Key-Word-Aware Network for Referring Expression Image Segmentation

Hengcan Shi, Hongliang Li, Fanman Meng, Qingbo Wu

School of Information and Communication Engineering,
University of Electronic Science and Technology of China
shihc@std.uestc.edu.cn, {hlli,fmmeng,qbwu}@uestc.edu.cn

Abstract. Referring expression image segmentation aims to segment out the object referred by a natural language query expression. Without considering the specific properties of visual and textual information, existing works usually deal with this task by directly feeding a foreground/background classifier with cascaded image and text features, which are extracted from each image region and the whole query, respectively. On the one hand, they ignore that each word in a query expression makes different contributions to identify the desired object, which requires a differential treatment in extracting text feature. On the other hand, the relationships of different image regions are not considered as well, even though they are greatly important to eliminate the undesired foreground object in accordance with specific query. To address aforementioned issues, in this paper, we propose a key-word-aware network, which contains a query attention model and a key-word-aware visual context model. In extracting text features, the query attention model attends to assign higher weights for the words which are more important for identifying object. Meanwhile, the key-word-aware visual context model describes the relationships among different image regions, according to corresponding query. Our proposed method outperforms state-of-the-art methods on two referring expression image segmentation databases.

Keywords: referring expression image segmentation, key word extraction, query attention, key-word-aware visual context

1 Introduction

Image segmentation expects to segment out objects of interest from an image, which is a fundamental step towards high-level vision tasks, such as object extraction [14, 23, 25], image captioning [21, 32, 34] and visual question answering [21, 22, 35]. This paper focuses on referring expression image segmentation, in which the objects of interest are referred by natural language expressions, as shown in Fig. 1. Beyond traditional semantic segmentation, referring expression image segmentation needs to analyze both the image and natural language, which is a more challenging task.

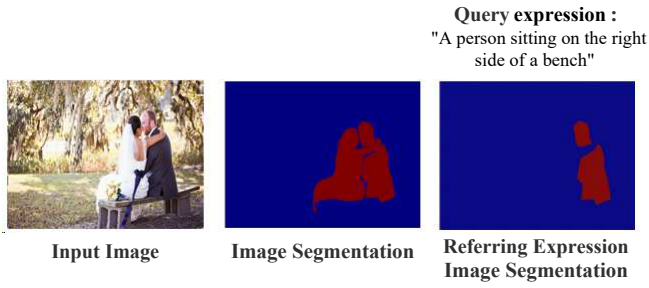

Fig. 1. Example of referring expression image segmentation task. Different from traditional image segmentation, referring expression image segmentation aims at segmenting out the object referred by a natural language query expression.

Previous works [9, 10, 18] formulate referring expression image segmentation task as a region-wise foreground/background classification problem. They combine each image region feature with whole query feature [9, 10] or every word feature [18] to classify the image region. However, each word in a query expression makes different contributions to identify the desired object, which requires a differential treatment in extracting text feature. Extracting key words is helpful to suppress the noise in the query and to highlight the desired objects. In addition, existing methods also ignore the visual context among different image regions. Visual context is important to localize and recognize objects. In Fig.1 we illustrate an example, which includes two foreground objects, i.e., the bride and the groom. It is clear that the groom is on the right side of the bench, which is important to match the query expression.

In this paper, we propose a key-word-aware network (KWAN) that extracts key words for each image region and models the key-word-aware visual context among multiple image regions in accordance with the natural language query. Firstly, we use a convolutional neural network (CNN) and a recurrent neural network (RNN) to encode the features of every image region and every word, respectively. Based on these features, we then find out the key words for each image region by a query attention model. Next, a key-word-aware visual context model is used to model the visual context among multiple image regions in accordance with corresponding key words. Finally, we classify each image region based on extracted visual features, key-word-aware visual context features and corresponding key word features. We verify the proposed method on the ReferItGame and Google-Ref datasets. The results show that our method outperforms previous state-of-the-art methods and achieves the best IoU and precision.

This paper is organized as follows. We introduce the related work in Section 2. In Section 3, we detail our proposed method for referring expression image segmentation. Experimental results are reported in Section 4 to validate the effectiveness of our method. Finally, Section 5 concludes this paper.

2 Related Work

In summary, there are three categories of works related with the task of this paper. The first is semantic segmentation, which is one of the most classic tasks in image segmentation and a foundation for referring expression image segmentation. The second is referring expression visual localization, which also needs to search object in an given image from natural language expressions. The third is referring expression image segmentation.

Semantic Segmentation. Semantic segmentation technologies have developed quickly in recent years, on which convolutional neural network (CNN)-based methods achieve state-of-the-art performance. The CNN-based semantic segmentation methods can be mainly divided into two types. The first is hybrid proposal-classifier models [1, 4–7], which first generate a number of proposals from the input image, and then segment out the foreground object in each proposal. The second is fully convolutional networks (FCNs) [2, 20, 27, 36], which segment the whole image end-to-end, without any pre-processing. Some methods [3, 15, 16, 19, 28, 39] leveraged visual context model to boost the semantic segmentation performance, which models the relationships among multiple image regions based on their spatial positions. Wang *et al.* [31] built an interaction between semantic segmentation and natural language. They extract an object relationship distribution from natural language descriptions, and then use the extracted distribution to constrain the object categories in semantic segmentation predictions. These semantic segmentation methods are foundations for referring expression image segmentation task.

