Multi-task Collaborative Network for Joint Referring Expression Comprehension and Segmentation

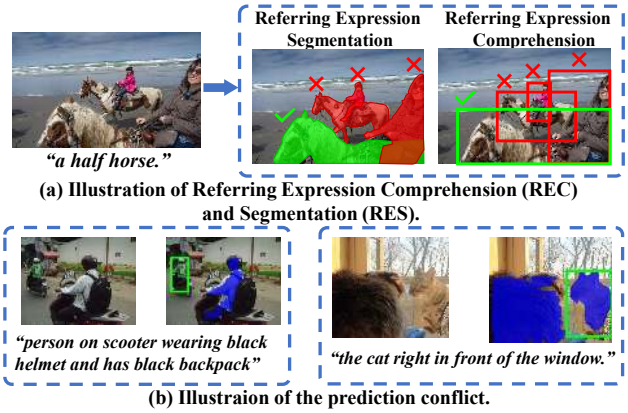
Gen Luo [1*], Yiyi Zhou[1*], Xiaoshuai Sun[1], Liujuan Cao[1], Chenglin Wu[2], Cheng Deng[3], Rongrong Ji[1†]
[1]Media Analytics and Computing Lab, Department of Artificial Intelligence,
School of Informatics, Xiamen University, 361005, China.
[2]DeepWisdom, China. [3]Xidian University, China.
{luogen, zhouyiyi}@stu.xmu.edu.cn, {xssun,caoliujuan}@xmu.edu.cn,
alexanderwu@fuzhi.ai,chdeng.xd@gmail.com, rrji@xmu.edu.cn

Abstract

Referring expression comprehension (REC) and segmentation (RES) are two highly-related tasks, which both aim at identifying the referent according to a natural language expression. In this paper, we propose a novel Multi-task Collaborative Network (MCN)[1] to achieve a joint learning of REC and RES for the first time. In MCN, RES can help REC to achieve better language-vision alignment, while REC can help RES to better locate the referent. In addition, we address a key challenge in this multi-task setup, i.e., the prediction conflict, with two innovative designs namely, Consistency Energy Maximization (CEM) and Adaptive Soft Non-Located Suppression (ASNLS). Specifically, CEM enables REC and RES to focus on similar visual regions by maximizing the consistency energy between two tasks. ASNLS supresses the response of unrelated regions in RES based on the prediction of REC. To validate our model, we conduct extensive experiments on three benchmark datasets of REC and RES, i.e., RefCOCO, RefCOCO+ and RefCOCOg. The experimental results report the significant performance gains of MCN over all existing methods, i.e., up to +7.13% for REC and +11.50% for RES over SOTA, which well confirm the validity of our model for joint REC and RES learning.

1. Introduction

Referring Expression Comprehension (REC) [11, 12, 19, 21, 44, 45, 48, 42, 37] and Referring Expression Segmentation (RES) [32, 16, 40, 25, 34] are two emerging tasks, which involves identifying the target visual instances according to a given linguistic expression. Their difference is that in REC, the targets are grounded by bounding boxes, while they are segmented in RES, as shown in Fig. 1(a).



(a) Illustration of Referring Expression Comprehension (REC) and Segmentation (RES).

(b) Illustraion of the prediction conflict.

Figure 1. (a) The RES and REC models first perceive the instances in an image and then locate one or few referents based on an expression. (b) Two typical cases of prediction conflict: wrong REC correct RES (left) and wrong RES correct REC (right).

REC and RES are regarded as two seperated tasks with distinct methodologies in the existing literature. In REC, most existing methods [11, 12, 19, 21, 23, 44, 45, 46, 48] follow a multi-stage pipeline, *i.e.,* detecting the salient regions from the image and selecting the most matched one through multimodal interactions. In RES, existing methods [32, 16] usually embed a language module, *e.g.*, LSTM or GRU [6], into a one-stage segmentation network like FCN [20] to segment the referent. Although some recent works like MAttNet [43] can simultaneously process both REC and RES, their multi-task functionality are largely attributed to their backbone detector, *i.e.*, MaskRCNN [43], rather than explicitly interacting and reincecing two tasks.

It is a natural thought to jointly learn REC and RES to reinforce each other, as similar to the classic endeavors in joint object detection and segmentation [9, 10, 7]. Compared with RES, REC is superior in predicting the poten-

*Equal Contribution. † Corresponding Author.
[1]Source codes and pretrained backbone are available at : https://github.com/luogen1996/MCN

tial location of the referent, which can compensate for the deficiency of RES in determining the correct instance. On the other hand, RES is trained with pixel-level labels, which can help REC obtain better language-vision alignments during the multimodal training. However, such a joint learning is not trivial at all. We attribute the main difficulty to *the prediction conflict*, as shown in Fig. 1 (b). Such prediction conflict is also common in general detection and segmentation based multi-task models [10, 8, 5]. However, it is more prominent in RES and REC, since only one or a few of the multiple instances are the correct referents.

To this end, we propose a novel Multi-task Collaborative Network (MCN) to jointly learn REC and RES in a one-stage fashion, which is illustrated in Fig. 2. The principle of MCN is a multimodal and multitask collaborative learning framework. It links two tasks centered on the language information to maximize their collaborative learning. Particularly, the visual backbone and the language encoder are shared, while the multimodal inference branches of two tasks remain relatively separated. Such a design is to take full account of the intrinsic differences between REC and RES, and avoid the performance degeneration of one task to accommodate the other, *e.g.*, RES typically requires higher resolution feature maps for its pixel-wise prediction.

To address the issue of prediction conflict, we equip MCN with two innovative designs, namely *Consistency Energy Maximization* (CEM) and *Adaptive Soft Non-Located Suppression* (ASNLS). CEM is a language-centric loss function that forces two tasks on the similar visual areas by maximizing the consistency energy between two inference branches. Besides, it also serves as a pivot to connect the learning processes of REC and RES. ASNLS is a post-processing method, which suppresses the response of unrelated regions in RES based on the prediction of REC. Compared with existing hard processing methods, *e.g.,* RoI-Pooling [30] or Rol-Align [10], the adaptive soft processing of ASNLS allows the model to have a higher error tolerance in terms of the detection results. With CEM and ASNLS, MCN can significantly reduce the effect of the prediction conflict, as validated in our quantitative evaluations.

