

SPC-GS: Gaussian Splatting with Semantic-Prompt Consistency for Indoor Open-World Free-view Synthesis from Sparse Inputs

Supplementary Material

In this appendix, we present additional materials to support and extend the findings and observations presented in the main body of this paper.

- Section A presents more details of the Scene-layout-based Gaussian Initialization (SGI) strategy.
- Section B elaborates on additional experimental details of our approach.
- Section C offers extended experimental analyses to showcase the effectiveness of our method.
- Section D presents more qualitative results to facilitate better visual comparisons.
- Section E provides per-scene quantitative evaluation results for more comprehensive comparisons.

A. More Details of Scene-layout-based Gaussian Initialization (SGI)

Our proposed SGI strategy includes two main components: *VGM-based Point Creation* and *Scene-layout Point Generation*. *i)* The first component aims to produce denser SfM points by leveraging additional view-changed images that are generated from the original sparse training views. *ii)* The second component yields an instructive scene-layout point distribution for enhanced Gaussian initialization.

Specifically, in the VGM-based Point Creation process, eight neighboring views $\{\tilde{I}_i^j\}_{j=1}^8$ are generated for each original training image I_i using the image-to-video generation mode of the advanced video generation model MotionCtrl [47]. Fig. 8 shows these additional view-changed images (denoted by the blue border) alongside the original training view (indicated by the red border). Subsequently, the generated images are combined with the original training images for SfM processing, yielding denser initialized SfM points. As shown in Table 5, we can see that sparse-input training images alone yield limited SfM points. Augmenting the training images with view-changed images increases the amount of SfM points. Furthermore, the Scene-layout Point Generation further produces a scene-layout-wise Gaussian point distribution, serving as an enhanced and instructive initialization prior. Moreover, these results also can be found in Fig. 9.

In general, these results demonstrate that our full SGI framework effectively provides dense and instructive points for Gaussian initialization, which promotes scene Gaussian representation, and consequently, enhances semantic Gaussian learning in sparse-input scenarios.

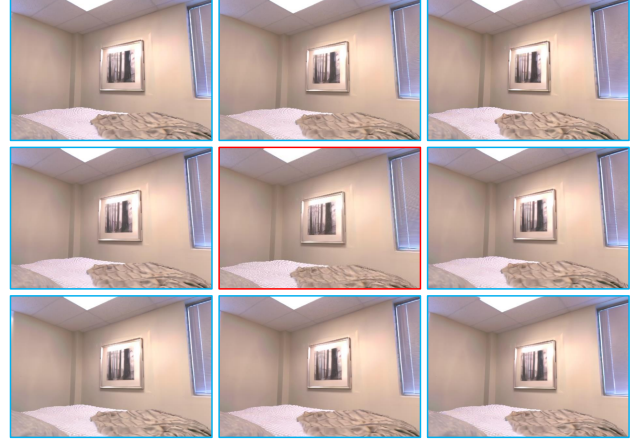


Figure 8. Illustration of training image (red border in the center) and corresponding generated images (blue border).

Scene	SfM Points	SfM Points with Generated Views	Scene Layout Points from SGI
Room0	122	6527	57524
Room1	52	5628	58334
Room2	22	2659	54308
Office0	474	5452	41989
Office2	140	2864	44836
Office4	110	3621	29115
Scene0004	70	4027	64016
Scene0389	19	1207	22901
Scene0494	6	3125	37195
Scene0693	48	2911	44060

Table 5. Number of Gaussian points in various settings. “SfM Points” refers to the initialized Gaussian points derived from the Structure-from-Motion (SfM) algorithm using sparse training images. “SfM Points with Generated Views” indicates the initialized SfM Gaussian points derived from the SfM algorithm using training and generated images. “Scene Layout Points from SGI” denotes the Gaussian points obtained from the Scene-layout Gaussian Initialization (SGI) strategy, which are treated as initialized Gaussian points to optimize the Gaussian radiance field.

B. Elaborated Experimental Details

B.1. Datasets

To evaluate the performance of our approach, we conduct sparse-input open-world free-view synthesis experiments on two widely-used benchmark indoor scene datasets: Replica [42] and ScanNet [8].

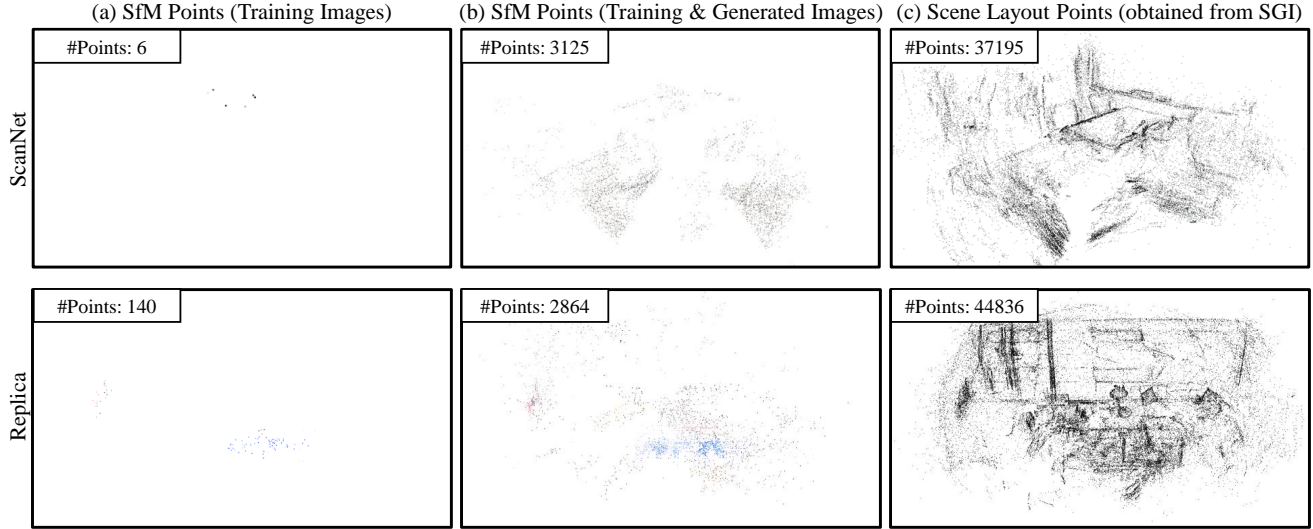


Figure 9. Visualization of Gaussian point distributions of different scenes: (a) Sparsely distributed SfM points from limited training views; (b) Denser SfM points distribution using generated images and training images; (c) Instructive point distribution obtained via our Scene-layout-based Gaussian Initialization (SGI) strategy, demonstrating improved scene layout coverage that enhances reconstruction quality and consequently improves segmentation accuracy.

