Structured Attention Network for Referring Image Segmentation

Liang Lin, Pengxiang Yan, Xiaoqian Xu, Sibei Yang, Kun Zeng, and Guanbin Li

***Abstract*—Referring image segmentation aims at segmenting out the object or stuff referred to by a natural language expression. The challenge of this task lies in the requirement of understanding both vision and language. The linguistic structure of a referring expression can provide an intuitive and explainable layout for reasoning over visual and linguistic concepts. In this paper, we propose a structured attention network (SANet) to explore the multimodal reasoning over the dependency tree parsed from the referring expression. Specifically, SANet implements the multimodal reasoning using an attentional multimodal tree-structure recurrent module (AMTreeGRU) in a bottom-up manner. In addition, for spatial detail improvement, SANet further incorporates the semantics-guided low-level features into high-level ones using the proposed attentional skip connection module. Extensive experiments on four public benchmark datasets demonstrate the superiority of our proposed SANet with more explainable visualization examples.**

***Index Terms*—referring image segmentation, vision and language, cross-modal reasoning**

I. Introduction

Referring image segmentation (RIS) is a challenging task that requires thorough understanding of both vision and natural language. Given an input image and a referring expression, the goal of RIS is to segment out the object or stuff referred to by the referring expression. Compared with an extensively studied task in computer vision, i.e., semantic segmentation, RIS is not limited to pre-defined object categories (e.g. "person", "train") but contains a wider range of linguistic concepts described by entities (e.g. "man"), stuff ("sky"), attributes (e.g. "pink" and "big") and relationships between objects (e.g. "right" and "next to"). Therefore, RIS is applied to more complex scenarios, such as interactive image editing and human-robot interaction.

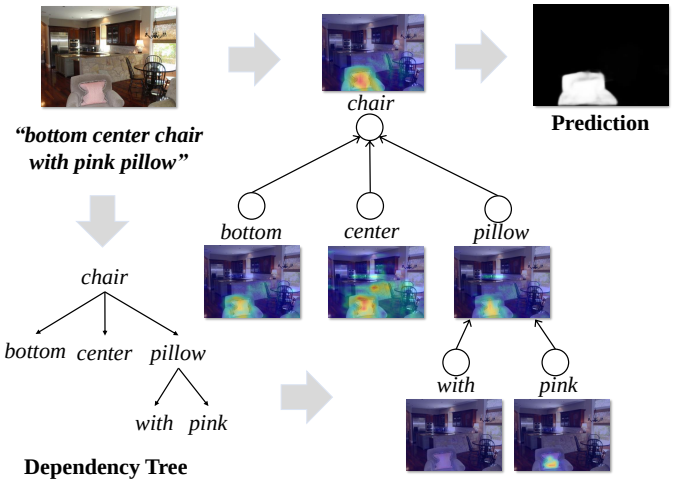


Fig. 1. Illustration of our proposed structured attention network for referring image segmentation. Given an input image and a referring expression, the proposed method works by performing bottom-up cross-modal reasoning over a dependency tree parsed from the referring expression. The visualization of attention maps demonstrates that each node can focus on specific spatial regions by integrating the features and attention maps of its child nodes.

Earlier solutions for RIS are achieved by extracting visual and linguistic features via convolutional neural networks (CNNs) and recurrent neural networks (RNNs) respectively and combining the features of two modals directly [1] or sequentially [2]. Recently, different attention mechanisms [3], [4], [5], [6], [7], [8] have been exploited to better align the concepts in visual and linguistic domains. Shi *et al.* [5] introduced a key-word-aware attention network to highlight key words and the visual context for the key words. Ye *et al.* [6] proposed a self-attention model to capture the long-range dependency of visual and linguistic features by exploring the significance of specific key words and image regions simultaneously.

Since RIS requires not only to roughly locate the referred regions but also to segment them out accurately, Li *et al.* [9] proposed a refinement network that takes pyramidal features as input to refine the details of segmentation via ConvLSTM [10]. Chen *et al.* [11] further proposed a ConvRNN model to iterate the steps of visual-textual co-embedding. More recently, Huang *et al.* [12] proposed to perform relation-aware reasoning on spatial region graph using graph convolutional networks. Hui *et al.* [13] proposed to use graph convolutional networks over a dependency parsing tree suppressed word graph for context modeling.

L. Lin and P. Yan contributed equally to this paper.

This work was supported in part by the National Key Research and Development Program of China under Grant No.2018YFC0830103, in part by the National Natural Science Foundation of China under Grant No.61976250 and No.U1811463, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant No.2020B1515020048, in part by the Natural Science Foundations of Guangdong under Grant No. 2017A03031335, in part by National High Level Talents Special Support Plan (Ten Thousand Talents Program). This work was also sponsored by CCF-Tencent Open Research Fund. (Corresponding author: Guanbin Li)

L. Lin, P. Yan, X. Xu and K. Zeng are with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, 510006, China (e-mail: linliang@ieee.org; yanpx@mail2.sysu.edu.cn; xuxq7@mail2.sysu.edu.cn; zengkun2@mail.sysu.edu.cn).

S. Yang is with the Department of Computing, the Hong Kong Polytechnic University, Hong Kong (e-mail: sibeiyang@comp.polyu.edu.hk).

G. Li is with the School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, 510006, China, and also with Pazhou Lab, Guangzhou, 510330, China. (e-mail: liguanbin@mail.sysu.edu.cn).

To achieve accurate segmentation of referred object, one feasible solution is to locate the referred object under the guidance of linguistic structure and refine the segmentation details by introducing semantics-guided low-level visual features. To realize the above solution, we proposed a structured attention network (SANet) to perform multimodal reasoning over a dependency tree parsed from the referring expression. As shown in Figure 1, given an input image and a referring expression, the referring expression is first parsed into a dependency tree via an off-the-shelf universal Stanford Parser [14]. Then, the extracted visual and linguistic features are fed into the attentional multimodal TreeGRU (AMTreeGRU) module for multimodal reasoning. Specifically, for each node of the dependency tree, the AMTreeGRU module updates its spatial attention map and feature by integrating the features and attention maps of its child nodes. And the output feature extracted from the root of the dependency tree serves as the high-level multimodal feature. In addition to the multimodal reasoning, we have proposed an attentional skip connection (ASC) module to further refine the details of referred regions indicated by our AMTreeGRU module. The ASC module introduces the semantics-guided low-level feature into high-level multimodal feature using a cross-modal attention mechanism.

The contributions of this paper can be summarized as follows.

- We propose an attentional multimodal TreeGRU (AMTreeGRU) module to effectively perform multimodal reasoning over a dependency tree parsed from the input referring expression, which can improve the interpretable ability of the process of referring image segmentation.
- We propose an attentional skip connection (ASC) module to selectively combine low-level features of CNNs under the guidance of high-level semantic ones, which can improve the details of segmentation results.
- Extensive experimental results demonstrate the effectiveness of our proposed AMTreeGRU and ASC modules, which make the proposed SANet comparable to the state-of-the-art RIS algorithms.

II. Related Work

A. Semantic Segmentation

Semantic segmentation aims to segment out the objects of predefined categories. Recently, with the development of deep convolutional neural networks, semantic segmentation has achieved great progress. In particular, due to the powerful end-to-end feature encoding ability and high efficiency, fully convolutional network (FCN) [15] and its variants have become dominant in this field. To enlarge the receptive field of convolutions without losing spatial details, Chen *et al.* [16] proposed to replace the standard convolutions with atrous convolutions in FCNs. More recently, Chen *et al.* [17] introduced an atrous spatial pooling pyramid pooling (ASPP) module, which exploits parallel atrous convolutions with different atrous rates to aggregate multi-scale context. Low-level features of CNNs have also been utilized to improve the spatial details of segmentation results [18]. Consequently, the development in semantic segmentation has laid a firm foundation for referring image segmentation.

