Potential of Incorporating Motion Estimation for Image Captioning

Kiyohiko Iwamura[1], Jun Younes Louhi Kasahara[1], Alessandro Moro[2], Atsushi Yamashita[1] and Hajime Asama[1]

***Abstract*— Automatic image captioning has various important applications such as indexing images on the Web or the depiction of visual contents for the visually impaired. Recently, deep learning based probabilistic frameworks have been greatly researched for image captioning. However the existing deep learning methods are only established on visual features, which have problems generating captions related to motions, because visual features from images do not include motion features. In this paper, we propose a novel, end-to-end trainable, deep learning image captioning model that estimates motion features from a image to help generate captions. Our proposed model was evaluated on two datasets, MSR-VTT2016-Image, and several copyright free images. We demonstrate that our proposed method using motion features improves performance on caption generation and that the quality of motion features is important to generate captions.**

I. INTRODUCTION

In recent years, automatically generating captions of images has been an active research topic in Computer Vision [1]. Various important applications exist such as the depiction of visual contents for the visually impaired and indexing services for images on social networks. Traditionally, such captions are generated manually by humans for each image, which is a labor intensive process. Therefore, there is a need for automatic image caption generation.

Recently, deep learning models have seen success for automatically generating image captions [2][3]. Such models generate accurate image captions for unknown images, without the need for humans to design features, by training using a large number of datasets. Vinyals et al. [2] proposed a model that consisted of Convolutional Neural Networks (CNN) and Recurrent Neural Network (RNN). Wang et al. [4] introduced bidirectional Long-Short Term Memory (LSTM) for RNN, which performed bidirectional calculations. These models can learn relationships between overall image features and corresponding words in the caption.

To generate highly accurate captions, it is often important to exploit some image regions features such as objects. Therefore, a visual attention captioning model has been proposed [5]. This model focuses on the regions of an image related to specific words in image caption. For that reason, attention captioning models have been developed and reported high performance in image captioning. The first approach to utilize attention mechanism was Xu et al. [5]. They improved caption quality but also provided the ability to visualize what the model use for traceability. Lu et al. [6] proposed adaptive attention mechanism, which decides when to rely on visual or non-visual features according to words in the caption.

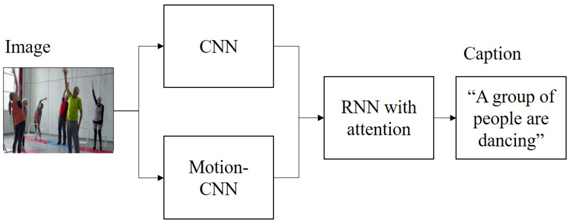


Fig. 1. Overview of our model. Input image is processed to CNN and Motion-CNN for incorporating image and motion features. Then RNN with attention is conducted for generating caption.

Most visual attention models used in image captioning improve performance by using the overall image or specific image regions. These method aim to learn directly relationships between image features and corresponding image caption. However, this is not relevant for words which do not have meaningful relationships with image features. For example, verbs that describe motion cannot be meaningfully put in relation with image features. Therefore generating captions including such verbs are challenging for captioning models.

If we get motion features from particular image regions, we can include the relation between motion features and verbs in the caption generation process. We directly incorporate motion features by motion estimation, which has been overlooked in related works. Therefore, the objective of this paper is to incorporate motion features in image captioning.

In this paper, we propose a method for image caption generation that focuses in incorporating motion estimation from images. For example, as illustrated in Fig. 1, our proposed method has 2 convolutional feature extractions: one is for image features and the other one is for motion features.

Overall, the main contributions of this paper are:

- We introduce a novel caption model with motion estimation that automatically extracts overall image features and motion features from images.
- We perform an analysis of our model, particularly on the effect of quality of motion features by comparing the performance of our proposed method without motion features, with estimated motion features and with high quality motion features.

[1] Kiyohiko Iwamura, Jun Younes Louhi Kasahara, Atsushi Yamashita and Hajime Asama are with the Department of Precision Engineering, The University of Tokyo, 113-8656 Tokyo, Japan. iwamura@robot.t.u-tokyo.ac.jp

[2] Alessandro Moro is with RITECS Inc., 190-0023 Tokyo, Japan.

