A Fast and Accurate One-Stage Approach to Visual Grounding

Zhengyuan Yang[2*] Boqing Gong[1†] Liwei Wang[1] Wenbing Huang[1] Dong Yu[1] Jiebo Luo[2]
[1]Tencent AI Lab [2]University of Rochester
{zyang39, jluo}@cs.rochester.edu, boqinggo@outlook.com
{liweiwang, dongyu}@tencent.com, hwenbing@126.com

Abstract

We propose a simple, fast, and accurate one-stage approach to visual grounding, inspired by the following insight. The performances of existing propose-and-rank two-stage methods are capped by the quality of the region candidates they propose in the first stage — if none of the candidates could cover the ground truth region, there is no hope in the second stage to rank the right region to the top. To avoid this caveat, we propose a one-stage model that enables end-to-end joint optimization. The main idea is as straightforward as fusing a text query's embedding into the YOLOv3 object detector, augmented by spatial features so as to account for spatial mentions in the query. Despite being simple, this one-stage approach shows great potential in terms of both accuracy and speed for both phrase localization and referring expression comprehension, according to our experiments. Given these results along with careful investigations into some popular region proposals, we advocate for visual grounding a paradigm shift from the conventional two-stage methods to the one-stage framework.

1. Introduction

We propose a simple, fast, and accurate one-stage approach to visual grounding, which aims to ground a natural language query (phrase or sentence) about an image onto a correct region of the image. By defining visual grounding at this level, we deliberately abstract away the subtle distinctions between phrase localization [30, 42], referring expression comprehension [15, 24, 48, 47, 22], natural language object retrieval [14, 16], visual question segmentation [9, 13, 20, 25], etc., each of which can be seen as a variation of the general visual grounding problem. We benchmark our one-stage approach for both phrase localization and referring expression comprehension. Results show that it is about 10 times faster than the state-of-the-art two-stage methods and meanwhile more accurate than them. Hence, we expect this work provides for visual grounding a new strong baseline, upon which one can conveniently build further to tackle variations (e.g., phrase localization) to the basic visual grounding problem by bringing in corresponding domain knowledge (e.g., attributes, relationship between phrases, spatial configuration of regions, etc.).

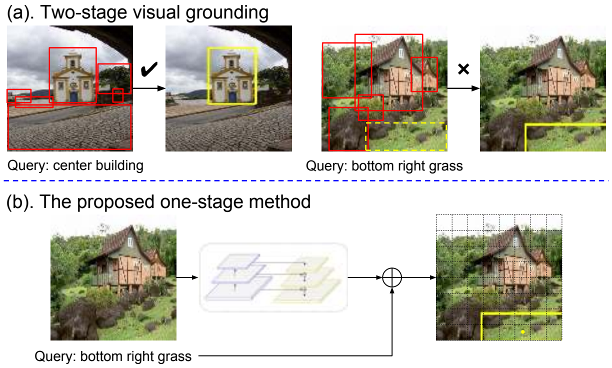

(a). Two-stage visual grounding
✔
×
Query: center building
Query: bottom right grass
(b). The proposed one-stage method
Query: bottom right grass

Figure 1. Visual grounding is the task of localizing a language query in an image. The output is often a bounding box as drawn in the yellow color. **(a).** Existing two-stage methods first extract region candidates and then rank them according to their similarities with the query. The inference speed is slow and the performance is capped by the quality of the region proposals (e.g., on the right, the "bottom right grass" is not covered by any of the region candidates). **(b).** Our proposed one-stage method directly predicts a grounding box given the input image and a query. It is hence significantly faster and also accurate in inference.

Visual grounding is key to machine intelligence and provides a natural channel for humans to communicate with machines about the physical world. Its potential applications include but are not limited to robotics, human-computer interaction, and early education. In addition, a good visual grounding model can benefit a variety of research problems such as visual question answering [53, 9, 17], image captioning [45, 1, 8], and image retrieval [37].

There are mainly two thriving threads of work in visual grounding: phrase localization [15, 30, 42] and referring expression comprehension [24, 48, 47, 14, 16] — plus some work on grounding as segmentation [13, 9, 20, 25]. The language query in the former is a local phrase of a full sentence

[*] Work done while Z.Yang was an intern at Tencent AI Lab at Bellevue.
[†] Now at Google.

describing an image, implying that multiple phrase queries could co-occur in the sentence. In the latter, the query is an expression referring to a particular region of an image through a combination of object categories, attributes, relationships with other objects, etc. Notably, in phrase localization, an image region linked to a phrase of one sentence can also be linked to a phrase of another sentence, establishing a coreference chain. Compared with phrase localization, referring expression has less ambiguity in general.

Recent advances in computer vision and natural language processing offer a rich set of tools, such as region proposals [54, 41], object detection [10, 34, 11], text embedding [28, 26, 4], syntactic parsing [39], etc., leading to methods [43, 42, 29, 2, 49, 47] exploiting various cues in the visual grounding problem. However, somehow surprisingly, the main bodies of these methods are remarkably alike: they propose multiple region candidates per image and then rank them according to their similarities with the language query. We contend that this propose-and-rank two-stage framework is flawed in at least two major ways.

- If none of the region candidates of the first stage hits the ground truth region, the whole framework fails no matter how good the second ranking stage could perform. We find that 200 Edgebox region proposals [54] per image can only hit 68% of the ground truth regions in ReferItGame [15], a benchmark dataset for referring expression comprehension. A hit is considered successful if any of the 200 proposals could reach 0.5 or higher intersection-over-union (IoU) [30] with the ground truth region.
- Most of the computation spent on the region candidates, such as generating proposals, extracting features, fusing with the query embedding, scoring similarities, and so on, are merely to rank them down to the list. After all, in most test cases, only one or two region proposals are correct. We believe this scheme is a waste of computation and should be improved.