Referring Expression Visual Localization. Referring expression visual localization expects to localize regions in an image from natural language expressions. The goal of this task is to find bounding boxes [11, 24, 26, 37, 38] or attention regions [21, 22, 32, 33, 35] referred by natural language queries. Methods in [11, 24, 26, 37, 38] first restored the natural language expressions from a number of pre-extracted proposals, and then took the proposal with the highest restoration score as the referred object. Methods proposed by [21, 22, 32, 33, 35] used visual attention models to measure the importance of each image region for image captioning [21, 32] or visual question answering [21, 22, 35] task. The most important regions were deemed as attention regions. The similarity between these localization methods and referring expression image segmentation methods is that they both need to find out objects referred by natural language queries. However, these localization methods only focus on generating bounding boxes or coarse attention maps, while referring expression image segmentation methods aim at obtaining fine segmentation masks.

Referring Expression Image Segmentation. Referring expression image segmentation have attracted increasing researchers' interest [9, 10, 18] in recent years. Beyond referring expression visual localization and semantic segmentation, referring expression image segmentation aims at generating fine segmentation masks from natural language queries. Hu *et al.* [9,10] combined the features of the natural language query and each image to determine whether the image region belongs to the referred object. Liu *et al.* [18] developed the referring ex-

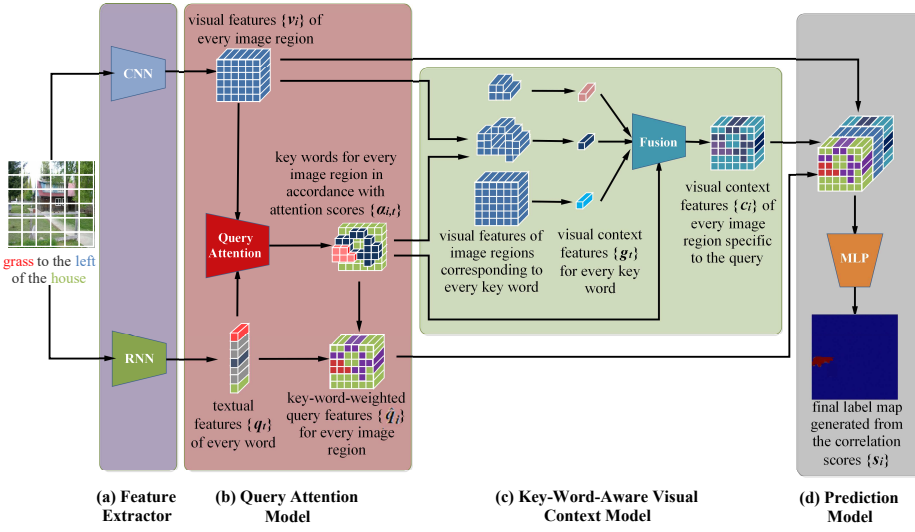

Fig. 2. Our proposed key-word-aware Network (KWAN) consists of four parts: (a) a CNN and an RNN that encode the features of every image region and every word in the nature language query, (b) a query attention model that extracts key words for each image region and use extracted key words to weight the original query, (c) a key-word-aware visual context model that models visual context based on corresponding key words, (d) a prediction model that predict the segmentation results based on visual features, key-word-aware visual context features and key-word-weight query features.

pression image segmentation technologies. Instead of directly using the feature of whole query, they concatenated the features of each word and each image region, and then used a multimodal LSTM to integrate these concatenated features. However, on one hand, these methods ignore that each word in a query makes different contributions to the segmentation. On the other hand, many queries need to compare multiple image regions, while these methods only separately tackle each image region. In contrast to previous methods, we propose a key-word-aware network, which extracts key words to suppress the noise in queries, and models key-word-aware visual context among multiple image regions to better localize and recognize objects.

3 Proposed Method

Overview. Given an image and a natural language query, our goal is to segment out the object referred by the query from the image. To this end, we propose a key-word-aware network (KWAN), which is composed of four parts as illustrated in Fig. 2. The first part is a feature extractor, which encodes features of the image and query. The second part is a query attention model, which extracts key words for each image region and leverages these key words to weight the query feature. The third part is a key-word-aware visual context model, which models

the visual context among multiple image regions based on the natural language query. The fourth part is a prediction model, which generates segmentation predictions based on the image features, the key-word-weighted query features and the key-word-aware visual context features. Below, we detail each part.

