Identity-aware Language Gaussian Splatting for Open-vocabulary 3D Semantic Segmentation

SungMin Jang   Wonjun Kim*
Konkuk University
{jsmbank, wonjkim}@konkuk.ac.kr

Abstract

Open-vocabulary 3D semantic segmentation has been actively studied by incorporating language features into 3D scene representations. Even though many methods have shown the notable improvement in this task, they still have difficulties to make language embeddings be consistent across different views. This inconsistency highly results in mis-labeling where different language embeddings are assigned to the same part of an object. To address this issue, we propose a simple yet powerful method that aligns language embeddings via the identity information. The key idea is to locate language embeddings for the same identity closely in the latent space while putting them apart otherwise. This approach allows the same object to have identical language embeddings in novel views with accurate semantic masks, which are well aligned with the input text. Furthermore, we propose a progressive mask expanding scheme that enables more accurate extraction of semantic mask boundaries. This scheme is very effective in preserving the boundary shape of the target region by allowing the model to consider the local relationship between segments. Experimental results on benchmark datasets demonstrate that our method delivers state-of-the-art performance in open-vocabulary 3D semantic segmentation. https://github.com/DCVL-3D/ILGS_release

1. Introduction

Open-vocabulary 3D semantic segmentation has begun to attract considerable attentions due to recent advances in 3D scene representations and vision-language models. This technique is able to handle open-ended language queries and segment corresponding regions in the 3D space, thus provides natural and flexible interactions for editing of 3D scenes. As demands from various applications, e.g., autonomous navigation [6], robotic manipulation [7], and VR/AR, increase, there has been the significant research progress in open-vocabulary 3D semantic segmentation in recent years. To do this, most previous methods have utilized high-quality 3D point clouds [19, 25], however, it is quite difficult to acquire data, which reflects various real-world environments, with language annotations. This limitation still makes the practical use of open-vocabulary 3D semantic segmentation challenging.

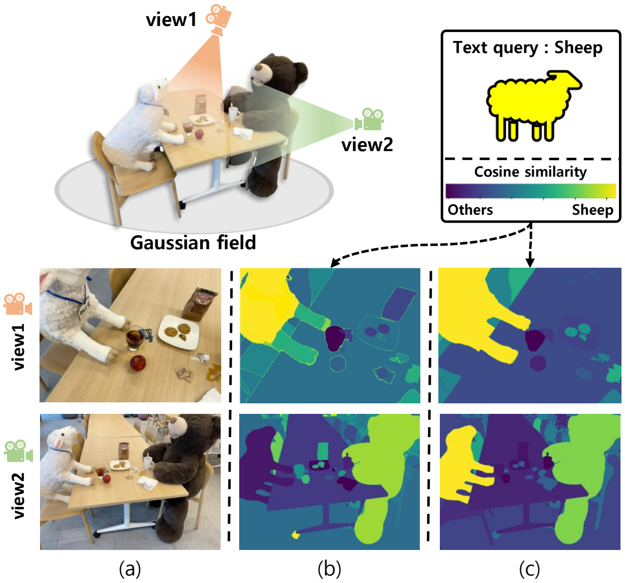

Figure 1. (a) RGB images. (b)(c) Results of the cosine similarity between the text embedding of the input query and language features by the previous method [20] and the proposed method, respectively.

To alleviate this issue, recent studies [10, 14, 20, 21, 24, 31, 34, 35] have adopted techniques of 3D scene representations such as neural radiance fields (NeRF) [16] and 3D Gaussian splatting (3DGS) [9]. Based on this, they have attempted to incorporate language features extracted from vision-language models, e.g., CLIP [22], LSeg [13], etc., into the result of 3D scene representations. Even though such language-aligned scene representations have brought the notable improvement in open-vocabulary 3D semantic segmentation, existing methods often fail to maintain

[*] Corresponding author

the consistency of language embeddings across different views as shown in Fig. 1(b). This issue arises as the same object exhibits varying appearances according to different views, which leads to inconsistencies in language features extracted by CLIP in this example.

In this paper, we propose an identity-aware language Gaussian field to resolve the aforementioned problem in open-vocabulary 3D semantic segmentation. The key idea is that language embeddings with the same identity are forced to be located close together in the latent space whereas those belonging to different identities are placed far apart. Specifically, we augment each Gaussian with both language and identity embeddings, which are learnable by the differentiable process of Gaussian splatting. Under the supervision of CLIP features and segmentation masks, these embeddings are rasterized as the form of a 2D feature map, respectively. Note that we utilize SAM [11] to generate segmentation masks from input images, and exploit a zero-shot tracker [4] to maintain the corresponding identity of each segment across different views. For open-vocabulary 3D semantic segmentation, after training, CLIP features encoded from the input query are compared with the rasterized 2D language feature map via the cosine similarity. The segment of the rasterized 2D identity feature map, which contains the highest cosine similarity, is selected as the seed of the target region. If the difference of the cosine similarity between neighbor segments and the seed is smaller than 10% of the similarity value in the seed, corresponding areas are added to the part of the target region. This progressive expanding scheme helps the model consider the local relationship between segments in the same target, enabling to extract the boundary more accurately. The main contribution of the proposed method can be summarized as follows:

- We propose a novel framework that enforces language embeddings in the Gaussian field to be located closer in the latent space for same identity embeddings, while pushing them apart otherwise. This approach makes language embeddings be consistent for the same object, even in different views.
- We propose a masking strategy that starts with the most relevant segment, determined by the highest cosine similarity between the input query embedding and the rasterized language feature map, and then iteratively adds neighboring segments to the seed segment based on their similarity. This progressive mask expanding scheme helps the model consider relationships between neighbor segments for accurately extracting boundaries of target regions.

2. Related Work

In this Section, we give a brief review of the previous studies for 3D scene representations and open-vocabulary 3D semantic segmentation.