The two caveats are left unresolved probably due to the long-standing pursuit of how to model different cues in visual grounding. In this paper, we take a step back and reexamine the visual grounding problem at an abstract level, without discriminating the query types. We propose to shift the paradigm from grounding as ranking multiple region candidates to directly proposing one region as the output.

To this end, we study an end-to-end one-stage approach to visual grounding. The main idea is as straightforward as fusing a text query's embedding into the YOLOv3 object detector [33]. Additionally, we augment the feature maps with spatial features to account for spatial mentions in the language queries (e.g., "the man on the right"). Finally, we replace the sigmoid output layer with a softmax function in order to enforce the network to generate only one image region in response to a query. Other cues explored in the two-stage methods, such as attributes, attention, bounding box annotations around extra objects, and so on, can be naturally added to our one-stage model. We focus on the vanilla model in the main text and examine its extensibilities to some other cues in supplementary materials.

The advantages of this one-stage approach are multiple-fold. First of all, it is fast in inference. It extracts features from the input image with only one pass and then directly predicts the coordinates of the output region. Without any code optimization, our implementation is about 10 times faster than state-of-the-art two-stage methods. Additionally, it is also accurate. Unlike the two-stage framework whose performance is capped by the region candidates, it enables end-to-end optimization. We show promising results on both phrase localization and referring expression comprehension. Finally, it generalizes better to different datasets than the two-stage methods because it does not depend on any additional tools or pre-trained models. Hence, we advocate this one-stage framework for future work on visual grounding and hope our approach in this work provides a new strong baseline.

2. Approach

In this section, we first review the existing two-stage frameworks for visual grounding [42, 30, 48, 24, 47, 35, 29] and then present our one-stage approach in detail.

2.1. Two-stage methods

Conventional methods for visual grounding, especially for the task of phrase localization [30, 42, 29, 2], are mainly composed of two separate stages. As shown in Figure 1, given an input image, the first step is to generate candidate regions using either unsupervised object proposal methods [54, 42, 29, 2] or a pre-trained object detection network [50, 47]. The second step is to rank the candidate regions conditioning on a language query about the image. Most existing two-stage methods differ from each other in the second step by scoring functions, network architectures, multi-task learning, and training algorithms. A number of studies [51, 42] cast the second step as a binary classification task, where a region-query pair is tagged "positive", "negative", or "ignored" based on the region's IoU with the ground truth region. The maximum-margin ranking loss is another popular choice for the second stage [24, 27, 43].

As a concrete example, we next describe the similarity network [42, 29] since it gives rise to state-of-the-art performance on benchmark datasets. The authors employ a Fast R-CNN [10, 34] pre-trained on Pascal [7] to extract visual features for each candidate region. To embed the text query, they find the Fisher encoding [28] works as well as or better than recurrent neural networks. The region features and query embeddings are fed through two network branches, respectively, before they merge by a layer of element-wise multiplication. A few nonlinear layers are added after they merge. Finally, the network outputs a similarity score by a sigmoid function. The authors train this network by a cross-entropy loss with positive labels for the matched pairs of regions and queries and negative labels for the mismatched pairs. A region is a match to the query if its IoU with the ground truth is greater than 0.7 and the regions with IoUs less than 0.3 are considered mismatches.

The overall performance of the two-stage framework is capped by the first stage. Besides, the candidate regions cause heavy computation cost. We next present a different paradigm, a one-stage visual grounding network which enables end-to-end optimization and is both fast and accurate.

2.2. Our one-stage approach

In short, our one-stage approach to the visual grounding is to fuse a text query's embedding into YOLOv3 [33], augment it with spatial features as the spatial configuration is frequently used by a query, replace its sigmoid output layer with a softmax function because we only need to return one region for a query, and finally train the network with YOLO's loss [31]. Despite being simple, this one-stage method signifies a paradigm shift away from the prevalent two-stage framework, and it gives rise to superior results in terms of both accuracy and speed.

We present this vanilla one-stage model as below and amend it in supplementary materials to account for some cues explored in the two-stage methods. Figure 2 illustrates the network architecture, mainly consisting of three feature encoding modules and three fusion modules.

Visual and text feature encoding. Our model is end-to-end, taking as input an image and a text query and then returning an image region as the response to the query. For the text query, we embed it to a 768D real-valued vector using the uncased version of Bert [4], followed by two fully connected layers either with 512 neurons. In addition, we also test the other embedding methods to fairly compare with the existing works. In particular, recent works [30, 42, 29] employ Fisher vectors of word2vec [28, 26]. Bidirectional LSTMs are adopted in [2, 35].

We use Darknet-53 [33] with feature pyramid networks [18] to extract visual features for the input image, which is resized to 256 × 256, at three spatial resolutions: $8\times 8\times D_1$, $16\times 16\times D_2$, and $32\times 32\times D_3$. In other words, the feature maps are respectively $\frac{1}{32}$, $\frac{1}{16}$, and $\frac{1}{8}$ of the original image size. There are $D_1 = 1024$, $D_2 = 512$, and $D_3 = 256$ feature channels at the three resolutions, respectively. We add a 1 × 1 convolution layer with batch normalization and RELU to map them all to the same dimension $D = 512$.

Spatial feature encoding. We find that the text queries often use spatial configurations to refer to objects, such as "the man on the left" and "the bottom right grass". However, the Darknet-53 features mainly capture visual appearances, lacking the position information. Hence, we explicitly encode some spatial features for each position of the three spatial resolutions. Specifically, as shown in Figure 2, we generate a coordinate map of size $W' \times H' \times D_{\text{spatial}}$ at each resolution, where W' and H' are the spatial size of a visual feature map, i.e., 8×8, 16×16, or 32×32, and $D_{\text{spatial}} = 8$ indicating we encode eight spatial features. If we place the feature map in a coordinate system such that its top-left and bottom-right corners lie at $(0,0)$ and $(1,1)$, respectively, the eight features for any position (i,j), $i \in \{0,1,\cdots,W'-1\}$ and $j \in \{0,1,\cdots,H'-1\}$, are calculated as follows:

$$\Big(\frac{i}{W'}, \frac{j}{H'}, \frac{i+0.5}{W'}, \frac{j+0.5}{H'}, \frac{i+1}{W'}, \frac{j+1}{H'}, \frac{1}{W'}, \frac{1}{H'}\Big),$$

which captures the coordinates of the top-left corner, center, and bottom-right corner of the grid at (i,j), along with the inverse of W' and H'.

