AG^{2}aussian: Anchor-Graph Structured Gaussian Splatting for Instance-Level 3D Scene Understanding and Editing

Supplementary Material

In this supplementary material, we first present the implementation detail of the physical simulation task in Sec. A. Then, we present more object query comparisons in Sec. B and Sec. C. In Sec. D, we demonstrate the robustness of our editing method and provide additional editing results on two scenes from the Mip-NeRF360 dataset [1]. Finally, in Sec. F and Sec. G, we provide more ablation study results, including both qualitative and quantitative analysis.

A. Application of Physical Simulation

In our experiment, we adopt PhyGaussian [12], a Gaussian-based simulator implemented via MLS-MPM [4], as our physical engine. The Gaussians are regarded as particles to perform the simulation. For computational efficiency purposes, we remove the background using a bounding box and retain only the foreground particles whose opacity $\alpha > 0.02$ for simulation. Specifically, in our experiments, we first use a query operation to select the object to be simulated. This object is then assigned Young's modulus $E = 2e^8$ and Poisson's ratio $\nu = 0.4$ to prevent deformation during simulation. The remaining particles within the bounding box, which serve as sticky boundary conditions with lower physical coefficients ($E = 2e^6$, $\nu = 0.3$), enable the simulated object to be easily separated from the surroundings. All of these particles are subsequently discretized into a grid 64^3. For all the physical simulation experiments, we simulate a total of 30 frames. All particles in this application are assigned von Mises Plasticity material.

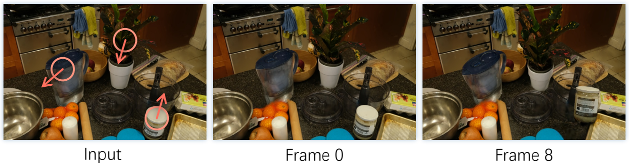

Figure 1. Physical simulation by applying the external forces (red arrows) to drag the selected objects.

B. More Text Query Results

We visualize more results of the open-vocabulary text query task in Figure 2, where our method demonstrates a clear advantage in selecting the complete 3D objects. By contrast, OpenGaussian [11], due to its codebook-based clustering approach, often fails to group an entire object into a single cluster, as seen with the ”waldo” in the first row and the ”stuffed bear” in the second row. Similarly, GsGrouping [13] frequently includes incorrect object IDs for the query, as seen with the ”stuffed bear” in the second row and the ”glass of water” in the third row. Meanwhile, SAGA [2] uses a limited number of clusters and is less aware of spatial information, making it prone to missing matches and selecting incorrect regions.

In Table 2 and Figure 6, we further report both the quantitative and qualitative results of open-vocabulary querying on Mip-NeRF360 [1], evaluated with the vocabulary provided by LEGaussian [10]. Our results consistently outperform existing approaches, achieving significant improvements in both mIoU and mBIoU. These gains hold across diverse scenes and object types, and are especially observed on thin, partially occluded, or clutter-surrounded objects. Qualitative results further validate that our selected regions can align well with the entire instance, whereas others always leave fragmented or jagged boundaries.

C. More Click Query Results

We report more object selection results on LLFF [8] in Figure 7 and Table 3, using the scribbles provided by NVOS [9]. As input, we first shrink the scribbles into skeleton lines and then use the pixels on the skeleton as click query points. By contrast, our method yields more accurate segmentation for complex objects like fern and dinosaur fossils, benefiting from the use of localized anchor-Gaussian and our anchor-graph-based strategy.

D. More Object Editing Results

Directly removing the selected Gaussians for the objects makes artifacts in the remaining scene, due to the missing observations of the occluded region across all views, as shown in the left column of Figure 3. Thus an inpainting operation is necessary to fill the holes.