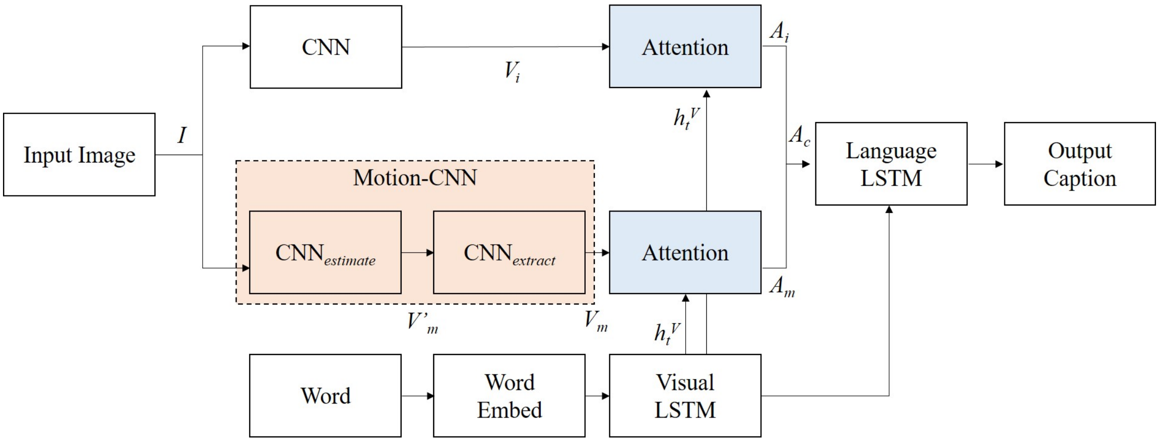


Fig. 2. Overview of our proposed model, which is composed of CNN and motion-CNN consisted of $\text{CNN}_{estimate}$ and $\text{CNN}_{extract}$. Two Attention and LSTM are performed to get attention features and caption. The model takes the images and the words estimated at last time step as input.

II. METHOD

We explain the concept about our model in Section II-A. We introduce our proposed captioning models in Sections II-B & II-C, then the generic probabilistic framework for image captioning is detailed in Section II-D.

A. Concept

In this paper, our concept is to incorporate motion estimation from images. Our key motivation is that humans can estimate motion from images by many years of experiences. For example, given an image, human can distinguish between actions such as ”walking” and ”standing”.

Furthermore, this is supported by corresponding neuroscience researches, such as Kourtzi et al. [7]. They report that the Medial Temporal/Medial Superior Temporal (MT/MST) cortex is one of the main brain regions engaged in the perceptual analysis of visual motion, and that MT/MST also engages in processing implied dynamic information from images.

Therefore, we introduce a motion estimation model within our image captioning model.

B. Overall Model Architecture

Following [1][5], our model consists of a CNN portion and a LSTM portion. The CNN portion is used for incorporating image features, and the LSTM portion is utilized for generating captions. However these models can not estimate motion features. Therefore we propose motion-CNN in the CNN portion.

Fig. 2 presents the overview of our model. Image $\boldsymbol{I}$ is passed to CNN and motion-CNN consisting of $\text{CNN}_{estimate}$ and $\text{CNN}_{extract}$. We get image features $\boldsymbol{V}_i$ and motion features $\boldsymbol{V}_m$ from each CNN. Next, attention features $\boldsymbol{A}_i$, $\boldsymbol{A}_m$ are calculated for each feature from the output of Visual LSTM (LSTM_V) $\boldsymbol{h}_t^V$ at time step t. Then, concatenated attention features $\boldsymbol{A}_c$ is inputted to Language LSTM (LSTM_L) and the caption is generated.

The flow of motion-CNN can be defined by the following formulas:

$$\boldsymbol{V}_m = \text{motion-CNN}(\boldsymbol{I}). \tag{1}$$

Different from [1][5], we get image features $\boldsymbol{V}_i$ and motion features $\boldsymbol{V}_m$. Therefore according to each feature, we use two separate attention mechanisms for focusing on the regions of an image related to specific words in image caption. The flow of attention is computed as:

$$\boldsymbol{A}_i = \text{Attention}(\boldsymbol{h}_t^V, \boldsymbol{V}_i), \tag{2}$$

$$\boldsymbol{A}_m = \text{Attention}(\boldsymbol{h}_t^V, \boldsymbol{V}_m), \tag{3}$$

where each attention shares learned parameters which are calculated by image features $\boldsymbol{V}_i$. Next, we need to fuse both $\boldsymbol{A}_i$ and $\boldsymbol{A}_m$. This is done as:

$$\boldsymbol{A}_c = \text{Concat}(\boldsymbol{A}_i, \boldsymbol{A}_m), \tag{4}$$

where $\text{Concat}(\cdot)$ is vector concatenation. From our experience, concatenation yields better results than summation for fusing image features $\boldsymbol{V}_i$ and motion features $\boldsymbol{V}_m$.

