Supplementary Material for D^{3}Net: A Unified Speaker-Listener Architecture for 3D Dense Captioning and Visual Grounding

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

In this supplementary material, we provide results on the ReferIt3D dataset in Sec. 1. To showcase the effectiveness of our speaker-listener architecture, we provide additional results on extra ScanNet [6] data in Sec. 2. We also include details about our PointGroup implementation as well as the detection and segmentation results in Sec. 3 and Sec. 4, respectively.

1 Experiments on ReferIt3D

1.1 Quantitative Results

We conduct additional experiments on the ReferIt3D Nr3D dataset [1]. It contains about 33k free-form object descriptions annotated by human experts for training and 8k for validation. We report our results on the validation split since there is no test set.

3D dense captioning We compare our 3D dense captioning and object detection results against the baseline Scan2Cap [3] in Tab. 1. Our method trained with the speaker MLE loss (marked "Ours (MLE)") outperforms Scan2Cap by a big margin, leveraging the improved object detection backbone. After training with the CIDEr reward (marked "Ours (CIDEr)"), our dense captioning results are further boosted. Training with the listener loss as the additional reward (marked "Ours (CIDEr+lis.)") further improves our results due to the explicit reinforcement of the discriminability of generated object descriptions. Here, our object detection mAP is also improved due to the end-to-end joint fine-tuning of our speaker-listener architecture. We showcase the effectiveness of training with extra ScanNet data in the last row in Tab. 1, where 3D dense captioning and object detection results are improved simultaneously.

3D visual grounding We compare our 3D visual grounding results against the baseline ScanRefer [2] and 3DVG-Transformer [14] in Tab. 2. As the descriptions in ReferIt3D dataset all refer to objects in the scene where multiple similar objects with the same class label are present, there is no such case that can be allocated to "Unique" subset where only one object with a specific class label can be found in the scene. Therefore, we allocate our results to "Multiple" and "Overall". Our method trained with the detector loss and the listener loss (marked "Ours(w/o fine-tuning)") clearly outperforms the baseline methods.

Table 1: Quantitative results on 3D dense captioning and object detection on ReferIt3D Nr3D dataset [1]. We average the conventional captioning evaluation metrics with the percentage of the predicted bounding boxes whose IoU with the GTs are higher than 0.5. Our method outperforms the baseline Scan2Cap [3] by a significant margin. We showcase the effectiveness of our speaker-listener architecture trained with partially annotated ScanNet data, where it achieves the best performance in all metrics.

	C@0.5IoU	B-4@0.5IoU	M@0.5IoU	R@0.5IoU	mAP@0.5
Scan2Cap [3]	22.38	13.87	20.44	47.96	33.21
Ours (MLE)	33.85	20.70	23.13	53.38	49.71
Ours (CIDEr)	36.79	21.12	23.91	53.83	50.89
Ours (CIDEr+lis.)	37.35	21.40	24.10	54.14	51.58
Ours (CIDEr+lis.+extra)	**38.42**	**22.22**	**24.74**	**54.37**	**52.69**

Table 2: Quantitative results on 3D visual grounding on ReferIt3D Nr3D dataset [1]. We adapt the evaluation setting as in Chen et al. [2] to be consistent with the main paper. We report results on "Multiple" and "Overall", as there is no case in ReferIt3D that is "Unique". Our base visual grounding network outperforms the baseline methods. Results are further improved after the joint fine-tuning with the speaker-listener architecture. Speaker-listener fine-tuning and semi-supervised training with partially annotated ScanNet data provide the best overall results.

	Acc@0.5IoU		
	Unique	Multiple	Overall
ScanRefer [2]	-	12.17	12.17
3DVG-Trans [14]	-	14.22	14.22
Ours (w/o fine-tuning)	-	19.64	19.64
Ours (w/ fine-tuning)	-	24.41	24.41
Ours (w/ fine-tuning + extra)	-	**25.23**	**25.23**

Our results (marked "Ours(w/ fine-tuning)") are significantly improved after fine-tuning jointly with the speaker. Our best results are obtained after jointly training with speaker-listener architecture on partially annotated ScanNet data, as demonstrated in the last row in Tab. 2.