We compare the two inpainting techniques adopted by GsGrouping [13] and our approach, which differ in localizing the artifact regions to be repaired. GsGrouping uses Deva Tracking [3]. As shown in the top row of the figure, due to ambiguous features and the difficulty of precisely identifying the hole regions, most viewpoints fail to maintain a stable artifact mask, resulting in suboptimal editing outcomes. By contrast, our anchor-graph structure enables an accurate selection of the object including the inner Gaussians, thus providing a precise localization of the artifact region by extending the boundary of the selected object,

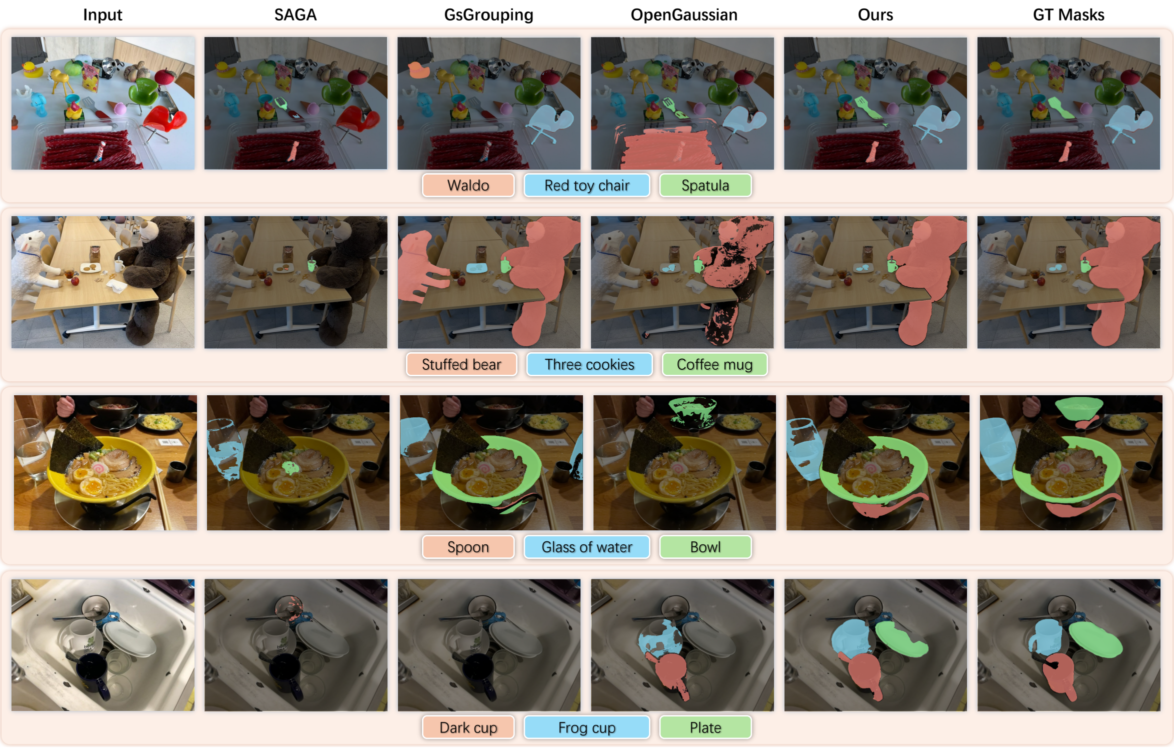

Figure 2. Open-vocabulary 3D object selection on the LERF dataset [5]. AG^{2}aussian outperforms other approaches in accurately identifying the clean and complete 3D objects corresponding to text queries.