C. Multiple CNN Architecture

In order to incorporate both image and motion features, we propose a multiple CNN model by feeding the image to CNN and motion-CNN. Fig. 3 presents the overview of our motion-CNN model. It consists of 2 types of CNNs. Given an image, convolutional motion estimation is first conducted and the output is image expressed motion of the same size as the input image. This expressed motion is estimated opticalflow. Estimated opticalflow is the pattern of apparent motion of objects or edges in a visual scene generated by Neural Network. Then, convolutional feature extraction is applied and motion features from image are obtained.

The process can be defined by the following formulas:

$$\boldsymbol{V}'_m = \text{CNN}_{estimate}(\boldsymbol{I}), \tag{5}$$

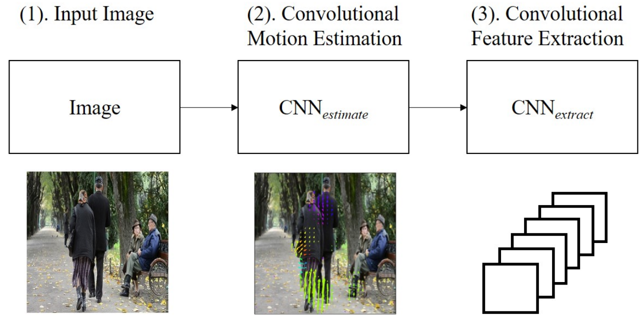

(1). Input Image
(2). Convolutional Motion Estimation
(3). Convolutional Feature Extraction
Image
CNN$_{estimate}$
CNN$_{extract}$

Fig. 3. Overall architecture of motion-CNN and examples corresponding to input image: (1) is an input image example, (2) is the flow visualization of estimated opticalflow, Following [10], we use same color coding to visualize flow for best viewed, (3) is the extract features.

$$\boldsymbol{V}_m = \text{CNN}_{\textit{extract}}(\boldsymbol{V}'_m), \tag{6}$$

where $\boldsymbol{V}'_m$ is the CNN$_{\textit{estimate}}$ output vector. CNN$_{\textit{estimate}}(\cdot)$ is the motion estimation and CNN$_{\textit{express}}(\cdot)$ is the feature extraction.

D. Probabilistic Framework for Image Captioning

We briefly mention the probabilistic framework for image captioning [2][6]. Given an image and corresponding caption, the probabilistic model directly maximizes the following formulation:

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \sum_{(\boldsymbol{I},\boldsymbol{S})} \log p(\boldsymbol{S}|\boldsymbol{I};\boldsymbol{\theta}), \tag{7}$$

where $\boldsymbol{\theta}$ are the parameters of the model, $\boldsymbol{I}$ is the image, and $\boldsymbol{S} = \{S_1, ..., S_t\}$ is the corresponding caption. Calculating the chain rule to model, the log likelihood of the joint probability is the following notation:

$$\log p(\boldsymbol{S}|\boldsymbol{I}) = \sum_{(t=1)}^{N} \log p(S_t|S_1, ..., S_{t-1}, \boldsymbol{I}), \tag{8}$$

where we drop the dependency on model parameters $\boldsymbol{\theta}$ for convenience.

In the probabilistic framework using RNN, RNN outputs $\boldsymbol{h}_t$ expressing the number of words according to time steps. The calculation of RNN over a single time step is operated by the following formula:

$$\boldsymbol{h}_t = f(\boldsymbol{h}_{t-1}, \boldsymbol{x}_t), \tag{9}$$

where f is a non-linear function. $\boldsymbol{x}_t$ is the RNN input vector at time t, and $\boldsymbol{h}_t$ is the RNN output vector at time t.

III. Experiments

A. Experimental Setup In Laboratory Conditions

In our experiments, we used ResNet101 [8] pretrained on ImageNet [9] for CNN and CNN$_{\textit{extract}}$ to extract image features. Im2Flow [10] was used for CNN$_{\textit{estimate}}$ to estimate motion features from images. We fine-tuned CNN and CNN$_{\textit{extract}}$. We used the Adam optimizer with base learning rate of 2×10^{-4} for the LSTMs and 1×10^{-5} for the CNN, and decay rate was set to 0.8 for every 5 epochs. The dimensions of embedding layers and both LSTMs were set to 512. We trained our model under cross entropy loss with doubly stochastic regularization [5]. In the decoding process, we used beam search with the beam size set to 3. We set the batch size to 16. All of our experiments were conducted on Intel Core i9-7900X cpu, Ubuntu 18.04, 64G RAM and GTX2080 Ti GPU with 12G memory.