Replica is a synthetic scene dataset comprising diverse, high-quality indoor room-scale environments. Each scene features photo-realistic textures, dense geometry, and semantic classes. For experiment evaluations, six commonly evaluated scenes from Replica are utilized: room0, room1, room2, office0, office2, and office4. Following 3DOVS [31], 45 categories are used for text queries: candle, book, vent, box, comforter, switch, bin, plant stand, bed, desk organizer, rug, bench, vase, bottle, ceiling, blanket, bowl, camera, wall, blinds, pillar, sculpture, tablet, chair, lamp, indoor plant, cabinet, stool, table, cushion, panel, plate, basket, pot, tissue paper, nightstand, sofa, window, picture, wall plug, tv screen, shelf, door, floor, clock.

ScanNet is a real-world indoor scene dataset that includes semantic segmentation labels and camera poses provided by BundleFusion [9]. For evaluation, four scenes are selected from ScanNet: scene0004_00, scene0389_00, scene0494_00, and scene0693_00. The commonly used 20 categories defined by ScanNet are used for text queries: wall, floor, cabinet, bed, chair, sofa, table, door, window, bookshelf, picture, counter, desk, curtain, refrigerator, shower curtain, toilet, sink, bathtub, other furniture.

Following the sparse-input experimental protocol outlined in [25, 60], we select every 10-*th* image in the sequence from each scene as the testing view, resulting in 22 to 30 testing images per scene for evaluation. From the remaining images, we uniformly sample 12 views to construct the sparse training set. The resolution of all images is set to 640×448 . During training, RGB images from the training set are utilized for scene reconstruction, with CLIP-derived

semantic features applied for semantic Gaussian learning. For evaluation, only the RGB images and ground-truth semantic labels from the testing set are utilized.

B.2. Implementation Details

Data Prereprocessing. Before training, CLIP features are pre-computed offline, following prior methods [31, 59]. The generated images of each training view are obtained using MotionCtrl [47] under the image-to-video mode.

Training. We set the learning rate as 0.0025 for Gaussian semantic parameters, while convolution layers ω_f and ω_s are optimized using Adam with the learning rate of 0.0005. The Outlier Gaussian Primitive Removal (OGR) strategy is implemented every 3k iterations. We first train our model to derive a scene-layout Gaussian point distribution through 10k iterations. These points are then used to initialize Gaussian positions, color attributes (using zero-order spherical harmonics), and semantic attributes, which undergo training for 10k iterations. The entire process costs 45 minutes on average per scene on one A100 GPU.

Inference. During inference, we project 3D Gaussians onto the 2D plane, concurrently producing rendered RGB images and rendered semantic features in novel views. Following previous 3D open-world segmentation methods [31, 35], a set of text queries are utilized to calculate the cosine similarity between these text features and the rendered features, generating the open-vocabulary segmentation results. Our approach achieves a rendering speed of over 300 FPS at a resolution of 640×448 .

Method	Replica [42]					ScanNet [8]				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	13.326	0.520	0.656	8.859	41.909	18.836	0.652	0.535	24.001	72.863
Feature 3DGS [59]	18.154	0.708	0.339	16.357	64.955	19.702	0.678	0.403	21.053	72.722
Gau-Grouping [50]	17.787	0.709	0.350	20.252	66.519	19.189	0.682	0.419	33.549	73.624
DNGaussian [25]	19.964	0.749	0.370	23.692	70.705	20.231	0.706	0.451	32.122	74.895
FSGS [60]	20.371	0.768	0.285	23.721	71.235	21.875	0.730	0.386	34.610	77.840
CoR-GS [56]	20.066	0.779	0.290	22.496	68.901	21.801	0.735	0.388	34.170	77.944
Ours	22.123	0.800	0.248	29.173	75.482	23.042	0.755	0.359	50.271	83.584

Table 6. Quantitative comparison of reconstruction and segmentation results on novel views in Replica and ScanNet datasets, using the CLIP-LSeg [24] to optimize Gaussian semantic attributes with 12 training views. Our approach achieves **superior results** across all metrics on various datasets.

C. Additional Analyses

In this section, we present additional experimental results to further validate the robustness of our proposed method. Specifically, we conduct comparative analyses using various vision-language foundation models, such as CLIP-LSeg [24] and APE [40], which have been adopted in prior works [36, 59] for optimizing Gaussian semantic attributes. Furthermore, we include a comparative study with ViewCrafter [53]. Additionally, we provide more comprehensive ablation studies to evaluate the effectiveness of different components within our framework.

C.1. Results using CLIP-LSeg Model

To assess the effectiveness and generalizability of our approach, we utilize CLIP-LSeg [24] for semantic Gaussian optimization and report the quantitative results in Table 6.

i) Comparison with 3D Open-vocabulary Segmentation Methods. As shown in Table 6, we see that our method consistently surpasses competitors across all metrics on various datasets when using CLIP-LSeg [24] to optimize Gaussian semantic attributes, while other approaches encounter significant challenges under the sparse input condition.

Specifically, the state-of-the-art method Gau-Grouping [50] demonstrates relatively low reconstruction quality and limited performance in open-world segmentation. This is because it uses sparse SfM points for Gaussian initialization, which hampers its ability to represent complex indoor scenes, resulting in inferior reconstruction quality and impaired segmentation precision. Additionally, Gau-Grouping only applies supervision to Gaussians within sparse training views, leading to the under-optimization problem. As a result, this method tends to overfit the training views while producing less accurate results for novel viewpoints. In comparison, our method utilizes dense scene-layout points and semantic-prompt consistency constraints, yielding improvements of 3.97 PSNR and 8.92% mIoU over the second-best method on the Replica benchmark. These results highlight the effectiveness of our approach for open-world free-view synthesis with sparse inputs.

ii) Comparison with Sparse-input Free-view Synthesis Methods. In Table 6, although methods like DNGaussian, FSGS, and CoR-GS achieve improved novel view quality by incorporating additional depth or color regularizations to optimize sparse-view Gaussian radiance fields, our approach consistently surpasses them in both reconstruction quality and semantic understanding accuracy. These improvements can be attributed to the effectiveness of our SGI strategy in enhancing Gaussian representation and the SPC regularization in boosting segmentation accuracy.