Fusion. We use the same operation to fuse the visual, text, and spatial features at the three spatial resolutions. In particular, we first broadcast the query embedding to each spatial location (i,j) and then concatenate it with the visual and spatial features, giving rise to a $512+512+8 = 1032$D feature vector. The visual, text, and spatial features are ℓ_2 normalized respectively before the concatenation. We add a 1 × 1 convolution layer to better fuse them at each location independently. We have also tested 3 × 3 convolution kernels hoping to make the fusion aware of the neighborhood structure, and yet the results are about the same as the 1 × 1 fusion. After this fusion step, we have a 512D feature vector for each location of the three spatial resolutions, i.e., three feature blobs of the sizes $8\times 8\times 512$, $16\times 16\times 512$, and $32\times 32\times 512$, respectively.

Grounding. The grounding module takes the fused features as input and generates a box prediction to ground the language query onto an image region. We design this module by following YOLOv3's output layer except that we 1) recalibrate its anchor boxes and 2) change its sigmoid layer to a softmax function.

There are $8\times 8+16\times 16+32\times 32 = 1344$ locations out of the three spatial resolutions — and each location is associated with a 512D feature vector as a result of the fusion module. YOLOv3 centers around each of the locations three anchor boxes. To better fit our grounding datasets, we customize the widths and heights of the anchors by K-means clustering over all the ground truth grounding boxes in the training set with $(1-\text{IoU})$ as the distance [32, 33].

There are (3 anchors per location × 1344 locations) $=$ 4032 anchor boxes in total. What YOLOv3 predicts is, out of each anchor box, four quantities by regression for shifting the center, width, and height of the anchor box, along with the fifth quantity by a sigmoid function about the confidence on this shifted box. We keep the regression branch as is. As only one region is desired as the output for grounding

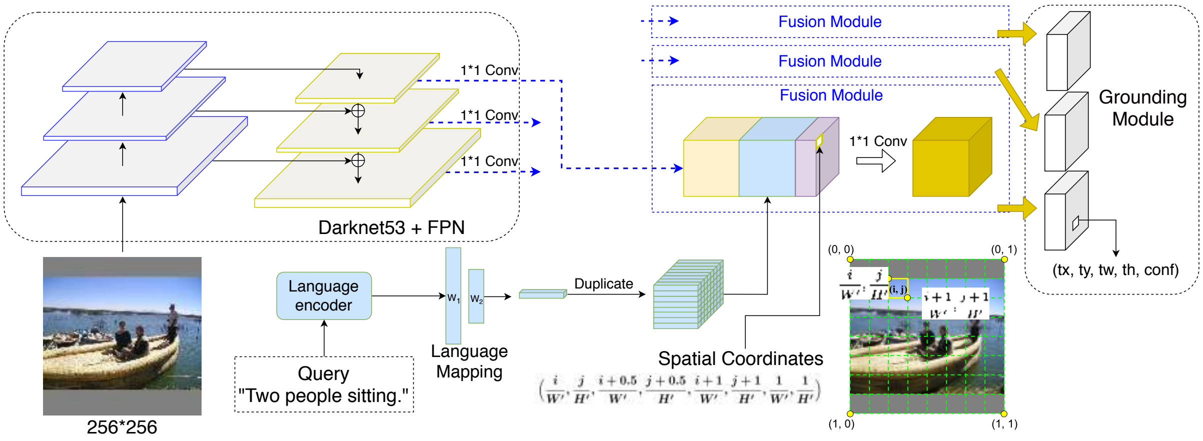


Figure 2. The proposed end-to-end one-stage visual grounding framework.

the query — at least according to the current formalization of the visual grounding problem, we replace the sigmoid functions with a softmax function over all the 4032 boxes. Accordingly, we replace the loss function over the confidence scores by a cross entropy between this softmax and a one-hot vector — the anchor box which has the highest IoU with the ground truth region is labeled 1 and all the others are labeled 0. We refer readers to [33] for more details.

2.3. Comparison to other one-stage grounding work

We contrast our approach to some closely related works, including two existing one-stage grounding methods [52, 44] and some on grounding as segmentation [13, 20, 25, 9].

The Interpretable and Globally Optimal Prediction (IGOP) [44] also tries to solve supervised visual grounding in a one-stage manner. IGOP employs feature maps from multiple vision tasks (e.g., object detection, semantic segmentation, pose estimation, etc.) and models the phrase localization task as finding a box on the feature maps which encapsulates the smallest energy. Since IGOP relies on multiple extra pre-trained vision models, it is not clear how to optimize it end-to-end.

The Multiple-scale Anchored Transformer Network (MATN) [52] is also a one-stage grounding model. However, many design consideratons of this network are to account for weakly supervised visual grounding. Besides, MATN directly predicts a single box as the output, essentially searches for one box out of a huge search space at the scale $O(W^2H^2)$, where W, H are width and height of the input image. This scheme has been shown inferior to those based on anchor boxes in object detection [32, 33], unless one has sufficiently big training sets.

We also briefly discuss some works on grounding text queries to segmentation masks [13, 20, 25, 9]. Due to the irregular shapes of the segmentation masks, it is hard to follow the propose-and-rank two-stage framework to output segmentation masks. Instead, they naturally employ one-stage frameworks. However, their network architectures, especially the output layer, are very different from ours.

