Kindai University and Kobe University at TRECVID 2019 AVS Task

Kimiaki Shirahama*, Daichi Sakurai*, Takashi Matsubara† Kuniaki Uehara†
* Department of Informatics, Kindai University
shirahama@info.kindai.ac.jp, sakurai.daichi@kindai.ac.jp
† Graduate School of System Informatics, Kobe University
matsubara@ai.cs.kobe-u.ac.jp, uehara@kobe-u.ac.jp

***Abstract*—This paper presents our system developed for Ad-hoc Video Search (AVS) task in TRECVID 2019. Our system is based on *embedding* that maps visual and textual information into a common space to measure the relevance of each shot to a topic. We devise three embedding models built on two sources of training data, MS-COCO [1] and Flickr 30k [2]. Image feature extractors and region detector internally used in these models are pre-trained on ImageNet [3] and Visual Genome [4], respectively. The following five variants of our system were submitted:**

1) ***F_M_C_D_kindai_kobe.19_1*: This run is an ensemble of three embedding models. The first and second models are respectively trained on MS-COCO and Flickr 30k to perform different coarse-grained embeddings between frames and a topic. The last model forms fine-grained embedding between regions in frames and words in a topic.**
2) ***F_M_C_D_kindai_kobe.19_2*: This run is the same to *F_M_C_D_kindai_kobe.19_1* except that the fine-grained embedding model normalises regional features.**
3) ***F_M_C_D_kindai_kobe.19_3*: This run only uses the fine-grained embedding model without the normalisation.**
4) ***F_M_C_D_kindai_kobe.19_4*: This run is an ensemble of only the two coarse-grained embedding models.**
5) ***F_M_N_D_kindai_kobe.19_5*: This run is the same to *F_M_C_D_kindai_kobe.19_3* except the fine-grained embedding model using the normalisation.**

The MAPs of *F_M_C_D_kindai_kobe.19_3*, *F_M_C_D_kindai_kobe.19_4* and *F_M_N_D_kindai_kobe.19_5* are 0.080, 0.059 and 0.081, respectively. This indicates that fine-grained embedding is much more effective than coarse-grained one. Considering that both *F_M_C_D_kindai_kobe.19_1*'s and *F_M_C_D_kindai_kobe.19_2*'s MAPs are 0.087, the ensemble of coarse-grained and fine-grained embeddings leads to small performance improvements, compared to *F_M_C_D_kindai_kobe.19_3* and *F_M_C_D_kindai_kobe.19_5*. This means that the performances of the former are mainly owing to the latter. Finally, *F_M_C_D_kindai_kobe.19_1* and *F_M_C_D_kindai_kobe.19_2* are ranked at the fifth position in terms of teams participating in the fully automatic category, and our runs achieve the best MAPs for three topics in this category.

I. Introduction

We are continuously participating in TRECVID to make an objective performance comparison between our system and systems developed all over the world [5]. Until last year our system for Ad-hoc Video Search (AVS) task was based on the *concept-based approach* that retrieves shots based on detection results of concepts related to a topic [6], [7], [8]. Two crucial processes in this approach are "concept selection" and "result fusion". The former selects concepts related to a topic, and the latter generates the final retrieval result by aggregating detection results of the selected concepts. However, it is difficult to devise concept selection and result fusion processes that are universally applicable for various topics. Some reasons are as follows: For concept selection, it is needed to consider concepts implicitly related to a topic and errors in concept detection. For result fusion, different approaches are needed depending on the relation among concepts and sentence structure of a topic.

To avoid such problematic concept selection and score fusion, this year our system is based on *embedding* that maps data in different modalities into a common space, so that their similarity can be directly computed. In our case, visual features of a shot and textual features of a topic's description are projected into a common space, where their similarity is used as the relevance score for the retrieval process. In other words, shots that have the highest similarities to the topic in the common space are retrieved.

