Supplementary Material for Scan2Cap: Context-aware Dense Captioning in RGB-D Scans

Dave Zhenyu Chen[1] Ali Gholami[2] Matthias Nießner[1] Angel X. Chang[2]
[1]Technical University of Munich [2]Simon Fraser University

In the supplemental, we provide additional details on the 2D captioning experiments to explain the choice of 2D input and captioning method that we use (Sec. 1). We also provide details about the 3d-to-2d projection (Sec. 2), additional experiments and ablation studies (Sec. 3.1) as well as qualitative examples (Sec. 3.2) for our 3D experiments.

Figure 1: We compare how each input choice affects the performance of our 2D captioning experiments with oracle bounding boxes. We show the caption generated using show and tell (S&T) for the best matching frame selected from the video recording (A+M, bottom left), rendered annotated viewpoint (A+R, bottom right), and from the bird's eye view (BEV, top). The BEV provides a good overview of large objects, but can miss smaller objects such as trashcans placed underneath desks. The matched frame may not fully capture the object of interest or provide enough context for informative captions (see Tab. 1 for quantitative comparisons).

Description Generation in 2D (rendered vs matched vs BEV)							
VF	VP	DET	CAP	C	B-4	M	R
G	A+R	-	S&T	49.61	11.41	15.64	40.59
G + T	A+R	O	S&T	**59.12**	**12.73**	**16.61**	**41.32**
G	A+M	-	S&T	11.50	1.63	5.64	13.86
G + T	A+M	O	S&T	16.76	2.01	6.14	14.23
G	BEV	-	S&T	19.94	8.74	14.64	36.53
G + T	BEV	O	S&T	24.21	9.69	14.41	37.38
T + C	A+R	O	TD	**51.35**	**13.09**	**15.88**	**43.52**
G + T + C	A+R	O	TD	18.10	5.65	11.37	33.10
T + C	A+M	O	TD	12.77	1.58	5.84	15.42
G + T + C	A+M	O	TD	14.00	1.68	5.74	15.41

Table 1: We compare captions for oracle bounding boxes from annotated viewpoints with rendered (A+R), matched frames (A+M), and from the birds-eye-view (BEV) on the ScanRefer [4] validation split. We observe that the rendered frames consistently result in better captions for different features (global (G), with target object features (T), and context object features (C)) and captioning methods (show and tell (S&T) vs top-down attention (TD)).

1. 2D experiments

1.1. Experimental setup

We conduct a series of experiments in 2D to select the input, captioning method, and visual features for our 2D baselines. We implement the models for the 2D experiments using PyTorch [12] and Detectron2 [16].

Choice of 2D input However, we find that it is often challenging to find a good matching frame (see Fig. 2), and using the rendered frames leads to better captioning performance (see Tab. 1) despite the rendering artifacts. Fig. 2 shows examples of viewpoints for which it is challenging to find a good matching frame from the video frames. We suspect that the poor performance of captioning with matched frames is due to the differences in viewpoints as well as the extremely limited field of view and motion blur found in the video frames. In addition, we also check the captioning performance from a bird-eye-view.

Captioning method For selecting a 2D captioning method, we experiment with a simple model, show and tell