In general, these quantitative results validate the effectiveness and generalizability of our approach in 3D indoor open-world free-view synthesis from sparse input images, optimized across different CLIP models. Moreover, the qualitative results using CLIP-LSeg to optimize Gaussian semantic attributes. can be found in Section D and Fig. 17.

C.2. Results using APE Model

Following GOI [36], we utilize the same Aligning and Prompting Everything All at Once (APE) model [40] to extract 2D semantic features from training views, which are treated as the 2D ground truth semantic features for optimizing Gaussian semantic attributes. Since GOI only supports single-query segmentation results using a vocabulary, rather than generating a relevancy map at a time as outlined in its paper [36], we adopt this single-query segmentation setting to produce sparse-input open-world free-view synthesis results. In line with GOI’s implementations, we employ the sparse-input training images for optimization, and then evaluate the single-query performance in novel views for comparisons.

The single-query segmentation results on ScanNet and Replica test data are illustrated in Fig. 10. It can be seen that GOI generates vague reconstruction results in novel viewpoints. This can be attributed to its limited Gaussian representation stemming from sparse Gaussian point initialization. Consequently, inheriting the bottleneck of inferior Gaussian representation, GOI easily faces challenges and produces incomplete and noisy segmentation results under sparse input conditions.

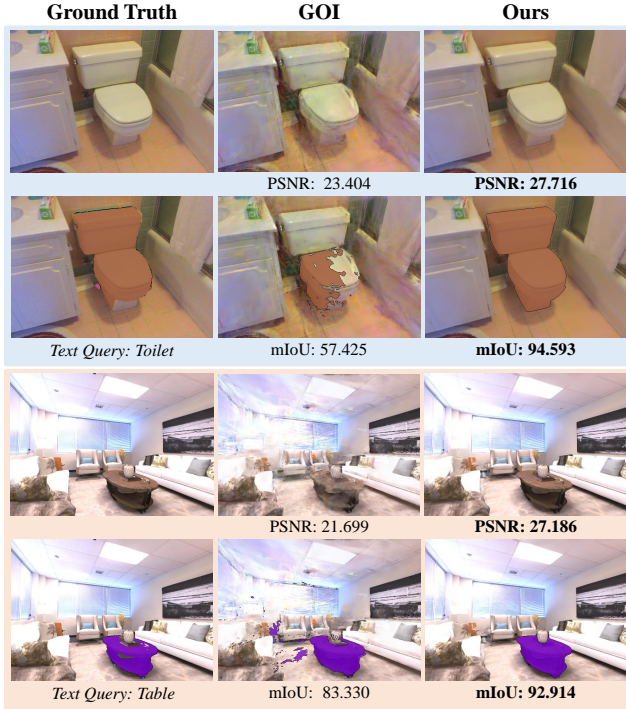


Figure 10. Comparison of reconstruction (rows 1, 3) and open-vocabulary segmentation (rows 2, 4) results on novel views across diverse scenes from the ScanNet and Replica datasets using APE [40] for optimizing Gaussian semantic attributes. Compared to GOI, our approach demonstrates photo-realistic appearance details and more complete segmentation results.

Method	Replica [42]			ScanNet [8]		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
ViewCrafter [53]	19.207	0.762	0.325	17.262	0.697	0.468
Ours	22.011	0.792	0.254	22.401	0.741	0.368

Table 7. Quantitative results of reconstruction on novel views.

C.3. Comparison with ViewCrafter

ViewCrafter [53] employs a pre-trained video generation model to synthesize additional views, aiming to enhance sparse-view 3DGS optimization. However, it inherits the *color shift* artifacts commonly associated with diffusion models, often generating reasonable structures with inaccurate color representations, as shown in the left part of Fig. 11. As a result, its reconstruction performance is inferior, as shown in Table 7 and Fig. 11.

C.4. More Ablation Studies

In this section, we present more ablation analyses of our approaches for a more comprehensive analysis.

Effectiveness of OGR. To assess the effectiveness of the Outlier Gaussian Primitive Removal (OGR), we present ablation results in Fig. 12 and #b of Table 8. As depicted in

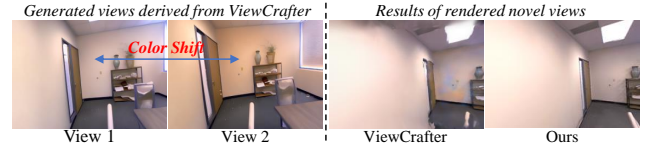


Figure 11. Visual results. Compared to ViewCrafter, our approach exhibits more photo-realistic appearance details.

Case	Configuration	Replica [42]				
		PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
#a	Ours (Full Setting)	22.011	0.792	0.254	23.960	63.262
#b	w/o OGR	21.889	0.790	0.258	22.501	62.085
#c	w \mathcal{L}_{GR}	20.296	0.761	0.304	19.864	57.812
#d	w 2D CLIP for \mathcal{L}_{inter}	21.906	0.790	0.259	23.139	62.596

Table 8. Further ablation results of our approach with various settings.

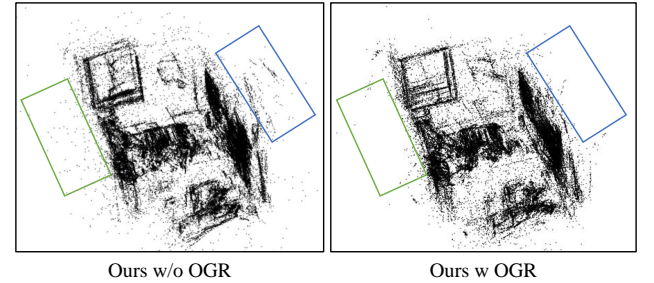


Figure 12. Comparison of Gaussian point clouds with and without Outlier Gaussian Primitive Removal (OGR) strategy. Applying the OGR can reduce outlier Gaussian primitives, facilitating the optimization of Gaussian representations and enhancing results.