In particular, two types of embeddings are considered in our system. The first is *coarse-grained* embedding that maps frames in a shot and the textual description of a topic into a common space [9]. This is useful for evaluating the overall relevance of the shot to the topic. However, coarse-grained embedding loses details in the shot and topic because it aims to find rough correspondences between shots and topics in the common space. Thus, we adopt as the second type *fine-grained* embedding that builds a common space to characterise correspondences between regions in frames of a shot and words in the textual description of a topic [10]. This is useful for examining whether the shot satisfies detailed requirements of the topic, such as object numbers, object types (e.g., male or female) and object characterisitcs (e.g., colour and pose). Finally, our system fuses the results by coarse-grained and fine-grained embeddings into a single result, where a shot and topic are matched based on both of their overall and detailed characteristics.

II. Our AVS System

This section presents our AVS system that first generates retrieval results by independently employing coarse-grained and fine-grained embeddings. Then, these results are fused into the final result where, for each shot, the relevance scores

computed by coarse-grained and fine-grained embeddings are simply summed up. Each of embeddings is detailed below.

A. Coarse-grained Embedding

Fig. 1 summarises our coarse-grained embedding method that is based on *Visual-Semantic Embedding* (VSE++) proposed in [9]. First of all, VSE++ is trained on a dataset consisting of image-caption pairs, each of which represents an image and caption that are relevant to each other. This kind of pair is called *positive pairs*, while every pair of image and caption that are irrelevant to each other is called *negative pair*. After training of VSE++ is finished, it is applied to frames in a shot and the textual description of a topic.

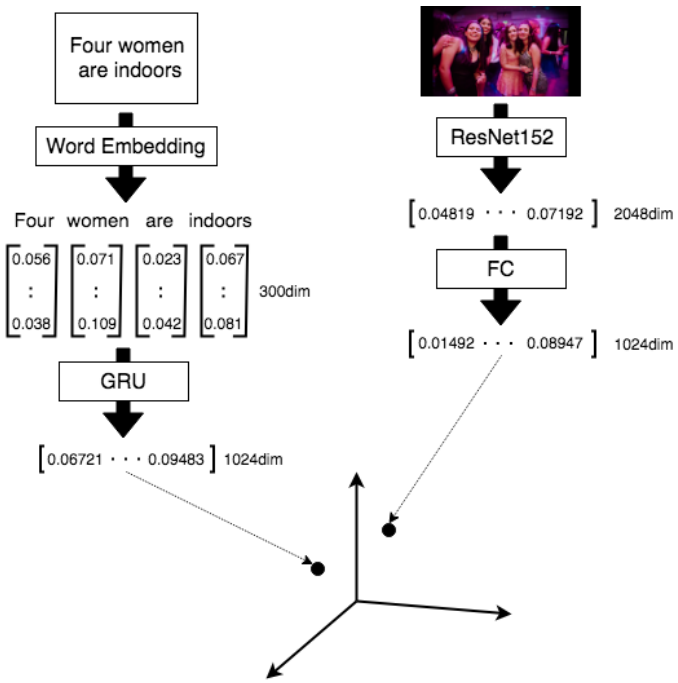

Fig. 1. An overview of our coarse-grained embedding method VSE++

As depicted in the top-right of Fig. 1, ResNet152 [11] trained on ImageNet [3] is used to extract a 2048-dimensional image feature from an image. Then, this feature is transformed into a 1024-dimensional feature via a Fully-Connected (FC) layer. Also, the top-left of Fig. 1 illustrates that each word in a text caption is encoded into a 300-dimensional feature using a word embedding layer. Subsequently, Gated Reccurrent Unit (GRU) is used to aggregate such features for all words into a single 1024-dimensional feature by considering the order of these words. In this framework, VSE++ optimises the FC layer in the image side and the word embedding layer and GRU in the text side, so that similar 1024-dimensional features are generated for images and captions in positive pairs, while producing dissimlar features for negative pairs. That is, images and captions in positive pairs are close to each other in the 1024-dimensinal common space, while those in negative pairs are distant from each other.

In the optimisation described above, VSE++ uses a triplet loss function to pay special attention to "hard" negative pairs each including an image and caption, which are irrelevant to each other but are located closely in the common space. VSE++ attempts to make images and captions in such hard negative pairs projected distant from each other in the common space. This is very useful for the retrieval process because images in hard negative pairs are nothing except for false positives, and frames in shots similar to those false positives become to be ranked at lower positions in a retrieval result.