B. Referring Image Localization and Segmentation

Referring image localization aims to locate the object instance referred to by a natural language expression with a bounding box. General solutions to this task can be achieved by first generating region proposals via an object detector (e.g. Faster R-CNN [19]) and then searching for the target object instance via natural language object retrieval methods [20], [21], [22]. [23], [24], [25] have further modeled the relationships between proposals and language to better match the target object instance. Different from referring image localization, referring image segmentation (RIS) aims to locate the referred object or stuff with a precise mask instead of a bounding box. Solutions to RIS can be divided into two categories, i.e., bottom-up and top-down. **Bottom-up** methods mainly focus on the modeling of multimodal features and directly generate the prediction masks. Hu *et al.* [1] used a straightforward concatenation of visual and linguistic feature for segmentation. The multimodal recurrent module [2], dynamic filters [26], key-word-aware visual context model [5], and cross-modal self-attention network [6] were further proposed to better model the features in two modals. Recurrent refinement network [9] was proposed to refine the high-level semantics via ConvLSTM [10]. Chen *et al.* [11] further improved the convolutional recurrent neural network by iteratively revealing segmentation cues via learned visual-textual co-embedding. **Top-down** methods mainly rely on object detectors pretrained large-scale detection dataset and segment the referred object from the retrieved box-level object proposal. MAttNet [23] and NMTree [27] use Mask R-CNN [28] to generate RoI proposals and generate the prediction mask within the selected proposal. MCN [29] and CGAN [30] use a multi-task learning framework to perform the RIL and RIS tasks simultaneously, where RIL can help improve the localization of the referred object in the segmentation mask.

In this paper, we propose a bottom-up method that considers the task of referring image segmentation by utilizing the dependency tree structure implied by natural language expressions to perform bottom-up reasoning across visual and linguistic domains. Moreover, we further leverage the high-level semantics as guidance to refine the spatial details of the referred regions.

C. Linguistic Structure Learning

Long-short terms memory (LSTM) [31] network and its variants have played a significant role in natural language processing (NLP) in the past decades due to its superior performance on sequence modeling. However, simply modeling the linguistic features via this kind of linear chain will ignore the structural information implied by natural language. Recently, learning and utilizing the tree structure of sentences are becoming more popular in NLP community. Chen *et al.* [14] proposed a fast and accurate dependency parser using neural networks, which serves as a powerful dependency tree parser in many NLP and vision-language tasks. Tail *et al.* [32] introduced a tree-structured LSTM network (TreeLSTM) to improve the semantic representations of phrases and sentences. Recently, serval RIL and RIS methods have also focused on the

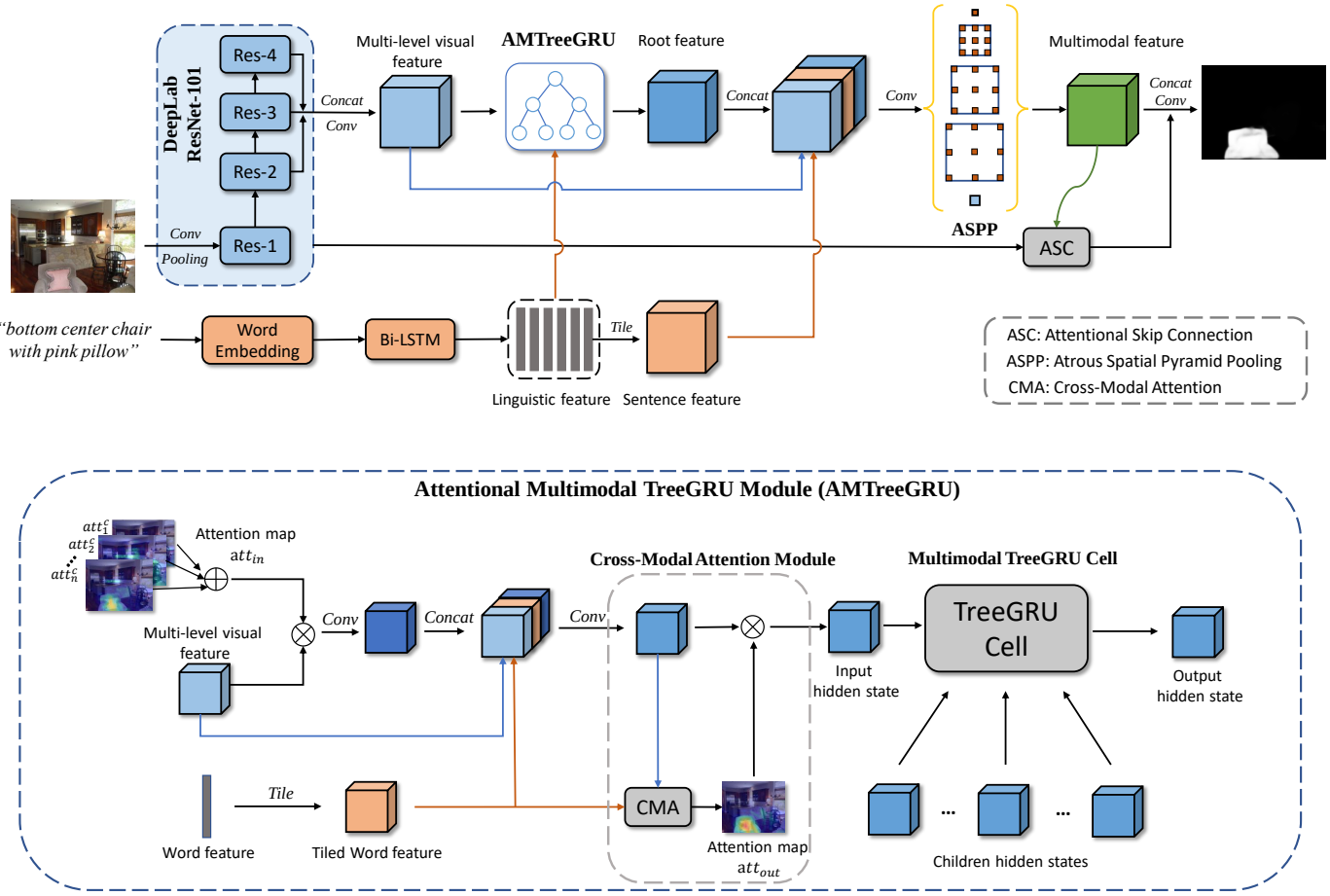


Fig. 2. The overall framework of our proposed structured attention network (SANet). The multi-level visual and linguistic features are extracted from CNN and LSTM respectively, both of which are then fed into the attentional multimodal TreeGRU (AMTreeGRU) module to perform cross-modal reasoning over a dependency tree parsed from the referring expression. Next, the combined features are fed into the ASPP [17] module to produce the multimodal feature by exploring the multi-scale information. Last, the multimodal feature serves as the high-level guidance to connect the low-level visual feature to the decoder via attentional skip connection (ASC) module. The bottom sub-figure shows the architecture of the proposed AMTreeGRU, the detailed description of which is provided in Section III-A.

utilization of linguistic structure information. [33], [34], [12] proposed to decompose the expression into serval semantic parts. [35], [27] introduced linguistic tree structure to perform bottom-up reasoning. [36], [25], [13] further utilize graph structure of language to perform fine-grained reasoning over referring expressions.

In this paper, we consider not only the extraction of linguist structure information but also a better way to align the visual and linguistic features and reason over the linguist tree structure. Thus, we propose an attentional multimodal tree-structured gated recurrent unit (AMTreeGRU) with convolutional operation in its connection layers and apply it to perform bottom-up co-reasoning between visual and linguistic features over the dependency tree parsed from the referring expression.