3.1 Image and Query Feature Extractor

The inputs in referring expression image segmentation task contain two parts: an image $I \in R^{H \times W \times C_{im}}$ and a natural language query $X \in R^{C_{text} \times T}$, where H and W are the height and width of the image, respectively; C_{im} is the number of image channels; T denotes the number of words in the query; and each word represented by an C_{text}-dimensional one-hot vector. We first use a convolutional neural network (CNN) to extract a feature map of the input image as follows:

$$\begin{aligned} F &= CNN(I) \\ &= \{f_1, f_2, ..., f_{hw}\} \end{aligned} \tag{1}$$

where $F \in R^{h \times w \times C_f}$ is the extracted feature map; h and w are the height and width of feature map, respectively; and C_f is the feature dimension. In the feature map F, each feature vector $f_i \in R^{C_f}$ encodes the appearance and semantic information of the i-th image region.

Since the referring expression image segmentation task also needs spatial position information, we extract a position feature from the spatial coordinates of the i-th image region:

$$p_i = [x_i, y_i] \tag{2}$$

where $p_i \in R^2$ is the position feature of the i-th image region, which is concatenated by the normalized horizontal and vertical coordinates x_i and y_i. The operator $[\cdot, \cdot]$ represents the concatenation of features. Therefore, the final visual feature of the i-th image region can be obtained as follows:

$$v_i = [f_i, p_i] \tag{3}$$

where $v_i \in R^{Cv}$ is a Cv-dimensional visual feature vector of the i-th image region, and $Cv = C_f + 2$. The visual feature contains appearance, semantic and spatial position information of the image region.

We use a recurrent neural network (RNN) to encode the feature of natural language query X as follows:

$$\begin{aligned} Q &= RNN(W_e X) \\ &= \{q_1, q_2, ..., q_T\} \end{aligned} \tag{4}$$

where $Q \in R^{C_q \times T}$ is the encoded feature matrix of the query X, in which each feature vector $q_t \in R^{C_q}$ encodes the textual semantic and contextual information for the t-th word. $W_e \in R^{C_e \times C_{text}}$ is a word embedding matrix to reduce the dimensionality of the word features.

3.2 Query Attention Model

After the feature encoding, we then extract key words by a query attention model. For the i-th image region, the query attention can be captured as follows:

$$z_{i,t} = w_z^T tanh(W_q q_t + W_v v_i) \tag{5}$$

$$\alpha_{i,t} = \frac{exp(z_{i,t})}{\sum_{r=1}^{T} exp(z_{i,r})} \tag{6}$$

where $W_q \in R^{C_z \times C_q}$, $W_v \in R^{C_z \times C_v}$ and $w_z \in R^{C_z}$ are parameters in query attention model; $\alpha_{i,t} \in [0, 1]$ is the query attention score of the t-th word for the i-th image region, and $\sum_{t=1}^{T} \alpha_{i,t} = 1$. A high score $\alpha_{i,t}$ means that the t-th word is important for i-th image region, i.e., word t is a key word for image region i.

Based on the learned query attention scores, the feature of query can be weighted as follows:

$$\hat{q}_i = \sum_{t=1}^{T} \alpha_{i,t} q_t \tag{7}$$

where $\hat{q}_i \in R^{C_q}$ is the weighted query feature for the i-th image region. In the weighted query feature, words are no longer equally important. Key words make more important contributions.

3.3 Key-Word-Aware Visual Context Model

The key-word-aware visual context model learns the context among multiple image regions for the natural language query. Towards this goal, we first aggregate the visual messages of image regions for each key word:

$$m_t = \begin{cases} \frac{\sum_{i=1}^{hw} v_i u(\alpha_{i,t} - Thr)}{\sum_{i=1}^{hw} u(\alpha_{i,t} - Thr)}, & \max\limits_{i=1,\ldots,hw}(\alpha_{i,t}) \geq Thr \\ \mathbf{0}, & otherwise \end{cases} \tag{8}$$

where $m_t \in R^{C_v}$ is the aggregated visual feature vector, and $u(\cdot)$ represents an unit step function. Thr is a threshold to select out the key word. $\alpha_{i,t} \geq Thr$ implies that the t-th word is a key word for the i-th image region. If the t-th word is a key word for at least one image region (i.e., $\max_{i=1,\ldots,hw}(\alpha_{i,t}) \geq Thr$), we average the visual features of image regions which take this word as a key word. Otherwise, the t-th word is a non-key word for whole image, hence the aggregated visual feature m_t is $\mathbf{0}$. The threshold Thr is set to $1/T$, since $\sum_{t=1}^{T} \alpha_{i,t} = 1$.

Based on the aggregated visual messages, we then use a fully-connected layer to learn visual context:

$$g_t = ReLU(W_g m_t + b_g) \tag{9}$$

where $g_t \in R^{C_g}$ is the learned visual context feature specific to the t-th word, $W_g \in R^{C_g \times C_v}$ and $b_g \in R^{C_g}$ are the parameters in the fully-connected layer, and ReLU denotes the rectified linear unit activation function.

Finally, we fuse the visual context features specific to each key words into the one specific to whole query as follows:

$$c_i = \sum_{t=1}^{T} g_t u(\alpha_{i,t} - Thr) \tag{10}$$

where $c_i \in R^{C_g}$ is the fused visual context feature specific to the query for the i-th image region.