Fig. 12, the omission of OGR results in an increased presence of outlier Gaussian primitives. These excessive outlier primitives, when used for subsequent Gaussian initialization, complicate the optimization stage, leading to performance degradation as shown in configuration #b of Table 8. These findings demonstrate the efficacy of the OGR strategy in mitigating outlier proliferation, preserving Gaussian representation quality, and enhancing rendering results.

Impact of Generated-view Color Supervision. To investigate the impact of color supervision \mathcal{L}_{GR} provided by generated images, we present the ablation results in #c of Table 8 and Fig. 13. It can be observed that the addition of color supervision \mathcal{L}_{GR} leads to performance degradation. This occurs because the reconstruction task is highly sensitive to erroneous color pixels in generated images, as these inaccuracies can significantly degrade reconstruction quality, with even minor color inconsistencies affecting reconstruction quality, as illustrated in Fig. 13.

Analysis of the Pseudo Supervision Signals. To analyze the effectiveness of using self-rendered semantic representations from training views for supervising pseudo views in

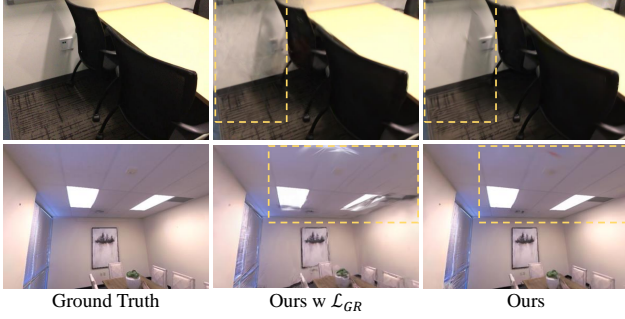


Figure 13. Visual ablation results for using generated-view color supervision \mathcal{L}_{GR} . Incorporating color supervision \mathcal{L}_{GR} (“Ours w \mathcal{L}_{GR} ”) provided from generated RGB images produces inferior results compared to without using \mathcal{L}_{GR} (“Ours”).

Case	Configuration	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
#a	Baseline	17.134	0.678	0.369	14.899	51.255
#b	#a + SGI*	20.914	0.782	0.284	18.865	57.032
#c	#b + SPC	21.806	0.792	0.254	23.202	62.057

Table 9. Ablation results of our SGI strategy incorporating only vanilla reconstruction loss \mathcal{L}_C and semantic loss \mathcal{L}_S (Eq. (3)).

\mathcal{L}_{inter} (i.e. Eq. (5)), we replace them with the 2D CLIP-derived semantics from training views. As shown in #d of Table 8, this replacement leads to degraded performance. This demonstrates the advantage of self-rendered semantic representations derived from the 3D radiance fields exhibit superior coherent and reliable information over the 2D representations, enhancing overall optimization.

Analysis of the View Constraint in SGI. We examine the effect of view constraints on 3D Gaussian densification within the SGI by utilizing only two components: the vanilla reconstruction loss \mathcal{L}_C and semantic loss \mathcal{L}_S (Eq. (3)), denoted as SGI* in Table 9. The experimental results demonstrate that our SGI strategy also achieves significant improvements, while the additional application of SPC further enhances performance. These findings indicate that our SGI can generate effective scene-layout Gaussian distributions without relying on specific view constraints.

Analysis of the Hyperparameter for Region Boundary Erosion. Table 10 presents an experimental analysis of the erosion hyperparameter used for Region Boundary Erosion. Specifically, the configuration 3×3 indicates that a pixel $M'(x, y)$ is retained as “True” only if all pixels within its 3×3 neighborhood in the input original mask M are “True”; otherwise, it is assigned “False”. This operation effectively contracts region boundaries inward.

The experimental results demonstrate that increasing the size of the erosion hyperparameter, i.e. using a larger kernel, helps reduce the number of ambiguous Gaussian primitives near object boundaries, thereby improving perfor-

Case	Configuration	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
#a	1×1	21.876	0.792	0.255	23.270	61.409
#b	3×3	21.901	0.791	0.255	23.474	62.485
#c	5×5	22.011	0.792	0.254	23.960	63.262
#d	7×7	21.920	0.792	0.255	23.388	62.314
#e	9×9	21.950	0.792	0.255	23.182	62.349

Table 10. Analysis of the erosion hyperparameter for Region Boundary Erosion.

Point Prompts	#Train	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
Number=1 (Base)	~45mins	22.011	0.792	0.254	23.960	63.262
Number=2	~50mins	22.044	0.792	0.254	24.089	63.365
Number=3	~55mins	22.034	0.792	0.254	24.364	63.440

Table 11. Performance for the Iterative Stochastic Prompting (ISP) approach with different settings.

mance. However, excessive erosion diminishes the number of available Gaussians for optimization, potentially limiting performance. Based on these empirical findings, we adopt the 5×5 configuration as the final parameter choice.

Analysis of the Number of Point Prompts in ISP. In Table 11, we evaluate the performance across varying quantities of point prompts used to establish region mask correspondences. The experimental results demonstrate that increasing point prompt quantities yields performance comparable to the baseline configuration. This finding suggests that the *iterative stochastic design* in the base ISP effectively generates a uniform distribution of points across the entire image space as the training processes. This enables efficient construction of region mask correspondences across diverse image regions. Overall, given the additional training time required for incorporating more point prompts and their marginal performance gains, we opt not to include further point sampling in the final framework.

Analysis of Lower-Order Spherical Harmonics (SH). As shown in Table 12, employing lower-order SH under generated view constraints (with \mathcal{L}_{GR}) partially alleviates the performance drop in evaluation metrics. However, a noticeable degradation still occurs, primarily due to color inaccuracies in the generated images.

Analysis of Varying Numbers of Training Views. As illustrated in Fig. 14, our method demonstrates strong robustness when using varying numbers of training views in Room0, ranging from very sparse (fewer than 10 views, where COLMAP fails) to dense view configurations. The superior segmentation performance is attributed to the proposed semantic-prompt consistency regularization strategy.

C.5. Limitation Analysis and Future Work

While our SPC-GS framework demonstrates notable advantages in sparse-input open-world free-view synthesis, it is currently limited to static 3D scenes, as it does not incorporate dynamic Gaussian modeling or time-dependent opti-

Loss	SH Setting	Room0 Scene				
		PSNR	SSIM	LPIPS	mIoU	mAcc
wo \mathcal{L}_{GR}	Ours (SH=3)	20.872	0.704	0.322	16.330	52.486
w \mathcal{L}_{GR}	SH=0	19.040	0.674	0.372	15.066	45.537
	SH=1	18.791	0.669	0.376	13.953	45.471
	SH=2	18.780	0.668	0.378	13.664	44.995
	SH=3	18.766	0.664	0.380	12.358	44.700

Table 12. Results of different spherical harmonics parameters.