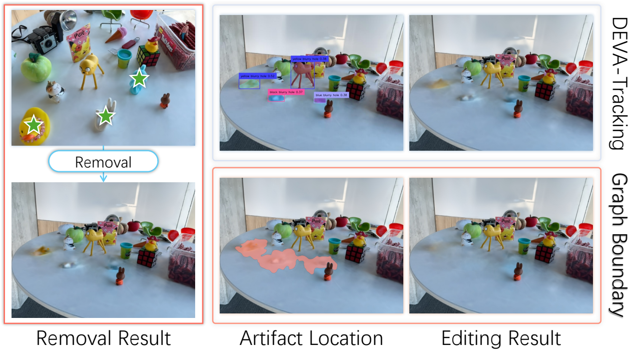

Figure 3. Object removing editing results with artifact regions localized and inpainted with different techniques. Compared to the DEVA-Tracking [3] adopted by GsGrouping [13] (top row), our anchor-graph structured representation (bottom row) enables an accurate localization of the artifact regions and thus makes realistic inpainting results without affecting the surrounding objects.

yielding more reliable and visually coherent editing results.

To further validate the performance of our artifact localization, we performed editing experiments on two scenes from the Mip-NeRF360 dataset [1]. For the counter scene, we removed three objects of varying sizes, including a transparent kettle. As for the kitchen scene, we evaluated our method's ability to repair large hole regions resulting from object removal. As shown in Figure 4, our approach accurately identifies and fills the hole regions, resulting in high-quality and consistent scene editing.

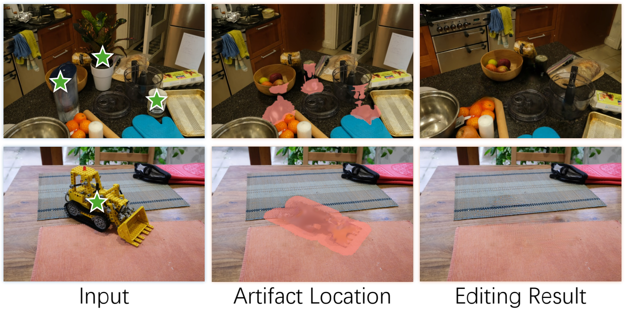

Figure 4. More editing results on MipNeRF360 [1] using our graph-based artifact localization technique.

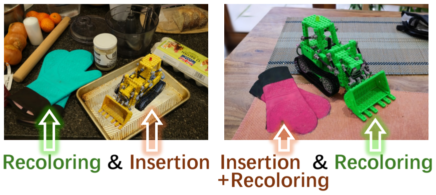

Figure 5. More object recoloring and insertion editing results on MipNeRF360 [1].

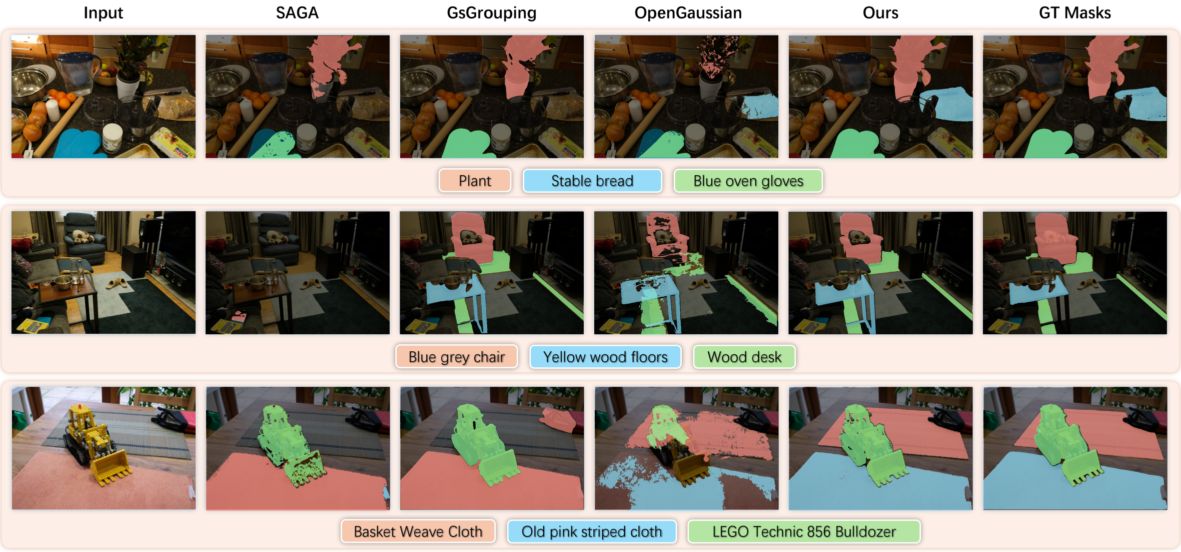

Figure 6. Open-vocabulary 3D object selection on the Mip-NeRF360 dataset [1].

