Text-guided Sparse Voxel Pruning for Efficient 3D Visual Grounding

Supplementary Material

We provide statistics and analysis for visual feature resolution (Sec. A), detailed comparisons of computational cost (Sec. B), detailed results on the ScanRefer dataset [1] (Sec. C), qualitative comparisons (Sec. D) and potential limitations (Sec. E) in the supplementary material.

A. Visual Feature Resolution of Different Architectures

To analyze the scene representation resolution of point-based and sparse convolutional architectures, we compare the resolution changes during the visual feature extraction process for EDA [11] and TSP3D-B, as illustrated in Fig. 1. For a thorough examination of the feature resolution of the sparse convolution architecture, we consider TSP3D-B without incorporating TGP and CBA. The voxel numbers for TSP3D-B are based on the average statistics from the ScanRefer validation set. In point-based architectures, the number of point features is fixed and does not vary with the scene size. In contrast, the number of voxel features in sparse convolutional architectures tends to increase as the scene size grows. This adaptive adjustment ensures that features do not become excessively sparse when processing larger scenes. As shown in Fig. 1, point-based architectures perform aggressive downsampling, with the first downsampling step reducing 50,000 points to just 2,048 points. Moreover, the final scene representation consists of only 1,024 points, leading to a relatively coarse representation. By contrast, convolution-based architectures progressively downsample and refine the scene representation through a multi-level structure. Overall, the sparse convolution architecture not only provides high-resolution scene representation but also achieves faster inference speed compared to point-based architectures.

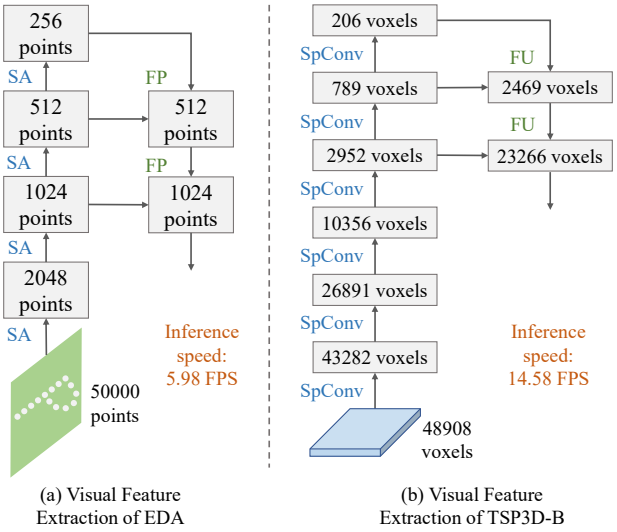

Figure 1. Feature resolution progression of point-based EDA and sparse convolutional TSP3D-B. SA, FP, SpConv, and FU represent set abstraction, feature propagation, sparse convolution, and feature upsampling, respectively. For the point-based architecture, the downsampling process is aggressive, with the first downsampling reducing 50,000 points directly to 2,048 points. Furthermore, the final scene representation consists of only 1,024 points. In contrast, the sparse convolutional architecture performs progressive downsampling and refines the scene representation through a multi-level structure. This approach not only provides a high-resolution scene representation but also achieves faster inference speed compared to the point-based architecture.

B. Detailed Computational Cost of Different Architectures

We provide a detailed comparison of the inference speed of specific components across different architectures, as shown in Tab. 1. Two-stage methods tend to have slower inference speed and are significantly impacted by the efficiency of the detection stage, which is not the primary focus of the 3DVG task. Therefore, we focus our analysis solely on the computational cost of single-stage methods. We divide the networks of existing methods and TSP3D into several components: text decoupling, visual backbone, text backbone, multi-modal fusion, and the head. The inference speed of each of these components is measured separately.

Backbone. Except for TSP3D, the visual backbone in other methods is PointNet++ [7], which has a high computational cost. This is precisely why we introduce a sparse convolution backbone, which achieves approximately three times the inference speed of PointNet++. As for the text backbone, both TSP3D and other methods use the pre-trained RoBERTa [5], so the inference speed for this component is largely consistent across the methods.

Multi-modal Fusion. The multi-modal feature fusion primarily involves the interaction between textual and visual features, with different methods employing different modules. For instance, the multi-modal fusion in SDSPS mainly includes the description-aware keypoint sampling (DKS) and target-oriented progressive mining (TPM) modules. And methods like BUTD-DETR, EDA, and MCLN rely on cross-modal encoders and decoders for their fusion process. In our TSP3D, the multi-modal fusion involves feature upsampling, text-guided pruning (TGP), and completion-based addition (CBA). Notably, even though

Table 1. Detailed comparison of computational cost for different single-stage architectures on the ScanRefer dataset [1]. The numbers in the table represent frames per second (FPS). TSP3D demonstrates superior processing speed across all components compared to other methods, with the inference speed of the sparse convolution backbone being three times faster than that of the point-based backbone.