3. Experiments

3.1. Datasets and experiment protocols

We evaluate the proposed one-stage visual grounding approach on the Flickr30K Entities dataset [30] and the ReferItGame dataset [15]. The supplementary materials contain additional results on RefCOCO [48]. Flickr30K Entities augments the original Flickr30K [46] with region-phrase correspondence annotations. It links 31,783 images in Flickr30K with 427K referred entities. We follow the same training/validation/test split used in the previous work [30] in our experiments. ReferItGame [15] has 20,000 images from the SAIAPR-12 dataset [6]. We employ a cleaned version of the split provided by [14], which has 9,000, 1,000 and 10,000 images in the training, validation, and test sets, respectively. Following the same evaluation protocol in prior works [30, 35], given a language query, an output image region is considered correct if its IoU is at least 0.5 with the ground truth bounding box.

Some details of our model architecture. We use Darknet-53 [33] pre-trained on COCO object detection [19] as the visual encoder. To embed the language queries, we test Bert [4], a bi-LSTM framework used in [2], and a Fisher vector encoding used in [30, 42, 29]. We generate the anchor boxes by K-means clustering following the procedure of [32, 33]. The anchors on ReferitGame are $(18 \times 22), (48 \times 28), (29 \times 52), (91 \times 48), (50 \times 91), (203 \times 57), (96 \times 127), (234 \times 100), (202 \times 175)$ and on Flickr30K Entities are $(17 \times 16), (33 \times 35), (84 \times 43), (50 \times 74), (76 \times 126), (125 \times 81), (128 \times 161), (227 \times 104), (216 \times 180)$.

Training details. We keep the original image ratio when we resize an input image. We resize its long edge to 256 and then pad the image pixels' mean value along the short edge so that the final image size is 256×256. We follow [33] for data augmentation, i.e., adding randomization to the color space (saturation and intensity), horizontal flip, and random affine transformations. We train the model with RMSProp [40] optimization. We start with a learning rate of 10^{-4} and follow a polynomial schedule with a power of 1. As the Darknet has been pre-trained, we multiply the main learning rate by 0.1 for the Darknet portion of our model. The batch size is 32 in all our experiments. We observe about 1% improvement when we use larger batch sizes on a workstation with eight P100 GPUs, but we opt to report the results of the small batch size (32) so that one can easily reproduce our results on a desktop with two GPUs.

Competing baselines beyond existing methods. We compare with state-of-the-art visual grounding methods, about which the descriptions are deferred to Section 3.2. Beyond them, we also systematically study the following baselines and variations of our approach.

- **Similarity Net-Darknet.** Previous two-stage methods often use detection networks with a VGG-16 backbone [38] to extract visual features, while Darknet is adopted in our model. Naturally, one may wonder about the influence of the backbones in addition to the framework change from the two stages to the one stage. To single out the influence of the backbone networks, we construct a baseline using the two-stage similarity network [42] based on the Darknet visual features, modifying the code released with [29]. We first pool region features from all three feature blobs output by the feature pyramid network in YOLOv3, then ℓ_2 normalize them, respectively, and finally concatenate them as the visual features.
- **Similarity Net-Resnet.** We also test in the similarity network visual features extracted by Mask R-CNN [11] with a Resnet-101 [12] backbone, which is pre-trained on COCO detection. The feature dimension is 2048.
- **CITE-Resnet.** Furthermore, we compare to CITE [29] with Resnet-101 features. We keep the number of embeddings as the default value $K = 4$ in CITE. Region proposals and visual and language encoders remain the same as "Similarity Net-Resnet".
- **Ours-FV.** The Fisher vector (FV) encoding [28] of word2vec [26] features is used in some state-of-the-art visual grounding methods [30, 42, 29]. We include it in our approach as well. A language query is encoded to a 6000D FV embedding.
- **Ours-LSTM.** The LSTM encoding of language queries is also frequently used in the literature [2, 35], so we investigate its effect in our approach as well. We use a bi-LSTM layer with 512D hidden states in this work. We do not use word2vec features to initialize the embedding layer.
- **Ours-Bert.** We use the uncased version of Bert [4] that outputs a 768D embedding as our main language query encoder. We do not update the Bert parameters during training.
- **Ours-Bert-no Spatial.** In this ablated version of our approach, we remove the spatial features and only fuse the visual and text features.

3.2. Visual grounding results

Flickr30K Entities. Table 1 reports the phrase localization results on the Flickr30K Entities dataset. The **top portion** of the table contains the numbers of several state-of-the-art visual grounding methods [14, 43, 35, 30, 44, 2, 42, 29]. The results of two additional versions of the similarity network [42], respectively based on Resnet and Darkent, are shown in the **middle** of the table. Finally, the four rows at the **bottom** are different variations of our own approach.

We list in the "Region Proposals" column different region proposal techniques used in the visual grounding methods, followed by the number of region candidates per image (e.g., N=100). Edgebox [54] and selective search [41] are two popular options for proposing the regions. In the "Visual Features" column, we list the backbone networks followed by the datasets on which they are pre-trained. The "Language Embedding" column indicates the query embedding adopted by each grounding method.

Among the two-stage methods, not surprisingly, "Similarity Net-Resnet" gives much better results than "Similarity Net" because the Resnet visual features are generally higher-quality than the VGG features.

Although Darknet-53 and Resnet-101 give rise to comparable results on ImageNet [36], the Darknet features lead to poor visual grounding results. This is reasonable because Darknet does not have a separate region proposal network, making it tricky to extract the region features. Furthermore, the large down-scale ratios (1/8, 1/16, and 1/32) and the low feature dimensions (256, 512, 1024) of Darknet make its region features not as discriminative as Resnet's.

Our one-stage method and its variations outperform the two-stage approaches with large margins. By the last two rows of the table, we investigate the effectiveness of our spatial features. It is clear that the spatial information boosts the accuracy of "Ours-Bert-no Spatial" by about 1.6%. Finally, we note that language embedding techniques only slightly influence the results within a small range.

