CSGaussian: Progressive Rate-Distortion Compression and Segmentation for 3D Gaussian Splatting - Supplementary Material

Yu-Jen Tseng[1], Chia-Hao Kao[2], Jing-Zhong Chen[1], Alessandro Gnutti[2],
Shao-Yuan Lo[3], Yen-Yu Lin[1], Wen-Hsiao Peng[1]
[1]National Yang Ming Chiao Tung University, Taiwan
[2]University of Brescia, Italy [3]National Taiwan University, Taiwan

A. Experiment Details

A.1. SAM Mask Preprocess

We follow the approach outlined in Langsplat [8] to extract 2D masks using the Segment Anything Model (SAM) [3] and language features provided by CLIP [9]. Specifically, we process the input images through SAM to generate masks at three levels: "subpart," "part," and "whole." For each mask produced by SAM, we calculate the corresponding bounding box, which is then used to crop the input images. These cropped images serve as input to CLIP, allowing us to obtain the language features associated with each mask. For the LERF [2] dataset, we select the third level, "whole," in accordance with our baseline models [5, 10] to ensure a fair comparison. Similarly, for the 3D-OVS [6] dataset, we also choose the third level, "whole," for all experiments except for the "bed" scene. In that case, we use the "part" level since SAM incorrectly labeled the hand and banana as the same class.

A.2. Training

Our training procedure includes three stages: color-only 3DGS optimization and compression, 3DGS segmentation learning, and 3D semantic feature compression. The first stage follows [7, 11] to optimize the color-only 3DGS through rate-distortion optimization with 40,000 iterations. At the 3,000th iteration, we employ the masking technique from [1, 4, 11] to eliminate irrelevant Gaussian primitives. This involves a learnable binary mask $M_{n,k}$ that is applied to suppress the k-th Gaussian primitive of the n-th anchor. The formulation for the mask is given by $M_{n,k} = \mathbf{1}(\text{sigmoid}(m_{n,k}) > \epsilon)$, where $m_{n,k}$ is a learnable parameter and ϵ is a masking threshold applied consistently across all Gaussian primitives to determine their existence. The first MLP model in the INR-based hyperprior is introduced at the 10,000th iteration. After completing the first stage of training, we fix the parameters, including anchor features, position, offset, and scaling. We then proceed with compression-guided segmentation learning for an additional 40,000 iterations. Finally, we employ the second MLP model in the INR-based hyperprior and focus on learning the compressed semantic features over another 30,000 iterations. Details regarding the hyperparameter settings can be found in Table 1.

Table 1. Details of hyperparameter settings

Parameter	Value
number of offset K	10
offset learning rate	0.01
mask learning rate	0.01
scaling learning rate	0.07
opacity MLP learning rate	0.002
color MLP learning rate	0.08
covariance MLP learning rate	0.004
semantic feature learning rate	0.001
masking threshold ϵ	0.01

B. Additional Experiments

B.1. Generalizability of Quantization-Aware Training (QAT)

To further validate the generalization of our proposed quantization-aware training, we apply it to OpenGaussian [10] and InstanceGS [5], both of which do not incorporate compression. As shown in Table 2, the mean Intersection over Union (%) was reported for the settings with and without quantization-aware training. QAT consistently enhances segmentation performance on both 3DGS segmentation methods, demonstrating its broad applicability outside of compression settings.

B.2. Comparison with Conditional Entropy Coding Approach

In our proposed framework, the semantic features $\boldsymbol{s}$ are entropy encoded independently of anchor features $\boldsymbol{f}$. To val-

Table 2. QAT generalizability comparison, reported in mIoU.

Method	**Without QAT**	**With QAT**
OpenGaussian	44.4	46.6
InstanceGS	45.9	47.5

idate our approach, we implement a scheme where decoded $\boldsymbol{f}$ is served as conditions for an additional multi-layer perceptron to help predict the distributions of semantic features $\boldsymbol{s}$. As shown in Figure 1, this approach results in inferior performance than our independent design, suggesting that anchor and semantic features are not highly correlated, while the additional MLP introduces unnecessary bitrate overhead.