III. Our Method

The overall framework of the proposed structured attention network (SANet) is shown in Figure 2. The visual and linguistic features are extracted from the visual backbone and language model, respectively. To better align the visual and linguistic domains, the attentional multimodal TreeGRU (AMTreeGRU) is proposed to perform bottom-up co-reasoning between visual and linguistic features (Section III-A). In addition, the attentional skip connection (ASC) module is proposed to improve the spatial details of the referred regions (Section III-B). The attention mechanisms mentioned in AMTreeGRU and ASC are both achieved by a cross-modal attention (CMA) module (Section III-C). Specifically, the CMA module weights the input feature map F under the guidance of a feature map G of different modals and outputs a guided attention map att_{out} and a softly weighted output feature map $\hat{F}$, which is formulate as $att_{out}, \hat{F} = \mathrm{CMA}(F, G)$. Finally, we introduce the rest modules in Section III-D to provide a complete description of the proposed SANet.

A. Attentional Multimodal TreeGRU Module

As shown in Figure 2, the proposed attentional multimodal TreeGRU (AMTreeGRU) module performs bottom-up co-reasoning between visual and linguistic features for each node of the dependency tree. The dependency tree is parsed from the

referring expression using an off-the-shelf universal Stanford Parser [14]. The AMTreeGRU module consists of two major steps, i.e., selective attention via cross-modal attention mechanism and node state updating via convolutional TreeGRU.

Specifically, we denote the multi-level visual features extracted from the visual backbone as $V \in \mathbb{R}^{H\times W\times C_v}$, where H, W and C_v are the height, width and channel dimension of the visual feature map, respectively. And the linguistic feature extracted from the language model is denoted as $L \in \mathbb{R}^{T\times C_l}$, where T and C_l are the length of the expression and the channel dimension of the linguistic feature, respectively. For any node i, the AMTreeGRU module takes as input the visual feature V, the corresponding linguistic feature $L_i \in \mathbb{R}^{C_l}$ on this node, and the attention maps att^c and hidden states h^c of its child nodes, which is formulated as

$$att_i, h_i = \mathrm{AMTreeGRU}(att^{C(i)}, h^{C(i)}, V, L_i), \tag{1}$$

where $att_i \in \mathbb{R}^{H\times W\times 1}$, $h_i \in \mathbb{R}^{H\times W\times C_h}$ denote the attention map and hidden state generated by AMTreeGRU on node i, respectively. C_h is the channel dimension of hidden state and $C(i)$ is the set of the child nodes of i.

Selective attention: for each node of the dependency tree, we obtain its attention map by considering the hidden states and the attention maps of its child nodes. The input attention map att_{in} is computed by subtracting the average of the attention maps of its child nodes $att^{C(i)}$ from 1. The multi-level visual feature map is then softly weighted by the input attention map to pay more attention to these regions that are neglected by the child nodes. The weighted visual feature is denoted as $\hat{V} \in \mathbb{R}^{H\times W\times C_v}$. Then, we tile the corresponding word feature of the node as $L_i^t \in \mathbb{R}^{H\times W\times C_l}, i \in \{1, ..., T\}$ and concatenate it with the original and weighted visual features. The concatenated feature is then fed into the CMA module with the tiled word feature L_i^t as a guidance feature. The output attention map $att_i \in [0, 1]$ of CMA serves as the output attention map of the AMTreeGRU module on the current node. The concatenated feature will be softly weighted by att_{out} and serves as the input state x of the core multimodal TreeGRU cell. The above process can be formulated as follows:

$$\begin{aligned} att_{in} &= 1 - \frac{\sum_{k\in C(i)} att_k^{C(i)}}{num(C(i))}, \\ \hat{V} &= att_{in} \circ V, \\ F_c &= Conv_{1\times 1}(Concat(V, L_i^t, \hat{V})), \\ att_i, x_i &= \mathrm{CMA}_{\mathrm{tree}}(F_c, L_i^t), \end{aligned} \tag{2}$$

where $num(.)$ denotes the number of the set, $Conv_{1\times 1}(.)$ denotes 1×1 convolution, and $Concat(.)$ denotes the concatenation of input tensors.

Node state updating: for each node of the dependency tree, we update its multimodal state by using the proposed multimodal TreeGRU (MTreeGRU). Compared to the traditional gated recurrent units (GRU) [37] adapted in NLP, the MTreeGRU has two differences. First, the GRU cell is applied to the tree structure instead of the sequence. Second, we replace all the fully connected layers in traditional GRU with convolutional layers, which is more suitable for segmentation task and can better handle features from multiple modals. Let $C(i)$ denote the set of the children of node i, x_i denote its input state, and h_i denote its hidden state. The node updating process via MTreeGRU can be formulated as follows:

$$\begin{aligned} \tilde{h_i} &= \sum_{k\in C(i)} h_k, \\ z_i &= \sigma(W_{xz} * x_i + W_{hz} * \tilde{h_i} + b_z), \\ r_{ik} &= \sigma(W_{xr} * x_i + W_{hr} * h_k + b_r), \\ \hat{h_i} &= \tanh(W_{xh} * x_i + W_{hh} * \sum_{k\in C(i)} r_{ik} \circ h_k + b_h), \\ h_i &= (1 - z_i) \circ \hat{h_i} + z_i \circ \tilde{h_i}, \end{aligned} \tag{3}$$

where '$*$' denotes the convolution operator, '$\circ$' denotes the Hadamard product, and $\sigma(.)$ represents the sigmoid activation function. W and b represent the learnable parameters and bias terms, respectively.

To conclude, the proposed AMTreeGRU updates the state of each node over a dependency tree parsed from the referring expression by selectively integrating the attention maps and features of its child nodes. In the end, it reaches to the top of the dependency tree and outputs the hidden state and attention map of the root node.

B. Attentional Skip Connection Module

To improve the segmentation details of RIS, previous work [6] has tried to apply the same processing to features from multiple levels and integrate them using gated fusion technique. Although it shows a significant performance improvement, repeating the same treatment to multi-level features will severely increase the computational cost. Another previous work [18] on semantic segmentation has demonstrated that connecting low-level visual features into the high-level ones can further improve segmentation details. However, in our observations, we found that directly introducing low-level visual features would not only improve segmentation details but also bring extra details that are irrelevant to the referred region. Therefore, we suggest that the low-level features should be further guided by high-level semantics and propose an attentional skip connection (ASC) module in this section.

As shown in Figure 2, the proposed ASC module connects the low-level feature $V_1 \in \mathbb{R}^{H_1\times W_1\times C_1}$ of ***Res-1*** from our visual backbone to the segmentation decoder under the guidance of the multimodal feature $F_m \in \mathbb{R}^{H\times W\times C_h}$ following ASPP. Specifically, it first reduces the channels of the low-level feature V_1 to 48 and then feeds the skipped low-level feature V_{skip} and the high-level multimodal feature F_m into the CMA module, which is formulated as

$$\begin{aligned} V_{skip} &= Conv_{1\times 1}(V_1), \\ att_{skip}, \hat{V}_{skip} &= \mathrm{CMA}_{\mathrm{skip}}(V_{skip}, F_m), \end{aligned} \tag{4}$$

where $Conv_{1\times 1}(.)$ denotes 1×1 convolution and $\hat{V}_{skip}$ denotes the skipped low-level feature weighted by the attention att_{skip}. As shown in Figure 3, since the high-level feature has a coarser resolution than low-level one, the low-level feature will be downsampled to the same scale as high-level one when computing the attention map and the computed attention map