3.4 Prediction Model and Loss Function

Once we extract the visual feature v_i, the key-word-weighted query feature $\hat{q}_i$ and the key-word-aware visual context feature c_i, a correlation score between the query and each image region can be obtained as follows:

$$s_i = sigmoid(MLP([\hat{q}_i, v_i, c_i])) \tag{11}$$

where MLP denotes a multi-layer perceptron, and sigmoid function are used to normalize the score. $s_i \in (0, 1)$ is the normalized correlation score between i-th image region and the natural language query. A high correlation score means that current image region is highly correlative with the query, i.e., this image region is belong to referred foreground object.

Scores of all image regions together form a label map. We upsample the label map into original image size as the segmentation result. A pixel-wise cross entropy loss is used to constrain the training:

$$\begin{aligned} Loss = -\frac{1}{N}\sum_{n=1}^{N}\frac{1}{H^{(n)}W^{(n)}}\sum_{j=1}^{H^{(n)}W^{(n)}} & [y_j^{(n)} \times logs_j^{(n)} \\ & + (1 - y_j^{(n)}) \times log(1 - s_j^{(n)})] \end{aligned} \tag{12}$$

where N is the number of images in total training set; $H^{(n)}$ and $W^{(n)}$ are the height and width of the n-th image, respectively; $s_j^{(n)}$ denotes the correlation score of the j-th pixel in the n-th image; and $y_j^{(n)} \in \{0, 1\}$ is the label indicating whether pixel j belongs to referred object.

4 Experiments

We conduct experiments to evaluate our method on two challenging referring expression image segmentation datasets, including the ReferItGame dataset and the Google-Ref dataset. Objective and subjective results are reported in this section.

Evaluation Metrics. We adopt two typical image segmentation metrics: the intersection-over-union (IoU) and the precision (Pr). The IoU is a ratio between

intersection and union areas of segmentation results and ground truth. The precision is the percentage of correctly segmented objects in the total dataset. The correctly segmented objects are defined as objects whose IoU passes a pre-set threshold. We use five different thresholds in experiments: 0.5, 0.6, 0.7, 0.8, 0.9. The precisions with these thresholds are represented by Pr@0.5, Pr@0.6, Pr@0.7, Pr@0.8, Pr@0.9, respectively.

Implementation Details. The proposed method can be implemented with any CNN and RNN. Since state-of-the-art methods [9, 18] often choose VGG16 [30] or Deeplab101 [2] as their CNN and use LSTM [8] as their RNN, to fairly compare our method with them, we also implement the proposed method with these CNN and RNN in our experiments. The dimensions of CNN and RNN features are both set to 1000 (i.e., $C_f = C_q = 1000$). The maximum number T of words in a query is 20, thus the key word threshold Thr in the key-word-aware visual context model is set to 0.05 (i.e., $1/T$). We train the proposed method in two stages. The first stage is low resolution training. In this stage, the predictions are not upsampled, and the loss is calculated with down-sampled ground truth. The second stage is high resolution training, in which the predictions are upsampled into the original image size. The model is trained with Adaptive Moment Estimation (Adam) in all stages, and the learning rate is set to 0.0001. We initialize the CNN from the weights pre-trained on ImageNet dataset [29], and initialize other parts from random weights. All experiments are conducted based on the Caffe [12] toolbox on a single Nvidia GTX Titan X GPU with 12G memory.

Table 1. Comparison with state-of-the-art methods on the ReferItGame testing.

Method	IoU	Pr@0.5	Pr@0.6	Pr@0.7	Pr@0.8	Pr@0.9
VGG16						
[9]	48.03%	34.02%	26.71%	19.32%	11.63%	3.92%
[18]	48.84%	**35.79%**	27.53%	20.90%	11.72%	3.83%
Ours	**52.19%**	35.61%	**28.50%**	**21.85%**	**12.87%**	**4.83%**
Deeplab101						
[9]	56.83%	43.86%	35.75%	26.65%	16.75%	6.47%
[18]	57.34%	44.33%	36.13%	27.20%	16.99%	6.43%
Ours	**59.09%**	**45.87%**	**39.80%**	**32.82%**	**23.81%**	**11.79%**

4.1 Results on ReferItGame Dataset

The ReferItGame dataset [13] is a public dataset, with 20000 natural images and 130525 natural language expressions. Totalling 96654 foreground regions are referred by these expressions, which contain not only objects but also stuff, such as *snow*, *mountain* and so on. The dataset are split into training, validation and testing sets, containing 9000, 1000, and 10000 images, respectively. Similar