2.1. 3D Scene Representations

Recently, the neural radiance fields (NeRF) [16] has significantly improved the performance of novel view synthesis, which encodes the appearance and geometry of a 3D scene into a neural network. Building upon these promising results, various studies have emerged, particularly focusing on enhancing the rendering quality [1, 26], accelerating the rendering processes [2, 5, 17, 29], and adapting the model to explore unconstrained scenes in the real world [15, 23]. Despite these advances, NeRF still suffers from substantial computational cost, which makes real-time applications impractical and hinders downstream tasks. To address this limitation, 3D Gaussian splatting (3DGS) [9] has emerged in literature. 3DGS explicitly represents scenes with a set of point-based 3D Gaussians. This representation scheme achieves real-time rendering by leveraging well-optimized rasterization techniques. Inspired by the promising performance of 3DGS, various follow-up studies have been introduced, improving the rendering quality [8, 32], enabling editing and manipulation [3], and extending the approach to dynamic scenes [28, 30].

2.2. Open-Vocabulary 3D Semantic Segmentation

While NeRF and 3DGS demonstrate outstanding results in novel view synthesis, they are not capable of performing even semantic understanding. To bridge this gap, researchers have explored various approaches to incorporate semantic information into 3D scene representations. Among NeRF-based methods, Kerr *et al.*[10] encoded multi-scale CLIP features into NeRF to achieve hierarchical vision-language alignment. Meanwhile, Kobayashi *et al.*[12] proposed a feature distillation approach that decomposes NeRF into separate feature fields for semantic understanding and editing of specific objects through language queries. However, the performance of these methods is limited by computationally intensive rendering processes. To overcome this limitation, several studies have introduced various 3DGS-based approaches. Specifically, Qin *et al.* [20] constructed a fine-grained 3D language field using SAM and CLIP features. In particular, Shi *et al.* [24] proposed quantization-based language embedding in 3DGS, which reduced memory usage and preserved feature consistency. Similarly, Zhou *et al.* [34] introduced structured low-dimensional feature fields to enhance segmentation efficiency through feature distillation. Furthermore, Qu *et al.* [21] optimized the semantic-space hyperplane to improve feature selection, leading to more accurate open-vocabulary segmentation. Ye *et al.* [31] incorporated identity encoding in 3DGS for spatial consistency, thereby facilitating object editing in 3D scenes.

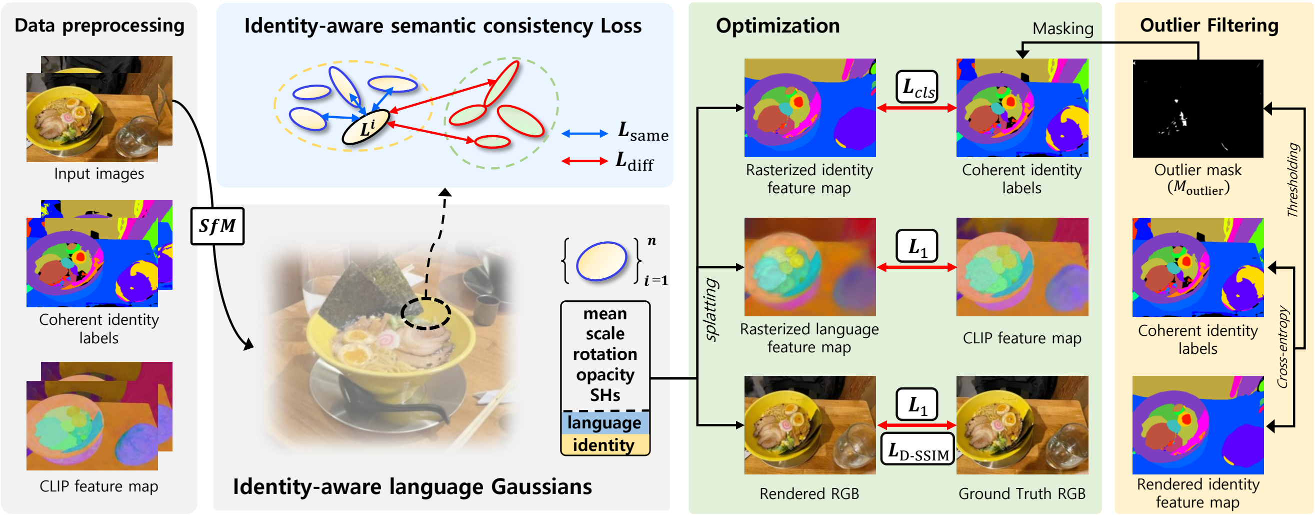

Figure 2. The overall framework of the proposed method. Language and identity embeddings are augmented into the Gaussian primitive. Coherent identity labels serve as pseudo ground truth (GT), representing automatically generated labels by a zero-shot-tracker [4] that approximates the true identity of objects but are not perfectly accurate. To compensate for this, we exclude regions with high cross-entropy between rendered identity features and coherent identity labels from loss computation.

3. Proposed Method

3.1. Preliminaries

3D Gaussian Splatting (3DGS) [9] represents a scene using a set of 3D Gaussians that are rasterized via a differentiable splatting process. Each Gaussian $\mathcal{G}_i$ is defined by its center $\boldsymbol{\mu}_i \in \mathbb{R}^3$, covariance matrix $\Sigma_i \in \mathbb{R}^{3\times 3}$, opacity α_i, and appearance encoded as spherical harmonics (SH) coefficients $\boldsymbol{c}_i$. The i-th Gaussian function is defined as:

$$\mathcal{G}_i(\mathbf{x}) = \exp\!\big(-\tfrac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^\top \Sigma_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\big). \tag{1}$$

During rendering, 3D Gaussians are projected onto the 2D image plane. The covariance undergoes transformation given by $\Sigma_i' = JW\Sigma_i W^\top J^\top$ to account for effects of perspective projection. W is the world-to-camera transformation matrix, and J represents the Jacobian matrix for perspective projection. The covariance Σ_i is decomposed into a rotation matrix R and a scaling matrix S as $\Sigma_i = RSS^\top R^\top$, which improves numerical stability during optimization. The final rendered color $C(v)$ at each pixel v is computed by splatting Gaussians onto the image plane, followed by alpha blending:

$$C(v) = \sum_{i\in\mathcal{N}} \boldsymbol{c}_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \tag{2}$$

where $\boldsymbol{c}_i$ represents the color of the i-th Gaussian. $\mathcal{N}$ is the set of Gaussians overlapping pixel v. $\alpha_i = \alpha_i^{2D}(v)$ denotes the opacity of the i-th Gaussian after projection to the image plane.