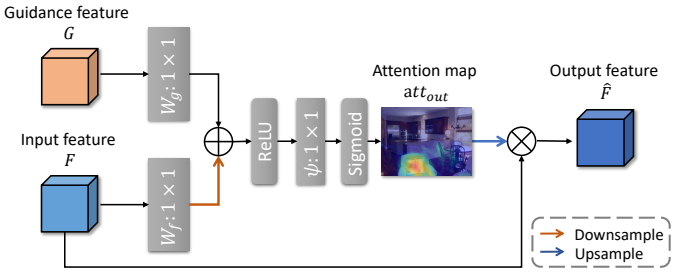


Fig. 3. The architecture of the cross-modal attention (CMA) module, which serves as the attention mechanism in both AMTreeGRU (Section III-A) and ASC (Section III-B) modules. The input feature map is softly weighted under the guidance of another feature map of different modals. Here, we use $\oplus$ and $\otimes$ to denote element-wise addition and element-wise multiplication, respectively.

will be upsampled to a higher resolution before weighting the low-level feature. Finally, the weighted skipped low-level feature V_{skip} is concatenated with the upsampled high-level feature F_m^{up} and fed into a segmentation decoder for referring image segmentation, i.e.,

$$S_{ref} = Decoder(Concat(\hat{V}_{skip}, F_m^{up})). \tag{5}$$

Specifically, the decoder is composed of two 3x3 and one 1x1 convolutional layers. The 3x3 convolutional layer is attached with BN and ReLU layers. The 1x1 convolutional layer is followed by sigmoid function to generate the segmentation map. Through the attention mechanism, the details of the segmentation results S_{ref} can be improved without introducing extra noise that is irrelevant to the referred region.

C. Cross-Modal Attention Module

In this section, we elaborate on the details of the CMA module. It works by generating an attention map for an input feature map under the guidance of another feature map of different modals to indicate the region of interest. Specifically, as shown in Figure 3, given an input feature map $F \in \mathbb{R}^{H_f \times W_f \times C_f}$ and a guidance feature map $G \in \mathbb{R}^{H_g \times W_g \times C_g}$ from different modals, the proposed CMA module can be formulated as follows:

$$q_{att} = \psi^T \left(\sigma_1 \left(W_f^T F + W_g^T G + b_g\right)\right) + b_\psi, \tag{6}$$

$$att_{out} = \sigma_2(q_{att}(F, G; \Theta_{att})), \tag{7}$$

$$\hat{F} = att_{out} \circ F, \tag{8}$$

where $\sigma_1(.)$, $\sigma_2(.)$ and $\circ$ denote the ReLU, sigmoid activation function, and element-wise multiplication, respectively. Θ_{att} denotes the parameters of the CMA module, which contains linear transformations $W_f \in \mathbb{R}^{F_f \times F_{int}}$, $W_g \in \mathbb{R}^{F_g \times F_{int}}$, $\psi \in \mathbb{R}^{F_{int} \times 1}$, and bias terms $b_g \in \mathbb{R}^{F_{int}}$, $b \in \mathbb{R}$. Here, the linear transformations are achieved via 1×1 convolutions instead of matrix multiplication.

Conclusively, the CMA module generates an attention map att_{out} and a feature map $\hat{F}$ that comes from the input feature map F weighted by this attention map.

D. Structured Attention Network

In this section, we will describe the rest modules to provide a complete description of our proposed SANet.

Feature extraction: as illustrated in Figure 2, for the input image, we adopt a deep convolutional network as the visual backbone. The visual feature $V \in \mathbb{R}^{H \times W \times C_v}$ is a combination of multi-level features from the visual backbone. For the referring expression, each word is represented by a word embedding. Then, we use a recurrent neural network to extract linguistic feature $L \in \mathbb{R}^{T \times C_l}$. **Cross-modal reasoning:** the visual and linguistic features are then fed into the AMTreeGRU module to perform bottom-up reasoning over the dependency tree parsed from the referring expression using an universal Stanford Parser [14]. **Feature decoding:** we regard the last hidden state of the Bi-LSTM as the sentence feature $L_s \in \mathbb{R}^{C_l}$. Then, we combine it with the feature of the root node of AMTreeGRU and visual feature V and feed them into ASPP [17] module to further explore the referred regions of multiple scales. Last, the semantics-guided low-level feature is embedded into the decoding processing via the ASC module to refine the details of referred regions.

IV. Experiments

A. Experimental Settings

Datasets: we conduct the evaluation of our proposed SANet on four public RIS benchmark datasets, including UNC [34], UNC+ [34], G-Ref [41], and ReferIt [42].

Both UNC and UNC+ were collected interactively from MS COCO [43] dataset via a two-player game [42]. To guarantee that the referred object should be determined not simply by the semantic categories but also the referring expression, only the images contain two or more objects of the same categories were collected in this game. Concretely, the UNC dataset contains 19,994 images with 142,209 expressions for 50,000 images. The UNC+ dataset has 141,564 expression referring to 49,856 objects in 19,992 images. The UNC+ dataset is similar to UNC except that no location words are allowed in the expressions of UNC+, which makes UNC+ more challenging than UNC. We use the same dataset splits as in [34] for training, validation, and testing.

The G-Ref dataset was also built upon MS COCO, which consists of 104,560 expressions referring to 54,822 objects in 26,711 images. It was collected via Amazon's Mechanical Turk instead of the two-player game, resulting in a longer average expression length (8.43 words) than other datasets (less than 4 words). We use the same splits as [41].

The ReferIt dataset contains 130,525 expressions referring to 96,654 distinct region masks from 19,894 natural images. It was built upon the IAPR TC-12 [44] dataset, which contains not only the object classes but also the stuff classes such as sky, ground, water. We use the same splits as in [42].

Evaluation criteria: following the experimental setup of prior works [9], [26], [1], we adopt the overall intersection-over-union (IoU) metric and precision metrics to evaluate the performance of our model. The overall IoU is the total intersection area divided by the total union area, where both intersection and union areas are accumulated over all

TABLE I
COMPARISON OF QUANTITATIVE RESULTS WITH EXISTING STATE-OF-THE-ART METHODS ON FOUR EVALUATION DATASETS USING OVERALL IOU (%).

Type	Method	Resolution	Visual Backbone	UNC			UNC+			G-Ref	ReferIt
				val	testA	testB	val	testA	testB	val	test
TD	MAttNet [23]	$\sim 1000 \times 600$	mrcnn-resnet101(coco-det)	56.51	62.37	51.70	46.67	52.39	40.08	n/a	-
	NMTree [27]	$\sim 1000 \times 600$	mrcnn-resnet101(coco-det)	56.59	63.02	52.06	47.40	53.01	41.56	n/a	-
	MCN [29]	416×416	DarkNet53(coco-det)	62.44	64.20	59.71	50.62	54.99	44.69	n/a	-
	CGAN [30]	416×416	DarkNet53(coco-det)	64.86	68.04	62.07	51.03	55.51	44.06	n/a	-
BU	LSTM-CNN [1]	320×320	FCN-VGG16(voc-seg)	-	-	-	-	-	-	28.14	48.03
	RMI [2]	320×320	DResNet101(voc-seg)	44.33	44.74	44.63	29.91	30.37	29.43	34.40	57.34
	KWA [5]	320×320	DResNet101(voc-seg)	-	-	-	-	-	-	36.92	59.09
	DMN [26]	320×320	DPN92(imagenet)	49.78	54.83	45.13	38.88	44.22	32.29	36.76	52.81
	RRN [9]	320×320	DResNet101(voc-seg)	54.26	56.21	52.71	39.23	41.68	35.63	36.32	63.12
	CMSA+DCRF [6]	320×320	DResNet101(voc-seg)	58.32	60.61	55.09	43.76	47.60	37.89	39.98	63.80
	STEP [11]	320×320	DResNet101(voc-seg)	60.04	63.46	57.97	48.19	52.33	40.41	46.40	64.13
	CMPC+DCRF [12]	320×320	DResNet101(voc-seg)	61.36	64.53	**59.64**	49.56	53.44	43.23	**49.05**	65.53
	LSCM+DCRF [13]	320×320	DResNet101(voc-seg)	61.47	64.99	59.55	49.34	53.12	**43.50**	48.05	**66.57**
	CGAN [30]	320×320	DResNet101(voc-seg)	59.25	62.37	53.94	46.16	51.37	38.24	46.54	-
	Ours	320×320	DPN92(imagenet)	**62.36**	**65.72**	57.62	50.18	54.87	43.00	42.06	65.62
	Ours	320×320	DResNet101(voc-seg)	61.84	64.95	57.43	**50.38**	**55.36**	42.74	44.53	65.88