1.2 Qualitative Analysis

3D dense captioning We compare our results with object captions from Scan2Cap [3] in Fig. 1. Object captions generated by Scan2Cap include more inaccurate spatial relationships. Also, those object captions cannot be used to uniquely localize the associated object. In contrast, our method produces more accurate and discriminative object captions with more spatial relationship information.

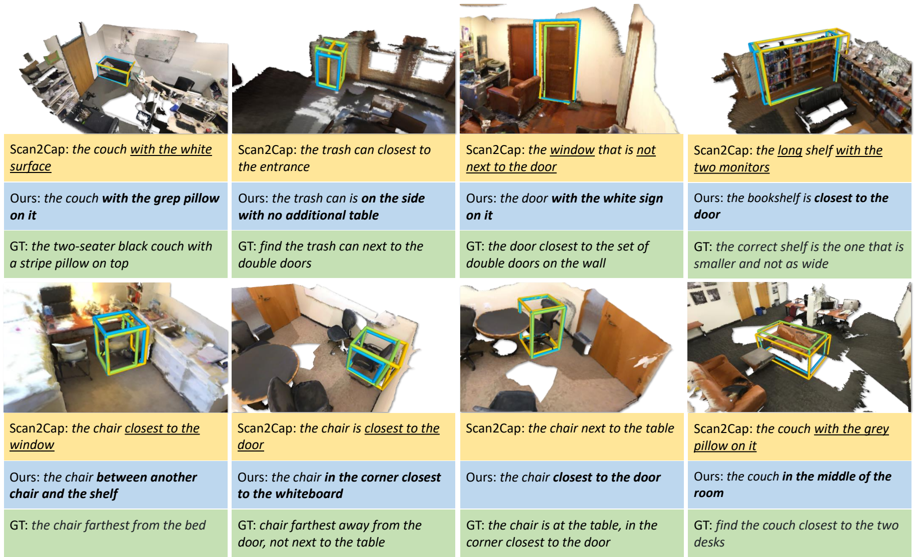

Scan2Cap: the couch with the white surface
Ours: the couch with the grep pillow on it
GT: the two-seater black couch with a stripe pillow on top
Scan2Cap: the trash can closest to the entrance
Ours: the trash can is on the side with no additional table
GT: find the trash can next to the double doors
Scan2Cap: the window that is not next to the door
Ours: the door with the white sign on it
GT: the door closest to the set of double doors on the wall
Scan2Cap: the long shelf with the two monitors
Ours: the bookshelf is closest to the door
GT: the correct shelf is the one that is smaller and not as wide
Scan2Cap: the chair closest to the window
Ours: the chair between another chair and the shelf
GT: the chair farthest from the bed
Scan2Cap: the chair is closest to the door
Ours: the chair in the corner closest to the whiteboard
GT: chair farthest away from the door, not next to the table
Scan2Cap: the chair next to the table
Ours: the chair closest to the door
GT: the chair is at the table, in the corner closest to the door
Scan2Cap: the couch with the grey pillow on it
Ours: the couch in the middle of the room
GT: find the couch closest to the two desks

Fig. 1: Qualitative results in 3D dense captioning task from Scan2Cap [3] and our method on ReferIt3D Nr3d dataset [1]. We underline the inaccurate words and mark the spatially discriminative phrases in bold.

Table 3: Comparison of the performance of our implementation of PointGroup using the Minkowski Engine against the original PointGroup (PG(*)) for instance segmentation. We report the mAP for IoU threshold 0.5 on the ScanNet v2 validation set. Our re-implementation using color gives comparable performance as the original PointGroup implementation. Using multiview features, we are able to further improve the performance.

