YORO - Lightweight End to End Visual Grounding

Chih-Hui Ho[1⋆], Srikar Appalaraju[2], Bhavan Jasani[2], R. Manmatha[2], and Nuno Vasconcelos[1]

UC San Diego[1] AWS AI Labs[2]
{chh279,nvasconcelos}@ucsd.edu {srikara,bjasani,manmatha}@amazon.com

A Limitation and Broader Impact

In this work, we designed a visual grounding (VG) model which has less parameters than state-of-the-art and has inference speed much faster than real-time (15 fps). YORO is the fastest VG methods in the literature and out-performs single-stage VG approaches by large margins. It also achieves better speed/accuracy trade-off than multi-stage VG models with 3.3× to 14× faster speed.

We believe our approach being light-weight will facilitate visual grounding (VG) in devices with less computation resources, such as edge devices, mobile robots, low-powered devices (e.g. Raspberry-Pi), and AR/VR devices. In addition, the proposed model requires less computation resources which is likely to be environmental friendly. These aspects warrant further research and consideration.

B Patch-Text Supervision

Figure 1 shows the predicted bounding box (blue) and ground truth box (red) of a same image across different epochs. It can be observed that the predicted box gradually becomes more accurate from epoch 0 to epoch 40 and shifts around various locations in different epochs. On the contrary, the green ground truth patches, that have IOU$\geq$ 0.5 with the ground truth box, stay consistent throughout the training epochs and make the proposed patch-text alignment loss a more stable. This shows the need for proposed novel patch-text loss.

C Parameter Size / Accuracy and Speed /Accuracy Plots

More comparisons on Parameter size vs Accuracy and Inference Speed vs Accuracy are provided here (w.r.t. Sec. 5 in main paper). We hope the community is inspired to consider other factors which make the technology useful (like inference speed and parameters) and not just on only Accuracy and the race to beat state-of-the-art in performance metrics.

⋆ Work done during an internship at Amazon.

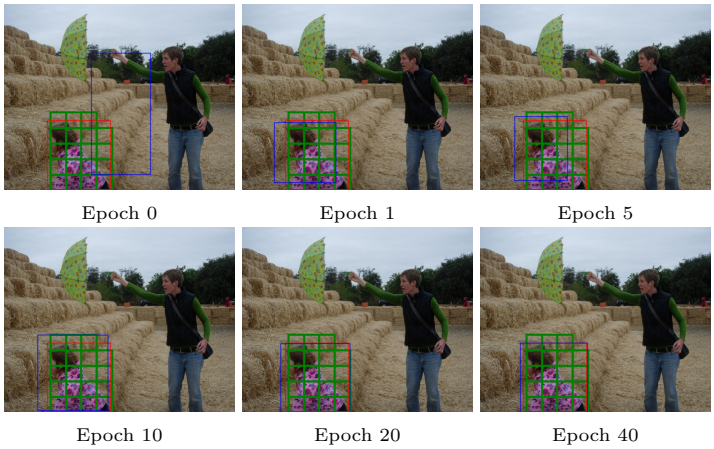


Fig. 1: **Patch-Text Supervision**: Need for consistent supervision for patch-alignment loss. Ground truth box is marked in red and green patches have overlapped with red box with IOU$\geq$ 0.5. Blue box is the prediction across different epochs.

Figure 2, 3 and 4 illustrate the trade-off between the accuracy and the inference speed in terms of FPS for RefCoco, RefCoco+ and RefCocog, respectively. YORO achieves better trade-off between the speed and accuracy in different datasets.

Figure 5, 6 and 7 illustrate the trade-off between the accuracy and the number of parameter used by each baseline for RefCoco, RefCoco+ and RefCocog, respectively. YORO achieves competitive accuracy with smaller model size.

This indicates YORO is more suitable for edge devices used in robotic applications and AR/VR devices.