To validate our approach, we conduct extensive experiments on three benchmark datasets, *i.e.*, RefCOCO, RefCOCO+ and RefCOCOg, and compare MCN to a set of state-of-the-arts (SOTAs) in both REC and RES [42, 38, 40, 16, 18, 37]. Besides, we propose a new metric termed *Inconsistency Error* (IE) to objectively measure the impact of *prediction conflict*. The experiments show superior performance gains of MCN over SOTA, *i.e.*, up to +7.13% in REC and +11.50% in RES. More importantly, these experimental results greatly validate our argument of reinforcing REC and RES in a joint framework, and the impact of prediction conflict is effectively reduced by our designs.

Conclusively, our contributions are three-fold:

- We propose a new multi-task network for REC and RES, termed Multi-task Collaborative Network (MCN), which facilitates the collaborative learning of REC and RES.
- We address the key issue in the collaborative learning of REC and RES, *i.e.*, the prediction conflict, with two innovative designs, *i.e.*, *Consistency Energy Maximization* (CEM) and *Adaptive Soft Non-Located Suppression* (ASNLS)
- The proposed MCN has established new state-of-the-art performance in both REC and RES on three benchmark datasets, *i.e.*, RefCOCO, RefCOCO+ and RefCOCOg. Notably, its inference speed is 6 times faster than that of most existing multi-stage methods in REC.

2. Related Work

2.1. Referring Expression Comprehension

Referring expression comprehension (REC) is a task of grounding the target object with a bounding box based on a given expression. Most existing methods [11, 12, 19, 21, 44, 45, 48, 42, 37] in REC follow a multi-stage procedure to select the best-matching region from a set of candidates. Concretely, a pre-trained detection network, *e.g.*, FasterRCNN [30], is first used to detect salient regions of a given image. Then to rank the query-region pairs, a multimodal embedding network [31, 36, 19, 3, 47] is used, or the visual features are included into the language modeling [23, 1, 21, 12, 44]. Besides, additional processes are also used to improve the multi-modal ranking results, *e.g.*, the prediction of image attributes [43] or the calculation of location features [45, 37]. Despite their high performance, these methods have a significant drawback in low computational efficiency. Meanwhile, their upper-bounds are largely determined by the pre-trained object detector [33].

To speedup the inference, some recent works in REC resort to a one-stage modeling [33, 38], which embeds the extracted linguistic feature into a one-stage detection network, *e.g.,* YoloV3 [29], and directly predicts the bounding box. However, their performance is still worse than the most popular two-stage approaches, *e.g.,* MattNet [42]. Conclusively, our work are the first to combine REC and RES in a one-stage framework, which not only boosts the inference speed but also outperforms these two-stage methods.

2.2. Referring Expression Segmentation

Referring expression segmentation (RES) is a task of segmenting the referent according to a given textual expression. A typical solution of RES is to embed the language encoder into a segmentation network, *e.g.*, FCN [20], which further learns a multimodal tensor for decoding the segmentation mask [32, 16, 25, 40, 34]. Some recent

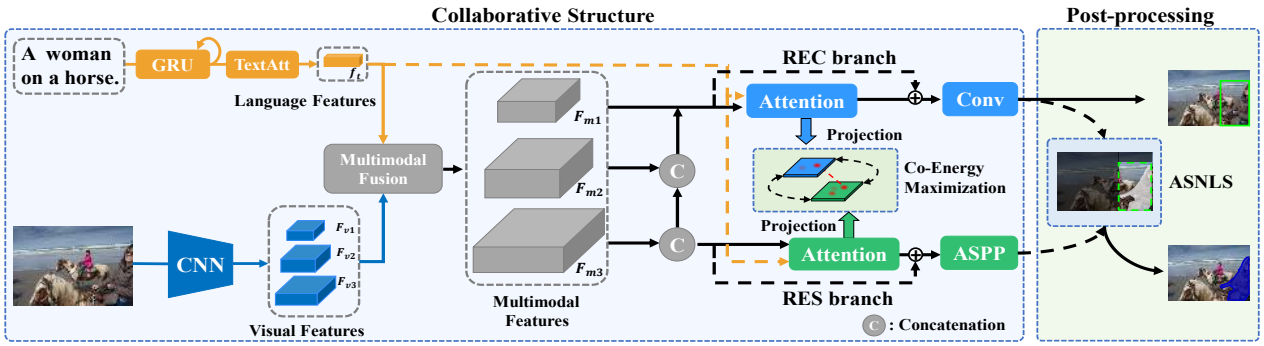


Figure 2. The framework of the proposed *Multi-task Collaborative Network* (MCN). The visual features and linguistic features are extracted by a deep convolutional network and a bi-GRU network respectively, and then fused to generate the multi-scale multimodal features. The bottom-up connection from the RES branch effectively promotes the language-vision alignment of REC. The two branches are further reinforced by each other through CEM. Finally, the output of RES is adaptively refined by ASNLS based on the REC result.

developments also focus on improving the efficiency of multimodal interactions, *e.g.*, adaptive feature fusions at multi-scale [32], pyramidal fusions for progressive refinements [16, 25], and query-based or transformer-based attention modules [34, 40].

Although relatively high performance is achieved in RES, existing methods are generally inferior in determining the referent compared to REC. To explain, the pixel-wise prediction of RES is easy to generate uncertain segmentation mask that includes incorrect regions or objects, *e.g.*, overlapping people. In this case, the incorporation of REC can help RES to suppress responses of unrelated regions, while activating the related ones based on the predicted bounding boxes.

2.3. Multi-task Learning

Multi-task Learning (MTL) is often applied when related tasks can be performed simultaneously. MTL has been widely deployed in a variety of computer vision tasks [8, 5, 27, 7, 10, 15]. Early endeavors [8, 5, 27] resort to learn multiple tasks of pixel-wise predictions in an MTL setting, such as depth estimation, surface normals or semantic segmentation. Some recent works also focus on combining the object detection and segmentation into a joint framework, *e.g.*, MaskRCNN [10], YOLACT [2], and RetinaMask [9]. The main difference between MCN and these methods is that MCN is an MTL network centered on the language information. The selection of target instance in REC and RES also exacerbates the issue of prediction conflicts, as mentioned above.