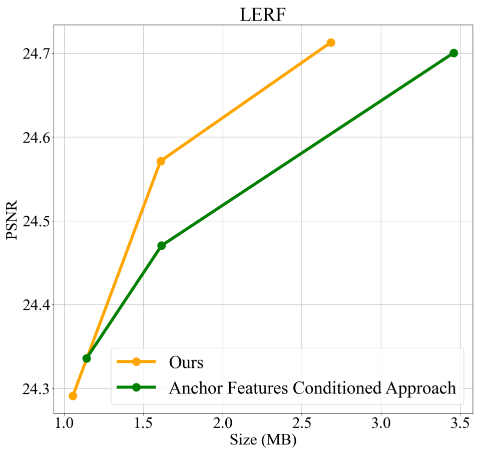

Figure 1. Anchor features conditioned rate-distortion comparison.

C. More Visualization

C.1. Quantization-Aware Training Analysis

We analyze the impact of quantization-aware training (QAT) using t-SNE visualization. We retrieve semantic features based on all provided text queries related to the scene “figurines” in LERF, comparing two settings: one with quantization-aware training and one without it. We then apply t-SNE visualization to both sets of features to examine the projected dimensions. As shown in Figure 3, the highlighted red boxes represent the features corresponding to the objects “rubber duck with hat” and “rubics cube”. With the help of QAT, these two objects are more distinctly separated. In contrast, without QAT, the model struggles to accurately differentiate between them. The object selection visualization in the bottom row of Figure 3 also demonstrates the effectiveness of our proposed QAT approach, showing a clear separation between the two objects.

C.2. 3D Scene Manipulation

Given a text query, we select the corresponding object and further illustrate various downstream applications of 3D scene manipulation that can be performed on these objects, such as removal (see Figure 4), insertion (Figure 5), and color modification (see Figure 6).

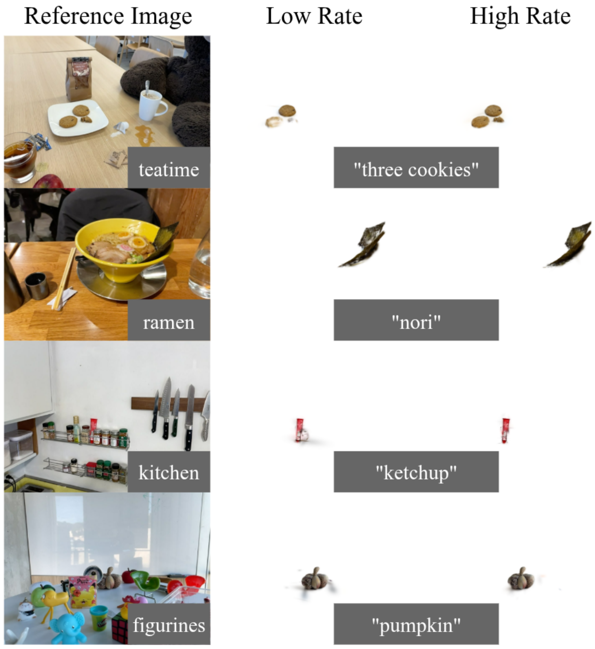

Figure 2. Low rate open-vocabulary segmentation results versus the high rate one.

C.3. Qualitative Results

We present a more qualitative comparison of our method on both LERF (see Figure 7) and 3D-OVS (see Figure 8) against the baseline approaches of OpenGaussian [10] and InstanceGS [5]. The visualizations demonstrate the effectiveness of our approach relative to these two methodologies.

D. Limitation

We further analyze some limitations within our proposed framework. We found a significant gap between performance at low rates and high rates, prompting us to investigate the differences illustrated in Figure 2. In the low-rate results, we identified two main factors affecting performance: (1) the objects tend to be more fragmented (as seen with “three cookies” and “nori”), and (2) the objects contain a greater amount of noisy Gaussian primitives (see “ketchup” and “pumpkin”). These limitations may stem from the lower quality of the foreground objects and the sparsity of Gaussian primitives in the low-rate setting. Addressing these limitations will be a focus of future work.