"TD" and "BU" denote "Top-Down" and "Bottom-Up" respectively.
"$\sim 1000 \times 600$" means resizing the input image to short side 600 pixels, but keeping maximum length within 1000.
"-" denotes no available results; "n/a" indicates that top-down methods do not use the same dataset split as bottom-up methods.
"DPN92(imagenet)" denotes DPN-92 [38] with ImageNet classfication pretrained weights.
"FCN-VGG16(voc-seg) / DResNet101(voc-seg)" denotes FCN-VGG16 [15] / Deeplab ResNet-101 [17] with PASCAL VOC segmentation [39] pretrained weights.
"mrcnn-resnet101(coco-det) / DarkNet53(coco-det)" denotes Mask R-CNN ResNet-101 [28] / DarkNet-53 [40] with MS COCO detection pretrained weights.

the test samples. The precision metrics ($prec@X$) calculate the proportion of test samples whose prediction masks have an IoU score higher than the threshold X, where $X \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$. Following [5], we use the optimal threshold of the validation or training set for the final segmentation results when calculating the IoU metric.

Implementation details: following the previous bottom-up RIS methods [6], [9], we employ DeepLab ResNet-101 [45], [17] pretrained on Pascal VOC segmentation task [39] as the visual backbone of SANet. The feature maps of *Res-2*, *Res-3*, and *Res-4* are resized to the same scale of *Res-4* and combined as the multi-level visual feature. The output stride of DeepLab ResNet-101 is set to 16, which means for an input image resized to 320×320, the scale of its feature map is 20×20. Following [12], [13], we use Glove word embeddings [46] pretrained on Common Crawl 840B tokens instead of the randomly initialized ones. And use Mutan [47] fusion to combine the sentence feature and visual feature in the feature decoding process. We set the maximum length of referring expressions to 20. Bi-directional LSTM (Bi-LSTM) [48], [31] network is adopted for linguistic feature extraction. For feature dimensions, we reduce the dimension of multi-level visual features to $C_v = 256$ and set $C_l = C_h = 256$. The dimension of the intermediate feature maps of the CMA module is set to 256 in the AMTreeGRU module and set to 48 in the ASC module. The input attention maps of leaf nodes in AMTreeGRU are filled with zero. We train the SANet using Adam optimizer [49] with an initial learning rate of $2.5e^{-5}$ for CNN, and $2.5e^{-4}$ for the rest modules. The training is conducted on a workstation with two NVIDIA GTX 1080Ti GPUs. The batch size is set to 16 on each GPU. We use sigmoid cross-entropy loss as the loss function and stop training after 10 epochs. The proposed SANet is implemented on PyTorch [50] without resorting to any post-processing techniques such as DenseCRF [51] and reaches a real-time speed of 32 fps on a single GPU when inference.

B. Comparison with State-of-the-Art

We compare our proposed SANet with ten existing state-of-the-art bottom-up RIS methods, including LSTM-CNN [1], RMI [2], KWA [5], DMN [26], RRN [9], CMSA [6], STEP [11], CMPC [12], LSCM [13], and CGAN [30]. Besides, we also make a comparison with four state-of-the-art top-down RIS methods, including MAttNet [23], NMTree [27], MCN [30], and CGAN [30], which rely on the RIL task and extra object detection datasets for training. We present the comparison of quantitative results and experimental settings with the above-mentioned methods in Table I.

For bottom-up methods, most of them use Deeplab ResNet-101 or FCN VGG-16 [52], [15] pretrained on Pascal VOC dataset as their visual backbone, except that DMN adopts DPN-92 [38] pretrained on ImageNet [53]. Besides, the resolution of input images is set to 320×320. For a fair comparison with these methods, we also adopt the same experimental settings and reported the quantitative results of our SANet with two different visual backbones, i.e., DPN-92 and Deeplab ResNet-101. As shown in Table I, our proposed SANet can achieve satisfactory performance that is comparable to the existing bottom-up RIS methods without resorting to extra time-consuming post-processing, such as DCRF [54]. In particular, our SANet outperforms the best-performing bottom-up RIS methods by 0.89%, 0.73%, 0.82%, 1.92% on the val, testA sets of UNC and UNC+, respectively. The performance on the challenging dataset UNC+, which contains no location words, also demonstrates that the proposed SANet can effectively reason between features of multiple modals by exploring the structural information implied by natural language.

TABLE II
ABLATION STUDIES ON THE VALIDATION SET OF UNC+ [34] USING $prec@X$ (%) AND OVERALL IOU (%). ALL THE MODELS USE THE SAME VISUAL BACKBONE DEEPLAB RESNET-101 AND ASPP DECODER AS THE BASELINE MODEL.

	Method	$prec$@0.5	$prec$@0.6	$prec$@0.7	$prec$@0.8	$prec$@0.9	overall IoU
1	Baseline (DResNet101+ASPP)	40.96	34.38	27.11	16.24	3.68	39.01
2	+SC	39.81	32.05	24.58	16.13	4.83	40.07
3	+ASC	42.19	35.25	28.12	19.26	5.37	40.62
4	+MGRU	46.59	39.43	30.42	17.21	3.50	43.84
5	+MTreeGRU	52.18	46.63	38.82	25.00	6.23	45.17
6	+AMTreeRNN	48.33	42.02	33.92	21.85	5.24	42.35
7	+AMTreeLSTM	53.12	47.15	38.55	24.30	5.98	45.45
8	+AMTreeGRU	53.93	48.44	40.47	26.26	6.96	46.05
9	+AMTreeGRU+SC	54.51	48.31	41.43	30.61	11.06	46.66
10	+AMTreeGRU+ASC (SANet)	54.95	49.18	41.91	31.17	11.10	47.27
11	SANet+Glove	57.10	51.01	43.74	33.06	12.63	48.56
12	SANet+Glove+Mutan (Ours)	**60.82**	**55.10**	**47.67**	**35.93**	**14.42**	**50.38**

TABLE III
RESULTS OF DIFFERENT VISUAL BACKBONES ON THE VALIDATION SET OF UNC+ [34] IN TERMS OF $prec@X$ (%) AND OVERALL IOU (%).

Visual Backbone	#Params	Pretrained	$prec$@0.5	$prec$@0.6	$prec$@0.7	$prec$@0.8	$prec$@0.9	overall IoU
Deeplab-MobileNetv2	14.39M	voc-seg	52.05	44.87	36.16	24.74	6.95	45.73
Deeplab-ResNet50	36.89M	voc-seg	55.76	49.67	42.20	31.10	11.90	47.55
DPN-92	48.10M	imagenet	60.34	55.01	**48.15**	**36.56**	14.31	50.18
Deeplab-ResNet101	55.88M	voc-seg	**60.85**	**55.10**	47.67	35.93	**14.42**	**50.38**

"#Params" denotes the number of parameters of SANet using different visual backbones.
"voc-seg" denotes that the visual backbone is pretrained on PASCAL VOC [39] segmentation task.
"imagenet" denotes that the visual backbone is pretrained on ImageNet [53] classification task.

