CCL-LGS: Contrastive Codebook Learning for 3D Language Gaussian Splatting

Supplementary Material

1. Qualitative Results on the Room Scene

As mentioned in the main paper, the Room scene in the 3D-OVS dataset includes annotation errors specifically in the "wood wall" region. This mislabeling affects the reliability of quantitative evaluation. Therefore, we present qualitative results in Fig. 1.

2. Efficiency Analysis

We conducted additional experiments on the Ramen scene from the LERF dataset using an Intel i7-14700KF CPU and an NVIDIA RTX 4090 GPU. The results are summarized in Tab. 1. The rendering speed (FPS) is measured by rendering the images with language features at a consistent resolution. Results show that our method achieves a favorable balance between multi-scale segmentation accuracy and efficiency under constrained GPU memory conditions.

3. Scale Number Analysis

To explore the impact of scale granularity on semantic segmentation accuracy, we compare our two-scale design with a baseline that merges all masks from three scales (subparts, parts, and whole objects) into a single set, as shown in Tab. 2. This single-scale baseline simplifies processing but sacrifices scale-specific representation.

Although using all three separate scales may provide marginal performance gains, we found that it leads to prohibitive GPU memory usage, system memory consumption, and preprocessing time due to the overwhelming number of fine-grained masks generated at the subpart level. Given these limitations, we do not include full three-scale experiments.

Our method merges subpart with part masks and whole with part masks, producing two non-overlapping, semantically meaningful sets. This strategy reduces redundancy and computational overhead while preserving scale-aware distinctions.

4. Language-based 3D Interaction Capability

Although our main paper emphasizes 2D supervision for semantic learning, our method indeed supports direct 3D interaction via language queries. Our codebook-based approach allows users to perform language-based querying and editing directly in 3D space.

Specifically, given a language query, we first match the query embedding with the codebook to select relevant semantic categories. Then, without any rasterization or alpha blending, we directly classify the semantic features of each 3D Gaussian using a lightweight linear classifier. Based on the predicted category distribution, we can identify Gaussians that correspond to the selected categories. This enables us to identify Gaussians in the 3D space that are semantically aligned with the input query. Once the relevant Gaussians are localized, we can directly perform various interaction operations on them, thereby enabling intuitive and interpretable 3D editing driven by natural language. Examples of such 3D interactions are illustrated in Fig. 2.

Method	mIoU	Pre-process	Training	Total	FPS	Memory
LangSplat	51.2	256min	36min	292min	42	7GB+4GB
LEGaussians	46.0	2min	40min	42min	65	19GB+9GB
Ours	62.3	16min	74min	90min	65	11GB+9GB

Table 1. Efficiency comparison on the Ramen scene from the LERF dataset.

Method	Ramen	Figurines	Teatime	Waldo Kitchen	Avg.
single scale	44.3	57.6	70.1	57.7	57.4
ours	62.3	61.2	71.8	67.1	65.6

Table 2. Comparison of different scale aggregation strategies.

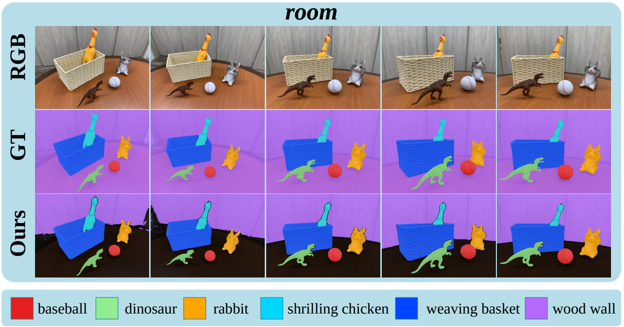

Figure 1. Qualitative results on the Room scene. The "wood wall" category contains obvious annotation errors.

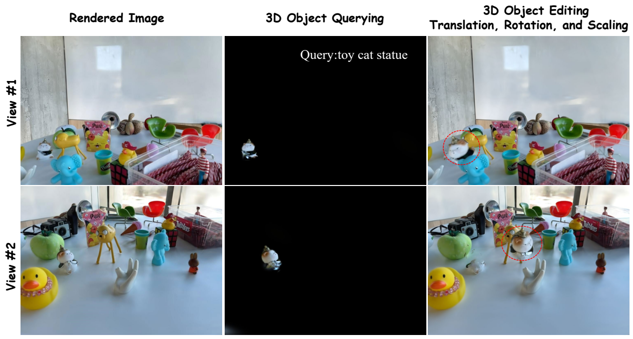

Figure 2. Examples of language-based 3D interaction and editing enabled by our method.

5. More Results

Beyond the specific cases discussed, we provide more qualitative visualizations of our method's semantic segmentation results across different scenes in Figs. 3, 4, and 5.

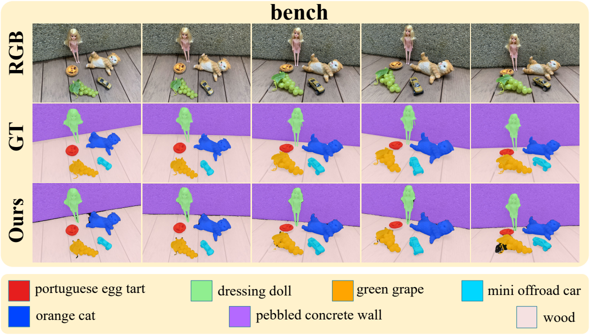

Figure 3. Qualitative semantic segmentation results on the Bench scene.

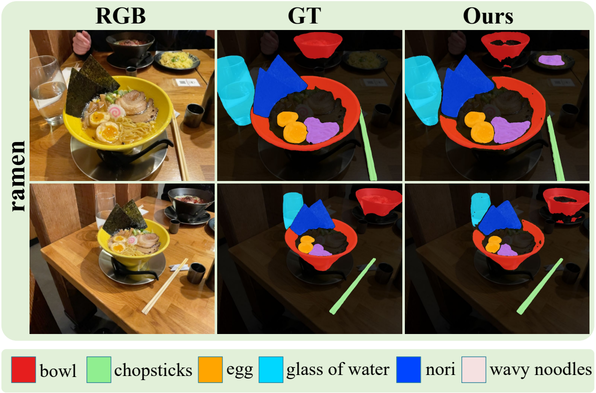

Figure 4. Qualitative semantic segmentation results on the Ramen scene.

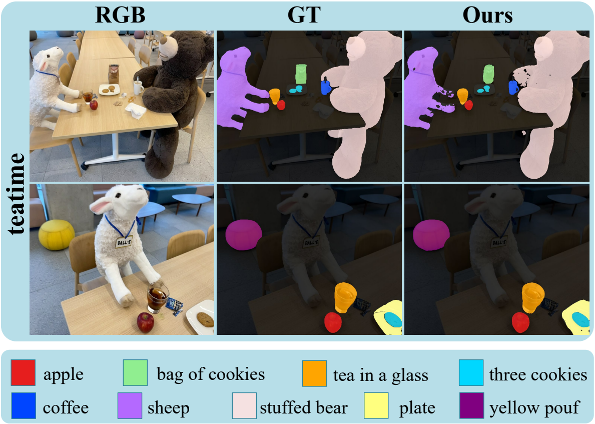

Figure 5. Qualitative semantic segmentation results on the Teatime scene.