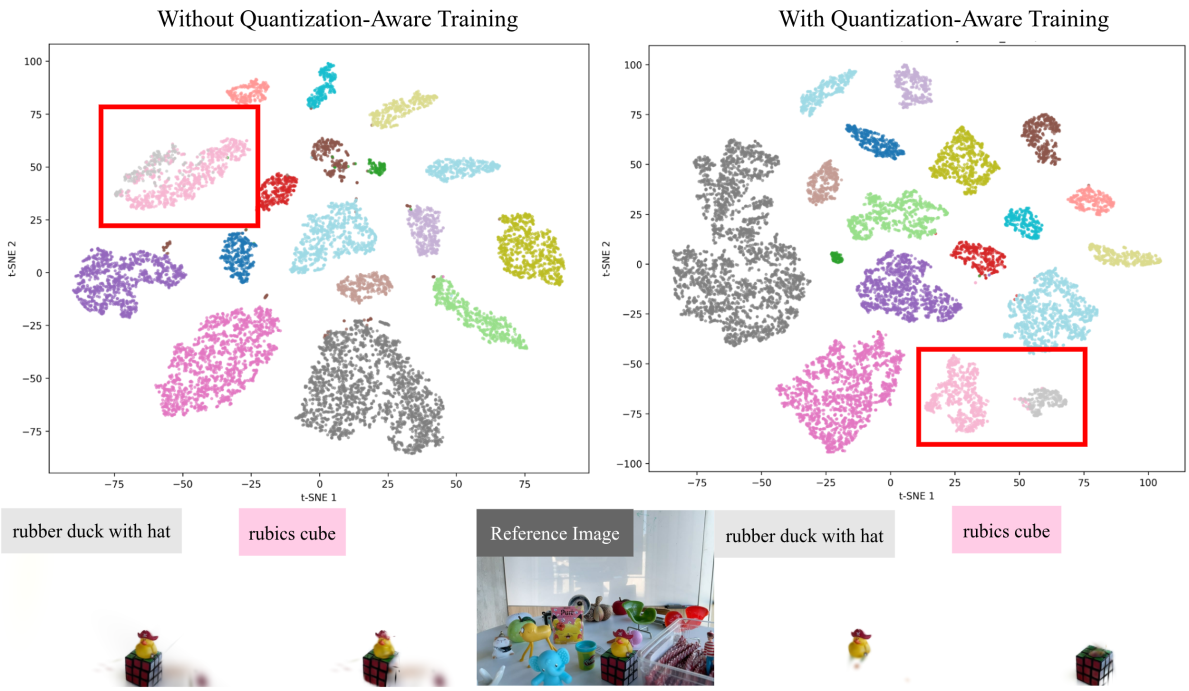

Figure 3. t-SNE Visualization

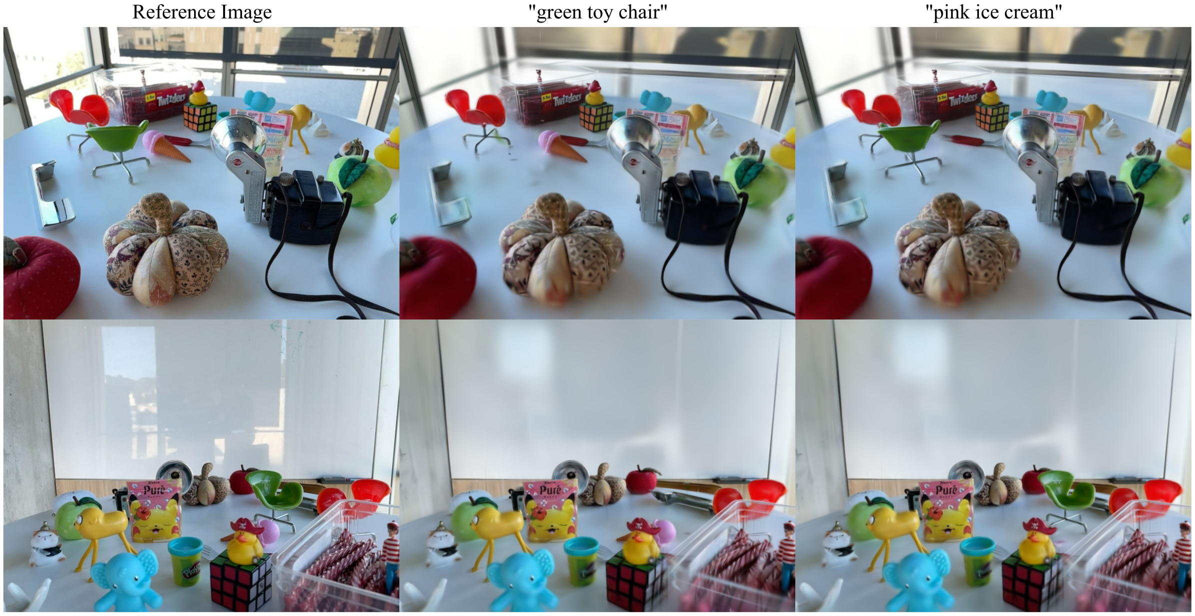

Figure 4. Object Removal: We remove the object