Method	mAP@0.5	cab.	bed	chair	sofa	tab.	door	wind.	booksh.	pic.	cntr	desk	curt.	refrige.	s. curt.	toil.	sink	batht.	other
PG (*)	56.9	48.1	69.6	**87.7**	71.5	62.9	42.0	46.2	54.9	37.7	22.4	41.6	44.9	37.2	64.4	98.3	**61.1**	80.5	53.0
PG (Color)	56.6	47.5	64.1	83.8	**75.4**	63.7	42.7	45.7	49.6	**43.7**	17.5	42.9	**47.9**	35.0	65.6	**100.0**	60.7	**81.9**	51.5
PG (Multiview)	**62.8**	**58.3**	**83.4**	86.9	66.3	**68.6**	**47.3**	**52.4**	**64.9**	38.3	**23.0**	**56.9**	46.3	**64.3**	**83.0**	98.3	57.0	71.4	**63.1**

3D visual grounding Fig. 2 compares our results with 3DVG-Transformer [14] on ReferIt3D Nr3D dataset [1]. 3DVG-Transformer clearly suffers from overfitting issue, as it tends to predict that same object bounding box given different queries as inputs (see the third and fourth examples in the first row). Leveraging the speaker-listener architecture, our method can better distinguish object from the same class than 3DVG-Transformer.

2 Additional Results on Extra ScanNet Data

Fig. 3 showcase the intermediate dense captioning and visual grounding results for scans where no GT object captions are provided in the ScanRefer dataset [2].

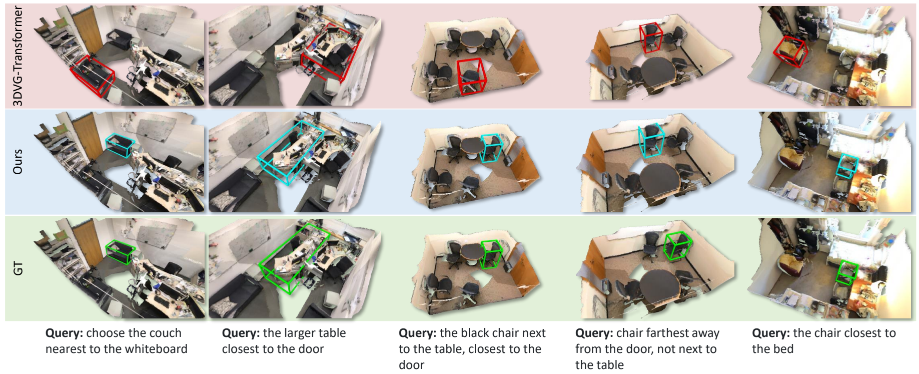

3DVG-Transformer
Ours
GT
Query: choose the couch nearest to the whiteboard
Query: the larger table closest to the door
Query: the black chair next to the table, closest to the door
Query: chair farthest away from the door, not next to the table
Query: the chair closest to the bed

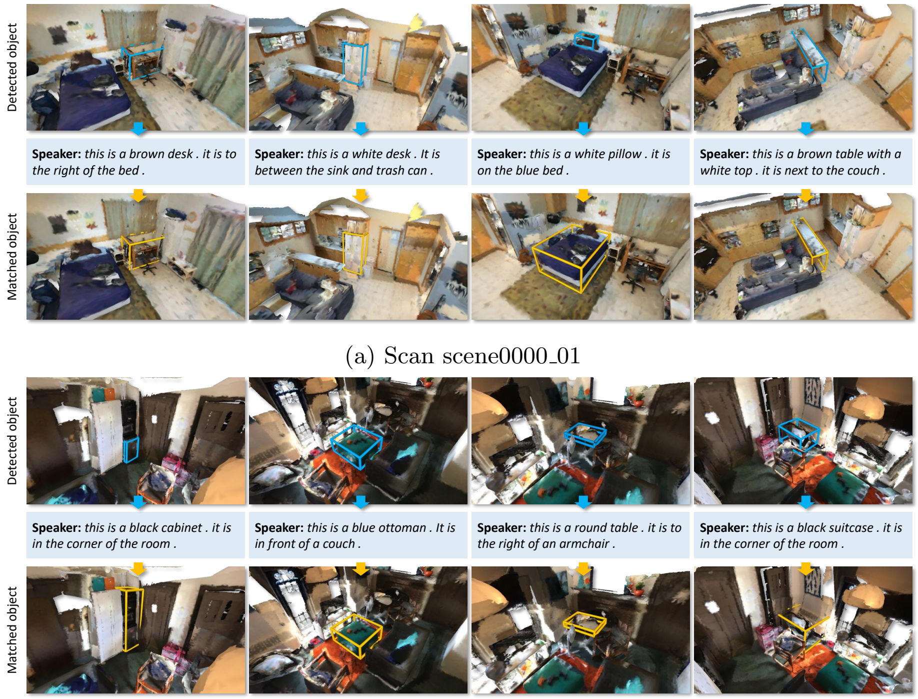
Fig. 2: 3D visual grounding results using 3DVG-Transformer [14] and our method on ReferIt3D Nr3D dataset [1].