D Qualitative Results

More qualitative visualizations from YORO are shown here (w.r.t Sec. 5 in main paper). Keeping transparency in mind, we show random correct and incorrect samples.

Correct Examples. Figure 8 contains correct detection results from YORO output. Examples from RefCoco and RefCoco+ have shorter query text, while examples from RefCocog are associated to more descriptive text. YORO can handle query text of various length and correctly detect the referent when the query contains digits. While the odd rows contain the predicted bounding box, the even rows show the patches that higher attention weight. More specifically, assuming the k^{th} detection token corresponds to the predicted bounding box, we extract the transformer attention weight between each patch and the k^{th} detection token. The attention weight is then converted to heatmap and overlaid on the input image. Take Figure 8(d) for example. Most patches around the bottom orange have higher attention weight.

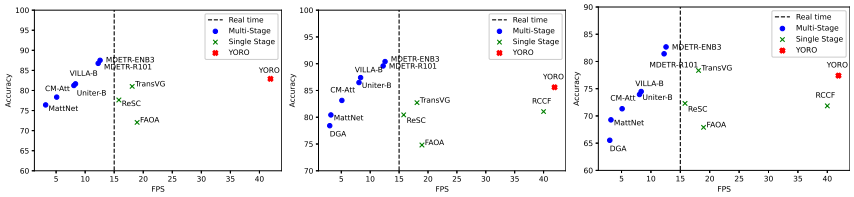


Fig. 2: Speed Accuracy Plot on RefCoco val, testA and testB dataset.

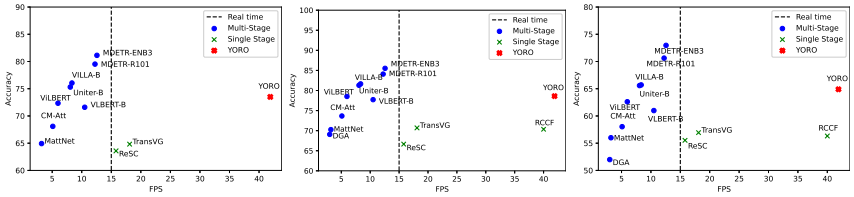


Fig. 3: Speed Accuracy Plot on RefCoco+ val, testA and testB dataset.

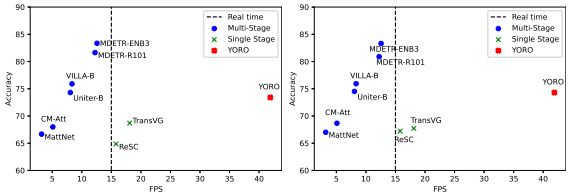


Fig. 4: Speed Accuracy Plot on RefCocog val, test dataset.