based on the given text query.

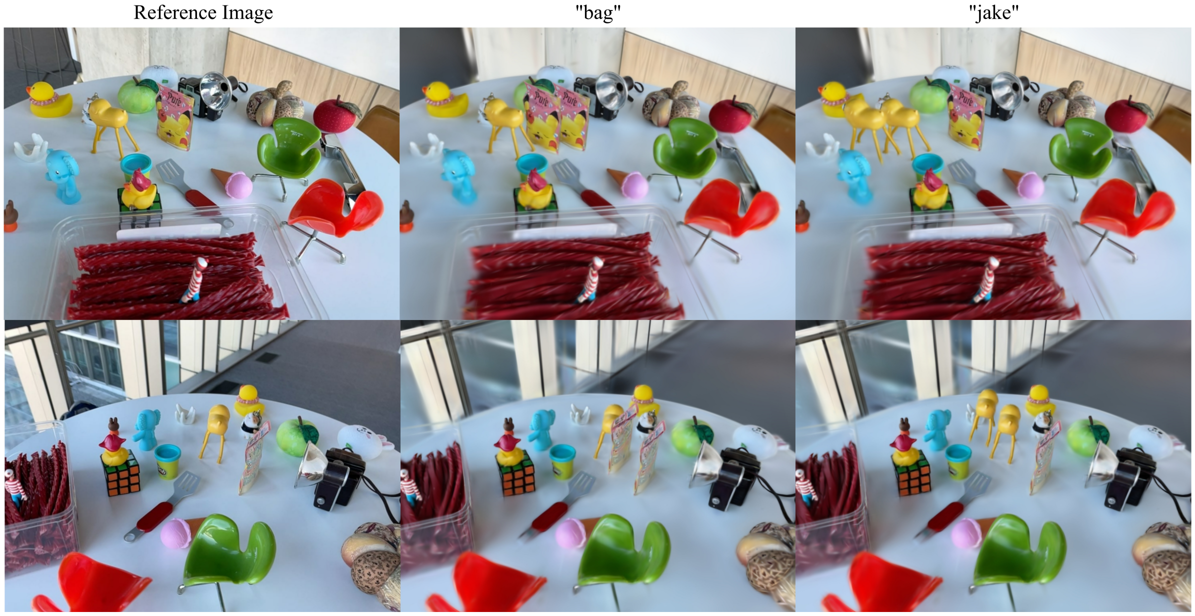

Figure 5. Object Insertion: We duplicate the chosen object and place it beside the original.