Those intermediate object captions and the matched object bounding boxes are used during the semi-supervised training of our speaker-listener architecture. Our architecture produces plausible object captions with adequate and discriminative spatial relationships that inherently enables visual grounding.

3 PointGroup Implementation Details

The official implementation of PointGroup uses SpConv [12], a spatially sparse convolution library devoted to 3D data, to build its SparseConv-based U-Net architecture to encode point and cluster representation. We migrated the implementation of PointGroup from SpConv to MinkowskiEngine [5], another auto-differentiation library for sparse tensors, since it outperformed SpConv by providing faster computation operations on GPU, user-friendly documentations and consistent code maintenance at the time the project was initiated.

Following Jiang et al. [8], we use the same hyperparameters for point voxelization and clustering. We set the maximum number of points per scene to $250,000$ by randomly adding small offsets to the point cloud and cropping out extra parts exceeding the predefined maximum scale of the scene if necessary. Limiting the number of points to $250,000$ allows us to fit the model on a RTX 3090. We augment each point cloud scene by jittering point coordinates slightly, mirroring about the YZ-plane, and rotation about the Z axis (up-axis) randomly from 0 to 360°. We also apply elastic distortion, which was used by Jiang et al. [8], to the scaled points. We share the same SparseConv-based U-Net architecture as Jiang et al. [8] for both backbone and ScoreNet except that the input data may contain mutiview features and normals instead of RGB colors. For each voxel, we encode the color, normal and multiview features extracted using ENet [7], giving us a total input dimension of 134. To adapt PointGroup as an object detector, we obtain axis-aligned bounding boxes using predicted instance clusters by simply calculating their sizes and centers from points assigned to


Detected object
Speaker: this is a brown desk . it is to the right of the bed .
Speaker: this is a white desk . It is between the sink and trash can .
Speaker: this is a white pillow . it is on the blue bed .
Speaker: this is a brown table with a white top . it is next to the couch .
Matched object
Detected object
Speaker: this is a black cabinet . it is in the corner of the room .
Speaker: this is a blue ottoman . It is in front of a couch .
Speaker: this is a round table . it is to the right of an armchair .
Speaker: this is a black suitcase . it is in the corner of the room .
Matched object

(a) Scan scene0000_01

(b) Scan scene0002_01

Fig. 3: Intermediate dense captioning and visual grounding results in the Speaker-listener architecture for RGB-D scans where no GT object descriptions are provided in ScanRefer dataset [2]

them. We set the thresholds of cluster scores as 0.09 and the minimum cluster point number as 100 to filter out bad cluster proposals. We train the PointGroup detector using Adam [9] with a learning rate of 2e-3, on the ScanNet train split with batch size 4 for 140k iterations until convergence.