Two experiments were conducted:

- Experiment 1 aiming at analyzing the effects of quality of motion features using MSR-VTT2016, a video captioning dataset. Concretely, we compared our model with no motion, estimated motion, and opticalflow calculated using 2 video frames.
- Experiment 2 performing image captioning with copyright free images freely available on the Internet.

The following methods are compared in our experiments:

- (A) Proposed method (without motion estimation): our proposed model without motion-CNN, which uses image features only.
- (B) Proposed method (with motion estimation): our proposed model with motion-CNN, which incorporates both image features and motion features.
- (C) Proposed method (with opticalflow): our proposed model with motion-CNN, which uses both image features and opticalflow for motion features. Opticalflow was calculated using two consecutive images and corresponds to high quality motion features.

B. Datasets

MSR-VTT2016-Image. To conduct an analysis of our model, particularly on the effects of quality of motion features, we created a image captioning dataset using MSR-VTT2016 [11]. MSR-VTT2016 is a large-scale video dataset, with 10,000 clips totaling 41.2 hours and 20 captions per clip. To evaluate the effects of quality of motion features, we needed high quality motion features. Therefore we used opticalflow in this paper. Opticalflow is the pattern of apparent motion of objects or edges in a visual scene calculated by two consecutive images. For conversion to an image captioning dataset, 4 frames, separated by 10 frames each, were extracted from each clip of MSR-VTT2016 and associated with 5 of the corresponding captions. To generate opticalflow, we used LiteflowNet2 [12]. This produced an image captioning dataset with the benefit of actual motion associated with each image, i.e., high quality motion features.

Copyright free images. This is used for qualitative analysis. This dataset was created by using copyright free web video clips, in the same fashion as for MSR-VTT2016-Image. Therefore those images contain true motion features but do not include captions.

Pre-processing. We truncated captions longer than 22 words for MSR-VTT2016-Image. We then built a vocabulary

TABLE I
EXPERIMENTAL1 RESULTS WITH MSR-VTT2016-IMAGE DATASET

Method	BLEU-1	BLEU-2	BLEU-3	BLEU-4	METEOR	ROUGE-L	CIDEr
(A) Without motion estimation	47.1	30.4	20.7	14.1	15.4	38.8	30.9
(B) With motion estimation	48.2	30.8	20.6	13.8	15.4	38.4	30.3
(C) With opticalflow	**49.6**	**32.4**	**22.0**	**15.1**	**16.1**	**39.8**	**33.6**

of words, removing words occurring less than 5 times, and obtained a vocabulary of 7,802 words.

C. Evaluation Metrics

To evaluate our model's performance on image captioning, the commonly used BLEU-N (N=1,2,3,4) metric [13], METEOR metric [14], ROUGE-L metrics [15] and CIDEr metric [16] were used.

BLEU-N metrics is calculated as:

$$BLEU_N = \min(1, e^{1-\frac{r}{c}}) \cdot e^{\frac{1}{N}\sum_{n=1}^{N}\log p_n}, \tag{10}$$

where r is reference sentence, c is generated sentence, p_n is the modified n-gram precision. We also used METEOR metric [14], ROUGE-L metric [15] and CIDEr metric [16] for comparison. These values basically translate the similarity of the generated caption with the ground truth caption. Therefore for all evaluation metrics, higher values show better results.

D. Results on Experiment 1

Table 1 presents the results obtained in experiment 1. For all evaluation metrics, higher values show better results and the best value for each metric is shown using bold fonts. The proposed method with optical flow obtained the best results for all considered metrics. The proposed method without motion estimation was the second best, except for BLEU-1,2 and the proposed method with motion estimation obtained overall the lowest values. The difference among those three variations of our proposed method was the quality of motion features: incorporating high quality motion features did improve image captioning performance.

E. Results on Experiment 2

Fig. 4 shows the copyright free images used in experiment 2. Fig. 4 (a) presents an image of people walking on the street and captions generated by each method. The proposed method without motion estimation generated "A group of people are walking on the street". The proposed method with motion estimation outputted "A group of people are walking down the street". The proposed method with opticalflow generated "A man is walking down the street". All methods succeeded here and managed to correctly generate **"walking"** in the cation.