For top-down methods, they usually rely on the RIL task for pre-training or multi-task learning and adopt the object detectors pretrained on a large-scale detection dataset MS COCO [43] as visual backbones. The scale of MS COCO (110K images) is much larger than that of Pascal VOC (10K) used in the bottom-up methods. Besides, the input resolution of the top-down methods ($\sim 1000 \times 600$ or 416×416) is usually larger than that of the bottom-up ones (320×320). As shown in Table I, CGAN, which is built upon MCN, provides results using two different experimental settings, i.e., top-down and bottom-up. We can find that CGAN in the top-down setting outperforms the one in the bottom-up setting by a large margin, which indicates the contribution of a larger input resolution and the visual backbone pretrained on COCO detection task. Moreover, the top-down methods are not applicable to ReferIt dataset since the referred regions might involve multiple objects, local regions with in a single object, or stuff regions (e.g., road, river, cloud in sky, etc.). Thus, the direct comparison between the bottom-up and top-down methods are not quite fair or appropriate. Nevertheless, our SANet is still comparable to MCN and even beats NMTree and MAttNet in the top-down settings. Moreover, under the same bottom-up setting, it can consistently outperform the CGAN by a large margin on both UNC and UNC+ datasets.

C. Ablation Studies

We conduct the ablation experiments on the validation set of UNC+ to verify the effectiveness of the proposed AMTreeGRU module and ASC module. Table II summarizes the ablation results in terms of $prec@X$ and overall IoU. We use DeepLab ResNet-101 and ASPP decoder as the baseline model (Row 1 of Table II). Moreover, we have also conducted experiments by replacing the visual backbone in the full model of SANet (Row 12 in Table II) to evaluate the contribution of different visual backbones. Table III summarizes the performance achieved by different visual backbones in terms of $prec@X$ and overall IoU.

Effectiveness of AMTreeGRU: to verify the effectiveness of the AMTreeGRU module, we conduct the experiments by adding AMTreeGRU module and its variants to the baseline model. As shown in rows 4 to 8 in Table II, MGRU refers to a sequential multimodal GRU module instead of the tree-structured GRU in AMTreeGRU. MTreeGRU is obtained by removing the cross-modal attention mechanism from AMTreeGRU. AMTreeRNN and AMTreeLSTM are obtained by replacing the GRU module in AMTreeGRU with vanilla RNN and LSTM respectively. Specifically, +MGRU improves the overall IoU of baseline by 4.83% via modeling visual features with the linguistic feature of each word simultaneously. The tree-structured +MTreeGRU outperforms +MGRU by 1.33% in terms of overall IoU, where it parses the referring expression into a dependency tree and performs multimodal reasoning in a tree-structured multimodal GRU, which indicates the effectiveness of exploring the structural information in natural language. By further introducing the cross-modal attention (CMA) mechanism into MTreeGRU, +AMTreeGRU improves the overall IoU achieved by +MTreeGRU by 0.88%, which indicates the effectiveness of CMA mechanism in

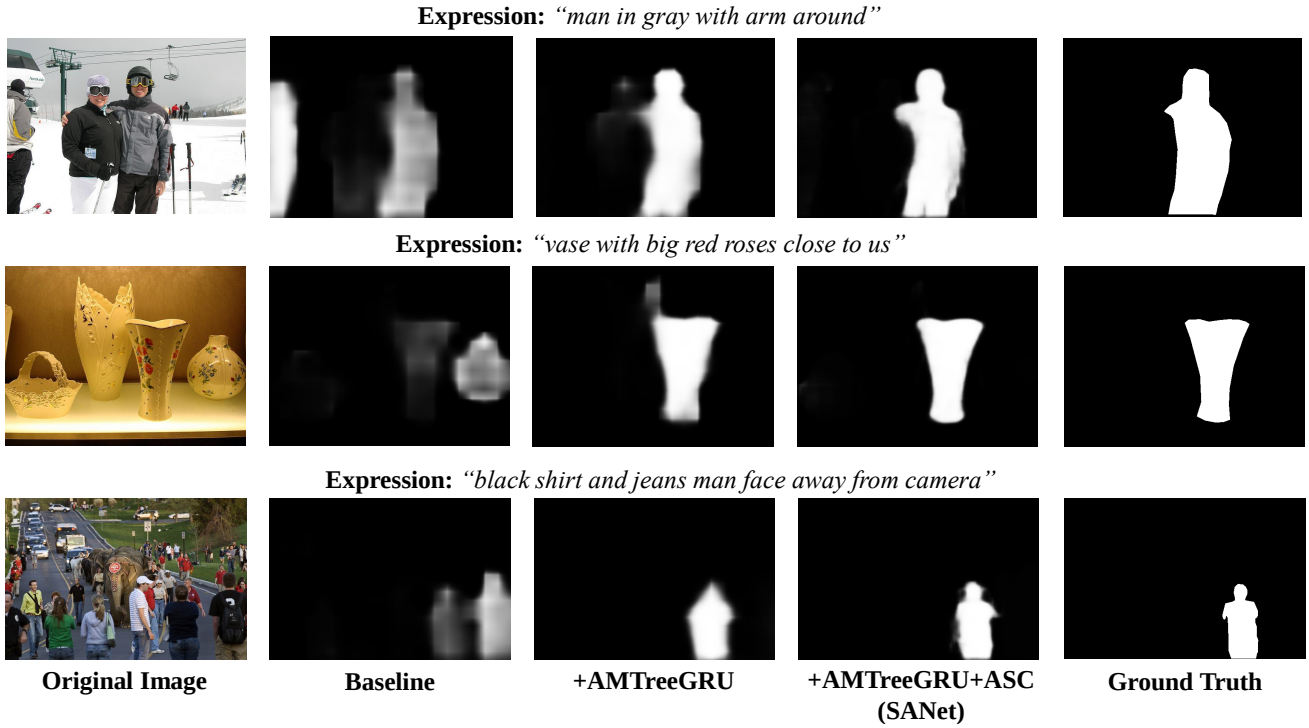


Fig. 4. Qualitative examples of different models from Table II on referring image segmentation. Our proposed SANet generates the segmentation results close to the ground truth with the help of AMTreeGRU that correctly locates the referred regions and ASC that further refines the spatial details of referred regions.

AMTreeGRU module. Besides, by replacing the GRU module in AMTreeGRU with different recurrent neural modules, we can find that both AMTreeLSTM and AMTreeGRU can outperform AMTreeRNN by a large margin. Under the same training iterations, AMTreeGRU can converge slightly better than AMTreeLSTM with fewer module parameters.

Effectiveness of ASC: we evaluate the effectiveness of the ASC module using two different base models, i.e., Baseline (Row 1 in Table II) and +AMTreeGRU (Row 8 in Table II). +SC and +ASC refer to the base model equipped with skip connection layer and the proposed attentional skip connection module respectively. As shown in the Table II, when using Baseline as the base model, +SC (Row 2) might introduce irrelevant noise and results in the decrease of precision metrics at the thresholds from 0.5 to 0.8, while +ASC (Row 3) consistently improves all precision metrics by 0.87%-3.02%. When using +AMTreeGRU as the base model, +AMTreeGRU+ASC (Row 10) consistently improves all precise metrics by 0.74%-4.91% and overall IoU by 1.22%. Moreover, under the precision metric, with higher thresholds, the base model equipped with ASC module achieves more improvements, which indicates that the semantics-guided low-level feature introduced by ASC can effectively refine the spatial details of referred regions. Row 11 and Row 12 show that incorporating with GloVe word embedding and Mutan fusion scheme can further boost the performance of the proposed SANet.

Sensitivity to different visual backbones: to evaluate the contribution of the visual backbone of SANet, we conduct the experiments by replacing the Deeplab-ResNet101 used in the full SANet (Row 12 in Table II), including Deeplab MobileNetv2 [55], Deeplab ResNet50 [45], Deeplab ResNet101 [45], and DPN-92 [38]. As shown in Table III, the performance of SANet will increase by adopting a stronger visual backbone. However, the performance gain of a better visual backbone is not significant. The Deeplab-ResNet101 can only improve the performance achieved by Deeplab-MobileNetv2 by 4.65% in terms of overall IoU, while it costs about four times parameters. Moreover, compared with Table I, we can find that the SANet with Deeplab-MobileNetv2 can already outperform some methods using stronger visual backbones, such as CMSA, RRN, DMN, and RMI on the val set of UNC+. This indicates that the major contribution to RIS methods comes from a better modeling scheme of cross-modal features.