3. Multi-task Collaborative Network

The framework of the proposed *Multi-task Collaborative Network* (MCN) is shown in Fig. 2. Specifically, the representations of the input image and expression are first extracted by the visual and the language encoders respectively, which are further fused to obtain the multimodal features of different scales. These multimodal features are then fed to the inference branches of REC and RES, where a bottom-up connection is built to strengthen the collaborative learning of two tasks. In addition, a language-centric connection is also built between two branches, where the *Consistency Energy Maximization* loss is used to maximize the consistency energy between REC and RES. After inference, the proposed *Adaptive Soft Non-Located Suppression* (ASNLS) is used to refine the segmentation result of RES based on the predicted bounding box by the REC branch.

3.1. The Framework

As shown in Fig. 2, MCN is partially shared, where the inference branches of RES and REC remain relatively independent. The intuition is two-fold: On one hand, the objectives of two tasks are still distinct, thus the full sharing of the inference branch can be counterproductive. On the other hand, such a relatively independent design enables the optimal settings of two tasks, *e.g.*, the resolution of feature map.

Concretely, given an image-expression pair (I, E), we first use the visual backbone to extract the feature maps of three scales, denoted as $\mathbf{F}_{v_1} \in \mathbb{R}^{h_1 \times w_1 \times d_1}$, $\mathbf{F}_{v_2} \in \mathbb{R}^{h_2 \times w_2 \times d_2}$, $\mathbf{F}_{v_3} \in \mathbb{R}^{h_3 \times w_3 \times d_3}$, where h, w and d denote the height, width and the depth. The expression is processed by a bi-GRU encoder, where the hidden states are weightly combined as the textual feature by using a self-guided attention module [39], denoted as $f_t \in \mathbb{R}^{d_t}$.

Afterwards, we obtain the first multimodal tensor by fusing $\mathbf{F}_{v_1}$ with f_t, which is formulated as:

$$f^l_{m_1} = \sigma(f^l_{v_1}\mathbf{W}_{v_1}) \odot \sigma(f_t\mathbf{W}_t), \tag{1}$$

where $\mathbf{W}_{v_1}$ and $\mathbf{W}_t$ are the projection weight matrices, and σ denotes *Leaky ReLU* [22]. $f^l_{m_1}$ and $f^l_{v_1}$ are the feature vector of $\mathbf{F}_{m_1}$ and $\mathbf{F}_{v_1}$, respectively. Then, the other two multimodal tensors, $\mathbf{F}_{m_2}$ and $\mathbf{F}_{m_3}$, are obtained by the fol-

lowing procedure:

$$\begin{aligned}\mathbf{F}_{m_{i-1}} &= UpSample(\mathbf{F}_{m_{i-1}}),\\ \mathbf{F}_{m_i} &= [\sigma(\mathbf{F}_{m_{i-1}}\mathbf{W}_{m_{i-1}}), \sigma(\mathbf{F}_{v_i}\mathbf{W}_{v_i})],\end{aligned} \tag{2}$$

where $i \in \{2, 3\}$, *UpSampling* has a stride of 2×2, and $[\cdot]$ denotes concatenation.

Such a multi-scale fusion not only propagates the language information through upsamplings and concatenations, but also includes the mid-level semantics to the upper feature maps, which is crucial for both REC and RES. Considering that these two tasks have different requirements for the feature map scales, *e.g.*, 13×13 for REC and 52×52 for RES, we use $\mathbf{F}_{m_1}$ and $\mathbf{F}_{m_3}$ as the inputs of REC and RES, respectively.

To further strengthen the connection of two tasks, we implement another bottom-up path from RES to REC. Such a connection introduces the semantics supervised by the pixel-level labels in RES to benefit the language-vision alignments in REC. Particularly, the new multimodal tensor, $\mathbf{F}'_{m_1}$ for REC, is obtained by repeating the down sampling and concatenations twice, as similar to the procedure defined in Eq. 2. Afterwards, $\mathbf{F}'_{m_1}$ and $\mathbf{F}'_{m_3}$ for REC and RES respectively are then refined by two *GARAN Attention* modules [41], as illustrated in Fig. 2.

Objective Functions. For RES, we implement the ASPP decoder [4] to predict the segmentation mask based on the refined multimodal tensor. Its loss function is defined by

$$\ell_{res} = -\sum_{l=1}^{h_3 \times w_3} \left[g_l \log(o_l) + (1 - g_l) \log(1 - o_l) \right], \tag{3}$$

where g_l and o_l represent the elements of the down-sampled ground-truth $\mathbf{G}' \in \mathbb{R}^{52\times52}$ and predicted mask $\mathbf{O} \in \mathbb{R}^{52\times52}$, respectively.

For REC, we add a regression layer after the multimodal tensor for predicting the confidence score and the bounding box of the referent. Following the setting in YoloV3 [29], the regression loss of REC is formulated as:

$$\ell_{rec} = \sum_{l=1}^{h_1 \times w_1 \times N} \ell_{box}(t_l^*, t_l) + \ell_{conf}(p_l^*, p_l), \tag{4}$$

where t_l and p_l are the predicted coordinate position of the box and confidence score. N is the number of anchors for each grid. t_l* and p_l* are the ground-truths. p_l^* is set to 1 when the anchor matches ground-truth. ℓ_{box} is a binary cross-entropy to measure the regression loss for the center point of the bounding box. For the width and height of the bounding box, we adopt the smooth-L1 loss [30]. ℓ_{conf} is the binary cross entropy.

3.2. Consistency Energy Maximization

We further propose a *Consistency Energy Maximization* (CEM) scheme to theoretically reduce the impact of prediction conflict. As shown in Fig. 3, CEM build a language-centered connection between two branches. Then, CEM loss defined in Eq. 9 is used to maintain the consistency of spatial responses for two tasks by maximizing the energy between their attention tensors.

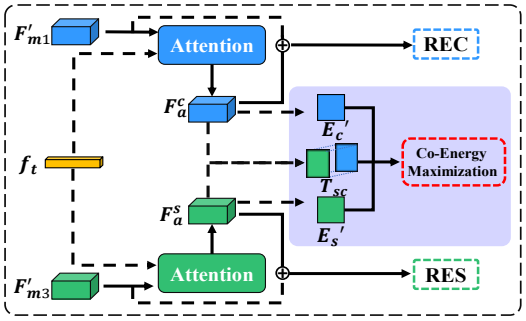


Figure 3. Illustration of the Consistency Energy Maximization (CEM). The CEM loss optimizes the attention features to maximize the consistency spatial responses between REC and RES.