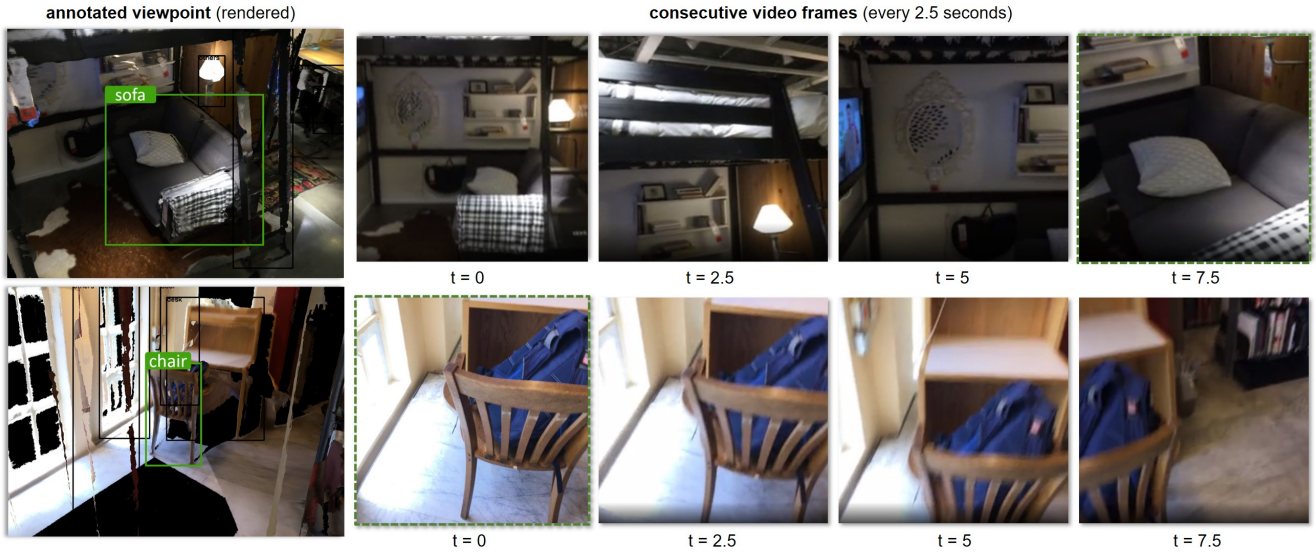

Figure 2: Examples of difficult to match viewpoints, with the rendered frame for the annotated viewpoint on the left, and sample frames from the video on the right (selected matched frame shown with dashed borders). The bounding box for the target object is shown in green. Due to a lack of video recording coverage, it is often impossible to match the exact viewpoint camera direction and origin. Frames from the video recording suffers from motion blur and have a view that is too close up, and missing contextual objects.

(S&T [15]), as well as the popular bottom-up and top-down attention model (TD [2]), and a recent state-of-the-art captioning method, the meshed-memory transformer (M^2 [6]). The S&T [15] and TD [2] models are similar to the original ones, but we replace LSTM [9] with GRU [5] due to the small size of the ScanRefer [4] dataset. In addition to the captioning methods above, we also compare our method against the retrieval baselines (Retr).

Visual features For visual features, we experiment with using the global visual features for the entire image (G), features from just the target object (T), and features from the context objects (C). For object-based features, we rely on object bounding boxes that are either oracle (O), detected using a 2D object detector (2DM), or back-projected from 3D (3DV). For our 2D detection, We use Mask R-CNN [8] with a pre-trained ResNet-101 [7] as our backbone and then fine-tune it on the ScanRefer training split using rendered viewpoints.

1.2. Results

In this section we evaluate our instance segmentation and captioning methods in 2D.

1.2.1 Object detection and instance segmentation

We evaluate the model performance on object detection and instance segmentation via *mAP* (mean average precision). Tab. 2 demonstrates our object detection and instance segmentation results.

1.2.2 Captioning

We evaluate the captions generated for 2D inputs using the well-established CiDEr [14], BLEU-4 [11], METEOR [3] and ROUGE [10], abbreviated as C, B-4, M, R, respectively. Tab. 3 shows our captioning experiment results and Fig. 3 shows examples from the different methods. Note that the captioning metrics reported here are not comparable to dense captioning metrics reported in the main paper, as these does not take into account the IoU, and we evaluate the predicted caption against the ground truth caption for each respective viewpoint.

Surprisingly, we find that the simple baseline of S&T outperforms other methods such as the top-down attention (TD) and meshed-memory transformer (M^2) on CiDEr and METEOR. We suspect that this is partly due to the limited amount of training data (MSCOCO has 113,287 training images with five captions each while ScanRefer has only 36,665 descriptions in the train split). Thus, for our 2D-based baselines in the main paper, we chose to use S&T with features from the global image and the target object.