3.2. Identity-aware Semantic Consistency Learning

Open-vocabulary 3D semantic segmentation faces challenges in keeping language embeddings consistent across different views. While Gaussian primitives are effectively optimized through the multi-view supervision, language embeddings often fail to be properly optimized, leading to inconsistent representations across different views. This inconsistency results in fragmented semantic masks in novel views.

To address this issue, we introduce an identity-aware semantic consistency learning scheme. The overall framework of the proposed method is shown in Fig. 2. Specifically, we incorporate the identity information into our framework, inspired by the concept of the identity encoding for segmentation and editing in 3D scenes [31]. In our approach, each Gaussian is augmented with a 16-dimensional identity embedding, which is learned to maintain consistent values for the same object across different views. In addition, we also augment each Gaussian with a 3-dimensional language embedding. Both language and identity embeddings are learned simultaneously through the differentiable process of Gaussian splatting.

During training, we randomly select a subset of Gaussians and compute the identity-aware semantic consistency loss $\mathcal{L}_{\text{cons}}$, which is defined as follows:

$$\mathcal{L}_{\text{cons}} = \mathcal{L}_{\text{same}} + \mathcal{L}_{\text{diff}}, \tag{3}$$

$$\begin{aligned}
\mathcal{L}_{\text{same}} &= \frac{1}{S}\sum_{i=1}^{M}\sum_{j=1}^{N} \mathbb{1}(F_i^{ID} = F_j^{ID})\Big(1-\cos(F_i^{Lang}, F_j^{Lang})\Big),\\
\mathcal{L}_{\text{diff}} &= \frac{1}{D}\sum_{i=1}^{M}\sum_{j=1}^{N} \mathbb{1}(F_i^{ID} \neq F_j^{ID})\Big(1+\cos(F_i^{Lang}, F_j^{Lang})\Big).
\end{aligned} \tag{4}$$

Here, $\mathbb{1}(\cdot)$ is the indicator function that returns 1 if the condition inside holds and 0 otherwise. The term $\cos(a,b) = \frac{a \cdot b}{|a||b|}$ represents the cosine similarity. F_i^{Lang} and F_j^{Lang} correspond to the language embeddings of the i-th and j-th Gaussians, respectively, while F_i^{ID} and F_j^{ID} refer to their identity embeddings. M is the number of selected Gaussians. N is the number of pairs considered for each Gaussian. S and D denote the total number of pairs satisfying $F_i^{ID} = F_j^{ID}$ and $F_i^{ID} \neq F_j^{ID}$, respectively. The loss term $\mathcal{L}_{\text{same}}$ enforces the consistency by maximizing the cosine similarity between language embeddings of Gaussians having the same identity. Meanwhile, $\mathcal{L}_{\text{diff}}$ minimizes the cosine similarity between Gaussians with different identities. By aligning language embeddings conditioned on the identity information, the proposed method yields the reliable segmentation result, which is well aligned with the input text query across different views as shown in Fig. 1(c).

However, the identity-aware semantic consistency loss relies on accurate coherent identity labels. These labels serve as pseudo ground truth, automatically generated by a zero-shot-tracker [4] to approximate true object identities. To compensate for the inaccuracy of coherent identity labels, we introduce a simple outlier filtering scheme. Specifically, we measure the per-pixel cross-entropy between the rendered identity feature maps $\widehat{ID}$ and the coherent identity labels ID, and filter out identity regions with high discrepancies, which is formulated as follows:

$$M_{\text{outlier}}(\mathbf{x}) = \begin{cases} 0, & \text{if } CE\big(\widehat{ID}(\mathbf{x}), ID(\mathbf{x})\big) > \tau(t), \\ 1, & \text{otherwise}, \end{cases} \tag{5}$$

where $CE(\cdot,\cdot)$ denotes the cross-entropy. $\mathbf{x}$ represents the spatial position of each pixel in the feature map. $\tau(t)$ is initially set to twice the average of cross-entropy values and decreases proportionally according to the number of iterations, reaching the average at 30,000 iterations. The example of $M_{\text{outlier}}(\mathbf{x})$ is shown in Fig. 2. This method facilitates identity-aware semantic consistency learning by improving the reliability of the identity information.

3.3. Progressive Mask Expanding

After training, open-vocabulary 3D semantic segmentation can be conducted in response to input text queries. To this end, most previous methods have relied on fixed empirical threshold values in generating semantic segmentation masks, however, they frequently fail to preserve object boundaries with intricate details.

To resolve this problem, we propose a progressive mask expanding scheme. Specifically, language and identity embeddings are rasterized from our identity-aware language Gaussian field. Given an input text query, we compute the cosine similarity between the text embedding of the input query and the rasterized language feature map. We then calculate the average cosine similarity for each segment in the rasterized identity feature map (see Fig. 2) and select the segment with the highest average similarity as the seed for the target region. Segments adjacent to the seed are incorporated into the target region when the difference of the cosine similarity between corresponding segments and the seed segment is smaller than 10% of the similarity value in the seed. This scheme proceeds iteratively while newly added neighbor segments become new seed segments themselves. The expansion continues until there are no remaining segments to add. The overall scheme is illustrated in Fig. 3.