Method	Text Decouple	Visual Backbone	Text Backbone	Multi-modal Fusion	Head	Overall
3D-SPS [6]	—	10.88	80.39	13.25	166.67	5.38
BUTD-DETR [4]	126.58	10.60	78.55	28.49	52.63	5.91
EDA [11]	126.58	10.89	81.10	28.57	49.75	5.98
MCLN [8]	126.58	10.52	76.92	23.26	41.32	5.45
TSP3D (Ours)	—	**31.88**	**81.21**	**28.67**	**547.32**	**12.43**

TSP3D progressively increases the resolution of scene features and integrates them with fine-grained backbone features, it still achieves superior inference speed. This is primarily due to the text-guided pruning, which significantly reduces the number of voxels and computational cost.

Head and Text Decouple. In the designs of methods such as BUTD-DETR, EDA, and MCLN, the input text needs to be decoupled into several semantic components. Additionally, their heads do not output prediction scores directly. Instead, they output embeddings for each candidate object, which must be compared with the embeddings of each word in the text to compute similarities and determine the final output. This can be considered additional preprocessing and post-processing steps, with the latter significantly impacting computational efficiency. In contrast, our TSP3D directly predicts the matching scores between the objects and the input text, making the head inference speed over ten times faster than these methods.

C. Detailed Results on ScanRefer

Due to page limitations, we report only the overall performances and inference speeds in the main text. To provide detailed results and analysis, we include the accuracies of TSP3D and other methods across various subsets on the ScanRefer dataset [1], as shown in Tab. 2. TSP3D achieves state-of-the-art accuracy, even when compared with two-stage methods, leading by $+1.13$ in Acc@0.5. TSP3D also demonstrates a level of efficiency that previous methods lack. In various subsets, TSP3D maintains comparable accuracy to both single-stage and two-stage state-of-the-art methods. Notably, the "multi-object" subset involves distinguishing the target object among numerous distractors of the same category within a more complex 3D scene. In this setting, TSP3D achieves a commendable performance of 42.37 in Acc@0.5, further demonstrating that TSP3D enhances attention to the target object in complex environments through text-guided pruning and completion-based addition, enabling accurate predictions of both the location and the shape of the target.

D. Qualitative Comparisons

To qualitatively demonstrate the effectiveness of our proposed TSP3D, we visualize the 3DVG results of TSP3D alongside EDA [11] on the ScanRefer dataset [1]. As shown in Fig. 2, the ground truth boxes are marked in blue, with the predicted boxes for EDA and TSP3D displayed in red and green, respectively. EDA encounters challenges in locating relevant objects, identifying categories, and distinguishing appearance and attributes, as illustrated in Fig. 2 (a), (c), and (d). In contrast, our TSP3D gradually focuses attention on the target and relevant objects under textual guidance and enhances resolution through multi-level feature fusion, showcasing commendable grounding capabilities. Furthermore, Fig. 2 (b) illustrates that TSP3D performs better with small or narrow targets, as our proposed completion-based addition can adaptively complete the target shape based on high-resolution backbone feature maps.

E. Limitations and Future Work

Despite its leading accuracy and inference speed, TSP3D still has some limitations. First, the speed of TSP3D is slightly slower than that of TSP3D-B. While TSP3D leverages TGP to enable deep interaction between visual and text features in an efficient manner, it inevitably introduces additional computational overhead compared to naive concatenation. In future work, we aim to focus on designing new operations for multi-modal feature interaction to replace the heavy cross-attention mechanism. Second, the current input for 3DVG methods consists of reconstructed point clouds. We plan to extend this to an online setting using streaming RGB-D videos as input, which would support a broader range of practical applications.

Table 2. Detailed comparison of methods on the ScanRefer dataset [1] evaluated at IoU thresholds of 0.25 and 0.5. TSP3D achieves state-of-the-art accuracy even compared with two-stage methods, with $+1.13$ lead on Acc@0.5. In various subsets, TSP3D achieves comparable accuracy to both single-stage and two-stage state-of-the-art methods. Additionally, TSP3D demonstrates a level of efficiency that previous methods lack.