2. 3D to 2D projection details

In order to caption the objects in the images using 3D detected information, we estimate the camera viewpoints from the 3D bounding boxes and project the 3D bounding boxes to the rendered single-view images for captioning. We show the example in Fig. 4.

	bath.	bed	bkshf.	cab.	chair	cntr.	curt.	desk	door	others	pic.	fridg.	showr.	sink	sofa	tabl.	toil.	wind.	mAP	mAP50	mAP75
DET	12.84	37.66	20.33	16.09	32.39	18.63	16.21	14.47	14.55	20.98	24.72	17.30	18.90	19.73	29.91	28.71	58.22	16.09	23.21	36.01	24.45
SEG	9.74	23.61	1.38	15.25	27.97	7.53	12.82	6.95	11.79	19.66	23.74	18.12	17.91	20.03	25.86	28.23	56.72	9.62	18.72	32.01	19.37

Table 2: 2D object detection (DET) and instance segmentation (SEG) results on the ScanRefer [4] validation split. Reported values for each object category is the *mAP* at IoU $= 0.50 : .05 : 0.95$ (averaged over 10 IoU thresholds). *mAP* is the class averaged precision at IoU $= 0.50 : .05 : 0.95$ (averaged over 10 IoU thresholds). *mAP50* is the class averaged precision at IoU $= 0.50$. *mAP75* is the class averaged precision at IoU $= 0.75$. We use a Mask R-CNN [8] with a pre-trained ResNet-101 [7] backbone and fine-tune it on the ScanRefer [4] training split.

Description Generation in 2D (Rendered Viewpoints)							
VF	VP	DET	CAP	C	B-4	M	R
G	A	-	Retr	12.07	4.58	11.50	29.37
G	A	-	S&T	49.61	11.41	15.64	40.59
T	A	O	Retr	23.00	7.28	13.44	33.82
T + C	A	O	TD	51.35	**13.09**	15.88	**43.52**
T + C	A	O	M^2	34.72	7.13	12.69	33.60
T + C	A	O	M^2 RL	42.77	9.03	14.34	36.27
G + T	A	O	S&T	**59.12**	12.73	**16.61**	41.32
G + T + C	A	O	TD	18.10	5.65	11.37	33.10
T + C	A	2DM	TD	35.65	**11.00**	14.30	**40.70**
T + C	A	2DM	M^2	31.02	7.19	12.28	33.22
T + C	A	2DM	M^2 RL	35.91	8.52	13.53	35.33
G + T	A	2DM	S&T	**41.44**	10.95	**15.08**	39.04
G + T + C	A	2DM	TD	14.84	4.95	10.85	31.52
G	E	-	S&T	28.52	24.03	18.92	47.76
T + C	E	3DV	TD	28.25	**30.11**	18.9	**52.14**
T + C	E	3DV	M^2	11.44	19.67	14.23	40.42
T + C	E	3DV	M^2 RL	11.83	24.79	15.47	42.69
G + T	E	3DV	S&T	**31.48**	25.35	**19.09**	47.06
G + T + C	E	3DV	TD	9.66	9.68	13.14	38.38

Table 3: Results of caption generation with rendered viewpoints on the ScanRefer [4] validation split. Captioning metrics are calculated by comparing the generated caption against the reference caption corresponding to the annotated viewpoint. VF is the input visual feature which can include the full image (G), context objects (C), and/or target object (T). VP is the viewpoint that can be annotated (A), estimated (E), or bird's eye viewpoint (BEV). DET is the object bounding box which can be the ground truth box (O), Mask R-CNN [8] detected in 2D (2DM) or back-projected VoteNet [13] detection in 3D (3DV). CAP is the captioning method which can be cosine retrieval (Retr), Show and tell (S&T) [15], Top-down attention [2] (TD), Meshed memory transformer [6] without and with self-critical optimization respectively (M^2) and (M^2 RL). Since S&T with global and target object features (G+T) gives the best CiDEr score, we select it as the 2D captioning method for the main paper.