D. Qualitative Analysis

Figure 4 shows some qualitative results of referring image segmentation to further verify the effectiveness of our proposed AMTreeGRU and ASC modules in SANet. As shown in the figure, the baseline model (Row 1 in Table II) cannot precisely locate the referred region in the complex scenarios. Equipped with AMTreeGRU, +AMTreeGRU (Row 8 in Table II) can better understand the referring expression and locate the referred region correctly. However, the boundaries of the referred region are still not accurate enough. While as shown in Figure 4, when further equipped with ASC module, +AMTreeGRU+ASC (Row 8 in Table II) can generate the masks with better spatial details by further introducing semantics-guided low-level visual features via the ASC module.

To provide a better explanation of the bottom-up reasoning process, we visualize attention maps over the nodes of the parsed dependency tree, and the visualization is shown in

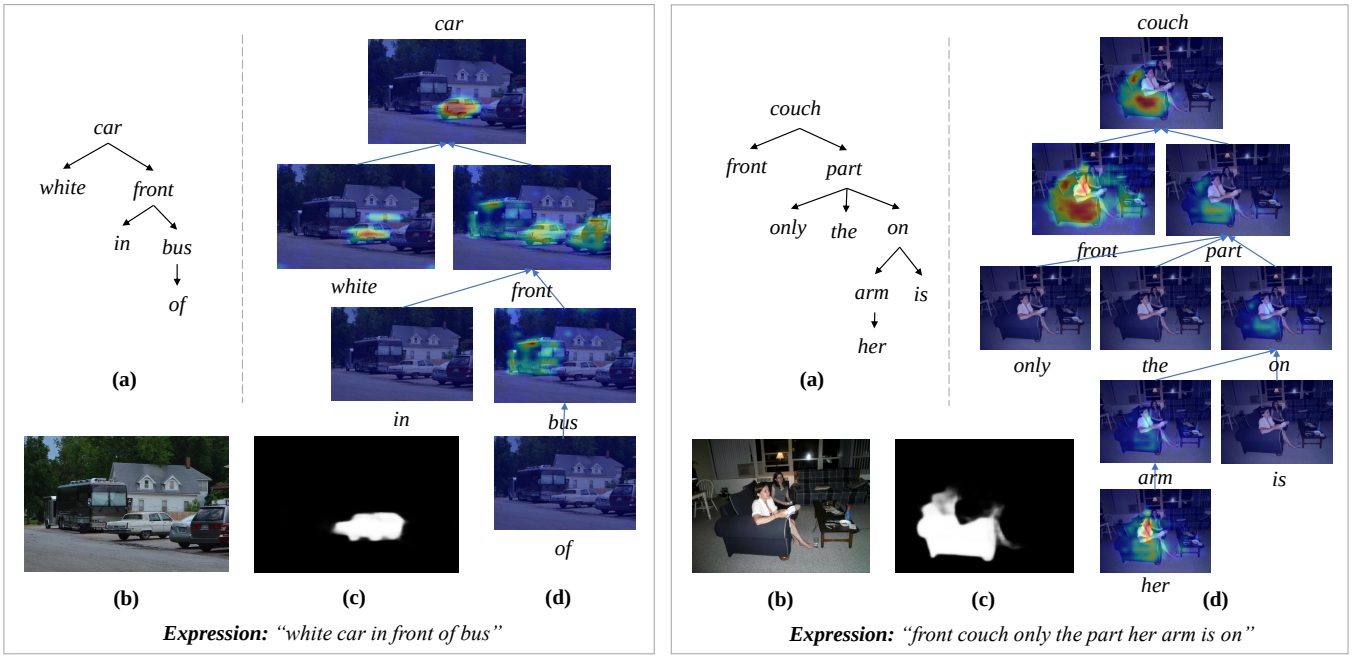


Fig. 5. Visualization of the attention maps in the bottom-up reasoning process of the AMTreeGRU module. (a) The dependency tree parsed from the referring expression. (b) Original image. (c) Prediction. (d) The attention maps in the bottom-up reasoning process of the AMTreeGRU module.

Figure 5. Each node of the dependency tree integrates the attention maps and hidden states of its child nodes and attends the referred region via reasoning at multiple steps. Specifically, as shown in the first example of Figure 5, the AMTreeGRU first locates a rough position of bus according to the nodes "of" and "bus". Then the node "front" focuses on the regions in front of bus by considering the linguistic concept of itself and the visual context implied by its child nodes. Finally, the root node "cat" locates the position of the car by considering both the cues of "white" and "in front of bus". The second example shows that the proposed AMTreeGRU first looks for women and then narrows the attention scope to the region around the arm of the front woman. Meanwhile, the node "front" highlights the whole front region of the image including the couch, women, and table. In the end, the root node focuses on the regions of couch that the arm touches.

V. Conclusion

In this paper, we address the task of referring image segmentation by leveraging the structural information implied by natural language expressions using the proposed SANet, which consists of two major components, i.e., AMTreeGRU and ASC modules. AMTreeGRU performs interpretable cross-modal reasoning over a dependency tree parsed from the referring expression. ASC improves the spatial details of the referred region by introducing low-level features under the guidance of high-level semantics. Benefiting from these two modules, our proposed SANet is comparable to the existing state-the-art methods on four benchmark datasets. Extensive ablation experiments have also demonstrated the effectiveness of our proposed model.

References

[1] R. Hu, M. Rohrbach, and T. Darrell, "Segmentation from natural language expressions," in *Proceedings of the European Conference on Computer Vision*, 2016, pp. 108–124.

[2] C. Liu, Z. Lin, X. Shen, J. Yang, X. Lu, and A. Yuille, "Recurrent multimodal interaction for referring image segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1271–1280.

[3] K. Cho, A. Courville, and Y. Bengio, "Describing multimedia content using attention-based encoder-decoder networks," *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1875–1886, 2015.

[4] L. Gao, Z. Guo, H. Zhang, X. Xu, and H. T. Shen, "Video captioning with attention-based lstm and semantic consistency," *IEEE Transactions on Multimedia*, vol. 19, no. 9, pp. 2045–2055, 2017.

[5] H. Shi, H. Li, F. Meng, and Q. Wu, "Key-word-aware network for referring expression image segmentation," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 38–54.

[6] L. Ye, M. Rochan, Z. Liu, and Y. Wang, "Cross-modal self-attention network for referring image segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 502–10 511.

[7] S. Yang, G. Li, and Y. Yu, "Relationship-embedded representation learning for grounding referring expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[8] Q. Cao, X. Liang, B. Li, G. Li, and L. Lin, "Visual question reasoning on general dependency tree," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7249–7257.

[9] R. Li, K. Li, Y.-C. Kuo, M. Shu, X. Qi, X. Shen, and J. Jia, "Referring image segmentation via recurrent refinement networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5745–5753.

[10] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-c. Woo, "Convolutional lstm network: A machine learning approach for precipitation nowcasting," in *Advances in Neural Information Processing Systems*, 2015, pp. 802–810.

[11] D.-J. Chen, S. Jia, Y.-C. Lo, H.-T. Chen, and T.-L. Liu, "See-through-text grouping for referring image segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 7454–7463.

[12] S. Huang, T. Hui, S. Liu, G. Li, Y. Wei, J. Han, L. Liu, and B. Li, "Referring image segmentation via cross-modal progressive comprehension," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 488–10 497.

[13] T. Hui, S. Liu, S. Huang, G. Li, S. Yu, F. Zhang, and J. Han, "Linguistic structure guided context modeling for referring image segmentation," in *European Conference on Computer Vision*, 2020, pp. 59–75.