B. Fine-grained Embedding

Stacked Cross Attention Network (SCAN) proposed in [10] is adopted for fine-grained embedding. Fig. 2 illustrates an overview of SCAN which is firstly trained on a dataset containing positive and negative pairs, and then used for shot frames and topic's textual description, similar to VSE++. As shown in the top-right of Fig. 2, an image is analysed to extract k "salient regions" ($k = 36$ in our case), each of which is likely to include a concept with an attribute, such as "blue water", "black hair", or "floral dress". To extract salient regions, we employ the bottom-up attention model [12] that is implemented with Faster R-CNN based on ResNet101 backbone and trained on Visual Genome dataset [4]. As a result, each of k regions is represented by a 2048-dimensional feature and transformed into a 1024-dimensional feature via an FC layer. On the other hand, the top-left of Fig. 2 shows the extraction of a 1024-dimensional feature for each word in a caption [10]: First, each word is encoded into a 300-dimensional feature using a word embedding layer, and then is represented by two 1024-dimensional features using a bi-directional GRU. Here, the first and second features are obtained by the forward and backward GRUs in the bi-directional GRU, respectively. The final feature for the word is computed as their average.

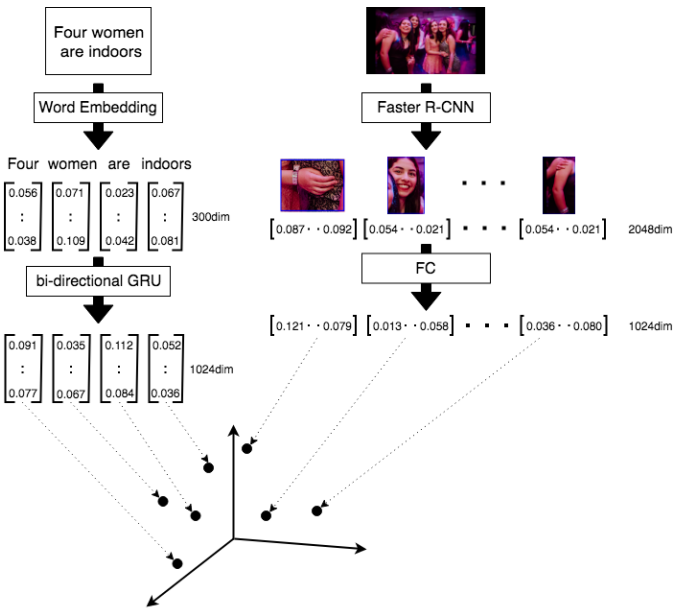

Fig. 2. An overview of our fine-grained embedding method SCAN

After the aforementioned feature extractions, n words in a caption C and k regions in an image I are now projected into the 1024-dimensional common space. SCAN aims to probabilistically make correspondences between words and regions through the *attention* mechanism where a pair of word and region relevant to each other gets a high attention. Specifically, the attention α_{ij} between the i-th region ($1 \leq i \leq k$) and the j-th word ($1 \leq j \leq n$) is computed as follows:

$$\alpha_{ij} = \frac{\exp(\lambda_1 s_{ij})}{\sum_{i=1}^{k} \exp(\lambda_1 s_{ij})} \tag{1}$$

where s_{ij} is the normalised cosine similarity between the feature e_j of the j-th word and the feature v_i of the i-th region (see [10] for more details). And, λ_1 is a hyperparameter to control the balance of attentions over k regions. Eq. (1) represents a probability of how relevant the i-th region is to the j-th word. Then, the "word-level" relevance of how suitable k regions in I are for the j-th word is defined as follows:

$$R(e_j, a_j^v) = \frac{e_j^T a_j^v}{||e_j|| ||a_j||} \text{ where } a_j^v = \sum_{i=1}^{k} \alpha_{ij} v_i \tag{2}$$

