning with transferred semantic attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 984–992.
[46] L. Baraldi, C. Grana, and R. Cucchiara, "Hierarchical boundary-aware neural encoder for video captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1657–1666.
[47] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[48] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu, "Video paragraph captioning using hierarchical recurrent neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 4584–4593.
[49] P. Pan, Z. Xu, Y. Yang, F. Wu, and Y. Zhuang, "Hierarchical recurrent neural encoder for video representation with application to captioning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 1029–1038.
[50] S. Venugopalan, L. A. Hendricks, R. J. Mooney, and K. Saenko, "Improving lstm-based video description with linguistic knowledge mined from text," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, Austin, TX, USA, Nov. 2016, pp. 1961–1966.
[51] J. Liang, L. Jiang, L. Cao, L. Li, and A. Hauptmann, "Focal visual-text attention for visual question answering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 6135–6143.
[52] P. Lu, L. Ji, W. Zhang, N. Duan, M. Zhou, and J. Wang, "R-VQA: Learning visual relation facts with semantic attention for visual question answering," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery & Data Mining*, 2018, pp. 1880–1889.
[53] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
[54] W. Zaremba and I. Sutskever, "Learning to execute," *CoRR*, vol. abs/1410.4615, 2014. [Online]. Available: http://arxiv.org/abs/1410.4615
[55] G. A. Sigurdsson, G. Varol, X. Wang, A. Farhadi, I. Laptev, and A. Gupta, "Hollywood in homes: Crowdsourcing data collection for activity understanding," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 510–526.
[56] J. Xu, T. Mei, T. Yao, and Y. Rui, "MSR-VTT: A large video description dataset for bridging video and language," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 5288–5296.
[57] A. Rohrbach, M. Rohrbach, N. Tandon, and B. Schiele, "A dataset for movie description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, Jun. 2015, pp. 3202–3212.
[58] R. Fakoor, A. Mohamed, M. Mitchell, S. B. Kang, and P. Kohli, "Memory-augmented attention modelling for videos," *CoRR*, abs/1611.02261, 2016.
[59] K. Simonyan and A. Zisserman. (2014). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556
[60] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, Dec. 2015, pp. 4489–4497.
[61] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global vectors for word representation," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2014, pp. 1532–1543.
[62] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2002, pp. 311–318.
[63] C. Lin and F. J. Och, "Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics," in *Proc. 42nd Annu. Meeting Assoc. Comput. Linguistics*, 2004, Art. no. 605.
[64] M. J. Denkowski and A. Lavie, "Meteor universal: Language specific translation evaluation for any target language," in *Proc. 9th Workshop Stat. Mach. Transl.*, 2014, pp. 376–380.
[65] R. Vedantam, C. L. Zitnick, and D. Parikh, "CIDEr: Consensus-based image description evaluation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 4566–4575.
[66] X. Zhang, K. Gao, Y. Zhang, D. Zhang, J. Li, and Q. Tian, "Task-driven dynamic fusion: Reducing ambiguity in video description," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 6250–6258.
[67] A. Rohrbach, M. Rohrbach, and B. Schiele, "The long-short story of movie description," in *Proc. 37th German Conf. Pattern Recognit. (GCPR)*, Aachen, Germany, Oct. 2015, pp. 209–221.



Bin Zhao is currently pursuing the Ph.D. degree with the School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China. His research interests include video summarization, video captioning, and machine learning.

Xuelong Li is currently a Full Professor with the School of Computer Science and Center for OPTical IMagery Analysis and Learning (OPTIMAL), Northwestern Polytechnical University, Xi'an, China.

Xiaoqiang Lu is currently a Full Professor with the Key Laboratory of Spectral Imaging Technology CAS, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China. His research interests include pattern recognition, machine learning, hyperspectral image analysis, cellular automata, and medical imaging.