Viewpoint estimation from 3D detections. We take several heuristics into account to estimate the viewpoints for the detected 3D boxes. To start with, we compute the average distance between the target objects and the recorded viewpoints (1.97 meters). Then, assuming the camera height as 1.70 meters, we compute the horizontal distance between the target objects and the viewpoints (0.99 meter). We randomly pick the points on the circle with the horizontal radius 0.99 meters to the target objects. We repeat the random selection process until the selected viewpoints are inside the scenes and the target objects are visible in the view.

Projecting 3D detections to the estimated views. We derive the camera extrinsics from the estimated viewpoints as we assume the cameras are always targeting at the center of the 3D bounding boxes. We keep the camera intrinsic as in ScanNet. Then, we use these camera parameters to render the single-view images for the 3D scans. The 3D bounding boxes are then projected into the image space as the targets and contexts for generating captions.

3. Additional 3D captioning results

3.1. Additional quantitative analysis

Does our method work on Nr3D? We evaluate our dense captioning method against the aforementioned baselines on Nr3D dataset [1], where each 3D object in the scene possesses several unique utterances. As shown in Tab. 4, our method outperforms all baselines with a significant margin, indicating the consistency of improvement on describing the 3D objects with respect to their appearance and spatial relationship.

Can the generated captions be used for localization? We conduct a reverse experiment using the generated captions to localize objects in the 3D scenes, where we use a pre-trained ScanRefer [4] with projected multiview features and point normals. As results shows in Tab. 5, ScanRefer using the generated descriptions achieves plausible localization accuracy when compared to using the original ScanRefer descriptions annotated by human experts. This further demonstrates the capability of our method of generating good descriptions with accurate appearance and spatial relationship aspects.

Do other 3D features help? We include colors and normals from the ScanNet meshes to the input point cloud features and compare performance against networks trained without them. As displayed in Tab. 6, our architecture trained with geometry, multi-view features and normal