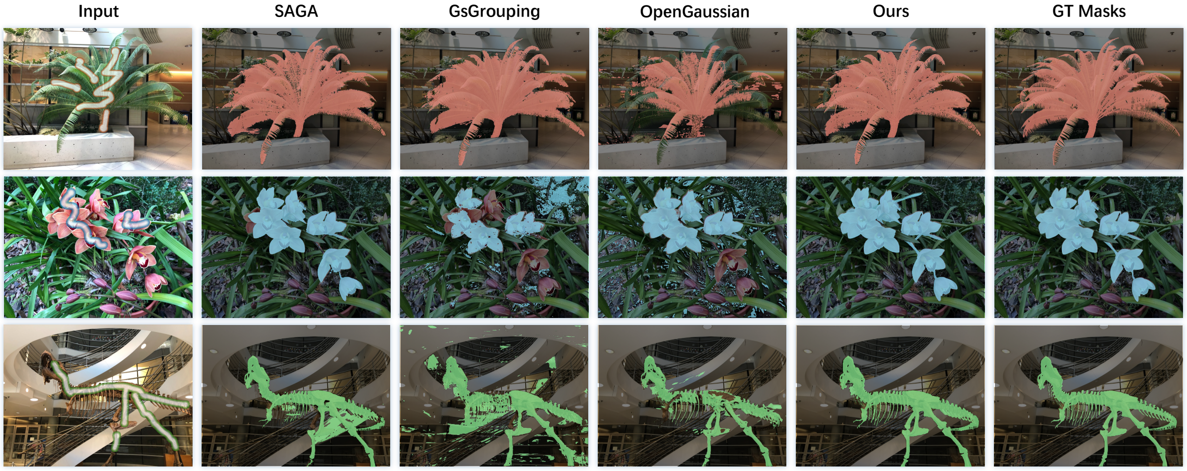

Figure 7. Scribbles-based 3D object selection on the LLFF dataset [1].

Additionally, we present the results of object recoloring and insertion of a complex scene in Figure 5, which contains many objects close to each other and has occlusions across multiple views.

E. Computation Overhead

The maximum reserved memory, training time, and rendering FPS are reported in Table 1. For our anchor-graph structure, we store the anchors only for the occupied voxels and the sparse edges between neighbor anchors, incurring minimal additional memory. On the other hand, this structure regularizes the Gaussian primitives to lie around the object surfaces, which largely reduces the number of Gaussians and thus the training time. Notably, we do not intend to claim a faster rendering speed, since we implemented a CUDA-based module to render RGB, feature map, and other outputs in one pass, while SAGA and OpenGaussian need to invoke the renderer multiple times.

Table 1. Computation Overhead on LERF dataset [5].

Methods	Memory↓	Train Time↓	Rendering FPS↑
SAGA	13.29 GB	**33.63 mins**	∼252
GsGrouping	20.21 GB	51.21 mins	∼114
OpenGaussian	16.81 GB	74.31 mins	∼96
w/ codebook	12.91 GB	69.97 mins	∼185
Ours	**7.56 GB**	39.55 mins	**∼515**

Table 2. Quantitative evaluation of text querying on Mip-NeRF360 dataset [1].