ReferitGame. Table 2 reports the referring expression comprehension results on ReferItGame [15]. Organizing the results in the same way as Table 1, the top portion of the table is about state-of-the-art grounding methods [14, 23, 50, 35, 44, 2, 42, 29], the middle is two versions of the similarity network, and the bottom shows our results.

We draw from Table 2 about the same observation as from Table 1. In general, our model with Darknet visual fea-

Table 1. Phrase localization results on the test set of Flickr30K Entities [30].

Method	Region Proposals	Visual Features	Language Embedding	Accu@0.5	Time (ms)
SCRC [14]	Edgebox N=100	VGG16-Imagenet	LSTM	27.80	-
DSPE [43]	Edgebox N=100	VGG19-Pascal	Word2vec, FV	43.89	-
GroundeR [35]	Selec. Search N=100	VGG16-Pascal	LSTM	47.81	-
CCA [30]	Edgebox N=200	VGG19-Pascal	Word2vec, FV	50.89	-
IGOP [44]	None	Multiple Network	N-hot	53.97	-
MCB + Reg + Spatial [2]	Selec. Search N=100	VGG16-Pascal	LSTM	51.01	-
MNN + Reg + Spatial [2]	Selec. Search N=100	VGG16-Pascal	LSTM	55.99	-
Similarity Net [42]	Edgebox N=200	VGG19-Pascal	Word2vec, FV	51.05	-
Similarity Net by CITE [29]	Edgebox N=200	VGG16-Pascal	Word2vec, FV	54.52	-
CITE [29]	Edgebox N=500	VGG16-Pascal	Word2vec, FV	59.27	-
CITE [29]	Edgebox N=500	VGG16-Flickr30K	Word2vec, FV	61.89	-
Similarity Net-Resnet [42]	Edgebox N=200	Res101-COCO	Word2vec, FV	60.89	184
CITE-Resnet [29]	Edgebox N=200	Res101-COCO	Word2vec, FV	61.33	196
Similarity Net-Darknet [42]	Edgebox N=200	Darknet53-COCO	Word2vec, FV	41.04	305
Ours-FV	None	Darknet53-COCO	Word2vec, FV	68.38	**16**
Ours-LSTM	None	Darknet53-COCO	LSTM	67.62	21
Ours-Bert-no Spatial	None	Darknet53-COCO	Bert	67.08	38
Ours-Bert	None	Darknet53-COCO	Bert	**68.69**	38

Table 2. Referring expression comprehension results on the test set of ReferItGame [15].

Method	Region Proposals	Visual Features	Language Embedding	Accu@0.5	Time (ms)
SCRC [14]	Edgebox N=100	VGG16-Imagenet	LSTM	17.93	-
GroundeR + Spacial [35]	Edgebox N=100	VGG16-Pascal	LSTM	26.93	-
VC [50]	SSD Detection [21]	VGG16-COCO	LSTM	31.13	-
CGRE [23]	Edgebox	VGG16	LSTM	31.85	-
MCB + Reg + Spatial [2]	Edgebox N=100	VGG16-Pascal	LSTM	26.54	-
MNN + Reg + Spatial [2]	Edgebox N=100	VGG16-Pascal	LSTM	32.21	-
Similarity Net by CITE [29]	Edgebox N=500	VGG16-Pascal	Word2vec, FV	31.26	-
CITE [29]	Edgebox N=500	VGG16-Pascal	Word2vec, FV	34.13	-
IGOP [44]	None	Multiple Network	N-hot	34.70	-
Similarity Net-Resnet [42]	Edgebox N=200	Res101-COCO	Word2vec, FV	34.54	184
CITE-Resnet [29]	Edgebox N=200	Res101-COCO	Word2vec, FV	35.07	196
Similarity Net-Darknet [42]	Edgebox N=200	Darknet53-COCO	Word2vec, FV	22.37	305
Ours-FV	None	Darknet53-COCO	Word2vec, FV	59.18	**16**
Ours-LSTM	None	Darknet53-COCO	LSTM	58.76	21
Ours-Bert-no Spatial	None	Darknet53-COCO	Bert	58.16	38
Ours-Bert	None	Darknet53-COCO	Bert	**59.30**	38

tures and Bert query embeddings outperforms the existing methods by large margins. Careful analyses reveal that the poor region candidates in the first stage are a major reason that the two-stage methods underperform ours. We present these analyses in Section 3.4.

3.3. Inference time comparison

A fast inference speed is one of the major advantages of our one-stage method. We list the inference time in the rightmost columns of Tables 1 and 2, respectively. We conduct all the tests on a desktop with Intel Core i9-9900K@3.60GHz and NVIDIA 1080TI. Typical two-stage approaches generally take more than 180ms to process one image-query pair, and they spend most of the time on generating region candidates and extracting features for them. In contrast, our one-stage approaches all take less than 40ms to ground a language query to an image — especially, "Ours-FV" takes only 16ms, making it potentially feasible for real-time applications.

3.4. Oracle analyses about the region candidates

Why could the one-stage methods achieve those big improvements over the two-stage ones? We conjecture that it is mainly because our one-stage framework can avoid imperfect region candidates. In contrast, the performances of the two-stage methods are capped by the **hit rate** of the region candidates they propose in the first stage. We say a

Table 3. Hit rates of region proposal methods.

Hit rate, N=200	Flickr30K Entities		ReferitGame	
	val set	test set	val set	test set
MRCN Detect. [11]	48.76	49.28	27.63	28.12
MRCN RP [11]	76.40	76.60	44.80	46.50
Edgebox [54]	82.91	83.69	68.62	68.26
Selec. Search [41]	84.85	85.68	81.67	80.34
Ours	**95.32**	**95.48**	**92.40**	**91.32**

ground truth region is hit by the region candidates if its IoU is greater than 0.5 with any of the candidates, and the hit rate is the number of ground truth regions hit by the candidates divided by the total number of ground truth regions.