4 Detection and Segmentation Results

4.1 Quantitative results

Instance segmentation. Tab. 3 compares the instance segmentation results of our PointGroup implementation against the original PointGroup (first row). With positions and colors as input, our implementation of PointGroup (second row) gives a similar performance as original PointGroup. When replacing colors with multiview features and normals (last row), our PointGroup implementation significantly outperforms the original one. Our multiview-based PointGroup gives mAP@0.5 of 62.8, which is close to the performance of the current state-of-the-art model HAIS [4], which achieves 64.1 on the validation set of ScanNet v2.

Table 4: Comparison of object detection performance of PointGroup (PG) and VoteNet. We report mAP with IoU threshold 0.5 on the ScanNet v2 validation set. PointGroup produces more accurate bounding boxes than VoteNet, and using multiview features further improves performance over incorporating color as input directly.

Method	mAP@0.5	cab.	bed	chair	sofa	tab.	door	wind.	booksh.	pic.	cntr	desk	curt.	refrige.	s. curt.	toil.	sink	batht.	other
VoteNet	33.5	8.1	76.1	67.2	**68.8**	42.4	15.3	6.4	28.0	1.3	9.5	37.5	11.6	27.8	10.0	86.5	16.8	**78.9**	11.7
PG (Color)	44.6	25.2	69.1	77.1	67.3	53.3	32.7	32.2	36.8	26.9	30.0	52.1	**33.5**	26.7	37.4	87.8	32.3	69.6	13.6
PG (Multiview)	**50.7**	**36.4**	**77.6**	**80.9**	66.1	**59.2**	**40.2**	**33.1**	**37.0**	**27.7**	**32.0**	**56.5**	32.2	**62.1**	**70.0**	**91.1**	**33.8**	60.2	**16.0**

Our implementation also surpasses the original PointGroup in training speed: given point coordinates and colors as input, it takes less than two days to train the model in our implementation, while the original one could take up to three days until convergence.

Object Detection. We compare our object detection results before fine-tuning with the speaker-listener architecture against the VoteNet [11] in Tab. 4. Given positions and colors as input, our PointGroup detector (second row) clearly outperforms VoteNet. Using multiview features and normals instead of RGB colors, our PointGroup based detector gives improved detection results of 50.7 mAP@0.5, which outperforms the current state-of-the-art detectors [10, 13] on the validation set of ScanNet v2 with gains of 3.7 and 2.6 respectively. Also, our PointGroup generates notably better detections for small and thin objects than VoteNet, such as picture ("pic.") and counter ("cntr").

4.2 Qualitative results

Instance segmentation. We present our instance segmentation results in Fig. 4. Our PointGroup trained with multiview features and normals clearly generates better instance segmentation masks than our model with raw point colors as input, as it better segments out tiny objects leveraging the higher resolution of the multiview images.

Object Detection. Fig. 5 showcases the effectiveness of our PointGroup in object detection over VoteNet. Our PointGroup implementation produces much more accurate object bounding boxes due to the fine-grained per-point segmentation. Also, training with multiview normal features can further improve the quality of the generated bounding boxes in comparison with PointGroup trained with the raw point colors (the third column vs. the first column).