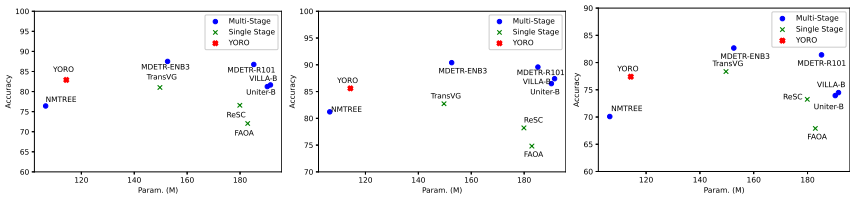


Fig. 5: Memory Accuracy Plot on RefCoco val, testA and testB dataset

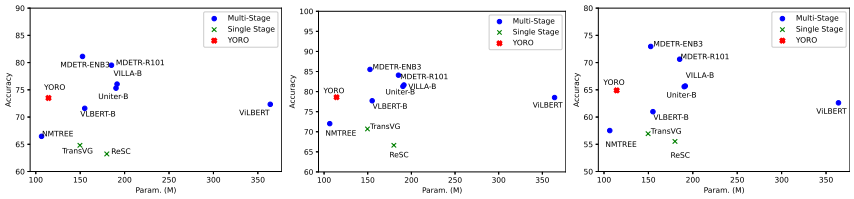


Fig. 6: Memory Accuracy Plot on RefCoco+ val, testA and testB dataset

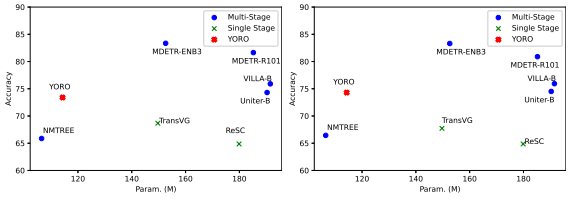
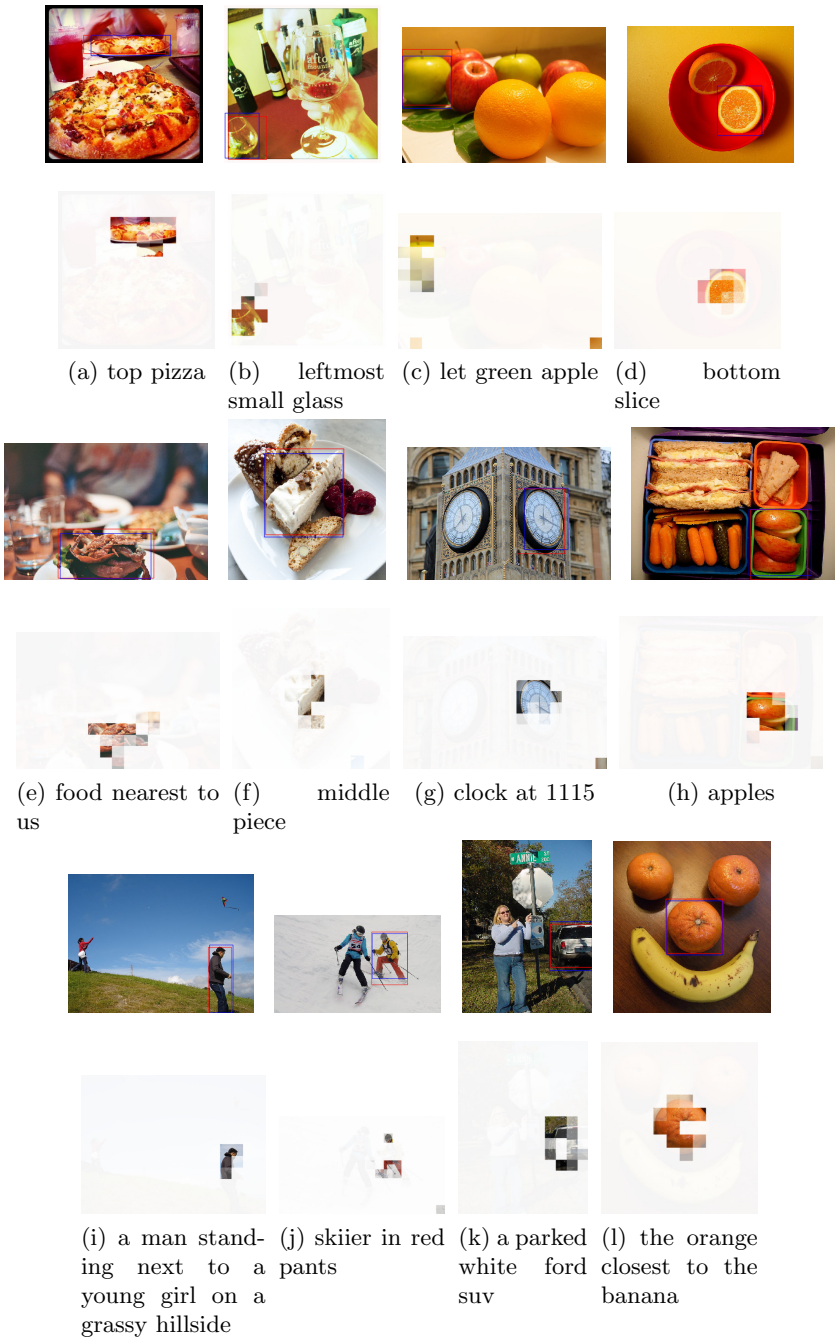