Concretely, given the attention tensors of RES and REC, denoted as $\mathbf{F}_a^s \in \mathbb{R}^{(h_3\times w_3)\times d}$ and $\mathbf{F}_a^c \in \mathbb{R}^{(h_1\times w_1)\times d}$, we project them to the two-order tensors by:

$$E_s = \mathbf{F}_a^s \mathbf{W}_s, \quad E_c = \mathbf{F}_a^c \mathbf{W}_c, \tag{5}$$

where $\mathbf{W}_s, \mathbf{W}_c \in \mathbb{R}^{d\times1}$, $E_s \in \mathbb{R}^{(h_3\times w_3)}$ and $E_c \in \mathbb{R}^{(h_1\times w_1)}$. Afterwards, we perform *Softmax* on E_c and E_s to obtain the energy distributions of REC and RES over the image, dentoed as E'_c and E'_s. Elements of E'_c and E'_s indicate the response degrees of the corresponding regions towards the given expression.

To maximize the co-energy between two tasks, we further calculate the inter-task correlation, $\mathbf{T}_{sc} \in \mathbb{R}^{(h_3\times w_3)\times(h_1\times w_1)}$, by

$$\mathbf{T}_{sc}(i,j) = s_w * \frac{f_i^{sT} f_j^c}{\|f_i^s\| \|f_j^c\|} + s_b, \tag{6}$$

where the $f_i^s \in \mathbb{R}^d$ and $f_j^c \in \mathbb{R}^d$ are elements of $\mathbf{F}_a^s$ and $\mathbf{F}_a^c$, respectively. The s_w and s_b are two scalars to scale the value in $\mathbf{T}_{sc}$ to $(0, 1]$. The co-energy C is caculated as:

$$\begin{aligned} C(i,j) =& \log\left[E'_s(i)\, \mathbf{T}_{sc}(i,j)\, E'_c(j)\right] \\ =& E_s(i) + E_c(j) + \log \mathbf{T}_{sc}(i,j) \\ & - \log\alpha_s - \log\alpha_c, \end{aligned} \tag{7}$$

where the α_s and α_c are two reguralization term to penalize the irrelevant responeses, denoted as:

$$\alpha_s = \sum_{i=1}^{h_3\times w_3} e^{E_s(i)}, \alpha_c = \sum_{i=1}^{h_1\times w_1} e^{E_c(i)}. \tag{8}$$

Finally, the CEM loss is formulated by

$$\ell_{cem} = -\sum_{i=1}^{h_3\times w_3} \sum_{j=1}^{h_1\times w_1} C(i,j). \tag{9}$$

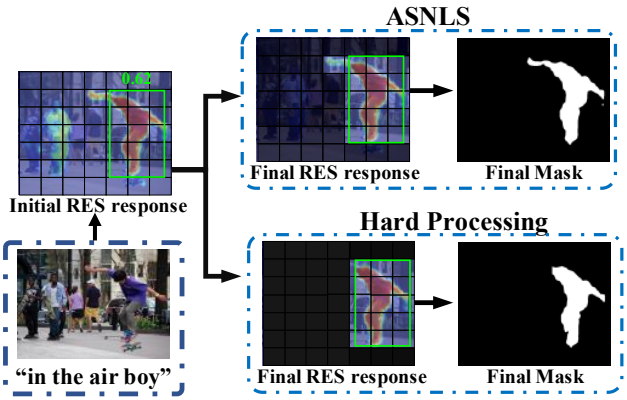


Figure 4. The comparison between ASNLS and conventional hard processing (bottom). Compared to the hard processing, ASNLS has a better error tolerance for REC predictions, which can well preserve the integrity of referent given an inaccurate box.

3.3. Adaptive Soft Non-Located Suppression

We further propose a soft post-processing method to methodically address the prediction conflict, termed as *Adaptive Soft Non-Located Suppression* (ASNLS). Based on the predcited bounding box by REC, ASNLS suppresses the response of unrelated regions and strengths the related ones. Compared to the existing hard processings, *e.g.*, ROI Pooling [30] and ROI Align [10], which directly crop features of the bounding box, the soft processing of ASNLS can obtain a better error tolerance towards the predictions of REC, as illustrated in Fig. 4.

In particular, given the predicted mask by the RES branch, $\mathbf{O} \in \mathbb{R}^{h_3 \times w_3}$, and the bounding box b, each element o_i in $\mathbf{O}$ is updated by:

$$m_i = \begin{cases} \alpha_{up} * o_i, & if\ o_i\ in\ b, \\ \alpha_{dec} * o_i, & else. \end{cases} \tag{10}$$

Here, $\alpha_{up} \in (1, +\infty)$ and $\alpha_{dec} \in (0, 1)$ are the enhancement and decay factors, respectively. We term this method in Eq. 10 as *Soft Non-Located Suppression* (Soft-NLS). After that, the updated RES result $\mathbf{O}$ is binarized by a threshold to generate the final mask.

In addition, we extend the Soft-NLS to an adaptive version, where the update factors are determined by the prediction confidence of REC. To explain, a lower confidence p indicates a larger uncertainty that the referent can be segmented integrally, and should increase the effects of NLS to eliminate the uncertainty as well as to enhance its saliency. Specifically, given the confidence score p, α_{up} and α_{dec} are calculated by

$$\begin{aligned} \alpha_{up} &= \lambda_{au} * p + \lambda_{bu}, \\ \alpha_{dec} &= \lambda_{ad} * p + \lambda_{bd}, \end{aligned} \tag{11}$$

where the λ_{au}, λ_{ad} , λ_{bu} and λ_{bd} are hyper-parameters[2] to control the enhancement and decay, respectively. We term this adaptive approach as *Adaptive Soft Non-Located Suppression* (ASNLS).

[2]In our experiments, we set $\lambda_{au} = -1$, $\lambda_{ad} = 1$, $\lambda_{bu} = 2$, $\lambda_{bd} = 0$.

3.4. Overall Loss

The overall loss function of MCN is formulated as:

$$\ell_{all} = \lambda_s \ell_{res} + \lambda_c \ell_{rec} + \lambda_e \ell_{cem}, \tag{12}$$

where, λ_s, λ_c and λ_e control the relative importance among the three losses, which are set to 0.1, 1.0 and 1.0 in our experiments, respectively.