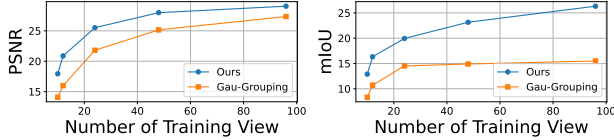


Figure 14. Results using 10, 12, 24, 48, and 96 training views.

mization mechanisms. Future research directions could explore the extension of our method to dynamic open-world free-view synthesis, enabling temporal scene modeling.

D. More Visualization Results

We provide additional qualitative results in Fig. 15, 16 and 17 to better illustrate the effectiveness of our method.

i) *Reconstruction*. As shown in Fig. 15, the results produced by 3DOVS and Gau-Grouping exhibit significant artifacts due to the limited radiance field representation quality when relying solely on sparse-input training data. Although FSGS and CoR-GS introduce additional geometric constraints to enhance Gaussian radiance fields under sparse input conditions, they struggle to obtain photo-realistic details. In contrast, our approach effectively reduces artifacts and ambiguity, presenting a more robust global structure with finer details, especially in rendered views distant from the training viewpoints (e.g. 3rd and 5th rows). These improvements can be attributed to our SGI and SPC strategies, which contribute to an enhanced Gaussian distribution and enforce effective view-consistency supervision, thereby improving the overall quality of Gaussian representation.

ii) *Open-world Segmentation*. As shown in Fig. 16, when using the CLIP model [37] for optimizing Gaussian semantic attributes, existing methods struggle to achieve precise object boundaries and maintain object integrity. Specifically, 3D open-vocabulary segmentation methods, such as LangSplat and Gau-Grouping, are hindered by the inferior Gaussian point distributions, which consequently impede the semantic Gaussian representation and easily lead to noisy rendered semantic results (e.g. *Chair* in the 1st row). Notably, LangSplat exhibits more pronounced noise (1st ~ 2nd rows) and even inaccurate semantic renderings (3rd ~ 4th rows) in sparse-input scenarios. This stems from a key factor: during semantic parameter opti-

mization, LangSplat inherits Gaussian attributes (*i.e.* position, scaling, and rotation) derived from sparse-input scene reconstruction using the vanilla 3DGS, and only optimizes the semantic parameter to obtain semantic Gaussian representation. By fixing the Gaussian primitives’ attributes, LangSplat prevents flexible adjustment of semantic representations, thereby inheriting the inherent bottlenecks of sparse-input scene reconstruction and significantly compromising segmentation performance.

Moreover, while sparse-input free-view synthesis methods, such as DNGaussian and CoR-GS, introduce geometric constraints to improve Gaussian representation, they still easily face challenges with semantic ambiguity due to insufficient semantic consistency supervision.

In contrast, our method delivers robust and accurate segmentation results, benefiting from the enhanced Gaussian distribution and effective semantic-prompt consistency supervision. Additional visualizations in Fig. 17 further demonstrate that, when optimized with the CLIP-LSeg model [24], our approach consistently outperforms others across diverse scenes, underscoring its robustness in 3D open-vocabulary semantic understanding.

In summary, our approach simultaneously delivers photorealistic rendering quality and superior segmentation performance across diverse scenes. This comprehensive evaluation highlights the effectiveness of our approach for sparse-input open-world free-view synthesis.

E. Per-scene Qualitative Results

In Tables 13 and 14, we present per-scene quantitative comparisons of reconstruction and segmentation results on novel views, utilizing CLIP and CLIP-LSeg for optimizing Gaussian semantic attributes, respectively. Overall, our approach consistently surpasses other state-of-the-art methods in terms of reconstruction and segmentation on synthetic and real-world scenes. These results highlight the effectiveness of our method in indoor open-world free-view synthesis using sparse-input data.