Methods	mIoU. ↑							mBIoU. ↑						
	bicycle	bonsai	counter	garden	kitchen	room	**Mean**	bicycle	bonsai	counter	garden	kitchen	room	**Mean**
SAGA	1.58	32.38	19.24	19.21	17.26	0.16	14.97	2.13	24.21	15.68	15.36	9.33	0.2	11.15
GsGrouping	10.52	**68.73**	47.73	**34.59**	61.7	41.22	44.08	8.89	**53.62**	44.61	29.52	**54.52**	36.04	37.86
OpenGaussian	25.97	33.2	47.52	25.87	41.42	41.7	35.94	15.41	26.34	41.59	20.87	21.7	35.84	26.95
Ours	**31.15**	53.47	**61.89**	34.46	**62.26**	**50.76**	**48.99**	**18.94**	48.49	**58.85**	**31.52**	41.77	**45.03**	**40.76**

Table 3. Quantitative evaluation of click querying on LLFF dataset [8].

Methods	mIoU. ↑									mBIoU. ↑								
	fern	flower	fortress	horns_c	horns_l	leaves	orchids	trex	Mean	fern	flower	fortress	horns_c	horns_l	leaves	orchids	trex	Mean
SAGA	**82.53**	95.15	98.15	92.83	94.57	92.88	88.82	83.99	91.61	75.12	80.87	78.18	68.44	72.2	77.89	74.76	70.25	75.04
GsGrouping	80.70	57.72	97.75	96.78	94.58	70.5	36.13	51.69	72.73	64.74	35.99	55.02	69.74	73.57	48.68	26.09	49.38	52.56
OpenGaussian	70.74	62.63	94.91	79.81	77.81	87.68	59.88	68.88	75.29	58.81	36.75	67.38	47.24	52.85	57.81	43.49	66.25	53.82
Ours	82.01	**95.38**	**98.59**	**97.36**	**96.31**	**93.89**	**90.76**	**87.02**	**92.66**	**77.85**	**81.73**	**91.06**	**81.24**	**83.54**	**80.71**	**80.42**	**85.24**	**82.64**

F. More Ablation Study Results

Table 4 presents the complete ablation study results on the LERF dataset [5]. Overall, our graph-related operations significantly improve both mask completeness and boundary quality, as evidenced by notable gains in mIoU and mBIoU.

To further assess the importance of these operations for the query task, we demonstrate the selected Gaussians and the remaining scenes. Figure 8 provides a full visualization of all ablation variants. Our graph-based region growing effectively prevents the selection of Gaussians outside the target object, as demonstrated by the comparison between the $w/o\ GraphSeg$ variant and our full method. Moreover, our graph propagation smooths the feature field within the object and enhances a clean Gaussian selection, effectively eliminating inner Gaussians in the remaining scenes, as shown by the comparison between $w/o\ \mathcal{L}_{prop}$ and our full method. Additionally, our anchor-Gaussian structure effectively constrains the local distribution of Gaussians, as demonstrated by the comparison between $w/o\ ag$ and $w/o\ Graph$. Overall, our full method not only enables the clean selection of objects but also ensures the comprehensive inclusiveness of the inner object Gaussians.

G. Comparison with Other Structured-GSes

Several recent works explore structured 3DGS, but for different goals and thus framework designs. Scaffold-GS [7] proposes the Anchor-Gaussian structure to distribute local 3D Gaussians and predicts their view-adaptive attributes. However, it does not localize the Gaussians to distribute within the voxel of the corresponding anchor, and eliminates the anchor-graph for the feature propagation. SuperGSeg [6] proposes to cluster the optimized Gaussians into Super-Gaussians and distill the semantic features to comprehensively understand 3D scenes. However, it lacks anchor-graph-based propagation to further refine the local feature fields and requires a much larger memory cost during training.