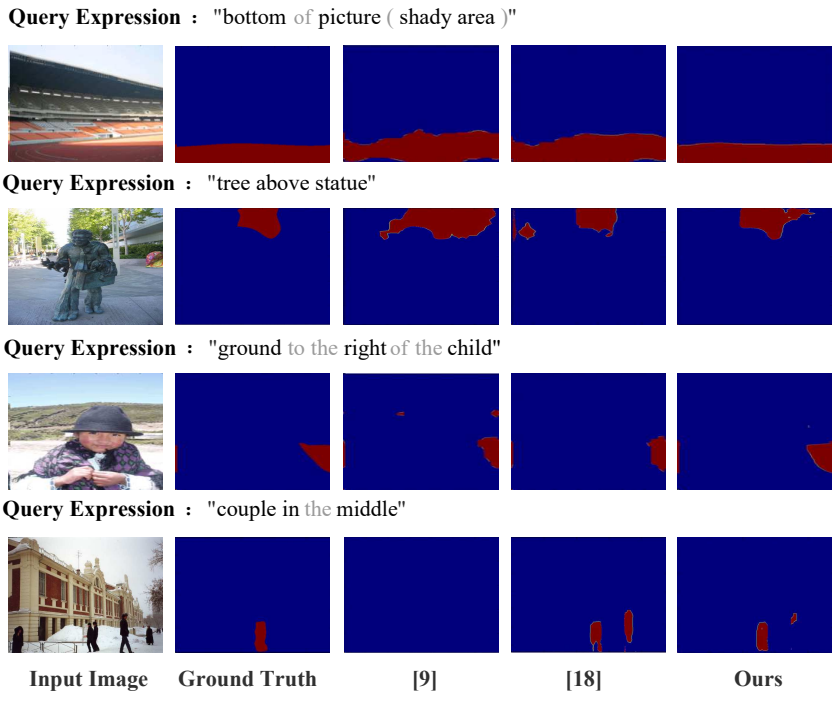

Fig. 3. Referring expression image segmentation results on the ReferItGame testing. Left to right: input images, ground truth, the segmentation results from [9], [18] and our method, respectively. All methods are implemented with *Deeplab101*. In query expressions, the black words mean key words our method predicted for foreground regions (red regions).

to [9, 18], we use training and validation sets to train, and use testing set to test our method.

The results are summarized in Table 1. All methods do not use additional training data and post processing like CRF. State-of-the-art methods in [9, 18] equally deal with every word in the natural language expressions and do not take into account the visual context. It can be observed from Table 1 that our proposed method outperforms these methods in terms of both IoU and precision, whether implemented with VGG16 or Deeplab101. Moreover, under the precision metric, with higher thresholds, our method achieves more improvements. This superior performance demonstrates the effectiveness of selectively extracting key words for every image region and modeling the key-word-aware visual context.

We depict some subjective referring expression image segmentation results on the ReferItGame dataset in Fig. 3. From the first and third images in Fig. 3, it can be seen that existing methods do not well segment out some objects when the query expression is too long or contains some noise, such as round brackets. Our method selects key words and filters out useless information in

the query, therefore can successfully segment out the referred objects in these images. Moreover, it can be observed that previous method localize and segment some desired objects wrongly when the query needs to compare multiple objects, such as the second and fourth images in Fig. 3. A major reason is that previous methods ignore the visual context among objects. Our method can generate better segmentation results by modeling the key-word-aware visual context.

Table 2. Comparison with state-of-the-art methods on the Google-Ref validation.

Method	IoU	Pr@0.5	Pr@0.6	Pr@0.7	Pr@0.8	Pr@0.9
VGG16						
[9]	28.14%	15.25%	8.37%	3.75%	1.29%	0.06%
[18]	28.60%	16.70%	8.77%	4.96%	1.79%	0.38%
Ours	**31.36%**	**17.71%**	**11.12%**	**7.90%**	**3.69%**	**1.07%**
Deeplab101						
[9]	33.08%	25.66%	18.23%	10.82%	4.17%	0.64%
[18]	34.40%	26.19%	18.46%	10.68%	4.28%	0.73%
Ours	**36.92%**	**27.85%**	**21.01%**	**13.42%**	**6.60%**	**1.97%**

4.2 Results on Google-Ref Dataset

The Google-Ref dataset [24] contains 26711 natural images with 54822 objects extracted from the MS COCO dataset [17]. There are 104560 expressions referring to these objects, and the average length of these expressions is longer than that in the ReferItGame dataset. We use the split from [24], which chose 44822 and 5000 objects for training and validation, respectively.

The objective and subjective results are shown in Table 2 and Fig. 4, respectively. From Table 2, it can be seen that our method outperforms previous methods under the both two metrics, IoU and precision. This demonstrates the effectiveness of our method. From Fig. 4, it can be observed that previous methods fail to segment some objects when the queries are too long, such as the first and second images in Fig. 4. In addition, previous methods find some wrong object instances when the queries need to compare different instances with the same class, such as the third and fourth images in Fig. 4. The proposed method can successfully segment out these objects, benefiting from the key word extraction and the key-word-aware visual context.

