PePESeg3D: Perception Prior Enhances Multi-Scale Segmentation for 3D Gaussian Splatting

Sungjae Choi Seunghee Koh Junmo Kim
KAIST
{sungjae579, seunghee1215, junmo.kim}@kaist.ac.kr

Abstract

*Recent advancements in 3D Gaussian Splatting (3DGS) have enabled multi-scale segmentation. Existing multi-scale segmentation methods often suffer from geometric ambiguity and view-inconsistent predictions, as they seperately reconstruct scenes and assign multi-scale semantics. To address these limitations, we propose **PePESeg3D**, a novel framework that injects perception priors into a multi-scale 3D Gaussian segmentation. We introduce PePE reconstruction, which incorporates monocular depth and segmentation constraints into Gaussian reconstruction to produce geometrically coherent structures. Building on this aligned geometry, we propose a PePE contrastive learning that integrates multi-scale masks, dense depth-color cues, and view-consistent centroid supervision to effectively compensate for the incompelete 2D masks and enforce global semantic consistency. Extensive experiments on the SPIn-NeRF and LERF-Mask benchmarks demonstrate that PePESeg3D achieves state-of-the-art performance in both multi-scale segmentation.*

1. Introduction

The emergence of 3D Gaussian Splatting (3DGS) introduces a new paradigm in 3D scene understanding. By lifting the capabilities of 2D foundation models onto Gaussian primitives, 3DGS facilitates segmentation [3, 15, 17–19] by allowing specific semantic feature to be assigned to each Gaussians. This design naturally extends from single-scale segmentation to multi-scale understanding, where the compositions of primitives can represent varying granularities from fine parts to whole objects [2, 16].

Typical single-scale segmentation methods jointly optimize Gaussian parameters and segmentation features using a unified ground truth from SAM [7]. In contrast, multi-scale segmentation requires sequential optimization of geometry and segmentation features due to complex ground truth with varying granularity. This leads to two inherent limitations of multi-scale segmentation in 3DGS.

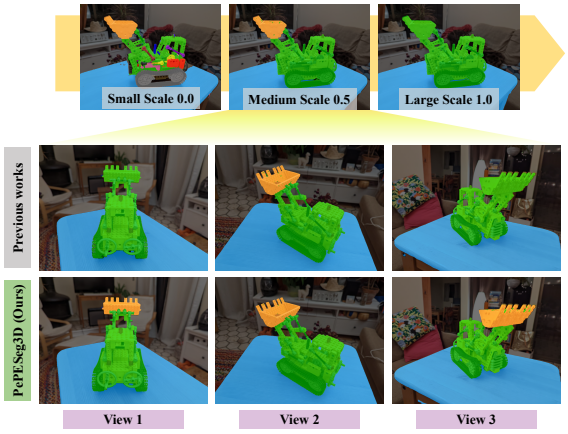

Figure 1. Comparison of view consistency in multi-scale segmentation. Unlike prior methods that produce view-inconsistent results, PePESeg3D achieves consistent segmentation across viewpoints by bridging geometry and semantics.

First, primitives initialized solely via photometric supervision without semantic awareness lead to a geometric mismatch. This misalignment hinders the optimization of segmentation features. Second, reliance on SAM introduces 3D inconsistency, as its grid-based prompting yields view-dependent masks with inconsistent granularity and missing entities. Introducing multiple granularities for multi-scale segmentation further exacerbates this inconsistency. As shown in Figure 1, existing methods fail to leverage underlying 3D structure, resulting in feature representations that lack robustness and consistency.

To address these limitations, we propose **PePESeg3D**, an integrated framework with two-stage optimization : Perception Prior Enhanced (PePE) Reconstruction and Perception Prior Enhanced (PePE) Contrastive Learning (see Figure 2). To resolve the spatial misalignment, PePE Reconstruction incorporates perception priors, including 2D segmentation masks and monocular depth, into the geometry optimization. This enables Gaussian primitives to respect both semantic boundaries and 3D structure, providing a well-aligned initialization for multi-scale feature learning.