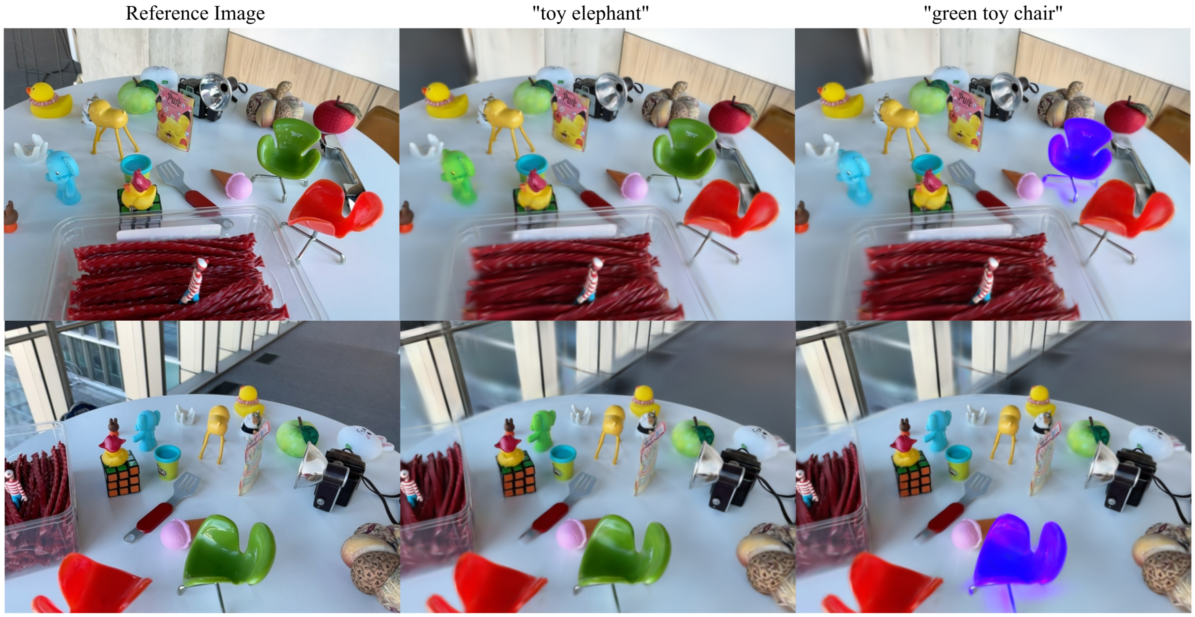

Figure 6. Object Color Modification: The chosen object is changed to a different color.