4. Experiments

We further evaluate the proposed MCN on three benchmark datasets, i.e., RefCOCO [13], RefCOCO+ [13] and RefCOCOg [24], and compare them to a set of state-of-the-art methods [43, 37, 38, 40, 16] of both REC and RES.

4.1. Datasets

RefCOCO [13] has 142,210 referring expressions for 50,000 bounding boxes in 19,994 images from MS-COCO [17], which is split into *train*, *validation*, *Test A* and *Test B* with a number of 120,624, 10,834, 5,657 and 5,095 samples, respectively. The expressions are collected via an interactive game interface [13], which are typically short sentences with a average length of 3.5 words. The categories of bounding boxes in TestA are people while the ones in TestB are objects.

RefCOCO+ [13] has 141,564 expressions for 49,856 boxes in 19,992 images from MS-COCO. It is also divided into splits of *train* (120,191), *val* (10,758), *Test A* (5,726) and *Test B* (4,889). Compared to RefCOCO, its expressions include more appearances (attributes) than absolute locations. Similar to RefCOCO, expressions of Test A in RefCOCO+ are about people while the ones in Test B are about objects.

RefCOCOg [24, 26] has 104,560 expressions for 54,822 objects in 26,711 images. In this paper, we use the UNC partition [26] for training and testing our method. Compared to RefCOCO and RefCOCO+, expressions in RefCOCOg are collected in a non-interactive way, and the lengths are longer (8.4 words on average), of which content includes both appearances and locations of the referent.

4.2. Evaluation Metrics

For REC, we use the precision as the evaluation metric. When the *Intersection-over-Union* (IoU) between the predicted bounding box and the ground truth is larger than 0.5, the prediction is correct.

For RES, we use IoU and Acc@X to evaluate the model. The Acc@X metric measures the percentage of test images

Table 1. Comparisons of the different post-processing methods on the validation set of RefCOCO. ↓ denotes the lower is better.

	IoU	Acc@0.5	Acc@0.6	Acc@0.7	Acc@0.8	Acc@0.9	IE ↓
w.o. post-processing	61.61	73.95	67.42	56.39	32.02	4.72	10.37%
RoI Crop [10, 30]	61.19	75.13	68.88	57.61	32.42	3.81	7.91%
Soft-NLS (ours)	62.27	75.92	69.48	58.21	33.20	5.11	7.28%
ASNLS (ours)	62.44	76.60	70.33	58.39	33.68	5.26	6.65%

Table 2. Ablation study on the *val* set of three datasets. The metric is Acc@0.5 for REC, and IoU for RES. *Base* indicates the network structure without any extra components.

	RefCOCO			RefCOCO+			RefCOCOg		
	REC	RES	IE ↓	REC	RES	IE ↓	REC	RES	IE ↓
MCN (Base)	77.45	58.24	13.80%	62.74	44.08	20.70%	62.29	44.58	19.87%
+TextAtt	77.65	58.44	13.44%	63.07	44.38	19.88%	64.51	46.58	18.71%
+GARAN	79.20	59.07	13.37%	66.22	47.89	17.12%	65.98	47.33	17.44%
+CEM	80.08	61.61	10.37%	67.16	49.55	13.51%	66.46	48.56	14.90%
+ASNLS	80.08	62.44	6.65%	67.16	50.62	7.54%	66.46	49.22	9.41%

Table 3. Comparisons of MCN with different network structures on the *val* set of RefCOCO. The structure of MCN can significantly improve the performance of both two tasks, and it is also superior than other single and multi-task frameworks.

Structure	REC	RES
Single_REC(scale[1]=13^2)	70.38	-
Single_REC(scale=52^2)	68.58	-
Single_RES(scale=13^2)	-	36.37
Single_RES(scale=52^2)	-	57.91
OnlyHeadDifferent(scale=13^2)	72.42	34.50
OnlyHeadDifferent(scale=52^2)	72.54	58.08
OnlyBackboneShared(REC_scale=13^2, RES_scale=52^2)	75.81	58.16
MCN (Base)	**77.45**	**58.24**

with an IoU score higher than the threshold X, while X higher than 0.5 is considered to be correct.

In addition, we propose a *Inconsistency Error* (IE) to measure the impact of the prediction conflict. The inconsistent results are considered to be the two types: 1) the results include wrong REC result and correct RES result. 2) the results include correct REC result and wrong RES result.

4.3. Implementation Details

In terms of the visual backbone, we train MCN with Darknet53 [29] and Vgg16 [35]. Following the setting of MattNet [43], the backbones are pre-trained on MS-COCO [17] while removing the images appeared in the val and test sets of three datasets. The images are resized to 416×416 and the words in the expressions are initialized with GLOVE embeddings [28]. The dimension of the GRU is set to 1,024. In terms of multimodal fusion, the project dimension in Eq. 1 and Eq. 2 is 512. For the Soft-NLS, we set α_{up} to 1.5 and set α_{dec} to 0.5. We set the maximum sentence length of 15 for RefCOCO and RefCOCO+, and 20 for RefCOCOg. To binarize the prediction of RES, we set a threshold of 0.35.

We use Adam [14] as the optimizer, and the batch size is set to 35. The initial learning rate is 0.001, which is multiplied by a decay factor of 0.1 at the 30th, the 35th and 40th epochs. We take nearly a day to train our model for 45 epochs on a single 1080Ti GPU.

[1]Scale denotes the resolution of the last feature map before prediction.

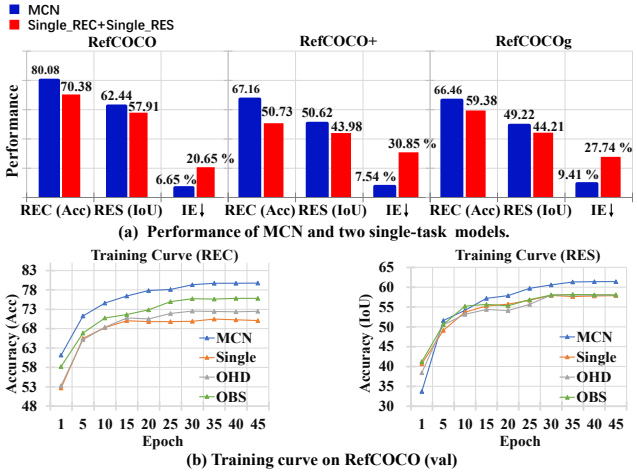


Figure 5. Comparisons of MCN and other structures. (a) MCN significantly improves the performance of both two tasks on three datasets. (b) The learning speed of MCN is superior to alternative structures. Here, all structures do not use the post-processing.