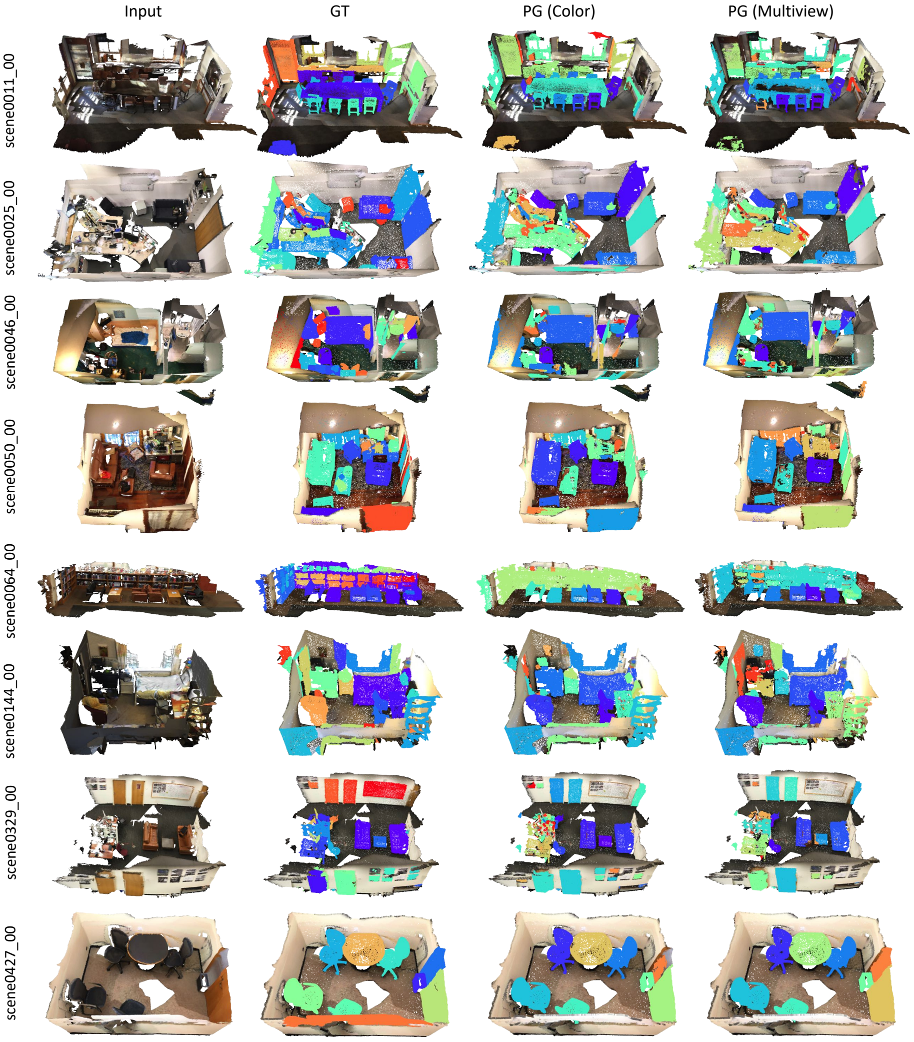


Fig. 4: Qualitative results in instance segmentation task on the ScanNet v2 validation set.