That is, a_j^v is the average of regional features weighted by their attentions to the j-th word. The word-level relevance $R(e_j, a_j^v)$ is calculated as the cosine similarity between a_j^v and e_j. The final relevance $R(C, I)$ between C and I is obtained by LogSumExp pooling below:

$$R(C, I) = \log \left(\sum_{j=1}^{n} \exp \left(\lambda_2 R(e_j, a_j^v) \right) \right)^{(1/\lambda_2)} \tag{3}$$

where λ_2 is a hyperparameter to control the balance of word-level relevances for n words in C. SCAN optimises the FC layer in the image side and the word embedding layer and bi-directional GRU in the caption side, so that $R(C, I)$s for positive pairs and those for negative pairs become high and low, respectively. This accordingly allows the attention mechanism to estimate high α_{ij} if the i-th region is relevant to the j-th word, and low α_{ij} for the other region-word pairs.

III. Experiments

This section provides the experimental results of our embedding-based AVS system. Some implementation details are firstly described, and then the results and a discussion about them are given.

A. Implementation Details

For coarse-grained embedding, two types of VSE++ are used, the one named "VSE++M" is trained on MS-COCO dataset [1] and the other termed "VSE++F" is built on Flickr 30k dataset [2]. For each shot, VSE++M and VSE++F are applied to the NIST-provided keyframe and 10 frames that are equi-distantly sampled over time. Assuming the case of VSE++M, it is used to encode the keyframe and 10 frames into 1024-dimensional features, which are subsequently aggregated into a single 1024-dimensional feature by average pooling. Meanwhile, using VSE++M, the textual description of a topic is encoded into a 1024-dimensional feature. Afterwords, shots are ranked by computing the cosine similarity between the feature of each shot and the one of the topic. Finally, the retrieval result by VSE++M is obtained as a set of 1000 shots that have the highest similarities to the topic. The same procedure is used to obtain the retrieval result by VSE++F. Lastly, the final result by coarse-grained embedding is obtained by fusing the results with VSE++M and VSE++F. Here, the overall similarity (relevance) of each shot to the topic is simply the sum of the cosine similarities computed in VSE++M and VSE++F.

For fine-grained embedding, SCAN is trained on MS-COCO dataset [1]. Because of its expensive computational cost, only the keyframe of each shot is processed. Furthermore, due to the GPU memory limitation, salient regions are extracted from each keyframe that is scaled to make the longer side (width or height) 480 pixels. SCAN is used to make correspondences between words in the text description of a topic and salient regions in the keyframe of a shot. Then, the relevance of the shot to the topic is computed using Eq. (3). Shots are ranked based on their relevances and 1000 shots with the highest relevances form a retrieval result. Finally, the fusion of retrieval results by coarse-grained and fine-grained emebeddings is done by simply summing the relevances computed by them.

Based on the above-mentioned fusion approaches, the five submitted runs are configured as follows:

1) *F_M_C_D_kindai_kobe.19_1* is an ensemble of VSE++M, VSE++F and SCAN.
2) *F_M_C_D_kindai_kobe.19_2* is an ensemble of VSE++M, VSE++F, and SCAN where the feature v_i of the i-th region is L2-normalised.
3) *F_M_C_D_kindai_kobe.19_3* is comprised only of SCAN to examine the effectiveness of fusing it with coarse-grained embedding (VSE++M and VSE++F).
4) *F_M_C_D_kindai_kobe.19_4* is an ensemble of VSE++M and VSE++F to compare the performance of coarse-grained embedding to the one of fine-grained embedding (SCAN).
5) *F_M_N_D_kindai_kobe.19_5* is comprised only of SCAN with L2-normalisation, and its purpose is the same to *F_M_C_D_kindai_kobe.19_3*.