We study the hit rates of some popular region proposal methods: Edgebox [54], selective search [41], the region proposal network in Mask R-CNN [11] pre-trained on COCO [11], Mask R-CNN itself whose detection results are regarded as region candidates, and our one-stage approach whose box predictions are considered the region candidates. We keep top N=200 region candidates for each of them or as many regions as possible if it outputs less than 200 regions.

Table 3 shows the hit rates on both Flickr30K Entities and ReferItGame. It is interesting to see that the hit rates are in general higher on Flickr30 than on ReferItGame, especially when Edgebox generated proposals are used, somehow explaining why the two-stage grounding results on Flickr30K Entities (Table 1) are better than those on ReferItGame (Table 2). Another notable observation is that the top 200 boxes of our approach have much higher hit rates than the other techniques, verifying the benefit of learning in an end-to-end way.

One may wonder under what scenarios the region proposal methods but ours fail to hit the ground truth regions. Figure 3 gives some insights by showing the Edgebox region candidates on ReferItGame. We find that the region candidates mainly fail to hit stuff ground truth regions (e.g., the “grass” in Figure 3 (c)). Tiny objects are also hard to hit (cf. Figure 3 (e) and (f)). Finally, when a query refers to more than one objects, it might fail region proposal methods which are mostly designed to place a tight bounding box around only one object (cf. Figure 3 (a) and (b)).

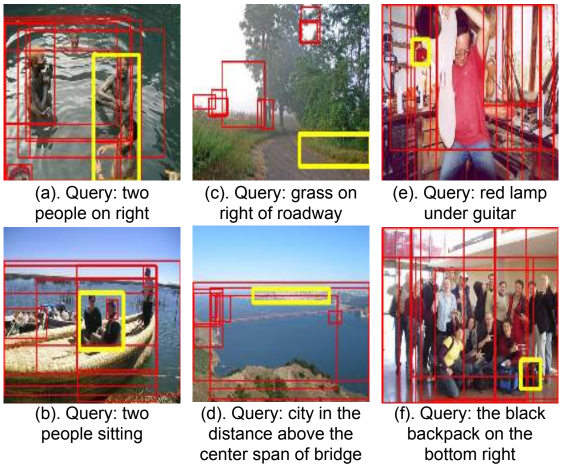
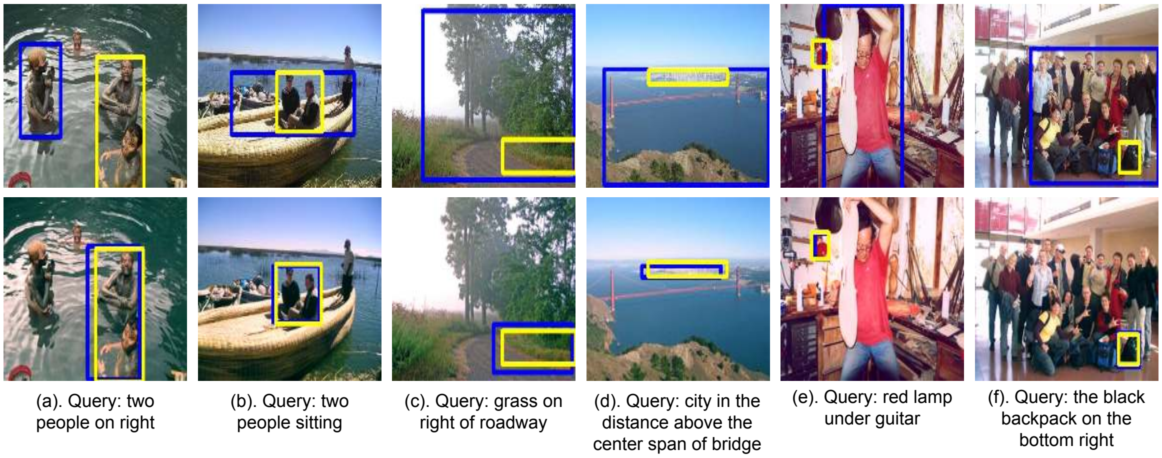

(a). Query: two people on right
(c). Query: grass on right of roadway
(e). Query: red lamp under guitar
(b). Query: two people sitting
(d). Query: city in the distance above the center span of bridge
(f). Query: the black backpack on the bottom right

Figure 3. Failure cases of the Edgebox region candidates (boxes shown in the red color) on ReferitGame. The yellow boxes are ground truths. For visualization purpose, we randomly hide some region candidates.

3.5. Qualitative results analyses

In this section, we analyze the success and failure cases of the two-stage similarity network as well as our model to show the advantages and limitations of the proposed one-stage method. Figure 4 shows the mistakes made by the similarity network that can be corrected by our method. The blue boxes are predictions, and the yellow boxes represent the ground truths. We group some common mistakes into the following scenarios.

- **Queries referring to multiple objects.** A language query in the visual grounding problem can refer to more than one objects, but the region proposals by design aim to each cover only one object. Check the queries “two people on right” and “two people sitting” in Figures 4 (a) and (b), respectively, for examples. it is (almost) impossible to overcome this type of mismatches by the existing propose-and-rank two-stage methods. In contrast, our approach is not restricted to one object per box at all and, instead, can flexibly adapt the box regression function according to the queries.
- **Queries referring to stuff as opposed to things.** The second kind of common errors made by the two-stage methods is on the queries referring to stuff as opposed to things, such like the “grass” and “city” shown in Figures 4 (c) and (d), respectively. This kind of errors is again due to that the region proposals mostly focus on thing classes — the “objectness” is often an important cue for proposing the regions. In sharp contrast, stuff regions generally have low “objectness” scores. Hence, we argue that the two-stage methods are incapable of handling such stuff regions given the status quo of region proposal techniques. Our one-stage method can instead learn to handle the stuff regions from the training set of the visual grounding datasets.
- **Challenging regions.** In the third kind of common errors, the two-stage methods fail to deal with challenging test cases, such as the small regions referred to by the queries in Figures 4 (e) and (f). There are mainly three reasons that fail the two-stage methods. First, the region candidates of the first stage may not provide a good coverage especially over small objects. Second, the visual features of small regions are not discriminative enough for the second stage to learn how to rank. Third, the image depicts complicated scenes or many duplicated objects. The last point could equally harm our approach as well as the two-stage ones.