4.3 Discussion

Ablation Study. To verify the effectiveness of each part in our method, a number of ablation studies are conducted on the ReferItGame dataset. We compare five different models as follows:

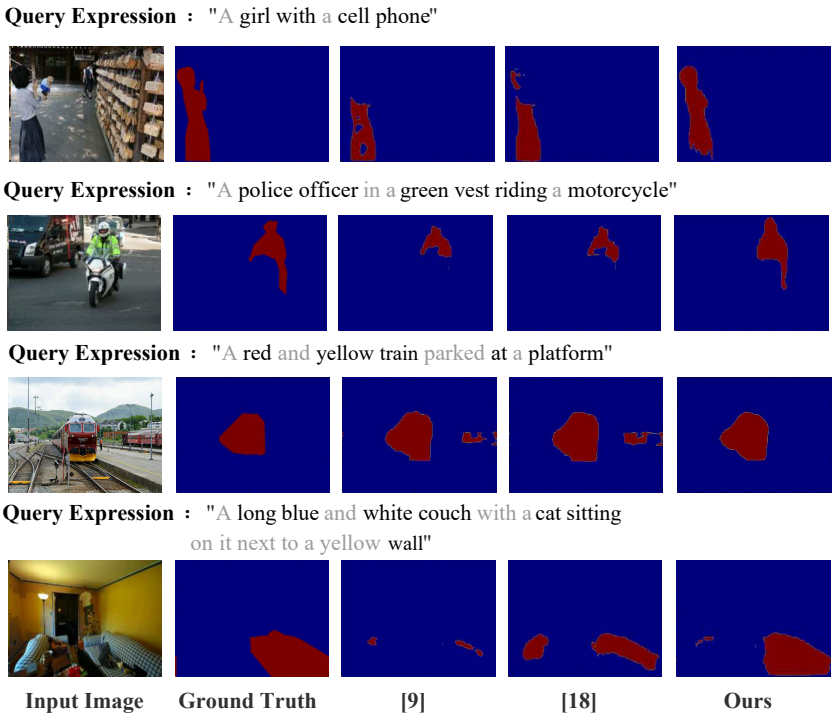

Fig. 4. Referring expression image segmentation results on the Google-Ref validation. Left to right: input images, ground truth, the segmentation results from [9], [18] and our method, respectively. All methods are implemented with *Deeplab101*. In query expressions, the black words mean key words our method predicted for foreground regions (red regions).

1. **Baseline**: We take the method in [9] as the baseline model, which classify each image region with whole query feature and do not model visual context.
2. **Key-word-model**: Instead of using whole query, we extract key words for every image region, but the visual context is not used in this model.
3. **Context-model**: We extract key words for every image region and leverage spatial pyramid pooling to model visual context, which is only based on the visual information.
4. **Full-model**: Full-model extracts key words for every image region and models key-word-aware visual context, which is not only based on vision but also the nature language query.
5. **Soft-model**: Soft-model also extracts key words and models key-word-aware visual context. In this model, we use a soft attention model to aggregate the context instead of the unit step function described in Section 3.3.

The results of ablation studies are shown in Table 3. It can be seen that (1) using key words is better than using whole query; (2) visual context is effective to improve the performance; (3) compared with the context only based on vision,

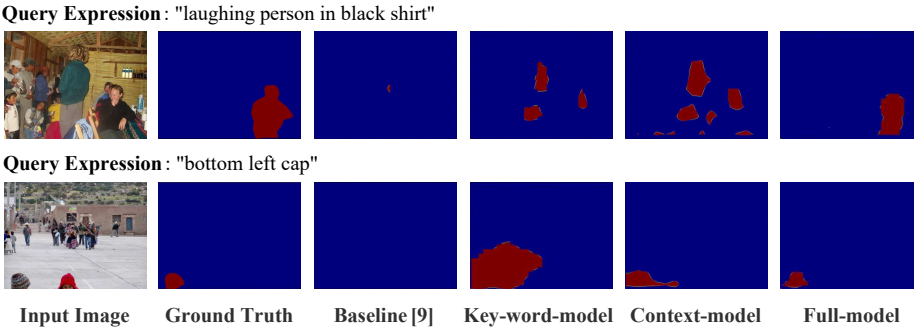

Fig. 5. Visualized results of the ablation studies on the ReferItGame testing. Left to right: input images, ground truth, the segmentation results from baseline model [9], key-word-model, context-model and full-model, respectively. All models are implemented with *VGG16*.

Table 3. Comparison of different ablation models on the ReferItGame testing. "Soft" means that the key-word-aware visual context is calculated by a soft attention model instead of the unit step function. All models are implemented with *VGG16*.

Method	Query Attention	Visual Context	Key-word-aware Visual Context	IoU
Baseline [9]				48.03%
Key-word-model	✓			50.28%
Context-model	✓	✓		51.01%
Full-model	✓		✓	**52.19%**
Soft-model	✓		soft	51.93%

key-word-aware visual context can further improve the referring expression image segmentation performance; (4) the performance of soft-attention-based model is comparable with that of the unit-step-function-based model. However, the computation cost of soft attention is much higher than that of unit step function. Therefore, we use the unit step function instead of the soft attention.