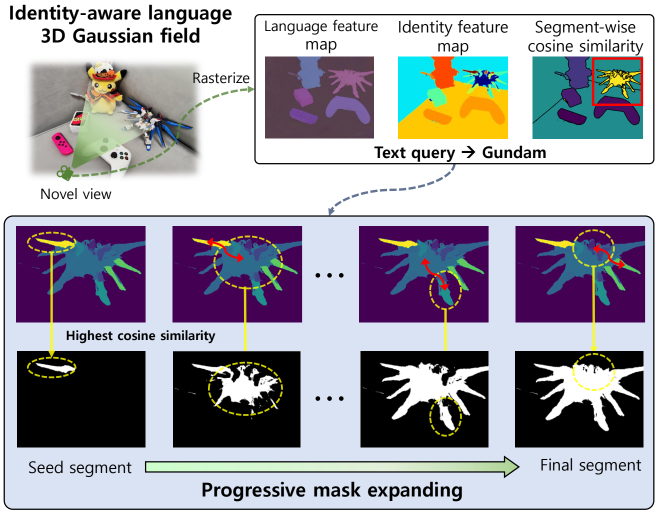
Identity-aware language 3D Gaussian field
Rasterize
Novel view
Language feature map
Identity feature map
Segment-wise cosine similarity
Text query → Gundam
Highest cosine similarity
Seed segment
Final segment
Progressive mask expanding

Figure 3. The proposed progressive mask expanding scheme. Rasterized language and identity feature maps are used to compute the cosine similarity with the input text query. Starting from a seed segment with the highest cosine similarity, the mask iteratively expands to neighboring segments.

This progressive expanding scheme helps the model consider the local relationship between segments in the same target, which ensures to extract segmentation boundaries more precisely. Additionally, it effectively handles variations in distributions of the cosine similarity in terms of different input queries without manual adjustments of threshold values as shown in Fig. 4.

3.4. Loss Function

The proposed method is trained based on four loss terms, i.e., color reconstruction loss $\mathcal{L}_{\text{rgb}}$, 2D identity loss $\mathcal{L}_{\text{cls}}$, CLIP loss $\mathcal{L}_{\text{clip}}$, and identity-aware semantic consistency loss $\mathcal{L}_{\text{cons}}$, which is newly introduced in subsection 3.2. The color reconstruction loss consists of L1 and D-SSIM terms, which measure the similarity of colors and structures between the rendered image $\hat{I}$ and the ground-truth image I, formulated as follows [9]:

$$\mathcal{L}_{\text{rgb}} = \mathcal{L}_1(\hat{I}, I) + \mathcal{L}_{\text{D-SSIM}}(\hat{I}, I). \tag{6}$$

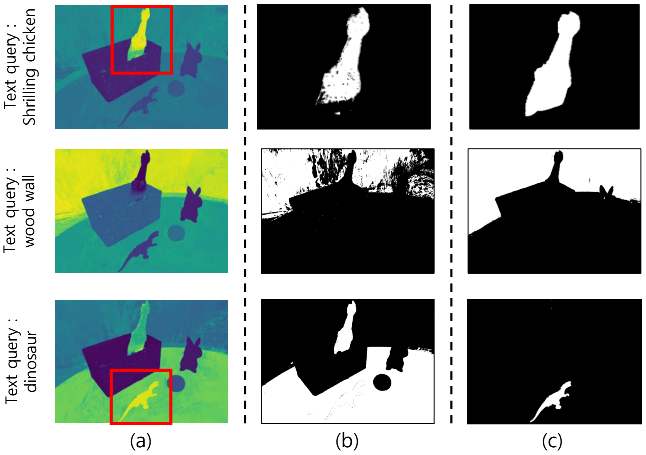

Figure 4. (a) Cosine similarity between the rasterized language feature map and the text embedding of the input query. (b)(c) Results of semantic segmentation by the previous method using the fixed threshold value [24] and the proposed method, respectively.

To learn identity and language embeddings augmented to Gaussians, we utilize the 2D identity loss $\mathcal{L}_{\text{cls}}$ and the CLIP loss $\mathcal{L}_{\text{clip}}$. The 2D identity loss is applied to enforce the identity be consistent across different views [31], given as

$$\mathcal{L}_{\text{cls}} = \frac{1}{|M_{\text{valid}}|} \sum_{\mathbf{x} \in M_{\text{valid}}} CE\big(\widehat{ID}(\mathbf{x}), ID(\mathbf{x})\big), \quad (7)$$

where M_{valid} represents the set of valid pixels after outlier filtering, which is introduced in subsection 3.2. The CLIP loss $\mathcal{L}_{\text{clip}}$ is computed by using the L1 norm between rasterized language feature maps and CLIP feature maps by following the approach in [20]. The total loss function is defined as follows:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{rgb}}\mathcal{L}_{\text{rgb}} + \lambda_{\text{cls}}\mathcal{L}_{\text{cls}} + \lambda_{\text{clip}}\mathcal{L}_{\text{clip}} + \lambda_{\text{cons}}\mathcal{L}_{\text{cons}}, \quad (8)$$

where $\lambda_{\text{rgb}}, \lambda_{\text{cls}}, \lambda_{\text{clip}}$, and λ_{cons} are balancing factors for each loss term, which are set to 1.0, 1.0, 1.0, and 0.5, respectively. For stable optimization, we do not apply $\mathcal{L}_{\text{cons}}$ during the first 15,000 iterations, allowing the model to focus on learning by $\mathcal{L}_{\text{clip}}$. After this warm-up phase, we remove $\mathcal{L}_{\text{clip}}$ and incorporate $\mathcal{L}_{\text{cons}}$ into the total loss for remaining iterations.

4. Experimental Results

4.1. Training

All experiments were conducted using the PyTorch framework [18] and custom CUDA kernels, built upon the 3DGS framework [9]. Our model is trained on an AMD EPYC 7352 24-Core Processor CPU and a single NVIDIA A100 GPU. We employed the Adam optimizer with the first and second moment decay rates of 0.9 and 0.999 to train all model parameters. Our model is trained in the end-to-end manner, that is, Gaussians, identity embeddings, and language embeddings are optimized simultaneously. The dimension of identity and language embeddings is set to 16 and 3, respectively.

4.2. Datasets and Evaluation Metrics

Datasets. For the performance evaluation of the proposed method, two representative benchmarks, i.e., LERF [10] and 3D-OVS [14], are employed. The LERF dataset consists of 3D scenes in the wild, which are captured by using the Polycam application on the iPhone. The 3D-OVS dataset consists of the diverse set of long-tail objects, which are captured with various poses under different backgrounds.