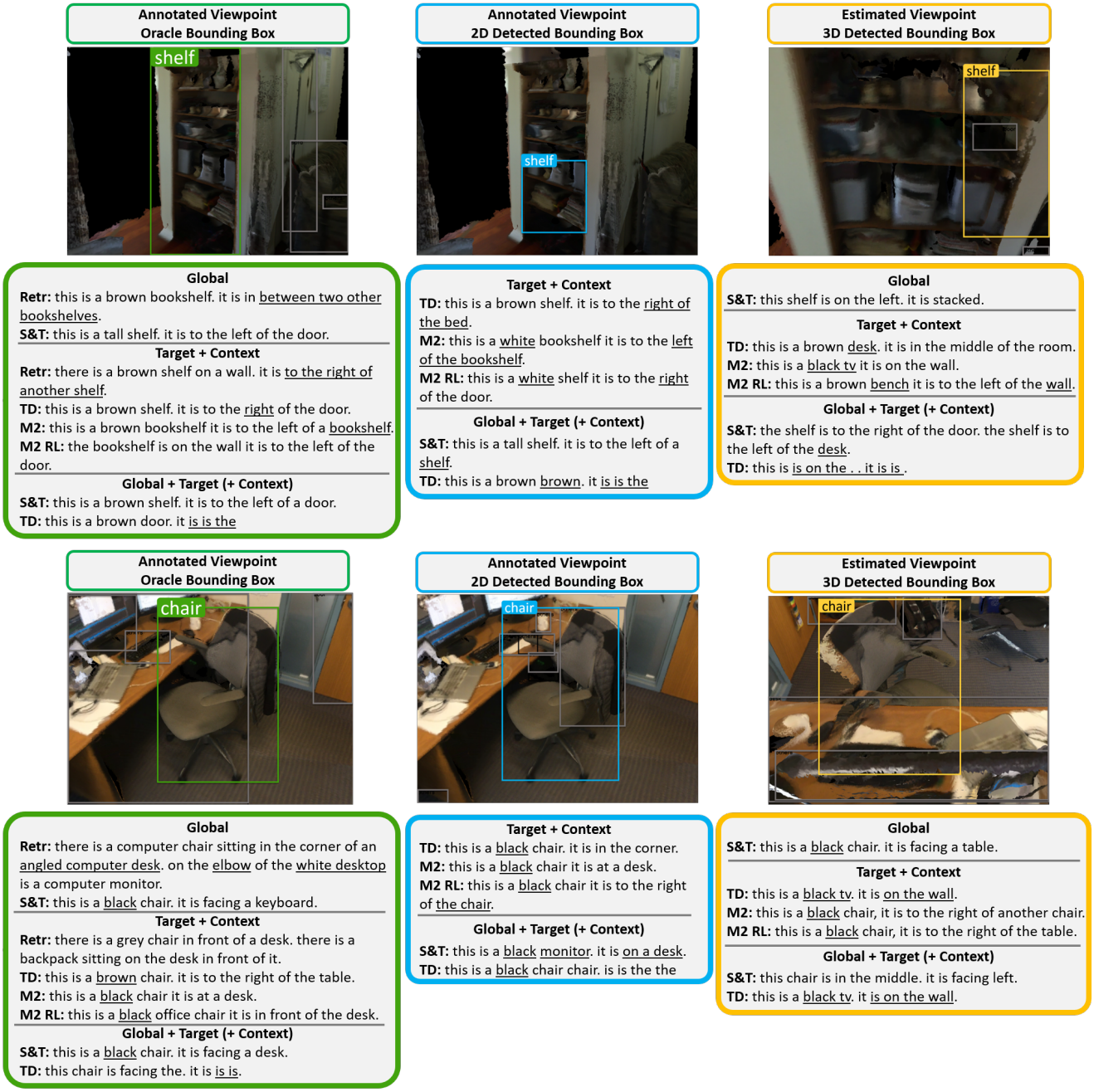

Figure 3: Examples of captions generated from 2D rendered frames with oracle bounding boxes (O-left), detected boxes from Mask-RCNN (2DM-middle), and projected bounding boxes from 3D to 2D (3DV-right). The oracle and Mask-RCNN predictions are from the annotated viewpoint, while the 3D to 2D projection is using an estimated viewpoint. The bounding box for the target object is shown in color, while the bounding box for the context objects are in gray. Inaccurate parts of the caption are underscored.

vectors (xyz+multiview+normal) achieves the best performance among all ablations. This matches the feature ablation from ScanRefer [4].

3.2. Additional qualitative analysis

Do graph and attention help with captioning? We compare our model (VoteNet+RG+CAC) with the basic descrip-

	Captioning	Detection	C@0.25IoU	B-4@0.25IoU	M@0.25IoU	R@0.25IoU	C@0.5IoU	B-4@0.5IoU	M@0.5IoU	R@0.5IoU	mAP@0.5IoU
2D-3D Proj.	2D	Mask R-CNN	-	-	-	-	-	-	-	-	-
3D-2D Proj.	2D	VoteNet	8.57	8.49	18.83	44.95	3.93	4.21	16.68	41.24	31.83
VoteNetRetr [13]	3D	VoteNet	12.60	10.36	20.73	45.53	7.68	7.11	18.83	42.71	31.83
Ours	3D	VoteNet	**42.21**	**24.43**	**25.07**	**55.88**	**24.10**	**15.01**	**21.01**	**47.95**	**32.21**

Table 4: Comparison of 3D dense captioning results obtained by Scan2Cap and other baseline methods on Nr3D [1]. 2D-3D Proj. is not performed on Nr3D due to the lack of viewpoint annotations. We average the scores of the conventional captioning metrics, e.g. CiDEr [14], with the percentage of the predicted bounding boxes whose IoU with the ground truth are greater than 0.25 and 0.5. Our method outperforms all baselines with a remarkable margin.