We visualize some results of different ablation models in Fig. 5. It can observed that the baseline model almost does not predict any foreground object region for some queries, due to that it fails to mine semantic from these query expressions. Key-word-model mines key words from the queries, thus it generates some foreground predictions. However, key-word-model still cannot segment out the referred objects, because it separately classify each image region, while these queries need to compare multiple regions. Context-model improves the segmentation results by modeling visual context among image regions, but it also fails to segment out these objects. A major reason is that the context-model ignores the relationship between visual context and the natural language queries. Our full-model extracts key words and models key-word-aware visual context, therefore successfully segments out these objects.

Fig. 6. Visualization of key words for some image regions on the ReferItGame testing. Left to right: input images, key words (black words) for image regions (red, green and blue points), and segmentation results from our full model implemented with *VGG16*.

Table 4. IoU for queries of different lengths on the ReferItGame testing. All methods are implemented with *VGG16*.

IoU Length / Method	1	2-3	4-6	7-20
[9]	62.64%	44.48%	34.56%	20.09%
[18]	63.19%	46.08%	35.43%	22.25%
Ours	**65.59%**	**48.03%**	**38.03%**	**26.61%**

Key Word. Table 4 and Table 5 show the segmentation performance for queries of different lengths. It can be observed that compared with existing methods, the proposed method yields more gains when deals with longer queries. This demonstrates that using key words instead of whole queries is effective, especially when tackling long queries. Fig. 6 depicts visualized examples of extracted key words for some image regions. For example, in the second image in Fig. 6, only according to the word *cap*, the green regions can be eliminated from the desired foreground object, because they are not caps.

Failure Case. Some failure cases are shown in Fig. 7. One type of failures occurs when queries contain low-frequency or new words. For example, in the first image in Fig. 7, *blanket* rarely appears in the training data. As a result, our method does not segment out the *blanket*, although it has already highlighted the *right white* regions in the *background*. Another case is that our method sometimes fails to segment out small objects. For instance, in the second image in Fig. 7, our method highlights the *left* of the *background*, but does not segment out the *person*, because it is very small. This problem may be alleviated by enlarging the scale of input images.

Table 5. IoU for different length queries on Google-Ref validation. All methods are implemented with *VGG16*.

IoU / Length / Method	1-5	6-7	8-10	11-20
[9]	28.67%	23.69%	23.44%	23.22%
[18]	31.05%	27.32%	26.23%	25.25%
Ours	**34.15%**	**28.79%**	**29.90%**	**28.33%**

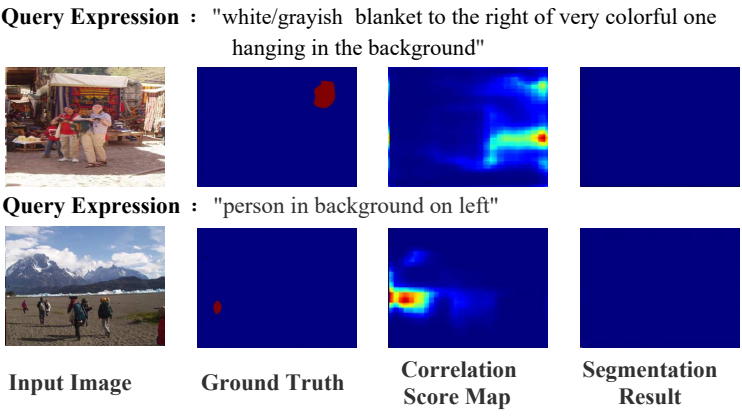

Fig. 7. Failure cases on the ReferItGame dataset. Left to right: input images, ground truth, correlation score maps and segmentation results from our method implemented with *VGG16*.

5 Conclusion

This paper has presented key-word-aware network (KWAN) for referring expression image segmentation. KWAN extracts key words by a query attention model, to suppress the noise in the query and to highlight the desired objects. Moreover, a key-word-aware visual context model is used to learn the relationships of multiple visual objects based on the nature language query, which is important to localize and recognize objects. Our method outperforms state-of-the-art methods on two common referring expression image segmentation databases. In the future, we plan to improve the capacity of the network to tackle objects of different sizes.

Acknowledgement. This work was supported in part by National Natural Science Foundation of China (No. 61525102, 61601102 and 61502084).