Evaluation metrics. For the quantitative evaluation, we use two metrics, i.e., the mean Intersection over Union (mIoU) and the mean Boundary IoU (mBIoU), which are commonly used in this field. Specifically, mIoU measures the segmentation accuracy by computing the ratio of the intersection between the predicted result and ground truth segmentation to their union. Meanwhile, mBIoU measures the alignment accuracy between predicted segmentation boundaries and ground truth. These metrics evaluate the accuracy of semantic segmentation masks corresponding to the input text queries.

4.3. Performance Evaluation

Quantitative evaluation. To demonstrate the effectiveness of the proposed method, we compare ours with previous methods for open-vocabulary 3D semantic segmentation, i.e., LangSplat [20], Feature-3DGS [34], GS-Grouping [31], GOI [21], and LEGaussian [24]. The result of the performance comparison is shown in Table 1. In all experiments, the proposed method shows the meaningful improvement compared to previous approaches. Specifically, the proposed method achieves 80.5 mIoU and 76.0 mBIoU on the LERF dataset, which outperforms the state-of-the-art methods by a considerable margin for all the metrics. The performance comparison on the 3D-OVS dataset is also shown in Table 2. As can be seen, the proposed method achieves 94.4 mIoU, which shows the superior performance compared to previous methods. Based on results presented in Tables 1 and 2, we could confirm that the proposed identity-aware semantic consistency learning scheme is effective to generate semantic segmentation masks in novel views. Furthermore, we also evaluate the performance of the proposed method with photometric metrics, such as peak signal-to-noise ratio (PSNR), structural similarity index (SSIM) [27], and learned perceptual image patch similarity (LPIPS) [33]. The corresponding results are shown in Table 3. We found that our method improves

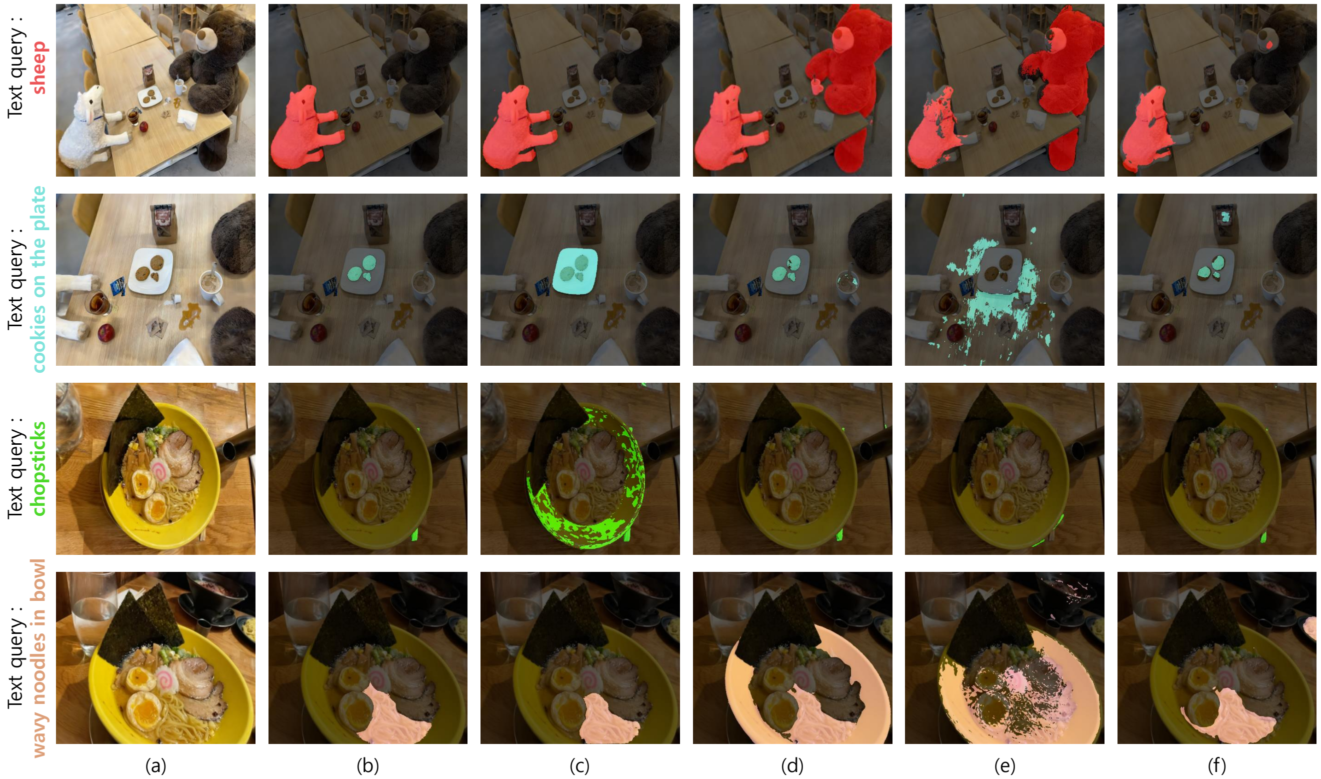
Text query : sheep
Text query : cookies on the plate
Text query : chopsticks
Text query : wavy noodles in bowl
(a)
(b)
(c)
(d)
(e)
(f)

Figure 5. Some examples of open-vocabulary 3D semantic segmentation on the LERF [10] dataset. (a) Rendered images by 3DGS [9]. Results by (b) the proposed method, (c) GOI [21], (d) Gaussian Grouping [31], (e) Feature 3DGS [34], and (f) LangSplat [20].