Fig. 4 (b) shows an image of a car running in an aisle. The proposed method without motion estimation generated "Someone is showing a car". The proposed method with motion estimation generated "A person is driving". The proposed method with opticalflow outputted "A car is being driven". Here, the proposed methods incorporating motion features successfully generated words such as **"driving"** and **"driven"** in their caption while our proposed method without motion features only generated the word **"showing"**.

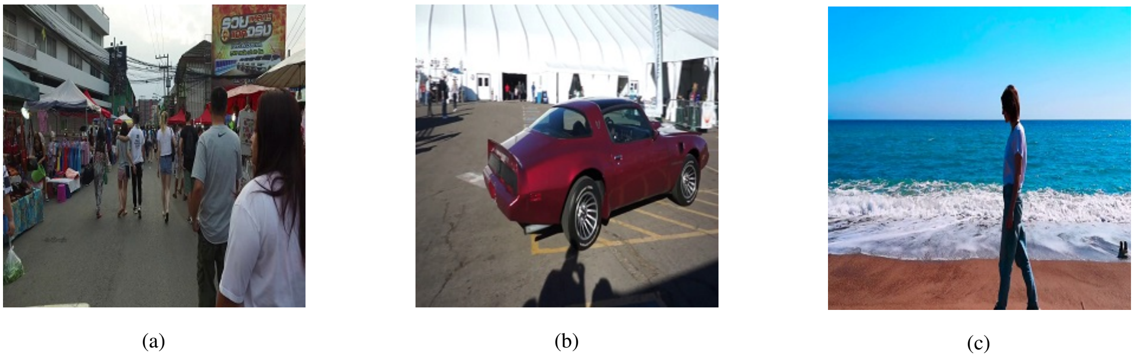
Fig. 4 (c) presents an image of a man walking on a beach. The proposed method without motion estimation generated "A man is dancing". The proposed method generated "A man in a blue shirt is swimming in the sea". The proposed method with opticalflow outputted "A man in a blue shirt is on the beach. Here, all methods failed to generate correctly the world **"walking"**.

F. Discussion

Results in experiment 1 show that high quality motion features improve image captioning performance, and that estimated motion features decrease image captioning performance. One possible explanation for the decreased performance is the lack of accuracy of our motion estimation. Proposed motion-CNN consisted of CNN$_{estimate}$ and CNN$_{extract}$. This CNN$_{estimate}$ was pretrained using an action recognition dataset. Therefore the performance of the estimation is variable depending on the target. For example, the performance for objects such as humans can be high. However, for other objects, such as the background, it may be poorer. Therefore, incorporating other CNN models for motion estimation may yield better captioning performance for our method.

Results in experiment 2 show examples of successes and failures for generating captions. Fig. 5 presents the flow visualization of each input image shown in Fig. 4. Following [10], we use same color coding to visualize flow for best viewed. The first and second rows show that estimated opticalflow and opticalflow matches for objects. Therefore our models using motion features generated correct words such as **"walking"**, **"driving"** and **"driven"**. The last row shows a failure case. Estimated opticalflow is erroneous compared to opticalflow, and this estimated opticalflow may hinder the generation of the word **"walking"**. However our model with opticalflow can not output the word **"walking"** either. Therefore, this case illustrates our model failing to use motion features.

As we have discussed, we can improve image captioning performance by utilizing motion features. However, it is possible that motion features can not be acquired from images that show very slow moving objects. This is because current model of CNN$_{estimate}$ has been pre-trained based on fast human motion. Therefore, to deal with such cases, it is necessary to pre-train with a different dataset containing slow moving objects.



(a)

(b)

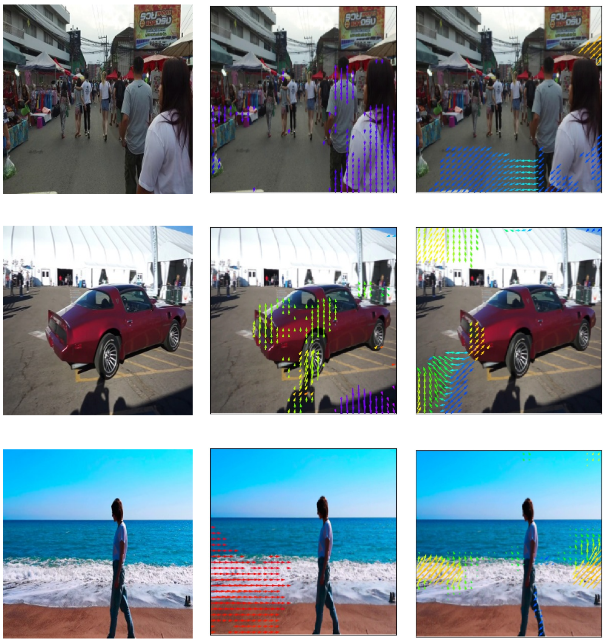
(c)

A group of people are walking on the street.
A group of people are walking down the street.
A man is walking down the street.