Object Localization Results						
	unique		multiple		overall	
Evaluation data	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5	Acc@0.25	Acc@0.5
ScanRefer [4]	76.33	53.51	32.73	21.11	41.19	27.40
Ours	64.54	45.70	26.93	18.14	35.58	24.48

Table 5: Comparison of localization results using a pretrained ScanRefer network [4] on the original ScanRefer validation split and the generated captions by our dense captioning method, respectively. We measure percentage of predictions whose IoU with the ground truth boxes are greater than 0.25 and 0.5. We also report scores on "unique" and "multiple" subsets; unique means that there is only a single object of its class in the scene. Using the generated captions achieves comparable localization results when compared to those using the annotated descriptions by human experts.

	C@0.25IoU	B-4@0.25IoU	M@0.25IoU	R@0.25IoU	C@0.5IoU	B-4@0.5IoU	M@0.5IoU	R@0.5IoU	mAP@0.5IoU
Ours (xyz)	47.21	29.41	24.89	50.74	32.94	20.63	21.10	41.58	27.45
Ours (xyz+rgb)	49.36	32.88	25.52	54.20	33.41	21.61	**22.12**	43.61	27.52
Ours (xyz+rgb+normal)	53.73	**34.25**	26.14	54.95	35.20	22.36	21.44	43.57	29.13
Ours (xyz+multiview)	54.94	32.73	25.90	53.51	36.89	21.77	21.39	42.83	31.43
Ours (xyz+multiview+normal)	**56.82**	34.18	**26.29**	**55.27**	**39.08**	**23.32**	21.97	**44.78**	**32.21**

Table 6: Ablation study with different features. We compute standard captioning metrics with respect to the percentage of the predicted bounding box whose IoU with the ground truth are greater than 0.25 and 0.5. The higher the better.

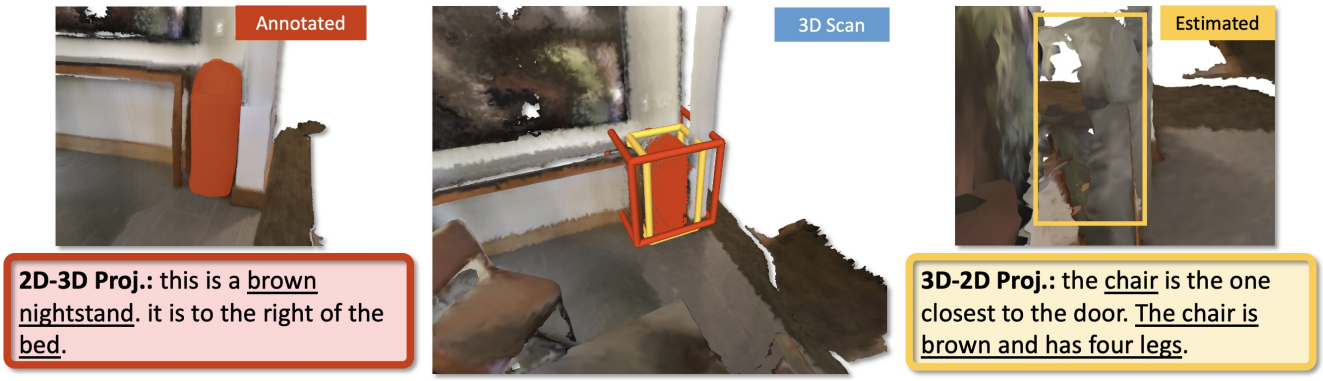

Figure 4: Comparison of generated captions based on 2D-3D and 3D-2D projected detections (2D-3D Proj. and 3D-2D Proj respectively). In 2D-3D Proj., we first detect object mask in the rendered annotated viewpoints using Mask R-CNN [8] (as shown in the red box on the left), and generate the caption for the detected object. While in 3D-2D Proj., we first detect object bounding boxes in 3D using VoteNet [13], then estimate a viewpoint for the detected 3D bounding box, and we back-project the detected bounding box to 2D. We then generate the caption based on the estimated viewpoint and the back-projected bounding box (see the yellow box on the right).

tion generation component (VoteNet+GRU) introduced in Vinyals et al. [15] and our model equipped only with the context-aware attention captioning (VoteNet+CAC). As shown in Fig. 5, though all three methods produce good