Fig. 7: Memory Accuracy Plot on RefCocog val, test dataset



(a) top pizza (b) leftmost small glass (c) let green apple (d) bottom slice

(e) food nearest to us (f) middle piece (g) clock at 1115 (h) apples

(i) a man standing next to a young girl on a grassy hillside (j) skiier in red pants (k) a parked white ford suv (l) the orange closest to the banana

Fig. 8: Visualization of the predicted box (blue) and ground truth box (red) in Refcoco (a-d)/Refcoco+(e-h)/Refcocog(i-l) dataset. The patches that have higher attention are highlighted in the even rows.

D.1 Full Comparison with State of the Art

Table 1 extends the comparison of YORO with both single stage and multi-stage methods. Note that comparing YORO with multi-stage methods in terms of accuracy is **not an apples-to-apples comparison**, because multi-stage methods tends to sacrifice speed for accuracy improvement with the use of resource consuming models. The table is divided into blocks with the top performance within each block in bold.

(Bi)LSTM language model Two-stage methods: These methods are based on small (LSTM-based) language models of relatively few parameters that severely under-perform transformer-based approaches. Note that, for these models, parameter size does not correlate with inference speed, due to recursive LSTM computations. In fact, these are the slowest models considered. YORO significantly out-performs all these models in most splits (up +7 abs points), for marginally higher memory complexity (7% increase) and much faster inference (10× speed up).

BERT language model Two-stage methods: The main difference between YORO and these methods is the elimination of the visual backbone. This drastically reduces parameter size (between 60% and 74% of the parameters) and boosts inference speed by a substantial amount (between 4× and 7× speed-up). Despite these gains, YORO is comparable to the best of these methods, achieving superior performance on Refcoco val/testB (between +1.25 and +2.9) and comparable in other splits.

Encoder-Decoder multi-stage methods: Like YORO , these methods are transformer-based but use more powerful encoder-decoder models. While it has been argued that encoder-decoder models have more powerful predictive capability than encoder-only approaches [23, 1, 9], YORO outperforms all the methods in this class other than MDETR by large margins (between abs +3.0% and +10.1%). The comparison to MDETR is more complex, because it also uses the more powerful RoBERTa [18] language model. This enables MDTER to achieve higher accuracies, but also makes it a lot larger (1.3×) and slower (3.4×) than YORO .

Single-stage methods: These are the methods directly comparable to YORO . YORO outperforms all these methods on seven splits (between abs +1.88% and +8.6%). The only exception is Refcoco testB (-0.95% lower than TransVG [6]). The most competitive method in this class is TransVG, which has 1.3× the size and is 2.3× slower than YORO but achieves significantly lower accuracies on most of the splits. On the other hand, the only method of speed comparable to YORO (RCCF) has significantly lower accuracy on all splits (up to a 14 point drop on Refcocog Val). Overall, YORO has a significantly better trade-off between speed, parameter size and accuracy than all these methods.

Table 1: **Refcoco, Refcoco+ and Refcocog**. YORO achieves SOTA performance when compared with single-stage models. YORO is one of the smallest models and is also the fastest. **Bold type** indicates the best model in each block.