Someone is showing a car.
A person is driving.
A car is being driven.

A man is dancing.
A man in a blue shirt is swimming in the sea.
A man in a blue shirt is on the beach.

Fig. 4. Images used in experiment 2 and captions generated by each method. Captions in black font were generated by our proposed without motion estimation. Captions in blue font were generated by our proposed with motion estimation. Captions in red font were generated by our proposed method with opticalflow. (a) is an image of people walking on the street. (b) is an image of a car running on the aisle. (c) is an image of a man walking on the beach.



(a) Input Image

(b) Estimated Opticalflow

(c) Opticalflow

Fig. 5. Flow visualization for Fig. 4. (a) is input image, (b) is opticalflow estimated by CNN$_{estimate}$, (c) is opticalflow calculated by two consecutive images. Following [10], we use same color coding to visualize flow for best viewed.

IV. CONCLUSIONS

In this paper, we proposed a novel method for image caption generation that focuses in incorporating motion estimation from images. We demonstrated that our proposed method using motion features improved performance on caption generation and that the quality of motion features was important to generate accurate captions.

Our future work will focus on how to better incorporate motion features and evaluating with other image captioning datasets.

ACKNOWLEDGMENT

This research was conducted using the SGI Rackable C2112-4GP3/C1102-GP8 (Reedbush-U/H/L) in the Information Technology Center, The University of Tokyo.

REFERENCES

[1] 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. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018.

[2] Oriol Vinyals, Alexander Toshev, Samy Bengio and Erhan. Show and Tell: A Neural Image Caption Generator. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2015.

[3] Andrej Karpathy and Li Fei-Fei. Deep Visual-Semantic Alignments for Generating Image Descriptions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3128–3137, 2015.

[4] Cheng Wang, Haojin Yang, Christian Bartz and Christoph Meinel. Image Captioning with Deep Bidirectional LSTMs. *Proceedings of the ACM International Conference on Multimedia*, pages 988–997, 2016.

[5] Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron-Courville, Ruslan Salakhutdinov, Richard S. Zemel and Yoshua Bengio. Show, Attend and Tell: Neural Image Caption Generation with Visual Attention. *Proceedings of the International Conference on Machine Learning*, pages 2048–2057, 2015.

[6] Jiasen Lu, Caiming Xiong, Devi Parikh and Richard Socher. Knowing When to Look: Adaptive Attention via A Visual Sentinel for Image Captioning. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 375–383, 2017.

[7] Zoe Kourtzi and Nancy Kanwisher. Activation in Human MT/MST by Static Images with Implied Motion. *Journal of Cognitive Neuroscience*, 12(1), pages 48–55, 2000.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren and Jian Sun. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[9] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. Imagenet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115(3), pages 211–252, 2015.

[10] Ruohan Gao, Bo Xiong and Kristen Grauman. Im2Flow: Motion Hallucination from Static Images for Action Recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5937–5947, 2018.

[11] Jun Xu, Tao Mei, Ting Yao and Yong Rui. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, pages 5288–5296, 2016.

[12] Tak-Wai Hui, Xiaoou Tang and Chen Change. A Lightweight Optical flow CNN-revisiting Data Fidelity and Regularization. *arXiv preprint arXiv:1903.07414*, 2019.

[13] Kishore Papineni, Salim Roukos, Todd Ward and Wei-Jing Zhu. BLEU: A Method for Automatic Evaluation of Machine Translation. *Proceedings of the Annual Meeting on Association for Computational Linguistics*, pages 311–318, 2002.

[14] Michael Denkowski and Alon Lavie. Meteor Universal: Language Specific Translation Evaluation for Any Target Language. *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, pages 376–380, 2014.

[15] Chin-Yew Lin, Guihong Cao, Jianfeng Gao and Jian-Yun Nie. An Information-theoretic Approach to Automatic Evaluation of Summaries. *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter Association for Computa-tional Linguistics*, pages 463–470, 2006.

[16] Ramakrishna Vedantam, C. Lawrence Zitnick and Devi Parikh. Consensus-based Image Description Evaluation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4566–4575, 2015.