Methods	**figurines**		**teatime**		**ramen**		**Avg.**	
	mIoU	mBIoU	mIoU	mBIoU	mIoU	mBIoU	mIoU	mBIoU
LangSplat [20]	52.8	50.5	69.5	65.6	50.4	44.7	57.6	53.6
Feature-3DGS [34]	58.8	52.5	40.5	36.8	43.7	38.3	47.7	45.7
Gaussian Grouping [31]	69.7	67.9	71.7	66.1	77.0	68.7	72.8	67.6
GOI [21]	63.7	-	44.5	-	52.6	-	53.6	-
Ours	**75.9**	**73.8**	**81.2**	**78.8**	**84.3**	**75.5**	**80.5**	**76.0**

Table 1. Performance comparisons of open-vocabulary 3D semantic segmentation on the LERF [10] dataset (the best results are shown in bold).

rendering performance for novel view synthesis compared to other end-to-end approaches.

Qualitative evaluation. Furthermore, the qualitative comparison with LangSplat [20], Feature-3DGS [34], GS-Grouping [31], GOI [21], and LEGaussian [24] is presented on LERF and 3D-OVS datasets in Figs. 5 and 6, respectively. These results demonstrate the effectiveness of the proposed method in open-vocabulary 3D semantic segmentation. Specifically, the misalignment between language embeddings and corresponding objects yields incorrect semantic segmentation masks. As a result, semantic segmentation masks contain incorrect regions, which are not aligned with input text queries as shown in Fig. 5(c)(d)(e) and 6(c)(f). In addition, previous methods often fail to extract boundaries accurately due to the use of fixed threshold values in generating semantic segmentation masks(see Fig. 5(e)(f) and Fig. 6(e)(f)). As can be seen, the proposed method successfully mitigates this problem by applying the identity-aware semantic consistency loss and the progressive mask expanding scheme. Consequently, our approach generates open-vocabulary 3D semantic masks more accurately compared to previous approaches. Furthermore, we

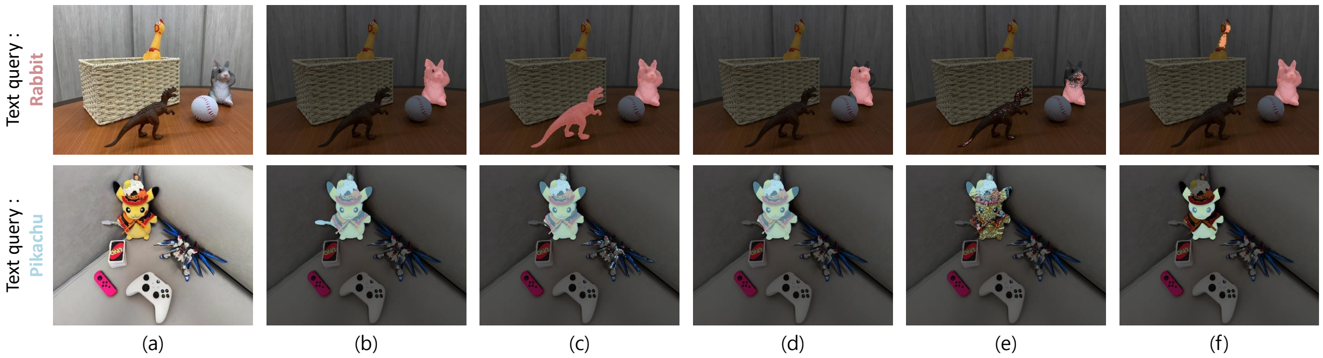

Figure 6. Some examples of open-vocabulary 3D semantic segmentation on the LERF [10] dataset. (a) Rendered images by 3DGS [9]. Results by (b) the proposed method, (c) GOI [21], (d) Gaussian Grouping [31], (e) Feature 3DGS [34], and (f) LangSplat [20].

Methods	bed	bench	Room	sofa	lawn	Avg.
LangSplat [20]	92.5	94.2	**94.1**	90.0	**96.1**	93.4
Feature-3DGS [34]	83.5	90.7	84.7	86.9	93.4	87.8
Gaussian Grouping [31]	83.0	91.5	85.9	87.3	90.6	87.7
GOI [21]	89.4	92.8	91.3	85.6	94.1	90.6
Ours	**96.4**	**95.5**	93.6	**92.2**	94.2	**94.4**

Table 2. Performance comparisons of open-vocabulary 3D semantic segmentation on the 3D-OVS [14] dataset (the best mIoU results are shown in bold).

Methods	PSNR	SSIM	LPIPS
3DGS	25.870	0.867	0.211
Gaussian Grouping [31]	25.710	0.852	0.235
Ours	**26.108**	**0.868**	**0.210**

Table 3. Performance comparisons of novel view rendering on the LERF [10] dataset (the best results are shown in bold).

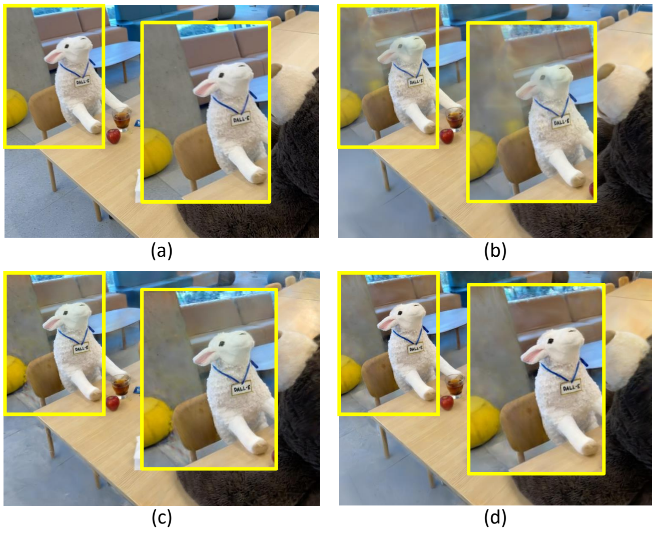

Figure 7. Example of novel view rendering. (a) Ground truth. Results by (b) 3DGS, (c) Gaussian grouping [31], and (d) Ours.

can see that the proposed method is able to render the target object without any severe distortion in background compared to previous methods as shown in Fig. 7.