4.4. Experimental Results

4.4.1 Quantitative Analysis

Comparisons of different network structures. We first evaluate the merit of the proposed multi-task collaborative framework, of which results are given in Tab. 3. In Tab. 3, *Single_REC* and *Single_RES* denote the single-task setups. *OnlyHeadDifferent* (OHD) and *OnlyBackboneShared* (OBS) are the other two types of multi-task frameworks. OHD denotes that the inference branches are also shared and only the heads are different, *i.e.*, the regression layer for REC and the decoder for RES. In contrast, OBS denotes that the inference branches of two tasks are completely independent. From the first part of Tab. 3, we ob-

Table 4. Comparisons of MCN with the state-of-the-arts on the REC task.

Model	Visual Features	RefCOCO			RefCOCO+			RefCOCOg		Speed* ↓
		val	testA	testB	val	testA	testB	val	test	
MMI [23] CVPR16	vgg16	-	64.90	54.51	-	54.03	42.81	-	-	-
CMN [31] CVPR16	vgg16	-	71.03	65.77	-	54.32	47.76	-	-	-
Spe+Lis+Rl [45] CVPR17	frcnn-resnet101	69.48	73.71	64.96	55.71	60.74	48.80	60.21	59.63	-
Spe+**Lis**+Rl [45] CVPR17	frcnn-resnet101	68.95	73.10	64.85	54.89	60.04	49.56	59.33	59.21	-
ParalAttn [49] CVPR18	frcnn-vgg16	-	75.31	65.52	-	61.34	50.86	-	-	-
LGRANs [37] CVPR19	frcnn-vgg16	-	76.60	66.40	-	64.00	53.40	-	-	-
NMTree [18] ICCV19	frcnn-vgg16	71.65	74.81	67.34	58.00	61.09	53.45	61.01	61.46	-
FAOA [38] ICCV19	darknet53	71.15	74.88	66.32	56.86	61.89	49.46	59.44	58.90	**39 ms**
MattNet [43] CVPR18	frcnn-resnet101	76.40	80.43	69.28	64.93	70.26	56.00	**66.67**	67.01	367 ms
MattNet [43] CVPR18	mrcnn-resnet101	76.65	81.14	69.99	65.33	71.62	56.02	66.58	**67.27**	378 ms
MCN (ours)	vgg16	75.98	76.97	73.09	62.80	65.24	54.26	62.42	62.29	48 ms
MCN (ours)	darknet53	**80.08**	**82.29**	**74.98**	**67.16**	**72.86**	**57.31**	66.46	66.01	56 ms

* The inference time is tested on the same hardware, i.e., GTX1080ti.

Table 5. Comparisons of MCN with the state-of-the-arts on the RES task.

Model	Visual Features	RefCOCO			RefCOCO+			RefCOCOg	
		val	testA	testB	val	testA	testB	val	test
DMN [25] ECCV18	resnet101	49.78	54.83	45.13	38.88	44.22	32.29	-	-
RRN [16] CVPR18	resnet101	55.33	57.26	53.93	39.75	42.15	36.11	-	-
CMSA [40] CVPR19	resnet101	58.32	60.61	55.09	43.76	47.60	37.89	-	-
MattNet [43] CVPR18	mrcnn-resnet101	56.51	62.37	51.70	46.67	52.39	40.08	47.64	48.61
NMTree [18] ICCV19	mrcnn-resnet101	56.59	63.02	52.06	47.40	53.01	41.56	46.59	47.88
MCN (ours)	vgg16	57.33	58.59	57.23	46.53	48.68	41.93	46.95	47.20
MCN (ours)	darknet53	**62.44**	**64.20**	**59.71**	**50.62**	**54.99**	**44.69**	**49.22**	**49.40**

serve that MCN significantly benefits both tasks. Besides, we notice that the two tasks have different optimal settings about the scales of the multimodal tensors, *i.e.*, 13×13 for REC and 52×52 for RES, suggesting the differences of two tasks. The second part of Tab. 3 shows that a completely independent or fully shared network can not maximize the advantage of the joint REC and RES learning, which subsequently validates the effectiveness of the collaborative connections built in MCN. Meanwhile, as shown in Fig. 5, MCN demonstrates its benefits of collaborative multi-task training and outperforms other single and multi-task models by a large margin.

Comparison of ASNLS and different post-processing methods. We further evaluate different processing methods, and give the results in Tab. 1. From Tab. 1, the first observation is that all the processing methods based on REC have a positive impact on both the RES performance and the IE score. But we also notice that the hard processing, *i.e.*, RoI Crop [10, 30], still reduces the performance of RES on some metrics, *e.g.*, IoU and Acc@0.9, while our soft processing methods, *i.e.* Soft-NLS and ASNLS, does not. This results greatly prove the robustness of our methods. Meanwhile, we observe that ASNLS can achieve more significant performance gains than Soft-NLS, which validates the effects of the adaptive factor design.

Ablation study. Next, we validate different designs in MCN, of which results are given in Tab. 2. From Tab. 2, we can observe significant performance gains by each design of MCN, *e.g.*, up to 7.04% gains for REC and 14.84% for RES. We also notice that CEM not only helps the model achieve distinct improvements on both the REC and the RES tasks, but also effectively reduces the IE value, *e.g.*, from 17.12% to 13.51%. Similar advantages can also be witnessed in ASNLS. Conclusively, these results confirm the merits of the collaborative framework, CEM and ASNLS again.