Therefore, we perform the ablation study experiments ($w/o\ localization$ and $w/\ codebook$) to validate the effectiveness of our design, as shown in Table 5. Specifically, for $w/o\ localization$, we remove the scaling constraint (Eq. 2) and structured spatial regularization (Eq. 3-4), to evaluate the effectiveness of our anchor-graph structure compared to ScaffoldGS and SuperGSeg. For $w/\ codebook$, we preserve our stage 1 and introduce a learnable codebook to emulate the Super-Gaussians proposed by SuperGSeg. Our full approach significantly outperforms both variants in segmentation accuracy, demonstrating the advantages of our anchor-graph–based localization and propagation.

References

[1] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 5470–5479, 2022. 1, 2, 3, 4

[2] Jiazhong Cen, Jiemin Fang, Chen Yang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Segment any 3d gaussians. In Proceedings of the AAAI Conference on Artificial Intelligence, pages 1971–1979, 2025. 1

[3] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Alexander Schwing, and Joon-Young Lee. Tracking anything with decoupled video segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 1316–1326, 2023. 1, 2

[4] Yuanming Hu, Yu Fang, Ziheng Ge, Ziyin Qu, Yixin Zhu, Andre Pradhana, and Chenfanfu Jiang. A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. ACM Transactions on Graphics (TOG), 37(4):150, 2018. 1

[5] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 19729–19739, 2023. 2, 3, 4, 5

[6] Siyun Liang, Sen Wang, Kunyi Li, Michael Niemeyer, Stefano Gasperini, Nassir Navab, and Federico Tombari. Su-