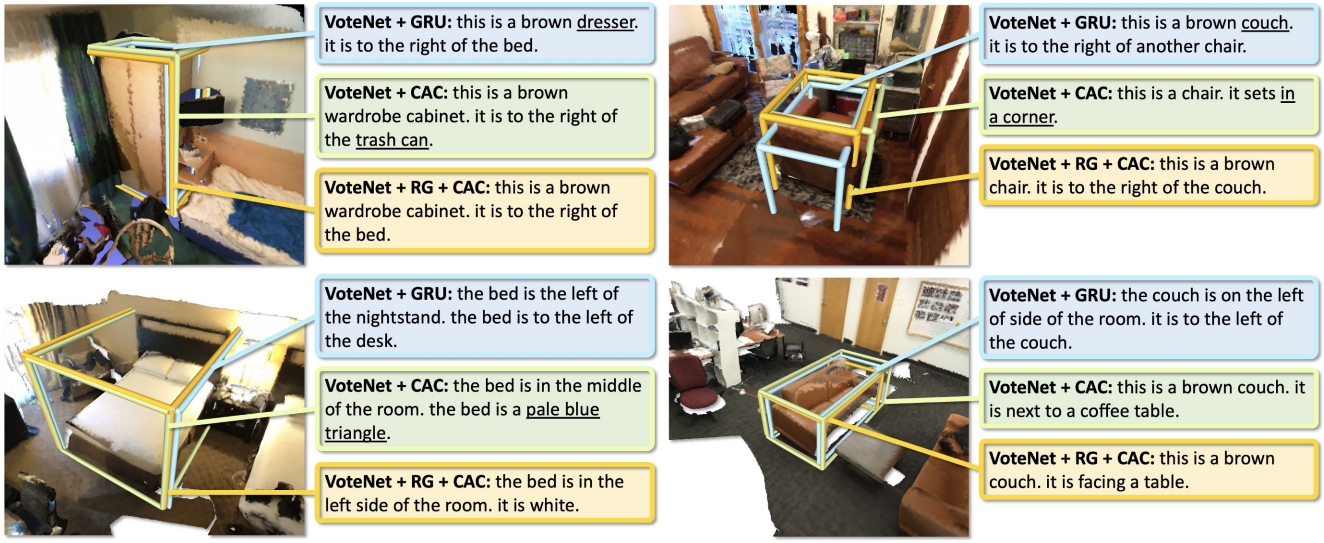

Figure 5: Ablation study with different components in our method: VoteNet [13] + GRU [5], which is similar to "show and tell" Vinyals et al. [15]; VoteNet + Context-aware Attention Captioning (CAC); VoteNet + Relational Graph (RG) + Context-aware Attention Captioning (CAC), namely Scan2Cap. We underscore the inaccurate aspects in the descriptions. Image best viewed in color.

bounding boxes (IoU>0.5), VoteNet+GRU makes mistakes when describing the target objects. VoteNet+CAC refers to the target and the objects nearby in the scene, but still fails to correctly reveal the relative spatial relationships. In contrast, VoteNet+RG+CAC can properly handle the interplay of describing the target appearance and the relative spatial relationships in the local environment.

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. ReferIt3D: Neural listeners for fine-grained 3D object identification in real-world scenes. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 3, 5

[2] 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. 2, 3

[3] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005. 2

[4] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. ScanRefer: 3D object localization in RGB-D scans using natural language. *16th European Conference on Computer Vision (ECCV)*, 2020. 1, 2, 3, 4, 5

[5] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014. 2, 6

[6] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10578–10587, 2020. 2, 3

[7] 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. 2, 3

[8] 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, 3, 5

[9] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 2

[10] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004. 2

[11] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 2

[12] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages

8024–8035. 2019. 1

[13] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3D object detection in point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9277–9286, 2019. 3, 5, 6

[14] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. CIDEr: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015. 2, 5

[15] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015. 2, 3, 5, 6

[16] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 1