Method	Backbone		Refcoco			Refcoco+			Refcocog		Param.	FPS
	Lang.	Visual	Val	TestA	TestB	Val	TestA	TestB	Val	Test	(M)	
Two-Stage *Vision-Language methods based on* ***(Bi)LSTM*** *for Language Model*												
S.L.R. [31]	LSTM	VGG16	-	72.94	62.98	-	58.68	47.68	-	-	-	-
Luo [22]	BiLSTM	VGG16	-	67.94	55.18	-	57.05	43.33	-	-	-	-
Deng [5]	LSTM	VGG16	**81.27**	81.17	**80.01**	65.56	68.76	60.63	-	-	-	-
VC [33]	LSTM	VGG16	-	73.33	67.44	-	58.40	53.18	-	-	-	-
LGRAN [25]	LSTM	VGG16	-	76.6	66.4	-	64.0	53.4	-	-	-	-
Liu [16]	LSTM	VGG19	-	72.08	57.29	-	57.97	46.20	-	-	-	-
MattNet [30]	BiLSTM	Res101	76.40	80.43	69.28	64.93	70.26	56.00	66.67	67.01	-	3.2
CM-Att-Erase [17]	LSTM	Res101	78.35	**83.14**	71.32	**68.09**	**73.65**	**58.03**	**67.99**	**68.67**	-	5.1
CMN [12]	LSTM	VGG16	-	71.03	65.77	-	54.32	47.76	-	-	-	-
DDPN [32]	LSTM	Res101	76.8	80.1	72.4	64.8	70.5	54.1	-	-	-	-
DGA [26]	BiLSTM	VGG16	-	78.42	65.53	-	69.07	51.99	-	63.28	-	3
PLAN [34]	LSTM	VGG16	-	75.31	65.52	-	61.34	50.86	-	-	-	-
RvGTree [11]	BiLSTM	Res101	75.06	78.61	69.85	63.51	67.45	56.66	66.95	66.51	-	-
NMTREE [15]	BiLSTM	Res101	76.41	81.21	70.09	66.46	72.02	57.52	65.87	66.44	106.5	-
Two-Stage *Vision-Language methods based on* ***Bert-base*** *for Language Model*												
Uniter [3]	Bert-B	Res101	81.24	86.48	73.94	75.31	81.3	65.58	74.31	74.51	190.3	8.1
VLBERT [24]	Bert-B	Res101	-	-	-	71.6	77.72	60.99	-	-	155.1	10.5
VILLA [10]	Bert-B	Res101	**81.65**	**87.4**	**74.48**	**76.05**	**81.65**	**65.7**	**75.9**	75.93	191.5	8.3
ERNIE ViLL [29]	Bert-B	Res101	-	-	-	74.02	80.33	64.74	-	-	-	-
12 in 1 [20]	Bert-B	ResXT152	-	80.58		-	73.25		-	**75.96**	-	-
ViLBERT [19]	Bert-B	Res101	-	-	-	72.34	78.52	62.61	-	-	364	6
Multi-Stage *Vision-Language methods* - Encoder-Decoder Transformers. *__Note__: these approaches are more powerful than encoder-only models.												
VGTR [8]	Bi-LSTM	Res50	78.29	81.49	72.38	63.29	70.01	55.64	64.19	64.01	-	-
VGTR [8]	Bi-LSTM	Res101	79.20	82.32	73.78	63.91	70.09	56.51	65.73	67.23	-	-
VL-T5 [4]	T5	Res101	-	-	-	-	-	-	-	71.3	304.9	6.7
VLT [7]	RNN	Res50	76.20	80.31	71.44	64.19	68.40	55.84	61.03	60.24	-	-
MDETR [13]	Roberta-B	Res101	86.75	89.58	81.41	79.52	84.09	70.62	81.64	80.89	185.2	12.24
MDETR [13]	Roberta-B	ENB3	**87.51**	**90.4**	**82.67**	**81.13**	**85.52**	**72.96**	**83.35**	**83.31**	152.6	12.57
Single-Stage *Vision-Language methods*												
FAOA [28]	Bert	Darknet53	72.05	74.81	67.91	-	-	-	-	-	182.9	18.9
RCCF [14]	BiLSTM	DLA34	-	81.06	71.85	-	70.35	56.32	-	-	-	40
SSG [2]	BiLSTM	Darknet53	-	76.51	67.50	-	62.14	49.27	58.80	-	-	-
MCN [21]	BiGRU	Darknet53	80.08	82.29	74.98	67.16	72.86	57.31	66.46	66.01	-	-
ReSC [27]	Bert-B	Darknet53	76.59	78.22	73.25	63.23	66.64	55.53	64.87	64.87	179.9	15.8
TransVG [6]	Bert-B	Res101	81.02	82.72	**78.35**	64.82	70.70	56.94	68.67	67.73	149.7	18.1
YORO	Bert-B	Linear	**82.9**	**85.6**	77.4	**73.5**	**78.6**	**64.9**	**73.4**	**74.3**	**114.3**	**41.9**