B. Results

Fig. 3 shows the ranking of 37 AVS methods developed in the fully automatic category. Each bar represents the MAP of one method, and the five red bars represent the MAPs of our submitted runs. As can be seen from Fig. 3, *F_M_C_D_kindai_kobe.19_1* and *F_M_C_D_kindai_kobe.19_2* are ranked at the 14-th and 15-th positions, respectively (both runs actually get the same MAPs 0.087). These positions correspond to the fifth position among nine teams participating in the fully automatic category. In Fig. 3, the MAPs of *F_M_C_D_kindai_kobe.19_3*, *F_M_C_D_kindai_kobe.19_4* and *F_M_N_D_kindai_kobe.19_5* are 0.080, 0.059 and 0.081, respectively. This indicates that fine-grained embedding based on SCAN (either using or not-using L2-normalisation)

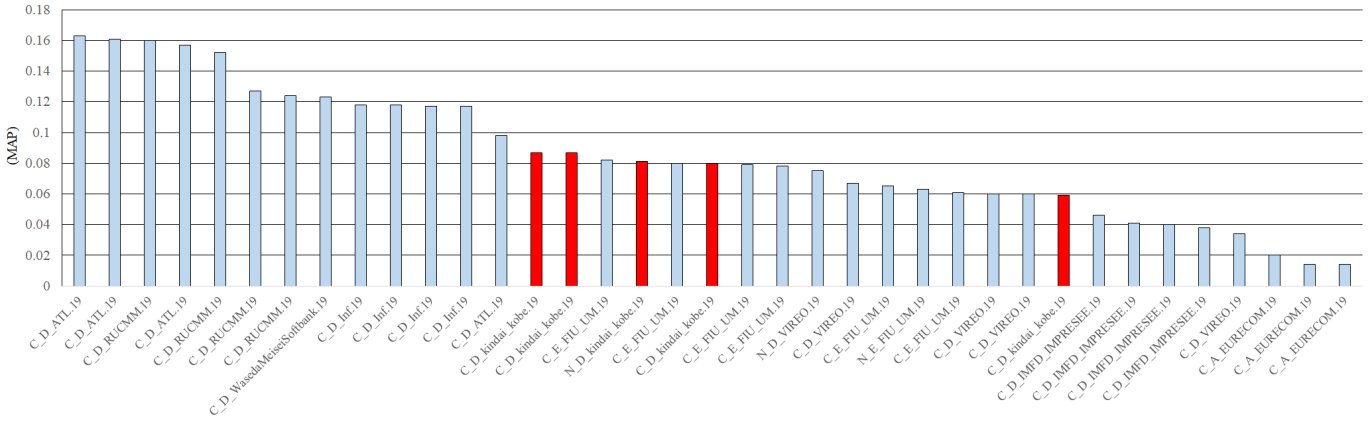

Fig. 3. Ranking of AVS methods developed in the fully automatic category with respect to their MAPs.

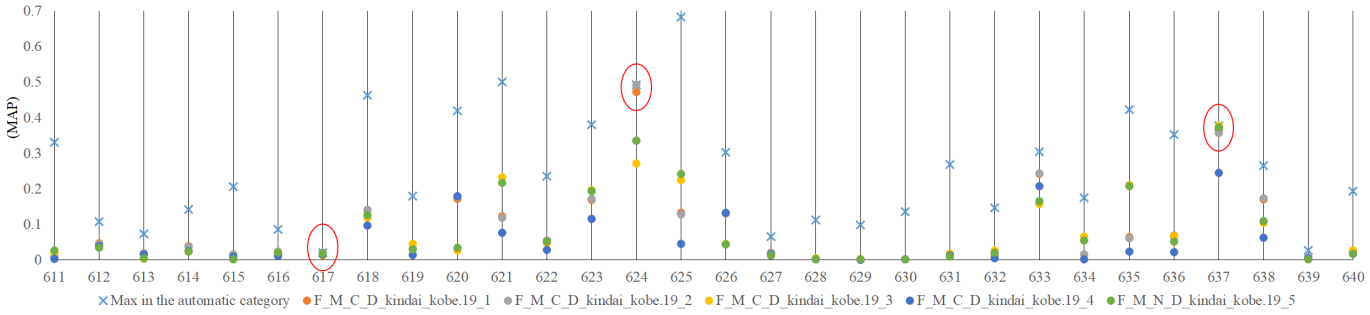

Fig. 4. Comparison between the best AP and APs of our submitted runs for each of 30 topics in the fully automatic category.