4.4. Ablation Study

In this subsection, we first demonstrate the comparative experimental results by changing the components of the proposed method based on the LERF dataset. Table 4 shows the contribution of such components. As can be seen, the performance of open-vocabulary 3D semantic segmentation is considerably improved as each component is added to the baseline. The best performance (mIoU: 80.5, mBIoU: 76.0) is achieved when identity-aware semantic consistency loss, progressive mask expanding, and outlier filtering are all applied.

In particular, we can see that the identity-aware semantic loss is most effective for improving the performance of open-vocabulary 3D semantic segmentation. The performance drop in the mBIoU value is observed when progressive mask expanding is not used. This confirms the effectiveness of our progressive mask expanding scheme in refining boundaries. Moreover, outlier filtering supports identity-aware semantic consistency loss and progressive

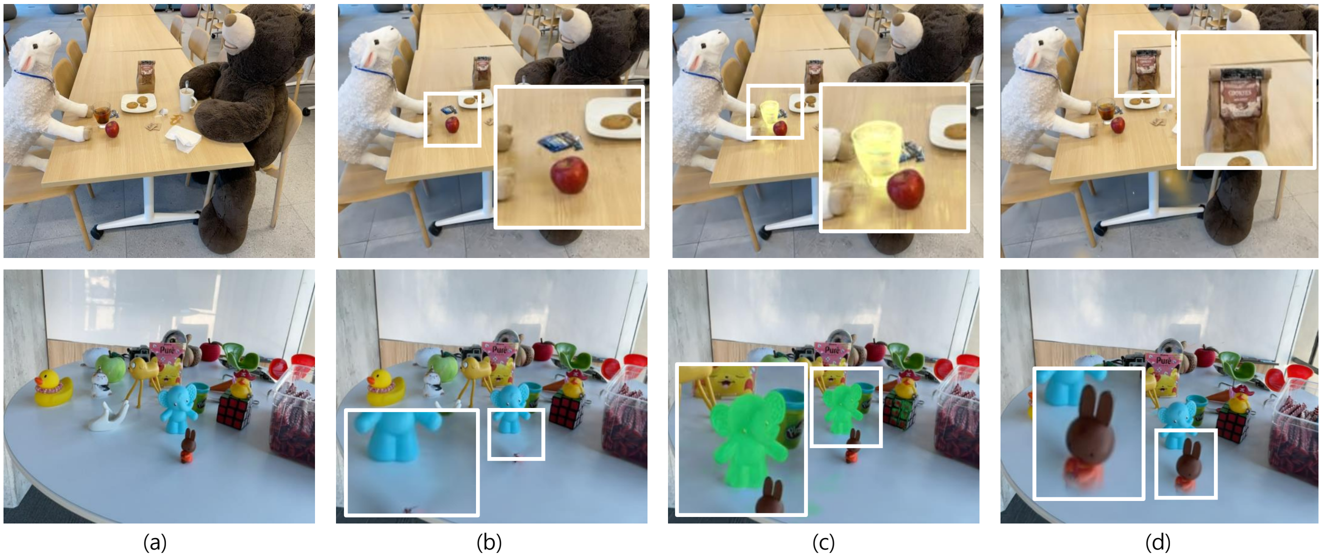

Figure 8. Some examples of 3D scene editing. (a) Input images. (b) Results of object removal (queries: "tea in glass" (top), "rabbit" (bottom)). (c) Results of color modification (queries: "tea in glass" (top), "blue elephant" (bottom)). (d) Results of object resizing (queries: "bag of cookies" (top), "rabbit" (bottom)).

Progressive mask expanding	Identity-aware Semantic loss	Outlier filtering	mIoU	mBIoU
✓	✗	✗	71.4	61.7
✗	✓	✗	73.3	67.6
✓	✗	✓	74.1	71.2
✓	✓	✗	75.8	72.8
✗	✓	✓	76.2	70.1
✓	✓	✓	**80.5**	**76.0**

Table 4. Performance analysis of the proposed method based on changes in progressive mask expanding, identity-aware semantic consistency loss, and outlier filtering on the LERF dataset (the best result are shown in bold).

mask expanding by providing more reliable identity labels. As a result, the performance of semantic segmentation is effectively improved. Consequently, synergy among the proposed components yields superior performance in open-vocabulary 3D semantic segmentation.

4.5. 3D Scene Editing

To further demonstrate the practical applicability of the proposed method, we apply it to 3D scene editing tasks. Specifically, we conduct object removal, color modification, and object resizing by leveraging our identity-aware language Gaussian field. Given an input text query, the segmentation mask is generated by using the rasterized language feature map (which is explained in subsection 3.3). We then obtain identity labels included in this segmentation mask from the rasterized identity feature map. Gaussians corresponding to these identity labels are selected for modification. For object removal, we can now exclude selected Gaussians from the 3D scene while preserving the surrounding background. For color modification, we adjust the appearance of the object by updating spherical harmonics (SH) coefficients in Gaussian representations. For object resizing, we apply a scaling transformation scheme to Gaussians and adjust their positions accordingly. These editing operations show that our identity-aware language Gaussian field can be effectively applied to applications for 3D scene editing as shown in Fig. 8.

5. Conclusion

In this paper, we introduce a simple yet powerful method for open-vocabulary 3D semantic segmentation. The key idea of the proposed method is to mitigate the multi-view inconsistency in language features through an identity-aware semantic consistency loss. This approach maintains consistent language features for the same object across different views. Additionally, the progressive mask expanding scheme provides more accurate boundaries. As a result, the proposed method produces precise semantic segmentation masks for input text queries. Experimental results demonstrate the effectiveness of our approach in open-vocabulary 3D semantic segmentation.

Acknowledgments. This work was supported by the National Research Foundation of Korea (NRF) funded by the Korea Government (MSIT) under Grant RS-2023-NR076462.

References

[1] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-NeRF: A Multiscale Representation for Anti-Aliasing Neural Radiance Fields. In *Proc. Int. Conf. Comput. Vis.*, pages 5855–5864, 2021.

[2] Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. TensoRF: Tensorial Radiance Fields. In *Proc. Eur. Conf. Comput. Vis.*, pages 333–350, 2022.