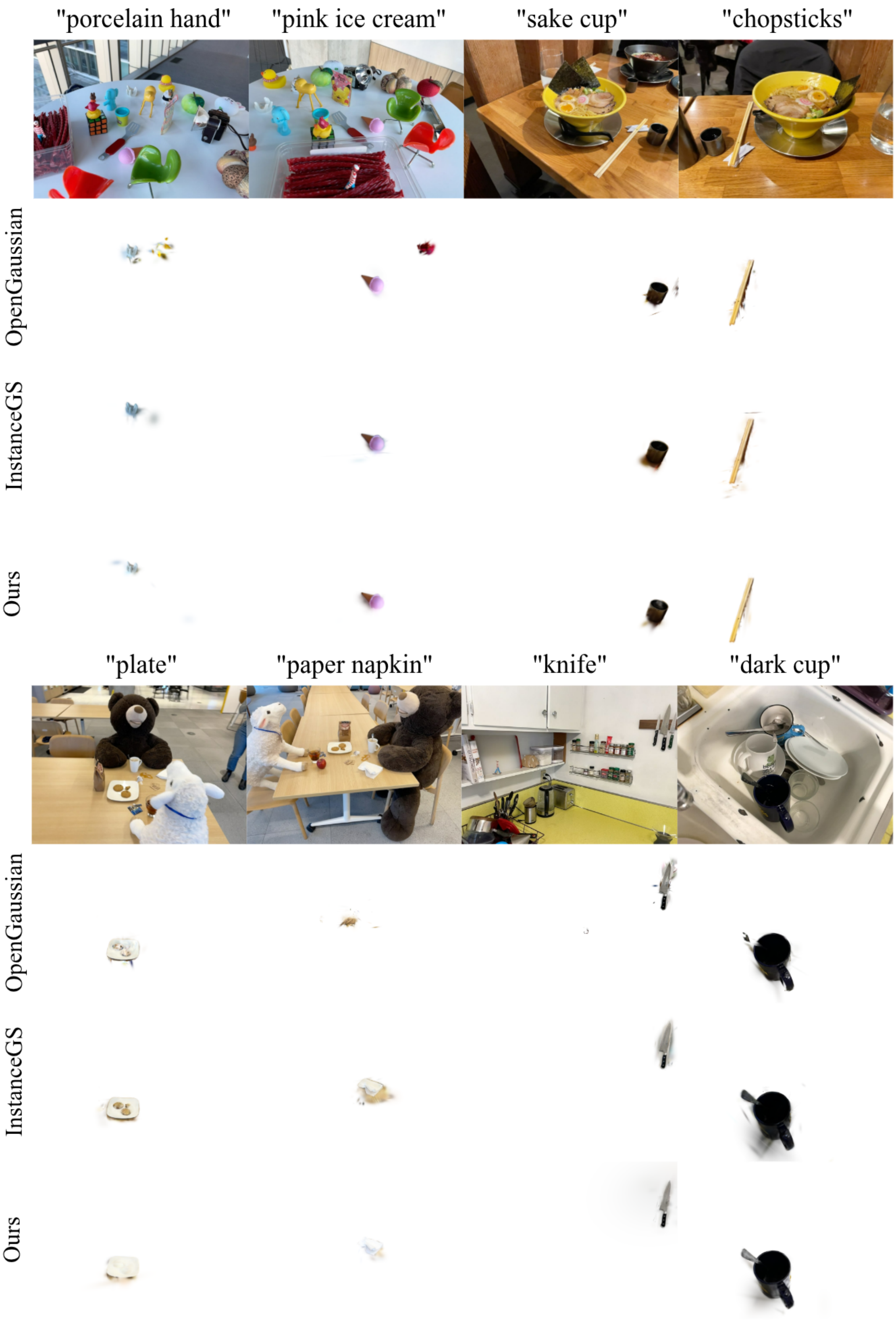

Figure 7. More visualization on LERF

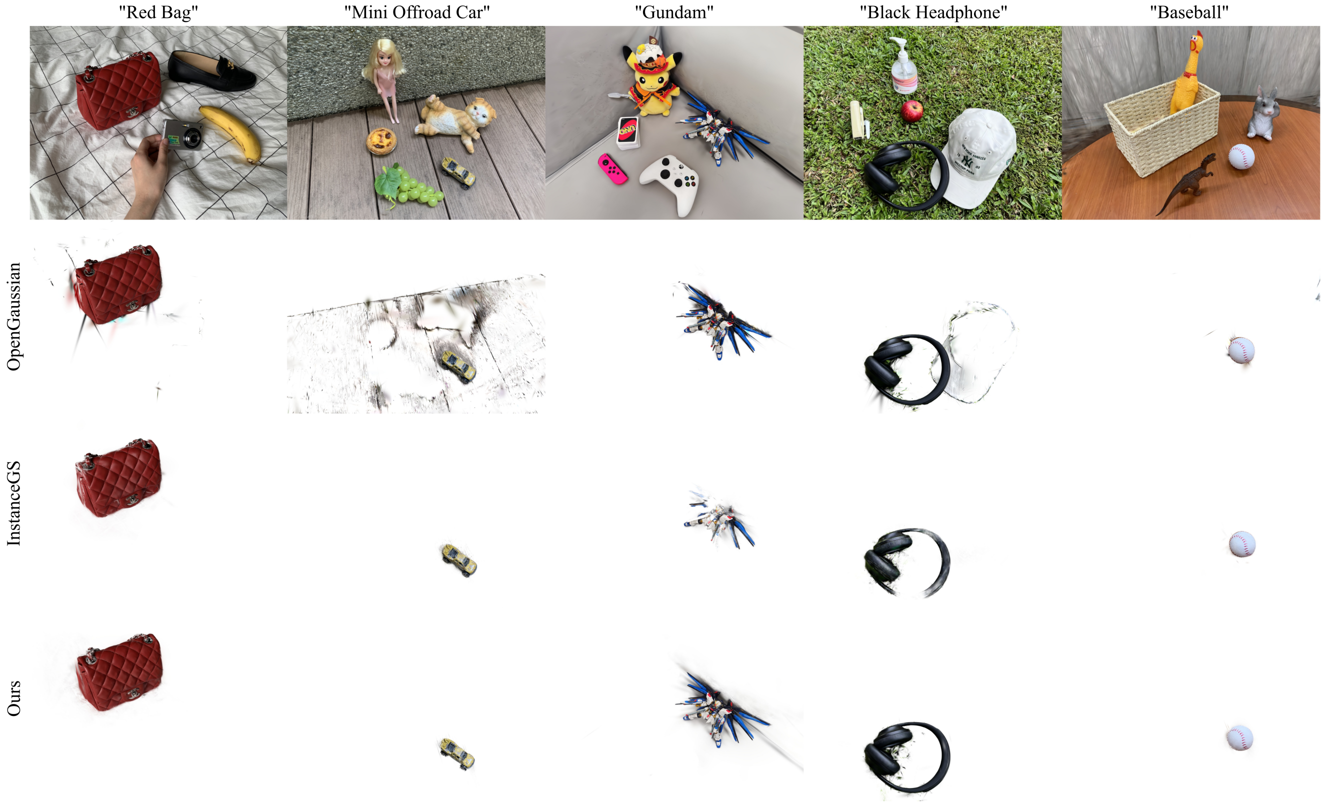

Figure 8. More visualization on 3D-OVS

References

[1] Yihang Chen, Qianyi Wu, Weiyao Lin, Mehrtash Harandi, and Jianfei Cai. Hac: Hash-grid assisted context for 3d gaussian splatting compression. In *European Conference on Computer Vision*, 2024. 1

[2] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. Lerf: Language embedded radiance fields. In *IEEE/CVF International Conference on Computer Vision*, 2023. 1

[3] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *IEEE/CVF International Conference on Computer Vision*, 2023. 1

[4] Joo Chan Lee, Daniel Rho, Xiangyu Sun, Jong Hwan Ko, and Eunbyung Park. Compact 3d gaussian representation for radiance field. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1

[5] Haijie Li, Yanmin Wu, Jiarui Meng, Qiankun Gao, Zhiyao Zhang, Ronggang Wang, and Jian Zhang. Instancegaussian: Appearance-semantic joint gaussian representation for 3d instance-level perception. In *IEEE/CVF Computer Vision and Pattern Recognition Conference*, 2025. 1, 2

[6] Kunhao Liu, Fangneng Zhan, Jiahui Zhang, Muyu Xu, Yingchen Yu, Abdulmotaleb El Saddik, Christian Theobalt, Eric Xing, and Shijian Lu. Weakly supervised 3d open-vocabulary segmentation. *Advances in Neural Information Processing Systems*, 2023. 1

[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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1

[8] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3d language gaussian splatting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2024. 1

[9] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, 2021. 1

[10] 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*, 2024. 1, 2

[11] Yu-Ting Zhan, Cheng-Yuan Ho, Hebi Yang, Yi-Hsin Chen, Jui Chiu Chiang, Yu-Lun Liu, and Wen-Hsiao Peng. Cat-3dgs: A context-adaptive triplane approach to rate-distortion-optimized 3dgs compression. In *International Conference on Learning Representations*, 2025. 1