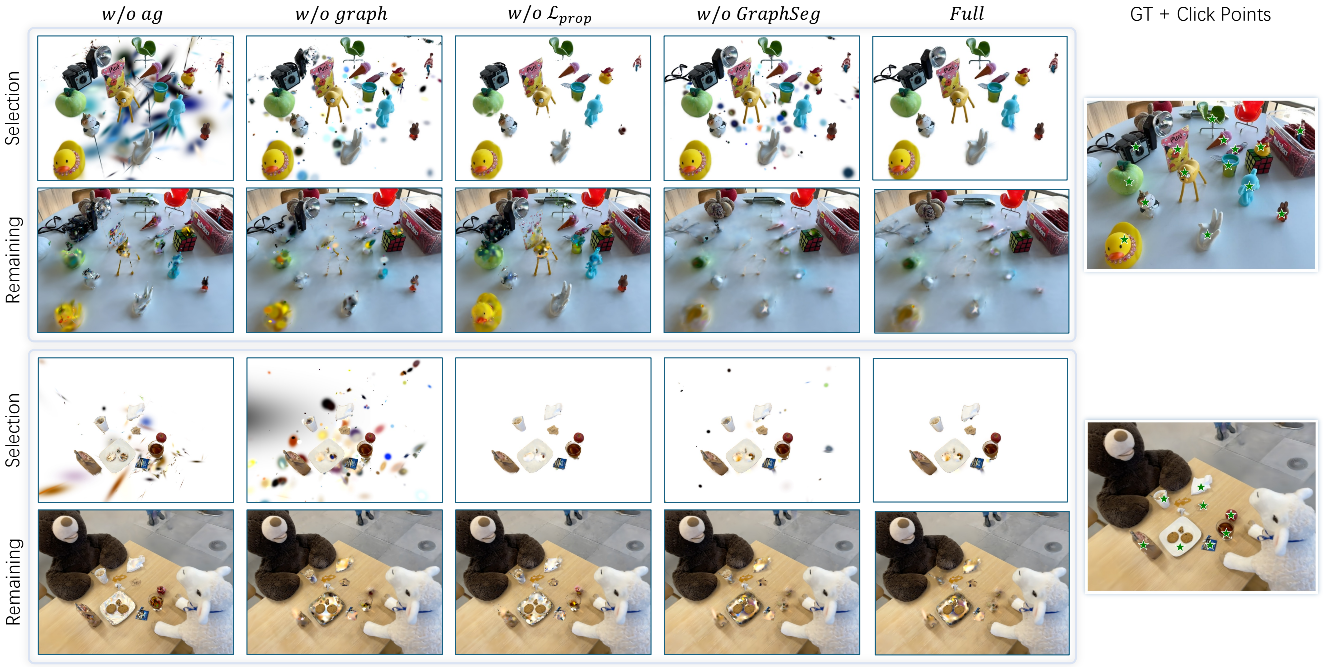

Figure 8. Ablation study results. We separately validated the importance of our key design for segmentation task, the Anchor-Gaussian structure, and the Graph-based operation. The advantage of the Anchor-Gaussian is demonstrated by comparing $w/o\ GraphSeg$ with a variant that uses 3DGS without our anchor-graph ($w/o\ ag$). The effectiveness of our Graph-based Operation respectively adopting $w/o\ graph$, $w/o\ \mathcal{L}_{prop}$ and $w/o\ GraphSeg$.

Table 4. Full ablation studies on the LERF-OVS dataset [5] about the key designed.

Case	$w/\ \mathcal{L}_{prop}$	$w/\ GraphSeg$	mIoU ↑					mBIoU. ↑				
			figurines	teatime	ramen	kitchen	**Mean**	figurines	teatime	ramen	kitchen	**Mean**
#1			57.62	64.72	26.39	22.14	42.72	56.72	61.30	26.05	16.73	40.20
#2		✓	55.95	66.54	31.45	29.50	45.85	58.59	63.13	31.02	21.32	43.51
#3	✓		65.08	71.16	28.15	**32.01**	49.10	63.61	67.33	26.46	21.46	44.72
Full	✓	✓	**66.98**	**71.62**	**47.99**	30.82	**54.35**	**65.30**	**67.83**	**42.45**	**22.15**	**49.43**

Table 5. Ablation Study of Structured-GS Design on LERF-OVS dataset [5]

Methods	mIoU ↑					mBIoU. ↑				
	figurines	teatime	ramen	kitchen	**Mean**	figurines	teatime	ramen	kitchen	**Mean**
w/ codebook	49.83	66.77	17.99	27.48	40.51	35.33	60.67	15.29	20.3	32.89
w/o localized	28.40	53.94	14.51	24.97	30.45	25.94	51.39	13.04	19.73	27.52
Ours	**66.98**	**71.62**	**47.99**	**30.82**	**54.35**	**65.30**	**67.83**	**42.45**	**22.15**	**49.43**

pergseg: Open-vocabulary 3d segmentation with structured super-gaussians, 2024. 4

[7] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 20654–20664, 2024. 4

[8] Ben Mildenhall, Pratul P. Srinivasan, Rodrigo Ortiz-Cayon, Nima Khademi Kalantari, Ravi Ramamoorthi, Ren Ng, and Abhishek Kar. Local light field fusion: Practical view synthesis with prescriptive sampling guidelines. ACM Transactions on Graphics (TOG), 2019. 1, 4

[9] Zhongzheng Ren, Aseem Agarwala†, Bryan Russell†, Alexander G. Schwing†, and Oliver Wang†. Neural volumetric object selection. In IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022. († alphabetic ordering). 1

[10] Jin-Chuan Shi, Miao Wang, Hao-Bin Duan, and Shao-Hua Guan. Language embedded 3d gaussians for open-vocabulary scene understanding. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5333–5343, 2024. 1

[11] Yanmin Wu, Jiarui Meng, Haijie Li, Chenming Wu, Yahao Shi, Xinhua Cheng, Chen Zhao, Haocheng Feng, Errui Ding, Jingdong Wang, et al. Opengaussian: Towards point-level 3d gaussian-based open vocabulary understanding. Advances in Neural Information Processing Systems, 37:19114–19138, 2024. 1

[12] Tianyi Xie, Zeshun Zong, Yuxing Qiu, Xuan Li, Yutao Feng, Yin Yang, and Chenfanfu Jiang. Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4389–4398, 2024. 1

[13] Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3d scenes. In ECCV, 2024. 1, 2