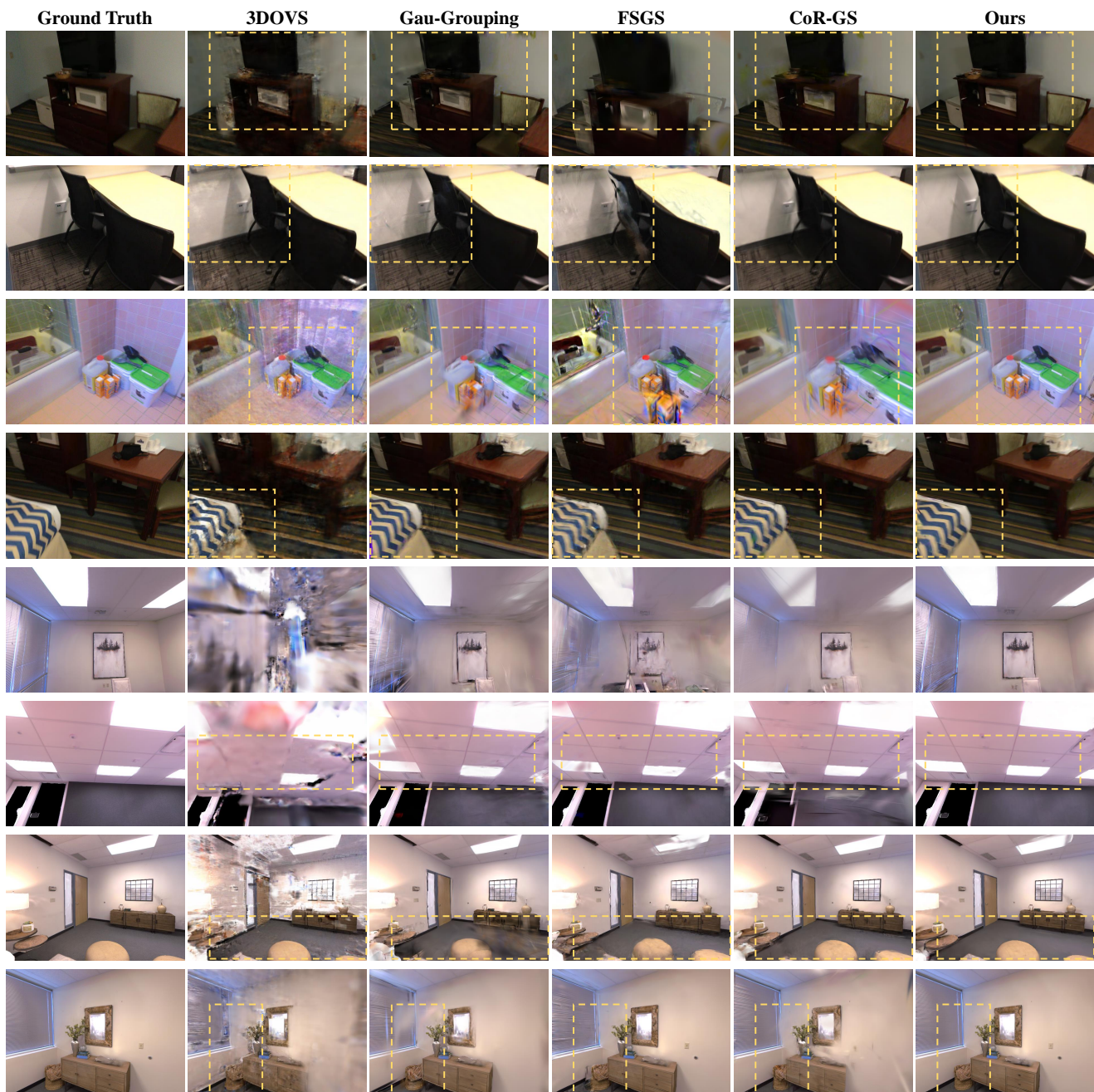


Figure 15. Visual reconstruction results on novel views from the ScanNet dataset ($1^{st} \sim 4^{th}$ Rows) and the Replica dataset ($5^{th} \sim 8^{th}$ Rows), using 12 input views for training. Our approach achieves superior global structure and photo-realistic details, attributed to our enhanced Gaussian representation and effective view-consistency constraints. More detailed analyses refer to Section D.

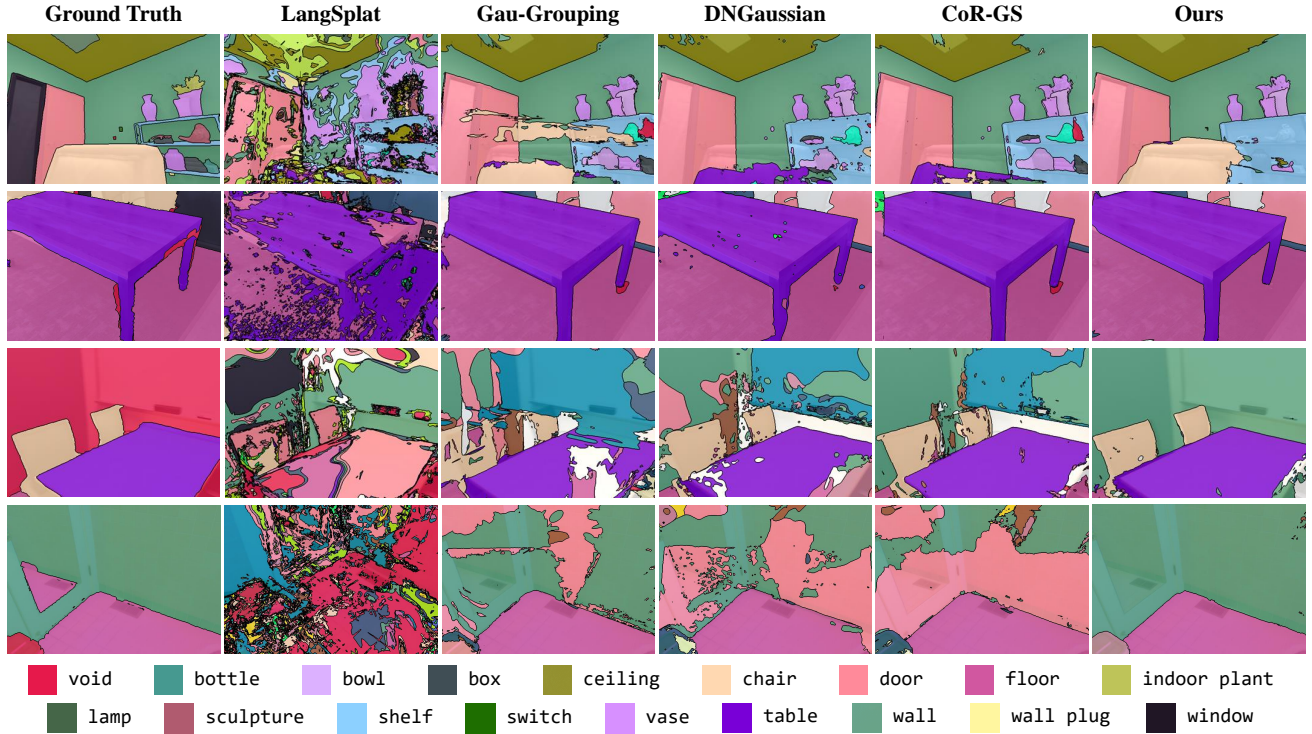


Figure 16. Visual open-world segmentation results on novel views from the Replica dataset (1st Row) and the ScanNet dataset (3rd ~ 4th Rows) when using the CLIP [37] to optimize Gaussian semantic attributes, with 12 training views. Our method produces more accurate and complete results thanks to the enhanced Gaussian points distribution and semantic consistency constraints.