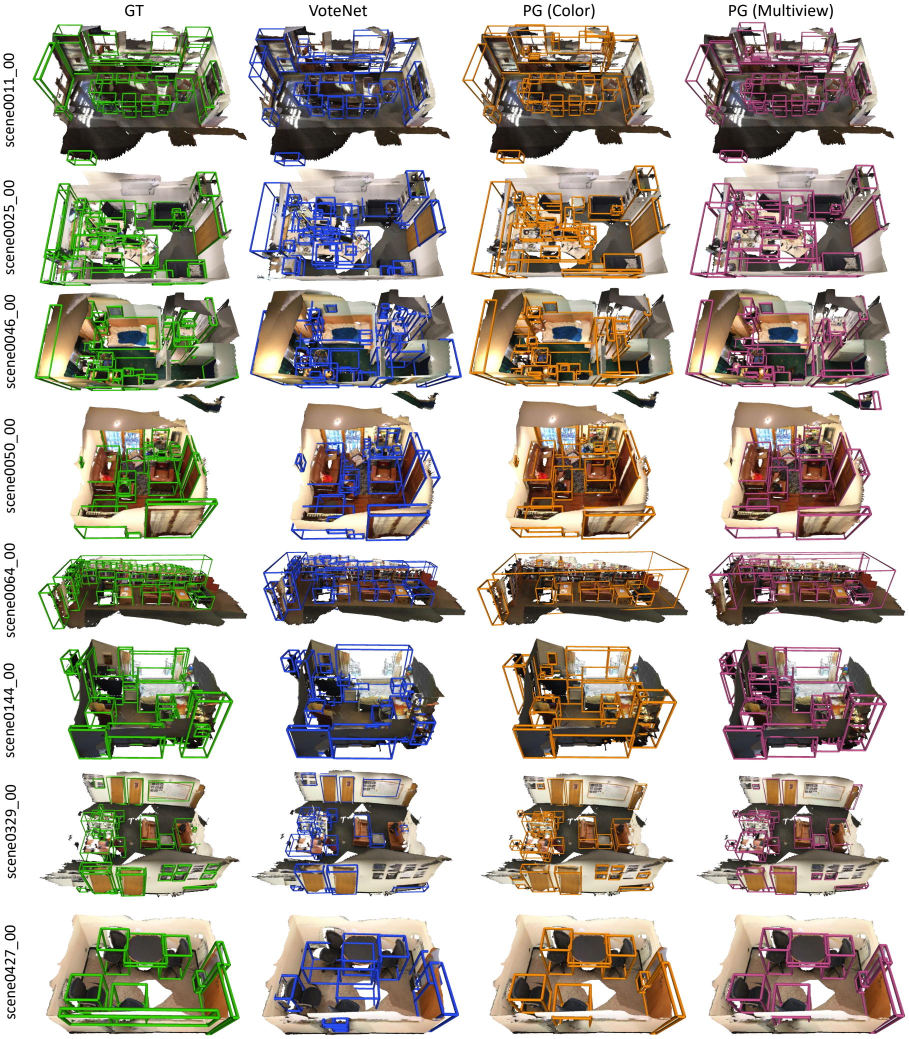


Fig. 5: Qualitative results in object detection task on the ScanNet v2 validation set.

References

1. Achlioptas, P., Abdelreheem, A., Xia, F., Elhoseiny, M., Guibas, L.: ReferIt3D: Neural listeners for fine-grained 3D object identification in real-world scenes. In: European Conference on Computer Vision, pp. 422–440, Springer (2020) 1, 2, 3, 4
2. Chen, D.Z., Chang, A.X., Nießner, M.: ScanRefer: 3D object localization in RGB-D scans using natural language. In: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX 16, pp. 202–221, Springer (2020) 1, 2, 3, 5
3. Chen, D.Z., Gholami, A., Nießner, M., Chang, A.X.: Scan2Cap: Context-aware dense captioning in RGB-D scans. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3193–3203 (2021) 1, 2, 3
4. Chen, S., Fang, J., Zhang, Q., Liu, W., Wang, X.: Hierarchical aggregation for 3D instance segmentation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), pp. 15467–15476 (October 2021) 5
5. Choy, C., Gwak, J., Savarese, S.: 4D spatio-temporal ConvNets: Minkowski convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3075–3084 (2019) 4
6. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 5828–5839 (2017) 1
7. Dai, A., Nießner, M.: 3DMV: Joint 3D-multi-view prediction for 3D semantic scene segmentation. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 452–468 (2018) 4
8. Jiang, L., Zhao, H., Shi, S., Liu, S., Fu, C.W., Jia, J.: PointGroup: Dual-set point grouping for 3D instance segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4867–4876 (2020) 4
9. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 5
10. Misra, I., Girdhar, R., Joulin, A.: An End-to-End Transformer Model for 3D Object Detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2021) 6
11. Qi, C.R., Litany, O., He, K., Guibas, L.J.: Deep Hough voting for 3D object detection in point clouds. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9277–9286 (2019) 6
12. traveller59: spconv. https://github.com/traveller59/spconv (2021) 4
13. Zhang, Z., Sun, B., Yang, H., Huang, Q.: H3DNet: 3D object detection using hybrid geometric primitives. In: European Conference on Computer Vision, pp. 311–329, Springer (2020) 6
14. Zhao, L., Cai, D., Sheng, L., Xu, D.: 3DVG-Transformer: Relation modeling for visual grounding on point clouds. In: Proceedings of the IEEE/CVF

International Conference on Computer Vision, pp. 2928–2937 (2021) 1, 2, 3, 4