(a). Query: two people on right

(b). Query: two people sitting

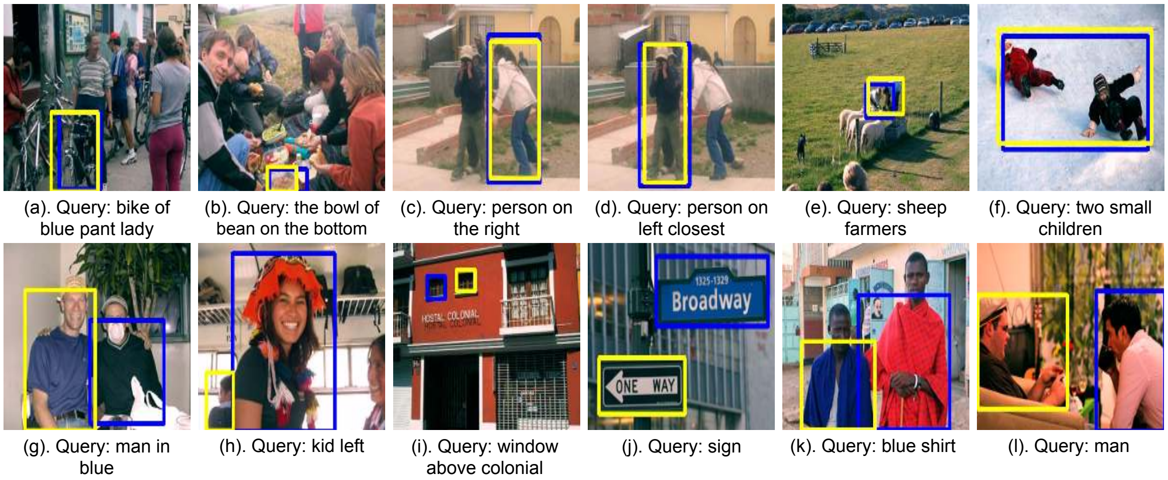
(c). Query: grass on right of roadway

(d). Query: city in the distance above the center span of bridge

(e). Query: red lamp under guitar

(f). Query: the black backpack on the bottom right

Figure 4. Mistakes made by the two-stage similarity network (top row) that can be corrected by our one-stage approach (bottom row). Blue boxes are predicted regions and yellow boxes are the ground truth. There are three types of common failures of the two-stage method: queries referring to multiple objects (a,b), queries referring to stuff regions (c, d), and challenging regions (e, f).



(a). Query: bike of blue pant lady

(b). Query: the bowl of bean on the bottom

(c). Query: person on the right

(d). Query: person on left closest

(e). Query: sheep farmers

(f). Query: two small children

(g). Query: man in blue

(h). Query: kid left

(i). Query: window above colonial

(j). Query: sign

(k). Query: blue shirt

(l). Query: man

Figure 5. Success cases on challenging instances (top row) and common failures (bottom row) of our one-stage method. Blue / yellow boxes are predicted regions / ground truths. The four columns on the left are from ReferitGame and the others are from Flickr30K Entities.

Failure cases of our approach. Figure 5 shows extra success and failure cases of our approach. The first row shows the typical success cases. The "bike of blue pant lady" in Figure 5 (a) queries an example image with multiple objects of the same class. Figure 5 (b) provides an example of correct predictions on tiny objects. (c) and (d) showcase our approach is able to interpret location information in the queries. The query in (e) contains a distracting noun, "sheep". Our model in (f) successfully predicts a region containing two objects.

Figures 5 (g)–(l) are some failure cases of our model. We find our model is insensitive to attributes, such as the "blue" in (g) and (k). It fails on (h) and (i) simply because those are very difficult test cases (e.g., one has to recognize the word "colonial" in the image in order to make the right prediction). Finally, (j) and (l) give two ambiguous queries for which our model happens to predict different boxes from those annotated by users.

4. Conclusion

We have proposed a simple and yet effective one-stage method for visual grounding. We merge language queries and spatial features into the YOLOv3 object detector and build an end-to-end trainable visual grounding model. It is about 10 times faster than state-of-the-art two-stage methods and achieves superior grounding accuracy. Besides, our analyses reveal that existing region proposal methods are generally not good enough, capping the performance of the two-stage methods and indicating the need of a paradigm shift to the one-stage framework. In future work, we plan to investigate the extensibility of the proposed one-stage framework for modeling other cues in the visual grounding problem.

Acknowledgment

This work is partially supported by NSF awards #1704337, #1722847, and #1813709.