Pretraining	RefCoco			CopsRef	
	val	testA	testB	val	test
w/o	73.4	78	67.1	64.4	69.5
w/	82.9 (+9.5)	85.6 (+7.6)	77.4 (+10.3)	68.08 (+3.68)	71.3 (+1.8)

Table 2: Ablation w/ and w/o pre-training.

D.2 Ablation on Model Pretraining

As mentioned in the main paper, YORO is pre-trained on the concatenated detection dataset curated by [13] to allow a good initialization of the detection branch. Table 2 further highlights the importance of the pre-training on RefCoco and CopsRef dataset. The averaged gains are 9.13% and 2.74% on RefCoco and CopsRef respectively. This supports the need of pre-trainng stage for the detection branch.

References

1. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. ArXiv **abs/2005.12872** (2020) 6
2. Chen, X., Ma, L., Chen, J., Jie, Z., Liu, W., Luo, J.: Real-time referring expression comprehension by single-stage grounding network. ArXiv **abs/1812.03426** (2018) 7
3. Chen, Y.C., Li, L., Yu, L., Kholy, A.E., Ahmed, F., Gan, Z., Cheng, Y., Liu, J.: Uniter: Universal image-text representation learning. In: ECCV (2020) 7
4. Cho, J., Lei, J., Tan, H., Bansal, M.: Unifying vision-and-language tasks via text generation. In: ICML (2021) 7
5. Deng, C., Wu, Q., Wu, Q., Hu, F., Lyu, F., Tan, M.: Visual grounding via accumulated attention. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018) 7
6. Deng, J., Yang, Z., Chen, T., Zhou, W., Li, H.: Transvg: End-to-end visual grounding with transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1769–1779 (October 2021) 6, 7
7. Ding, H., Liu, C., Wang, S., Jiang, X.: Vision-language transformer and query generation for referring segmentation. In: Proceedings of the IEEE International Conference on Computer Vision (2021) 7
8. Du, Y., Fu, Z., Liu, Q., Wang, Y.: Visual grounding with transformers. CoRR **abs/2105.04281** (2021), https://arxiv.org/abs/2105.04281 7
9. Fang, Y., Liao, B., Wang, X., Fang, J., Qi, J., Wu, R., Niu, J., Liu, W.: You only look at one sequence: Rethinking transformer in vision through object detection. arXiv preprint arXiv:2106.00666 (2021) 6
10. Gan, Z., Chen, Y.C., Li, L., Zhu, C., Cheng, Y., Liu, J.: Large-scale adversarial training for vision-and-language representation learning. In: NeurIPS (2020) 7
11. Hong, R., Liu, D., Mo, X., He, X., Zhang, H.: Learning to compose and reason with language tree structures for visual grounding. IEEE transactions on pattern analysis and machine intelligence (2019) 7
12. Hu, R., Rohrbach, M., Andreas, J., Darrell, T., Saenko, K.: Modeling relationships in referential expressions with compositional modular networks. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp. 4418–4427 (2017) 7
13. Kamath, A., Singh, M., LeCun, Y., Misra, I., Synnaeve, G., Carion, N.: Mdetr–modulated detection for end-to-end multi-modal understanding. arXiv preprint arXiv:2104.12763 (2021) 7, 8
14. Liao, Y., Liu, S., Li, G., Wang, F., Chen, Y., Qian, C., Li, B.: A real-time cross-modality correlation filtering method for referring expression comprehension. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 10877–10886 (2020) 7
15. Liu, D., Zhang, H., Wu, F., Zha, Z.J.: Learning to assemble neural module tree networks for visual grounding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2019) 7
16. Liu, J., Wang, L., Yang, M.H.: Referring expression generation and comprehension via attributes. In: 2017 IEEE International Conference on Computer Vision (ICCV). pp. 4866–4874 (2017). https://doi.org/10.1109/ICCV.2017.520 7
17. Liu, X., Wang, Z., Shao, J., Wang, X., Li, H.: Improving referring expression grounding with cross-modal attention-guided erasing. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019) 7

18. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach. ArXiv **abs/1907.11692** (2019) 6
19. Lu, J., Batra, D., Parikh, D., Lee, S.: Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In: Advances in Neural Information Processing Systems. pp. 13–23 (2019) 7
20. Lu, J., Goswami, V., Rohrbach, M., Parikh, D., Lee, S.: 12-in-1: Multi-task vision and language representation learning. In: The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020) 7
21. Luo, G., Zhou, Y., Sun, X., Cao, L., Wu, C., Deng, C., Ji, R.: Multi-task collaborative network for joint referring expression comprehension and segmentation. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) pp. 10031–10040 (2020) 7
22. Luo, R., Shakhnarovich, G.: Comprehension-guided referring expressions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (July 2017) 7
23. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research **21**(140), 1–67 (2020), http://jmlr.org/papers/v21/20-074.html 6
24. Su, W., Zhu, X., Cao, Y., Li, B., Lu, L., Wei, F., Dai, J.: Vl-bert: Pre-training of generic visual-linguistic representations. In: International Conference on Learning Representations (2020), https://openreview.net/forum?id=SygXPaEYvH 7
25. Wang, P., Wu, Q., Cao, J., Shen, C., Gao, L., Hengel, A.v.d.: Neighbourhood watch: Referring expression comprehension via language-guided graph attention networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2019) 7
26. Yang, S., Li, G., Yu, Y.: Dynamic graph attention for referring expression comprehension. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2019) 7
27. Yang, Z., Chen, T., Wang, L., Luo, J.: Improving one-stage visual grounding by recursive sub-query construction. In: ECCV (2020) 7
28. Yang, Z., Gong, B., Wang, L., Huang, W., Yu, D., Luo, J.: A fast and accurate one-stage approach to visual grounding. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (October 2019) 7
29. Yu, F., Tang, J., Yin, W., Sun, Y., Tian, H., Wu, H., Wang, H.: Ernie-vil: Knowledge enhanced vision-language representations through scene graphs. Proceedings of the AAAI Conference on Artificial Intelligence **35**(4), 3208–3216 (May 2021), https://ojs.aaai.org/index.php/AAAI/article/view/16431 7
30. Yu, L., Lin, Z., Shen, X., Yang, J., Lu, X., Bansal, M., Berg, T.L.: Mattnet: Modular attention network for referring expression comprehension. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018) 7
31. Yu, L., Tan, H., Bansal, M., Berg, T.L.: A joint speaker-listener-reinforcer model for referring expressions. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 3521–3529 (2017). https://doi.org/10.1109/CVPR.2017.375 7
32. Yu, Z., Yu, J., Xiang, C., Zhao, Z., Tian, Q., Tao, D.: Rethinking diversified and discriminative proposal generation for visual grounding. ArXiv **abs/1805.03508** (2018) 7

33. Zhang, H., Niu, Y., Chang, S.F.: Grounding referring expressions in images by variational context. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018) 7
34. Zhuang, B., Wu, Q., Shen, C., Reid, I., van den Hengel, A.: Parallel attention: A unified framework for visual object discovery through dialogs and queries. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2018) 7