[3] Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. GaussianEditor: Swift and Controllable 3D Editing with Gaussian Splatting. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 21476–21485, 2024.

[4] Ho Kei Cheng, Seoung Wug Oh, Brian Price, Alexander Schwing, and Joon-Young Lee. Tracking Anything with Decoupled Video Segmentation. In *Proc. Int. Conf. Comput. Vis.*, pages 1316–1326, 2023.

[5] Stephan J Garbin, Marek Kowalski, Matthew Johnson, Jamie Shotton, and Julien Valentin. FastNeRF: High-Fidelity Neural Rendering at 200FPS. In *Proc. Int. Conf. Comput. Vis.*, pages 14346–14355, 2021.

[6] Qiao Gu, Ali Kuwajerwala, Sacha Morin, Krishna Murthy Jatavallabhula, Bipasha Sen, Aditya Agarwal, Corban Rivera, William Paul, Kirsty Ellis, Rama Chellappa, et al. ConceptGraphs: Open-Vocabulary 3D Scene Graphs for Perception and Planning. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 5021–5028, 2024.

[7] Chenguang Huang, Oier Mees, Andy Zeng, and Wolfram Burgard. Visual Language Maps for Robot Navigation. In *Proc. IEEE Int. Conf. Robot. Autom.*, pages 10608–10615, 2023.

[8] Letian Huang, Jie Guo, Jialin Dan, Ruoyu Fu, Shujie Wang, Yuanqi Li, and Yanwen Guo. Spectral-GS: Taming 3D Gaussian Splatting with Spectral Entropy. *arXiv preprint arXiv:2409.12771*, 2024.

[9] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM Trans. Graph.*, 42(4):1–14, 2023.

[10] Justin Kerr, Chung Min Kim, Ken Goldberg, Angjoo Kanazawa, and Matthew Tancik. LERF: Language Embedded Radiance Fields. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 19729–19739, 2023.

[11] 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 *Proc. Int. Conf. Comput. Vis.*, pages 4015–4026, 2023.

[12] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing NeRF for Editing via Feature Field Distillation. In *Proc. Adv. Neural Inform. Process. Syst.*, pages 23311–23330, 2022.

[13] Boyi Li, Kilian Q Weinberger, Serge Belongie, Vladlen Koltun, and René Ranftl. Language-driven Semantic Segmentation. In *Proc. Int. Conf. Learn. Represent.*, pages 1–13, 2022.

[14] 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. In *Proc. Adv. Neural Inform. Process. Syst.*, pages 53433–53456, 2023.

[15] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 7210–7219, 2021.

[16] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *Proc. Eur. Conf. Comput. Vis.*, pages 405–421, 2020.

[17] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. *ACM Trans. Graph.*, 41(4):1–15, 2022.

[18] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *Proc. Adv. Neural Inform. Process. Syst.*, pages 1–4, 2017.

[19] Songyou Peng, Kyle Genova, Chiyu Jiang, Andrea Tagliasacchi, Marc Pollefeys, Thomas Funkhouser, et al. OpenScene: 3D Scene Understanding with Open Vocabularies. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 815–824, 2023.

[20] Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. Langsplat: 3D Language Gaussian splatting. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 20051–20060, 2024.

[21] Yansong Qu, Shaohui Dai, Xinyang Li, Jianghang Lin, Liujuan Cao, Shengchuan Zhang, and Rongrong Ji. GOI: Find 3D Gaussians of Interest with an Optimizable Open-vocabulary Semantic-space Hyperplane. In *Proc. ACM Int. Conf. Multimedia*, pages 5328–5337, 2024.

[22] 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 *Proc. Int. Conf. Mach. Learn.*, pages 8748–8763, 2021.

[23] Weining Ren, Zihan Zhu, Boyang Sun, Jiaqi Chen, Marc Pollefeys, and Songyou Peng. NeRF On-the-go: Exploiting Uncertainty for Distractor-free NeRFs in the Wild. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8931–8940, 2024.

[24] Jin-Chuan Shi, Miao Wang, Hao-Bin Duan, and Shao-Hua Guan. Language Embedded 3D Gaussians for Open-Vocabulary Scene Understanding. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5333–5343, 2024.

[25] Ayca Takmaz, Elisabetta Fedele, Robert Sumner, Marc Pollefeys, Federico Tombari, and Francis Engelmann. Open-Mask3D: Open-Vocabulary 3D Instance Segmentation. In *Proc. Adv. Neural Inform. Process. Syst.*, pages 68367–68390, 2023.

[26] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-NeRF: Structured View-Dependent Appearance for Neural Radiance Fields. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 5481–5490, 2022.

[27] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. Image Process.*, 13(4): 600–612, 2004.

[28] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4D Gaussian Splatting for Real-Time Dynamic Scene Rendering. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 20310–20320, 2024.

[29] Xiufeng Xie, Riccardo Gherardi, Zhihong Pan, and Stephen Huang. HollowNeRF: Pruning Hashgrid-Based NeRFs with Trainable Collision Mitigation. In *Proc. Int. Conf. Comput. Vis.*, pages 3480–3490, 2023.

[30] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time Photorealistic Dynamic Scene Representation and Rendering with 4D Gaussian Splatting. In *Proc. Int. Conf. Learn. Represent.*, pages 1–18, 2024.

[31] Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian Grouping: Segment and Edit Anything in 3D Scenes. In *Proc. Eur. Conf. Comput. Vis.*, pages 162–179, 2024.

[32] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-Splatting: Alias-free 3D Gaussian Splatting. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 19447–19456, 2024.

[33] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 586–595, 2018.

[34] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. Feature 3DGS: Supercharging 3D Gaussian Splatting to Enable Distilled Feature Fields. In *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, pages 21676–21685, 2024.

[35] Xingxing Zuo, Pouya Samangouei, Yunwen Zhou, Yan Di, and Mingyang Li. FMGS: Foundation Model Embedded 3D Gaussian Splatting for Holistic 3D Scene Understanding. *Int. J. Comput. Vis.*, 133(2):611–627, 2025.