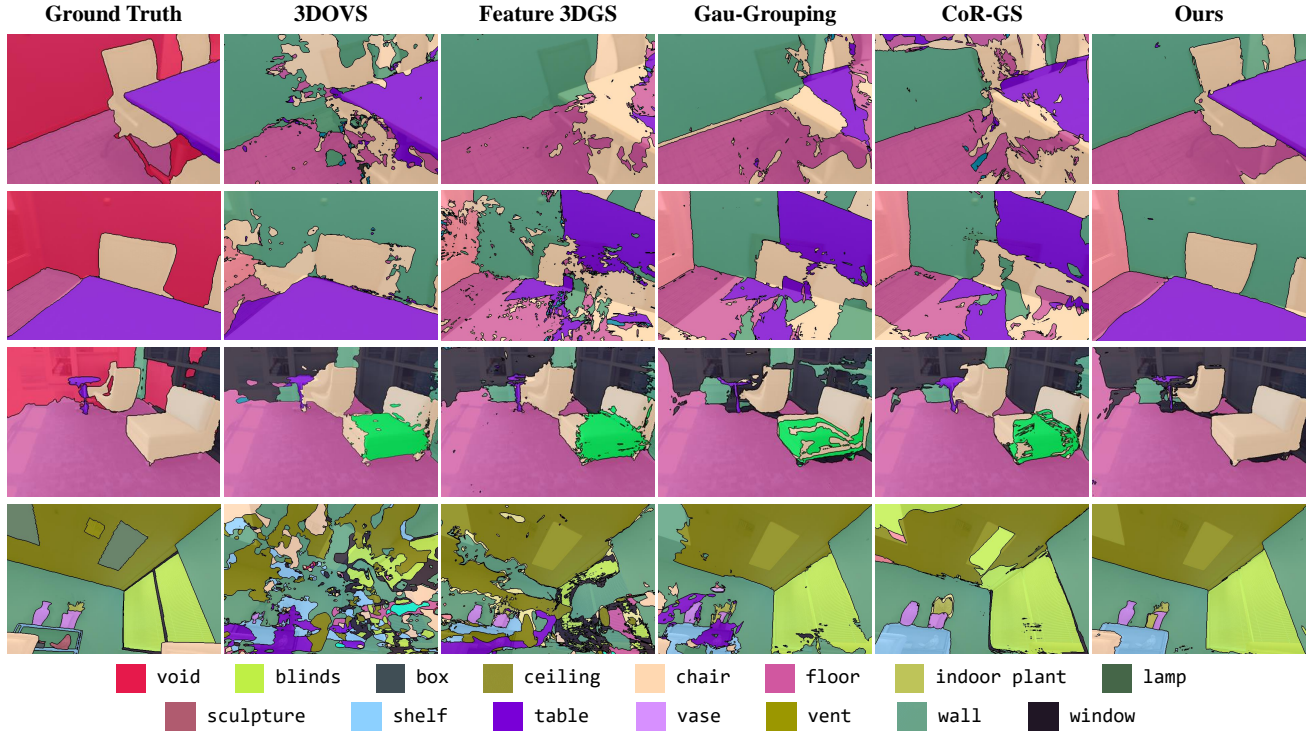


Figure 17. Visual open-world segmentation results on novel views from the ScanNet dataset (1st ~ 3rd Rows) and the Replica dataset (4th Row) when using the CLIP-LSeg model [24] for optimizing Gaussian semantic attributes, with 12 training views. Our approach consistently demonstrates superior precision.

Method	Room0					Room1				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	16.198	0.554	0.467	5.547	32.699	14.643	0.496	0.674	3.166	18.421
Feature 3DGS [59]	15.991	0.595	0.425	6.402	32.734	17.485	0.674	0.373	5.959	22.927
LangSplat [35]	16.481	0.580	0.407	2.924	20.102	17.116	0.639	0.393	2.167	17.038
Gau-Grouping [50]	15.978	0.599	0.436	10.712	40.774	16.400	0.639	0.410	12.709	46.494
DNGaussian [25]	16.005	0.601	0.492	11.274	42.377	19.353	0.691	0.406	16.883	53.758
FSGS [60]	18.437	0.650	0.365	12.695	44.934	18.516	0.671	0.356	16.065	56.217
CoR-GS [56]	18.813	0.688	0.357	12.747	44.289	19.526	0.725	0.327	16.158	51.927
Ours	20.872	0.704	0.322	16.330	52.486	22.363	0.750	0.264	29.022	74.166

Method	Room2					Office0				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	13.536	0.567	0.669	4.633	25.459	14.317	0.485	0.680	2.156	9.580
Feature 3DGS [59]	15.484	0.705	0.384	7.139	29.271	20.719	0.768	0.279	6.430	21.632
LangSplat [35]	18.092	0.740	0.314	4.277	27.811	21.543	0.774	0.277	1.852	11.097
Gau-Grouping [50]	19.359	0.756	0.313	17.657	69.003	20.128	0.738	0.334	11.485	29.045
DNGaussian [25]	20.626	0.767	0.353	17.229	70.091	20.552	0.759	0.362	13.055	32.788
FSGS [60]	18.295	0.740	0.343	15.562	65.899	21.762	0.789	0.262	13.548	31.720
CoR-GS [56]	18.648	0.783	0.303	15.270	64.398	22.056	0.804	0.251	13.020	31.989
Ours	23.267	0.835	0.209	21.112	73.906	23.382	0.821	0.246	16.488	34.438

Method	Office2					Office4				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	9.716	0.496	0.760	3.228	18.629	10.789	0.490	0.726	4.748	23.844
Feature 3DGS [59]	16.758	0.752	0.338	8.096	34.719	15.915	0.708	0.352	10.416	39.970
LangSplat [35]	18.237	0.779	0.779	3.802	12.832	14.821	0.694	0.348	4.729	22.844
Gau-Grouping [50]	16.133	0.716	0.381	19.108	67.423	15.784	0.695	0.380	18.500	49.792
DNGaussian [25]	19.100	0.765	0.369	21.779	76.004	15.438	0.686	0.429	19.723	51.380
FSGS [60]	18.261	0.777	0.310	19.737	71.854	16.893	0.724	0.334	18.998	50.224
CoR-GS [56]	17.313	0.781	0.325	18.187	70.760	17.374	0.768	0.309	20.220	51.490
Ours	23.095	0.858	0.204	35.109	87.375	19.087	0.785	0.275	25.697	57.201

Method	scene0004					scene0389				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	17.894	0.615	0.496	10.406	51.940	21.550	0.731	0.525	7.919	45.842
Feature 3DGS [59]	16.489	0.600	0.434	8.447	51.692	21.061	0.741	0.383	10.063	46.725
LangSplat [35]	16.656	0.605	0.427	5.499	43.347	23.720	0.791	0.352	4.545	34.339
Gau-Grouping [50]	17.837	0.634	0.424	10.416	54.014	23.241	0.785	0.366	24.414	78.311
DNGaussian [25]	17.810	0.670	0.506	9.460	53.999	20.818	0.739	0.442	20.590	70.258
FSGS [60]	19.011	0.645	0.419	9.934	55.340	24.681	0.796	0.354	21.265	73.597
CoR-GS [56]	18.961	0.662	0.407	10.582	54.445	25.098	0.812	0.338	23.462	75.366
Ours	19.131	0.677	0.404	13.411	64.718	27.232	0.838	0.329	48.243	82.017