is much more effective than coarse-grained embedding based on the ensemble of VSE++M and VSE++F[1]. One characteristic example showing the superiority of fine-grained embedding over coarse-grained one is topic 636 "Find shots of a man and a baby both visible", for which the former appropriately retrieves shots including both of man's and baby's appearances, while the latter retrieves many shots only including man's or baby's appearance. In addition, *F_M_C_D_kindai_kobe.19_1*'s and *F_M_C_D_kindai_kobe.19_2*'s MAPs (both are 0.087) demonstrate the effectiveness of an ensemble of coarse-grained and fine-grained embeedings. But, the improvement is only 0.006 compared to *F_M_N_D_kindai_kobe.19_5*'s MAP 0.081. This reveals the significant contribution of fine-grained embedding to the ensemble result.

Finally, for each of 30 topics, Fig. 4 shows the comparison between the best AP marked by cross and the APs of our five submitted runs marked by circle. The three red circles indicates that our submitted runs achieve the best APs for the corresponding three topics.

IV. Conclusion and Future Work

This paper introduced our AVS system that is an ensemble of coarse-grained embedding VES++ and fine-grained embedding SCAN. The results show the effectiveness of the ensemble, to which fine-grained embedding has a much more contribution than coarse-grained embedding. This is also supported by their individual performances, that is, fine-grained embedding is much more accurate than coarse-grained one.

Based on the obtained results, we plan to put a more focus on fine-grained embedding in the future. Specifically, this embedding currently takes very long time (about 13.3 hours) to finish the retrieval for a tpoic. One reason is that matching between regions and words is exhaustively performed for all shots, most of which are clearly irrelevant to the topic. Thus, we will develop a fast matching method that efficiently filters out many irrelevant shots by considering the structure of the topic's textual description and the relation among regions. This may also lead to a performance improvement, because general meanings like phrases can be matched with regions. Another future work will address how to deal with the "out-of-vocabulary" problem. Words in textual descriptions of some topics do not exist in neither MS-COCO's nor Flickr 30k's vocabulary. Only regarding such words as "unknown words" yields a poor retrieval performance. To overcome this, we plan to exploit Web images annotated with captions including unknown words.

[1]Our preliminary experiment showed that the ensemble of VSE++M and VSE++F is better than only using VSE++M or VSE++F.

References

[1] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in

context," in *Proc. of ECCV 2014*, 2014, pp. 740–755.
[2] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier, "From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions," in *Proc. of ACL 2014*, 2014, pp. 67–78.
[3] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
[4] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International Journal of Computer Vision*, vol. 123, no. 1, pp. 32–73, 2017.
[5] G. Awad, A. Butt, K. Curtis, Y. Lee, J. Fiscus, A. Godil, A. Delgado, A. F. Smeaton, Y. Graham, W. Kraaij, and G. Quénot, "Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval," in *Proceedings of TRECVID 2019*. NIST, USA, 2019.
[6] Y. Matsumoto, T. Shinozaki, K. Shirahama, M. Grzegorzek, and K. Uehara, "Kobe university, NICT and university of siegen at TRECVID 2016 avs task," in *Proc. of TRECVID 2016*, 2016.
[7] Z. He, T. Shinozaki, K. Shirahama, M. Grzegorzekz, and K. Uehara, "Kobe university, NICT and university of siegen at TRECVID 2017 avs task," in *Proc. of TRECVID 2017*, 2017.
[8] K. Shirahama, Z. He, and K. Uehara, "Kobe university and kindai university at TRECVID 2018 avs task," in *Proc. of TRECVID 2018*, 2018.
[9] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler, "Vse++: Improving visual-semantic embeddings with hard negatives," in *Proc. of BMVC 2018*, 2018.
[10] K.-H. Lee, X. Chen, G. Hua, H. Hu, and X. He, "Stacked cross attention for image-text matching," in *Proc. of ECCV 2018*, 2018, pp. 212–228.
[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. of CVPR 2016*, 2016, pp. 770–778.
[12] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proc. of CVPR 2018*, 2018, pp. 6077–6086.