[14] D. Chen and C. Manning, "A fast and accurate dependency parser using neural networks," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 740–750.

[15] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[16] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," *arXiv preprint arXiv:1412.7062*, 2014.

[17] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *arXiv preprint arXiv:1706.05587*, 2017.

[18] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European Conference on Computer vision*, 2018, pp. 801–818.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99.

[20] R. Hu, H. Xu, M. Rohrbach, J. Feng, K. Saenko, and T. Darrell, "Natural language object retrieval," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4555–4564.

[21] Y. He, S. Xiang, C. Kang, J. Wang, and C. Pan, "Cross-modal retrieval via deep and bidirectional representation learning," *IEEE Transactions on Multimedia*, vol. 18, no. 7, pp. 1363–1377, 2016.

[22] J. Dong, X. Li, and C. G. Snoek, "Predicting visual features from text for image and video caption retrieval," *IEEE Transactions on Multimedia*, vol. 20, no. 12, pp. 3377–3388, 2018.

[23] L. Yu, Z. Lin, X. Shen, J. Yang, X. Lu, M. Bansal, and T. L. Berg, "Mattnet: Modular attention network for referring expression comprehension," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1307–1315.

[24] X. Li and S. Jiang, "Bundled object context for referring expressions," *IEEE Transactions on Multimedia*, vol. 20, no. 10, pp. 2749–2760, 2018.

[25] S. Yang, G. Li, and Y. Yu, "Cross-modal relationship inference for grounding referring expressions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 4145–4154.

[26] E. Margffoy-Tuay, J. C. Pérez, E. Botero, and P. Arbeláez, "Dynamic multimodal instance segmentation guided by natural language queries," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 630–645.

[27] D. Liu, H. Zhang, F. Wu, and Z.-J. Zha, "Learning to assemble neural module tree networks for visual grounding," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4673–4682.

[28] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2961–2969.

[29] G. Luo, Y. Zhou, X. Sun, L. Cao, C. Wu, C. Deng, and R. Ji, "Multi-task collaborative network for joint referring expression comprehension and segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 10 034–10 043.

[30] G. Luo, Y. Zhou, R. Ji, X. Sun, J. Su, C.-W. Lin, and Q. Tian, "Cascade grouped attention network for referring expression segmentation," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 1274–1282.

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[32] K. S. Tai, R. Socher, and C. D. Manning, "Improved semantic representations from tree-structured long short-term memory networks," in *Proceedings of the Annual Meeting of Association for Computational Linguistics*, 2015, pp. 1556–1566.

[33] R. Hu, M. Rohrbach, J. Andreas, T. Darrell, and K. Saenko, "Modeling relationships in referential expressions with compositional modular networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1115–1124.

[34] L. Yu, P. Poirson, S. Yang, A. C. Berg, and T. L. Berg, "Modeling context in referring expressions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 69–85.

[35] V. Cirik, T. Berg-Kirkpatrick, and L.-P. Morency, "Using syntax to ground referring expressions in natural images," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[36] R. Hong, D. Liu, X. Mo, X. He, and H. Zhang, "Learning to compose and reason with language tree structures for visual grounding," *IEEE transactions on pattern analysis and machine intelligence*, 2019.

[37] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[38] Y. Chen, J. Li, H. Xiao, X. Jin, S. Yan, and J. Feng, "Dual path networks," in *Advances in Neural Information Processing Systems*, vol. 30, 2017, pp. 4467–4475.

[39] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

[40] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv preprint arXiv:1804.02767*, 2018.

[41] J. Mao, J. Huang, A. Toshev, O. Camburu, A. L. Yuille, and K. Murphy, "Generation and comprehension of unambiguous object descriptions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 11–20.

[42] S. Kazemzadeh, V. Ordonez, M. Matten, and T. Berg, "Referitgame: Referring to objects in photographs of natural scenes," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2014, pp. 787–798.

[43] 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 *Proceedings of the European Conference on Computer Vision*, 2014, pp. 740–755.

[44] H. J. Escalante, C. A. Hernández, J. A. Gonzalez, A. López-López, M. Montes, E. F. Morales, L. E. Sucar, L. Villaseñor, and M. Grubinger, "The segmented and annotated iapr tc-12 benchmark," *Computer vision and image understanding*, vol. 114, no. 4, pp. 419–428, 2010.

[45] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[46] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.

[47] H. Ben-Younes, R. Cadene, M. Cord, and N. Thome, "Mutan: Multimodal tucker fusion for visual question answering," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 2612–2620.

[48] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[50] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in pytorch," 2017.

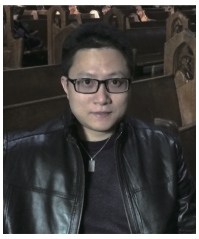
[51] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected crfs with gaussian edge potentials," in *Advances in Neural Information Processing Systems*, 2011, pp. 109–117.

[52] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proceedings of International Conference on Learning Representations*, 2015.

[53] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.

[54] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[55] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

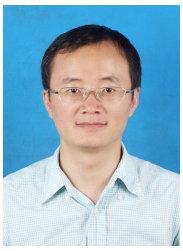


Liang Lin (M'09, SM'15) is a full Professor of Sun Yat-sen University. He is an Excellent Young Scientist of the National Natural Science Foundation of China. From 2008 to 2010, he was a Post-Doctoral Fellow at the University of California, Los Angeles. From 2014 to 2015, as a senior visiting scholar, he was with The Hong Kong Polytechnic University and The Chinese University of Hong Kong. He currently leads the SenseTime R&D teams to develop cutting-edge and deliverable solutions on computer vision, data analysis and mining, and intelligent robotic systems. He has authored and co-authored more than 100 papers in top-tier academic journals and conferences. He has been serving as an associate editor of IEEE Trans. Human-Machine Systems, The Visual Computer and Neurocomputing. He served as area/session chairs for numerous conferences, such as ICME, ACCV, ICMR. He was the recipient of the Best Paper Runners-Up Award in ACM NPAR 2010, the Google Faculty Award in 2012, the Best Paper Diamond Award in IEEE ICME 2017, and the Hong Kong Scholars Award in 2014. He is a Fellow of IET.

Pengxiang Yan received his B.E. degree in software engineering from Sun Yat-sen University, Guangzhou, China, in 2018. He is currently working toward an M.S. degree in computer science and technology at Sun Yat-sen University. His research interests include computer vision and deep learning.

Xiaoqian Xu received her B.E. degree in software engineering from Sun Yat-sen University, Guangzhou, China, in 2019. She is currently working toward an M.S. degree in computer science at Sun Yat-sen University. Her research interests include image understanding and machine learning.

Sibei Yang is currently a research assistant professor in Department of Computing, The Hong Kong Polytechnic University. She received the PhD degree from The University of Hong Kong in 2020. Her current research interests include computer vision, natural language processing, and deep learning. She has been serving a reviewer for numerous academic journals and conferences such as TIP, CVPR, and NeurIPS.



Kun Zeng received the Ph.D. degree from the National Laboratory of Pattern Recognition Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2008. He is currently an Associate Professor with Sun Yat-Sen University, Guangzhou, China. His current research interests include multimedia, computer vision, and machine learning.

Guanbin Li (M'15) is currently an associate professor in School of Data and Computer Science, Sun Yat-sen University. He received his PhD degree from the University of Hong Kong in 2016. His current research interests include computer vision, image processing, and deep learning. He is a recipient of ICCV 2019 Best Paper Nomination Award. He has authorized and co-authorized on more than 70 papers in top-tier academic journals and conferences. He serves as an area chair for the conference of VISAPP. He has been serving as a reviewer for numerous academic journals and conferences such as TPAMI, IJCV, TIP, TMM, TCyb, CVPR, ICCV, ECCV and NeurIPS.