References

1. Caesar, H., Uijlings, J., Ferrari, V.: Region-based semantic segmentation with end-to-end training. In: European Conference on Computer Vision. pp. 381–397. Springer (2016)
2. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)
3. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. CoRR (2017)
4. Dai, J., He, K., Sun, J.: Convolutional feature masking for joint object and stuff segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3992–4000 (2015)
5. Gupta, S., Arbeláez, P., Girshick, R., Malik, J.: Indoor scene understanding with rgb-d images: Bottom-up segmentation, object detection and semantic segmentation. International Journal of Computer Vision 112(2), 133–149 (2015)
6. Gupta, S., Arbelaez, P., Malik, J.: Perceptual organization and recognition of indoor scenes from rgb-d images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 564–571 (2013)
7. Gupta, S., Girshick, R., Arbeláez, P., Malik, J.: Learning rich features from rgb-d images for object detection and segmentation. In: European Conference on Computer Vision. pp. 345–360. Springer (2014)
8. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation 9(8), 1735–1780 (1997)
9. Hu, R., Rohrbach, M., Darrell, T.: Segmentation from natural language expressions. European Conference on Computer Vision (2016)
10. Hu, R., Rohrbach, M., Venugopalan, S., Darrell, T.: Utilizing large scale vision and text datasets for image segmentation from referring expressions. CoRR (2016)
11. Hu, R., Xu, H., Rohrbach, M., Feng, J., Saenko, K., Darrell, T.: Natural language object retrieval. In: Computer Vision and Pattern Recognition. pp. 4555–4564 (2016)
12. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proceedings of the 22nd ACM international conference on Multimedia. pp. 675–678. ACM (2014)
13. Kazemzadeh, S., Ordonez, V., Matten, M., Berg, T.: Referitgame: Referring to objects in photographs of natural scenes. In: Conference on Empirical Methods in Natural Language Processing. pp. 787–798 (2014)
14. Li, H., Meng, F., Wu, Q., Luo, B.: Unsupervised multiclass region cosegmentation via ensemble clustering and energy minimization. IEEE Transactions on Circuits and Systems for Video Technology 24(5), 789–801 (2014)
15. Li, Z., Gan, Y., Liang, X., Yu, Y., Cheng, H., Lin, L.: Lstm-cf: Unifying context modeling and fusion with lstms for rgb-d scene labeling. In: European Conference on Computer Vision. pp. 541–557. Springer (2016)
16. Liang, X., Shen, X., Feng, J., Lin, L., Yan, S.: Semantic object parsing with graph lstm. In: European Conference on Computer Vision. pp. 125–143. Springer (2016)
17. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)

18. Liu, C., Lin, Z., Shen, X., Yang, J., Lu, X., Yuille, A.: Recurrent multimodal interaction for referring image segmentation. IEEE International Conference on Computer Vision (2017)
19. Liu, W., Rabinovich, A., Berg, A.C.: Parsenet: Looking wider to see better. CoRR abs/1506.04579 (2015)
20. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3431–3440 (2015)
21. Lu, J., Xiong, C., Parikh, D., Socher, R.: Knowing when to look: Adaptive attention via a visual sentinel for image captioning. Proceedings of the IEEE conference on computer vision and pattern recognition (2016)
22. Lu, J., Yang, J., Batra, D., Parikh, D.: Hierarchical question-image co-attention for visual question answering. NIPS (2016)
23. Luo, B., Li, H., Meng, F., Wu, Q., Huang, C.: Video object segmentation via global consistency aware query strategy. IEEE Transactions on Multimedia PP(99), 1–1 (2017)
24. Mao, J., Huang, J., Toshev, A., Camburu, O., Yuille, A., Murphy, K.: Generation and comprehension of unambiguous object descriptions. In: CVPR (2016)
25. Meng, F., Li, H., Wu, Q., Luo, B., Huang, C., Ngan, K.: Globally measuring the similarity of superpixels by binary edge maps for superpixel clustering. IEEE Transactions on Circuits and Systems for Video Technology PP(99), 1–1 (2016)
26. Nagaraja, V.K., Morariu, V.I., Davis, L.S.: Modeling context between objects for referring expression understanding. European Conference on Computer Vision pp. 792–807 (2016)
27. Noh, H., Hong, S., Han, B.: Learning deconvolution network for semantic segmentation. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1520–1528 (2015)
28. Peng, Z., Zhang, R., Liang, X., Liu, X., Lin, L.: Geometric scene parsing with hierarchical lstm. In: Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence. pp. 3439–3445 (2016)
29. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision 115(3), 211–252 (2015)
30. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations (2015)
31. Wang, G., Luo, P., Lin, L., Wang, X.: Learning object interactions and descriptions for semantic image segmentation. In: CVPR (2017)
32. Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A., Salakhutdinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. International Conference on Machine Learning pp. 2048–2057 (2015)
33. Yang, Z., He, X., Gao, J., Deng, L., Smola, A.: Stacked attention networks for image question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2016)
34. Yao, B.Z., Yang, X., Lin, L., Lee, M.W., Zhu, S.C.: I2t: Image parsing to text description. Proceedings of the IEEE 98(8), 1485–1508 (2010)
35. Yu, D., Fu, J., Rui, Y., Mei, T.: Multi-level attention networks for visual question answering. In: Proceedings of the IEEE conference on computer vision and pattern recognition (July 2017)
36. Yu, F., Koltun, V.: Multi-scale context aggregation by dilated convolutions. In: International Conference on Learning Representations (2016)

37. Yu, L., Poirson, P., Yang, S., Berg, A.C., Berg, T.L.: Modeling context in referring expressions. European Conference on Computer Vision (2016)
38. Zhang, Y., Yuan, L., Guo, Y., He, Z., Huang, I., Lee, H.: Discriminative bimodal networks for visual localization and detection with natural language queries. Proceedings of the IEEE conference on computer vision and pattern recognition (2017)
39. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2017)