Method	scene0494					scene0693				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	16.446	0.663	0.577	6.162	31.342	18.996	0.561	0.580	6.516	16.300
Feature 3DGS [59]	14.304	0.649	0.445	5.571	31.384	20.497	0.635	0.436	5.309	22.209
LangSplat [35]	15.235	0.646	0.646	2.409	15.807	19.511	0.610	0.479	0.903	4.469
Gau-Grouping [50]	15.311	0.648	0.447	10.529	46.358	19.633	0.635	0.446	15.330	50.737
DNGaussian [25]	15.820	0.699	0.473	12.564	50.876	19.120	0.656	0.516	13.036	44.122
FSGS [60]	16.511	0.659	0.433	11.857	53.064	22.094	0.682	0.417	13.069	45.399
CoR-GS [56]	16.303	0.706	0.416	11.175	49.935	22.897	0.703	0.410	12.628	43.501
Ours	18.244	0.725	0.386	26.651	67.365	24.998	0.723	0.354	27.472	61.194

Table 13. Quantitative results of reconstruction and segmentation on novel views across various scenes from Replica and ScanNet datasets, using the CLIP [37] for optimizing Gaussian semantic attributes with 12 training views. Our approach achieves **superior performances** in both reconstruction quality and segmentation accuracy.

Method	Room0					Room1				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	16.277	0.557	0.464	17.789	62.604	14.717	0.513	0.661	7.771	41.968
Feature 3DGS [59]	16.992	0.599	0.405	14.700	58.249	17.532	0.650	0.373	12.257	57.894
Gau-Grouping [50]	17.548	0.636	0.392	20.618	65.525	18.857	0.681	0.348	16.899	63.053
DNGaussian [25]	17.168	0.633	0.467	19.914	62.975	20.731	0.722	0.378	21.399	68.646
FSGS [60]	19.720	0.696	0.336	21.967	66.214	20.969	0.731	0.303	19.626	67.325
CoR-GS [56]	19.362	0.707	0.351	19.377	61.534	19.603	0.732	0.329	16.332	60.915
Ours	20.944	0.716	0.311	25.740	70.296	22.541	0.761	0.252	23.539	71.692

Method	Room2					Office0				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	13.540	0.570	0.660	7.886	44.913	14.580	0.483	0.677	4.347	25.848
Feature 3DGS [59]	18.338	0.742	0.342	16.536	67.860	21.738	0.771	0.285	13.247	56.463
Gau-Grouping [50]	17.910	0.737	0.341	18.196	67.612	20.530	0.751	0.313	16.006	54.395
DNGaussian [25]	20.991	0.784	0.333	24.145	76.987	21.700	0.785	0.352	18.991	59.040
FSGS [60]	20.149	0.781	0.272	24.636	76.413	22.580	0.810	0.252	19.137	59.542
CoR-GS [56]	19.893	0.805	0.265	23.611	74.854	22.715	0.808	0.253	19.845	59.545
Ours	22.879	0.838	0.210	26.995	78.006	23.134	0.819	0.249	23.166	62.666

Method	Office2					Office4				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	9.687	0.497	0.753	6.912	40.524	11.157	0.502	0.721	8.448	35.599
Feature 3DGS [59]	18.812	0.788	0.281	20.073	79.211	15.513	0.701	0.346	21.328	70.054
Gau-Grouping [50]	17.130	0.756	0.323	26.287	78.060	14.748	0.692	0.384	23.505	70.468
DNGaussian [25]	21.082	0.822	0.311	27.715	81.715	18.115	0.748	0.381	29.989	74.869
FSGS [60]	20.486	0.817	0.251	27.564	82.127	18.324	0.772	0.294	29.397	75.789
CoR-GS [56]	20.354	0.830	0.255	26.757	82.480	18.467	0.792	0.286	29.053	74.079
Ours	23.413	0.868	0.200	39.293	88.195	19.825	0.798	0.266	36.304	82.039

Method	scene0004					scene0389				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	17.829	0.616	0.498	23.079	75.464	21.882	0.739	0.515	24.401	81.412
Feature 3DGS [59]	18.038	0.618	0.406	16.215	77.017	24.682	0.806	0.337	31.594	89.220
Gau-Grouping [50]	17.975	0.631	0.411	28.301	74.673	22.931	0.779	0.365	43.316	83.081
DNGaussian [25]	17.853	0.636	0.487	25.833	73.883	25.363	0.819	0.374	39.106	89.642
FSGS [60]	19.399	0.672	0.411	28.743	78.000	26.784	0.844	0.328	39.576	89.337
CoR-GS [56]	19.331	0.673	0.412	29.863	79.074	26.573	0.828	0.333	37.277	89.190
Ours	19.631	0.681	0.403	40.120	84.554	27.512	0.847	0.320	63.541	89.646

Method	scene0494					scene0693				
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	mIoU \uparrow	mAcc \uparrow
3DOVS [31]	16.703	0.676	0.560	26.038	75.165	18.930	0.577	0.566	22.487	59.412
Feature 3DGS [59]	15.448	0.659	0.426	17.940	67.083	20.641	0.629	0.443	18.461	57.569
Gau-Grouping [50]	15.594	0.670	0.428	34.902	67.838	20.255	0.649	0.471	27.678	68.902
DNGaussian [25]	15.240	0.664	0.475	32.891	67.215	22.469	0.704	0.468	30.657	68.838
FSGS [60]	17.058	0.691	0.413	37.701	74.877	24.257	0.713	0.391	32.422	69.146
CoR-GS [56]	17.386	0.716	0.406	38.263	75.702	23.913	0.724	0.399	31.276	67.810
Ours	19.711	0.756	0.355	52.016	81.288	25.315	0.737	0.357	45.407	78.849

Table 14. Quantitative results of reconstruction and segmentation on novel views across various scenes from Replica and ScanNet datasets, using CLIP-LSeg [24] for optimizing Gaussian semantic attributes with 12 training views. Our approach achieves **superior performances** in both reconstruction quality and segmentation accuracy.