Comparison with the State-of-the-arts. Lastly, we compare MCN with the state-of-the-arts (SOTAs) on both REC and RES, of which results are given in Tab. 4 and Tab. 5. As shown in Tab. 4, MCN outperforms most existing methods in REC. Even compared with the most advanced methods, like MattNet [43], MCN still achieves a comprehensive advantage and has distinct improvements on some splits, *e.g.* +7.13% on the testB split of RefCOCO and +2.80% the val split of RefCOCO+. In addition, MCN obviously merits in the processing speed to these multi-stage methods, *e.g.*, 6 times faster than MattNet, which also suggests that the improvements by MCN are valuable. Meanwhile, MCN are signficantly better than the most advanced one-stage model, *e.g.*, FAOA [38], which confirms the merit of the joint REC and RES learning again. In Tab. 5, we further observe that the performance leads of MCN leads in RES task is more distinct, which is up to +8.39% on

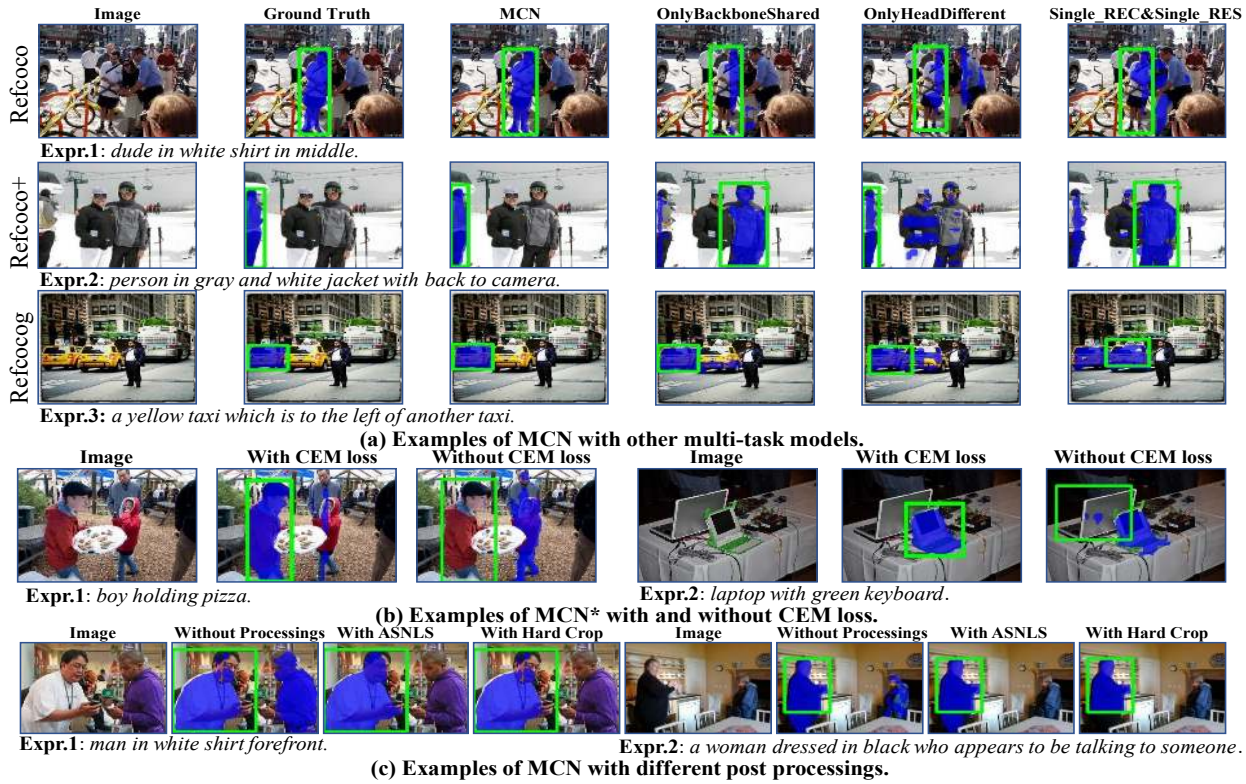


Figure 6. Visualizations of the inference and prediction by the proposed MCN. We compare the results of MCN with three multi-task networks in (a) and compare the effects of our design in (b) and (c). * denotes that the post-processings is not used in these example.

RefCOCO, +11.50% on RefCOCO+ and +3.32% on RefCOCOg. As previously analyzed, such performance gains stem from the collaborative learning structure, CEM loss and ASNLS, greatly confirming the designs of MCN.

4.4.2 Qualitative Analysis

To gain deep insights into MCN, we visualize its predictions in Fig. 6. The comparisons between MCN and alternative structures are shown in Fig. 6 (a). From Fig. 6 (a), we can observe that the collaborative learning structure of MCN significantly improves the results of both REC and RES. Besides, MCN is able to predict high-quality boxes and masks for the referent in complex backgrounds, which is often not possible by alternative structures, *e.g.*, Expr.1. Fig. 6 (b) displays the effect of the proposed CEM loss. Without it, the model tends to focus on different instances of similar semantics, resulting the prediction conflicts of the REC and RES branches. With CEM, the two inference branches can have a similar focus with respect to the expression. Fig. 6 (c) shows results of the model without and with different post-processing methods. From these examples, we can observe that the proposed ASNLS helps to preserve the integrity of an object, *e.g.,* Exp.(2). It can be seen that the part of referent outside the bounding box is preserved by our ASNLS, while it will be naturally cropped by the hard methods, *e.g.,* ROI-Pooling [30] and RoI-Align [10]. Conclusively, these visualized results reconfirm the effectiveness of the novel designs in MCN, *i.e.*, the collaborative learning structure, CEM and ASNLS.

5. Conclusion

In this paper, we propose a novel *Multi-task Collaborative Network* (MCN) for the first attempt of joint REC and RES learning. MCN maximizes the collaborative learning advantages of REC and RES by using the properties of two tasks to benefit each other. In addition, we introduce two designs, *i.e.*, *Consistency Energy Maximization* (CEM) and *Adaptive Soft Non-Located Suppression* (ASNLS), to address a key issue in this multi-task setting *i.e.*, the prediction conflict. Experimental results on three datasets not only witness the distinct performance gains over SOTAs of REC and RES, but also prove that the prediction conflict is well addressed.

Acknowledgements. This work is supported by the Nature Science Foundation of China (No.U1705262, No.61772443, No.61572410, No.61802324 and No.61702136), National Key R&D Program (No.2017YFC0113000, and No.2016YFB1001503), and Nature Science Foundation of Fujian Province, China (No. 2017J01125 and No. 2018J01106).