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018. 1

[2] Kan Chen, Rama Kovvuri, Jiyang Gao, and Ram Nevatia. Msrc: Multimodal spatial regression with semantic context for phrase grounding. In *Proceedings of the 2017 ACM on International Conference on Multimedia Retrieval*, pages 23–31. ACM, 2017. 2, 3, 4, 5, 6

[3] Kan Chen, Rama Kovvuri, and Ram Nevatia. Query-guided regression network with context policy for phrase grounding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 824–832, 2017. 12, 13

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 2, 3, 4, 5

[5] Pelin Dogan, Leonid Sigal, and Markus Gross. Neural sequential phrase grounding (seqground). In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4175–4184, 2019. 12, 13

[6] Hugo Jair Escalante, Carlos A Hernández, Jesus A Gonzalez, Aurelio López-López, Manuel Montes, Eduardo F Morales, L Enrique Sucar, Luis Villaseñor, and Michael Grubinger. The segmented and annotated iapr tc-12 benchmark. *Computer Vision and Image Understanding*, 114(4):419–428, 2010. 4

[7] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 2

[8] Yang Feng, Lin Ma, Wei Liu, and Jiebo Luo. Unsupervised image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2019. 1

[9] Chuang Gan, Yandong Li, Haoxiang Li, Chen Sun, and Boqing Gong. Vqs: Linking segmentations to questions and answers for supervised attention in vqa and question-focused semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1811–1820, 2017. 1, 4

[10] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 2

[11] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2, 5, 7

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 5

[13] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. In *European Conference on Computer Vision*, pages 108–124. Springer, 2016. 1, 4

[14] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. Natural language object retrieval. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4555–4564, 2016. 1, 4, 5, 6

[15] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014. 1, 2, 4, 5, 6, 12, 13

[16] Jianan Li, Yunchao Wei, Xiaodan Liang, Fang Zhao, Jianshu Li, Tingfa Xu, and Jiashi Feng. Deep attribute-preserving metric learning for natural language object retrieval. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 181–189. ACM, 2017. 1

[17] Qing Li, Jianlong Fu, Dongfei Yu, Tao Mei, and Jiebo Luo. Tell-and-answer: Towards explainable visual question answering using attributes and captions. In *Proceedings of the conference on empirical methods in natural language processing (EMNLP)*, 2018. 1

[18] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017. 3

[19] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 4, 12

[20] Chenxi Liu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, and Alan Yuille. Recurrent multimodal interaction for referring image segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1271–1280, 2017. 1, 4

[21] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016. 6, 12

[22] Xihui Liu, Zihao Wang, Jing Shao, Xiaogang Wang, and Hongsheng Li. Improving referring expression grounding with cross-modal attention-guided erasing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1950–1959, 2019. 1

[23] Ruotian Luo and Gregory Shakhnarovich. Comprehension-guided referring expressions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7102–7111, 2017. 5, 6

[24] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 11–20, 2016. 1, 2

[25] Edgar Margffoy-Tuay, Juan C Pérez, Emilio Botero, and Pablo Arbeláez. Dynamic multimodal instance segmentation

guided by natural language queries. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 630–645, 2018. 1, 4

[26] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013. 2, 3, 5

[27] Varun K Nagaraja, Vlad I Morariu, and Larry S Davis. Modeling context between objects for referring expression understanding. In *European Conference on Computer Vision*, pages 792–807. Springer, 2016. 2

[28] Florent Perronnin, Jorge Sánchez, and Thomas Mensink. Improving the fisher kernel for large-scale image classification. In *European conference on computer vision*, pages 143–156. Springer, 2010. 2, 3, 5

[29] Bryan A. Plummer, Paige Kordas, M. Hadi Kiapour, Shuai Zheng, Robinson Piramuthu, and Svetlana Lazebnik. Conditional image-text embedding networks. In *ECCV*, 2018. 2, 3, 4, 5, 6

[30] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *International journal of computer vision*, 123(1):74–93, 2017. 1, 2, 3, 4, 5, 6, 12, 13

[31] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 3

[32] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017. 3, 4

[33] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2, 3, 4, 5

[34] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015. 2, 12

[35] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision*, pages 817–834. Springer, 2016. 2, 3, 4, 5, 6

[36] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015. 5

[37] Amaia Salvador, Xavier Giró-i Nieto, Ferran Marqués, and Shin'ichi Satoh. Faster r-cnn features for instance search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 9–16, 2016. 1

[38] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 5

[39] Richard Socher, John Bauer, Christopher D Manning, et al. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 455–465, 2013. 2

[40] Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012. 5

[41] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013. 2, 5, 7, 12

[42] Liwei Wang, Yin Li, Jing Huang, and Svetlana Lazebnik. Learning two-branch neural networks for image-text matching tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(2):394–407, 2018. 1, 2, 3, 4, 5, 6, 12

[43] Liwei Wang, Yin Li, and Svetlana Lazebnik. Learning deep structure-preserving image-text embeddings. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5005–5013, 2016. 2, 5, 6

[44] Raymond Yeh, Jinjun Xiong, Wen-Mei Hwu, Minh Do, and Alexander Schwing. Interpretable and globally optimal prediction for textual grounding using image concepts. In *Advances in Neural Information Processing Systems*, pages 1912–1922, 2017. 4, 5, 6

[45] Quanzeng You, Hailin Jin, Zhaowen Wang, Chen Fang, and Jiebo Luo. Image captioning with semantic attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4651–4659, 2016. 1

[46] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. 4

[47] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1307–1315, 2018. 1, 2, 12

[48] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *European Conference on Computer Vision*, pages 69–85. Springer, 2016. 1, 2, 4, 12, 13

[49] Licheng Yu, Hao Tan, Mohit Bansal, and Tamara L Berg. A joint speaker-listener-reinforcer model for referring expressions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7282–7290, 2017. 2, 12

[50] Hanwang Zhang, Yulei Niu, and Shih-Fu Chang. Grounding referring expressions in images by variational context. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4158–4166, 2018. 2, 5, 6, 12

[51] Yuting Zhang, Luyao Yuan, Yijie Guo, Zhiyuan He, I-An Huang, and Honglak Lee. Discriminative bimodal networks for visual localization and detection with natural language queries. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 557–566, 2017. 2

[52] Fang Zhao, Jianshu Li, Jian Zhao, and Jiashi Feng. Weakly supervised phrase localization with multi-scale anchored transformer network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5696–5705, 2018. 4

[53] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4995–5004, 2016. 1

[54] C Lawrence Zitnick and Piotr Dollár. Edge boxes: Locating object proposals from edges. In *European conference on computer vision*, pages 391–405. Springer, 2014. 2, 5, 7, 12