Method	Venue	Unique (∼19%)		Multiple (∼81%)		Accuracy		Inference Speed (FPS)
		0.25	0.5	0.25	0.5	0.25	0.5	
Two-Stage Model								
ScanRefer [1]	ECCV'20	76.33	53.51	32.73	21.11	41.19	27.40	**6.72**
TGNN [3]	AAAI'21	68.61	56.80	29.84	23.18	37.37	29.70	3.19
InstanceRefer [13]	ICCV'21	77.45	66.83	31.27	24.77	40.23	30.15	2.33
SAT [12]	ICCV'21	73.21	50.83	37.64	25.16	44.54	30.14	4.34
FFL-3DOG [2]	ICCV'21	78.80	67.94	35.19	25.7	41.33	34.01	Not released
3D-SPS [6]	CVPR'22	84.12	66.72	40.32	29.82	48.82	36.98	3.17
BUTD-DETR [4]	ECCV'22	82.88	64.98	44.73	33.97	50.42	38.60	3.33
EDA [11]	CVPR'23	85.76	68.57	49.13	37.64	54.59	42.26	3.34
3D-VisTA [14]	ICCV'23	77.40	70.90	38.70	34.80	45.90	41.50	2.03
VPP-Net [9]	CVPR'24	86.05	67.09	50.32	39.03	55.65	43.29	Not released
G^3-LQ [10]	CVPR'24	**88.09**	**72.73**	51.48	**40.80**	56.90	**45.58**	Not released
MCLN [8]	ECCV'24	86.89	**72.73**	**51.96**	40.76	**57.17**	45.53	3.17
Single-stage Model								
3D-SPS [6]	CVPR'22	81.63	64.77	39.48	29.61	47.65	36.43	5.38
BUTD-DETR [4]	ECCV'22	81.47	61.24	44.20	32.81	50.22	37.87	5.91
EDA [11]	CVPR'23	86.40	69.42	48.11	36.82	53.83	41.70	5.98
G^3-LQ [10]	CVPR'24	**88.59**	**73.28**	50.23	39.72	55.95	44.72	Not released
MCLN [8]	ECCV'24	84.43	68.36	49.72	38.41	54.30	42.64	5.45
TSP3D (Ours)	——	87.25	71.41	**51.04**	**42.37**	**56.45**	**46.71**	**12.43**

Figure 2. Qualitative results of EDA [11] and our TSP3D on the ScanRefer dataset [1]. In each description, the red annotations indicate the target object. The orange annotations in (a) refer to relevant objects, while the yellow annotations in (d) denote the appearance or attributes of the target. TSP3D demonstrates exceptional performance in locating relevant objects, narrow or small targets, identifying categories, and distinguishing appearance and attributes.

References

[1] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *ECCV*, pages 202–221. Springer, 2020. 1, 2, 3, 4

[2] Mingtao Feng, Zhen Li, Qi Li, Liang Zhang, XiangDong Zhang, Guangming Zhu, Hui Zhang, Yaonan Wang, and Ajmal Mian. Free-form description guided 3d visual graph network for object grounding in point cloud. In *ICCV*, pages 3722–3731, 2021. 3

[3] Pin-Hao Huang, Han-Hung Lee, Hwann-Tzong Chen, and Tyng-Luh Liu. Text-guided graph neural networks for referring 3d instance segmentation. In *AAAI*, pages 1610–1618, 2021. 3

[4] Ayush Jain, Nikolaos Gkanatsios, Ishita Mediratta, and Katerina Fragkiadaki. Bottom up top down detection transformers for language grounding in images and point clouds. In *ECCV*, pages 417–433. Springer, 2022. 2, 3

[5] Yinhan Liu. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 1

[6] Junyu Luo, Jiahui Fu, Xianghao Kong, Chen Gao, Haibing Ren, Hao Shen, Huaxia Xia, and Si Liu. 3d-sps: Single-stage 3d visual grounding via referred point progressive selection. In *CVPR*, pages 16454–16463, 2022. 2, 3

[7] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *NeurIPS*, 30, 2017. 1

[8] Zhipeng Qian, Yiwei Ma, Zhekai Lin, Jiayi Ji, Xiawu Zheng, Xiaoshuai Sun, and Rongrong Ji. Multi-branch collaborative learning network for 3d visual grounding. In *ECCV*, pages 381–398. Springer, 2025. 2, 3

[9] Xiangxi Shi, Zhonghua Wu, and Stefan Lee. Viewpoint-aware visual grounding in 3d scenes. In *CVPR*, pages 14056–14065, 2024. 3

[10] Yuan Wang, Yali Li, and Shengjin Wang. Gˆ 3-lq: Marrying hyperbolic alignment with explicit semantic-geometric modeling for 3d visual grounding. In *CVPR*, pages 13917–13926, 2024. 3

[11] Yanmin Wu, Xinhua Cheng, Renrui Zhang, Zesen Cheng, and Jian Zhang. Eda: Explicit text-decoupling and dense alignment for 3d visual grounding. In *CVPR*, pages 19231–19242, 2023. 1, 2, 3, 4

[12] Zhengyuan Yang, Songyang Zhang, Liwei Wang, and Jiebo Luo. Sat: 2d semantics assisted training for 3d visual grounding. In *ICCV*, pages 1856–1866, 2021. 3

[13] Zhihao Yuan, Xu Yan, Yinghong Liao, Ruimao Zhang, Sheng Wang, Zhen Li, and Shuguang Cui. Instancerefer: Cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring. In *ICCV*, pages 1791–1800, 2021. 3

[14] Ziyu Zhu, Xiaojian Ma, Yixin Chen, Zhidong Deng, Siyuan Huang, and Qing Li. 3d-vista: Pre-trained transformer for 3d vision and text alignment. In *ICCV*, pages 2911–2921, 2023. 3