References

[1] Jonathan Baxter. A model of inductive bias learning. In *JAIR*, 2000. 2

[2] Daniel Bolya, Chong Zhou, Fanyi Xiao, and Yong Jae Lee. Yolact: Real-time instance segmentation. In *ICCV*, 2019. 3

[3] Kan Chen, Rama Kovvuri, and Ram Nevatia. Query-guided regression network with context policy for phrase grounding. In *ICCV*, 2017. 2

[4] Liangchieh Chen, George Papandreou, Iasonas Kokkinos, Kevin P Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. In *PAMI*, 2018. 4

[5] Liangfu Chen, Zeng Yang, Jianjun Ma, and Zheng Luo. Driving scene perception network: Real-time joint detection, depth estimation and semantic segmentation. In *WACV*, 2018. 2, 3

[6] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *arXiv preprint*, 2014. 1

[7] Nikita Dvornik, Konstantin Shmelkov, Julien Mairal, and Cordelia Schmid. Blitznet: A real-time deep network for scene understanding. In *ICCV*, 2017. 1, 3

[8] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *ICCV*, 2015. 2, 3

[9] Cheng-Yang Fu, Mykhailo Shvets, and Alexander C. Berg. RetinaMask: Learning to predict masks improves state-of-the-art single-shot detection for free. In *arXiv preprint*, 2019. 1, 3

[10] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross B Girshick. Mask r-cnn. In *ICCV*, 2017. 1, 2, 3, 5, 6, 7, 8

[11] Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. Modeling relationships in referential expressions with compositional modular networks. In *CVPR*, 2017. 1, 2

[12] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. Natural language object retrieval. In *CVPR*, 2016. 1, 2

[13] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara L Berg. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014. 5

[14] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *arXiv preprint*, 2014. 6

[15] Iasonas Kokkinos. Ubernet: Training a 'universal' convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *CVPR*, 2017. 3

[16] Ruiyu Li, Kaican Li, Yichun Kuo, Michelle Shu, Xiaojuan Qi, Xiaoyong Shen, and Jiaya Jia. Referring image segmentation via recurrent refinement networks. In *CVPR*, 2018. 1, 2, 3, 5, 7

[17] Tsungyi Lin, Michael Maire, Serge J Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5, 6

[18] Daqing Liu, Hanwang Zhang, Feng Wu, and Zheng-Jun Zha. Learning to assemble neural module tree networks for visual grounding. In *ICCV*, 2019. 2, 7

[19] Jingyu Liu, Liang Wang, and Minghsuan Yang. Referring expression generation and comprehension via attributes. In *ICCV*, 2017. 1, 2

[20] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1, 2

[21] Ruotian Luo and Gregory Shakhnarovich. Comprehension-guided referring expressions. In *CVPR*, 2017. 1, 2

[22] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *ICML*, 2013. 3

[23] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Maria Camburu, Alan L Yuille, and Kevin P Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016. 1, 2, 7

[24] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Maria Camburu, Alan L Yuille, and Kevin P Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016. 5

[25] Edgar A Margffoytuay, Juan C Perez, Emilio Botero, and Pablo Andres Arbelaez. Dynamic multimodal instance segmentation guided by natural language queries. In *ECCV*, 2018. 1, 2, 3, 7

[26] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *ECCV*, 2016. 5

[27] Vladimir Nekrasov, Thanuja Dharmasiri, Andrew Spek, Tom Drummond, Chunhua Shen, and Ian Reid. Real-time joint semantic segmentation and depth estimation using asymmetric annotations. In *ICRA*, 2019. 3

[28] Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *EMNLP*, 2014. 6

[29] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. In *arXiv preprint*, 2018. 2, 4, 6

[30] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. In *TPAMI*, 2017. 2, 4, 5, 6, 7, 8

[31] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In *ECCV*, 2016. 2, 7

[32] Trevor Darrell Ronghang Hu, Marcus Rohrbach. Segmentation from natural language expressions. In *ECCV*, 2016. 1, 2, 3

[33] Arka Sadhu, Kan Chen, and Ram Nevatia. Zero-shot grounding of objects from natural language queries. In *ICCV*, 2019. 2

[34] Hengcan Shi, Hongliang Li, Fanman Meng, and Qingbo Wu. Key-word-aware network for referring expression image segmentation. In *ECCV*, 2018. 1, 2, 3

[35] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *arXiv preprint*, 2014. 6

[36] Liwei Wang, Yin Li, and Svetlana Lazebnik. Learning deep structure-preserving image-text embeddings. In *CVPR*, 2016. 2

[37] Peng Wang, Qi Wu, Jiewei Cao, Chunhua Shen, Lianli Gao, and Anton van den Hengel. Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In *CVPR*, 2019. 1, 2, 5, 7

[38] Zhengyuan Yang, Boqing Gong, Liwei Wang, Wenbing Huang, Dong Yu, and Jiebo Luo. A fast and accurate one-stage approach to visual grounding. In *ICCV*, 2019. 2, 5, 7

[39] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. Hierarchical attention networks for document classification. In *ACL*, 2016. 3

[40] Linwei Ye, Mrigank Rochan, Zhi Liu, and Yang Wang. Cross-modal self-attention network for referring image segmentation. In *CVPR*, 2019. 1, 2, 3, 5, 7

[41] Zhou Yiyi, Ji Rongrong, Gen Luo, Sun Xiaoshuai, Jinsong Su, Chia-Wen Lin, and Qi Tian. A real-time global inference network for one-stage referring expression comprehension. In *arXiv preprint*, 2019. 4

[42] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*, 2018. 1, 2

[43] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. Mattnet: Modular attention network for referring expression comprehension. In *CVPR*, 2018. 1, 2, 5, 6, 7

[44] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *ECCV*, 2016. 1, 2

[45] Licheng Yu, Hao Tan, Mohit Bansal, and Tamara L Berg. A joint speaker-listener-reinforcer model for referring expressions. In *CVPR*, 2017. 1, 2, 7

[46] Zhou Yu, Jun Yu, Chenchao Xiang, Zhou Zhao, Qi Tian, and Dacheng Tao. Rethinking diversified and discriminative proposal generation for visual grounding. In *IJCAI*, 2018. 1

[47] Hanwang Zhang, Yulei Niu, and Shihfu Chang. Grounding referring expressions in images by variational context. In *CVPR*, 2018. 2

[48] Yuting Zhang, Luyao Yuan, Yijie Guo, Zhiyuan He, Ian Huang, and Honglak Lee. Discriminative bimodal networks for visual localization and detection with natural language queries. In *CVPR*, 2017. 1, 2

[49] Bohan Zhuang, Qi Wu, Chunhua Shen, Ian D Reid, and Anton Van Den Hengel. Parallel attention: A unified framework for visual object discovery through dialogs and queries. In *CVPR*, 2018. 7