Second, PePE Contrastive Learning complements scale-

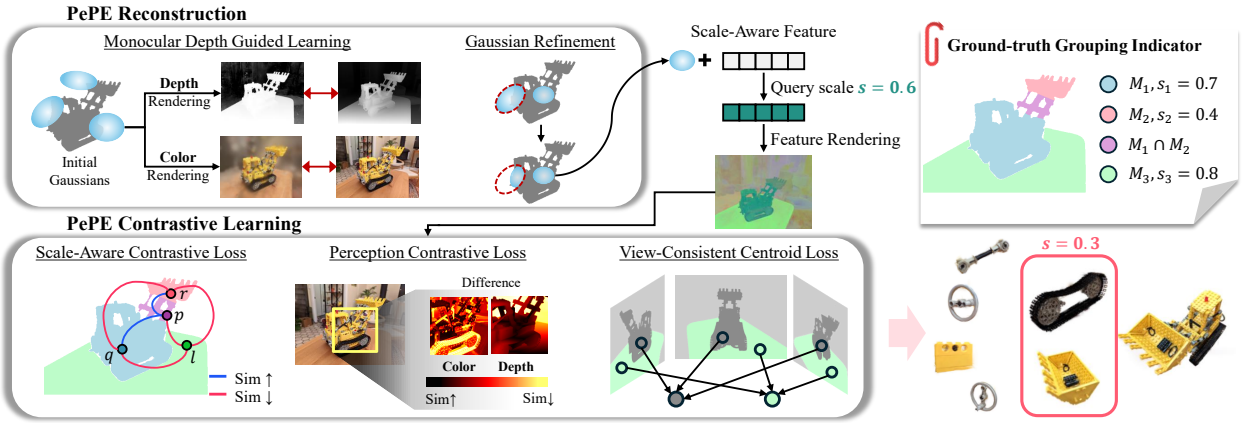

Figure 2. The two-stage pipeline of PePESeg3D. In PePE Reconstruction (top), monocular depth and segmentation masks guide Gaussian geometry to align with semantic boundaries and 3D structure. In PePE Contrastive Learning (bottom), scale-aware masks, depth-RGB cues, and scene-level centroids jointly supervise feature learning, enabling consistent multi-scale 3D segmentation.

aware mask supervision by introducing perceptual cues to compensate for incomplete SAM masks. Motivated by the observation that object boundaries correspond to abrupt depth and color changes [8], we exploit these signals to provide dense supervision and enforce feature distinctiveness, even in ambiguous regions. To further ensure consistency across views and scales, we introduce view-consistent centroid supervision that aligns local features with dynamically maintained global centroids. Extensive experiments demonstrate that PePESeg3D outperforms existing multi-scale methods while remaining competitive with single-scale approaches, highlighting the effectiveness of perception priors in improving segmentation.

2. Methods

2.1. PePE Reconstruction

2.1.1. Gaussian Refinement Guided Initialization

We refine the Gaussian parameters in the early stage to align with segmentation boundaries. For a given view with SAM [7] mask set $\mathcal{M}$, we sort $\mathcal{M}$ in ascending order of mask size to prioritize fine-grained details and process them sequentially. Each 3D Gaussian g is projected into a 2D Gaussian g', and we check whether its major axis v' intersects the boundary of $M_k \in \mathcal{M}$. If so, the Gaussian is refined following [4]. We restrict refinement to Gaussians near surfaces, since refining background regions can severely distort scene geometry. To identify them, we measure the depth gap between the projected z coordinate $\mathbf{z}_k$ of each Gaussian g_k and the rendered depth at the corresponding pixel, $\hat{D}_p = \sum_{i \in N_p} \mathbf{z}_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j)$. We retain only the top λ_{close} of Gaussians with the smallest depth gaps.

2.1.2. Monocular Depth Constrained Learning

We incorporate monocular depth map D from Depth-Anything-V2 [14] as a geometric prior. Since it provides relative depth, we follow [12] to estimate optimal scale parameters α^*, β^* via least-squares alignment between rendered depth $\hat{D}$ and prior D. The depth loss is defined over all N_d pixels in the depth map:

$$\mathcal{L}_{depth} = \frac{1}{N_d} \sum_{p \in D} |\alpha^* \hat{D}_p + \beta^* - D_p|, \tag{1}$$

Combined with photometric loss [5], the total objective is:

$$\mathcal{L}_{recon} = (1-\lambda_{ph})\mathcal{L}_1 + \lambda_{ph}\mathcal{L}_{\text{D-SSIM}} + \lambda_d \mathcal{L}_{depth}, \tag{2}$$

where λ_{ph} and λ_d are hyperparameters. We optimize the 3DGS representation using this objective as a pretraining stage. The resulting geometry serves as a foundation for subsequent physical scale calculation and feature learning.

2.2. PePE Contrastive Learning

We optimize Gaussian features $\mathbf{f}$ and a scale gate ψ using a scale-aware contrastive objective on rendered features. Given a query scale $s \in [0, 1]$, normalized from the spatial extent of masks, we define the scale-aware feature [2, 6] as $\mathbf{f}^s = \psi(s) \odot \mathbf{f}$. For a pixel p, its rendered feature is obtained by alpha blending $\mathbf{F}^s(p) = \sum_{i \in N_p} \mathbf{f}_i^s \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j)$. We then compute the cosine similarity between a pixel pair (p, q) as $C_{pq}^s = \langle \mathbf{F}^s(p), \mathbf{F}^s(q) \rangle / \left(\|\mathbf{F}^s(p)\|_2 \, \|\mathbf{F}^s(q)\|_2 \right)$. For supervision at scale s, let $\mathcal{M}_p = \{M_k \in \mathcal{M} \mid p \in M_k, \; s_{M_k} \geq s\}$ be the set of masks at pixel p whose scale is no smaller than s. We assign the active label as the finest mask in this set $\Lambda(s, p) = \arg\min_{M_k \in \mathcal{M}_p} \{s_{M_k}\}$. Using $\Lambda(s, p)$, we define the grouping indicator G_{pq}^s as $G_{pq}^s = \mathbf{1}\Big[(\Lambda(s,p) = \Lambda(s,q)) \vee (\exists k : p, q \in M_k \wedge s_{M_k} < s) \Big]$. This additionally treats pixels co-occurring in a finer-scale mask as a positive pair, preserving hierarchical consistency across scales. The scale-aware contrastive objective is computed over query scales

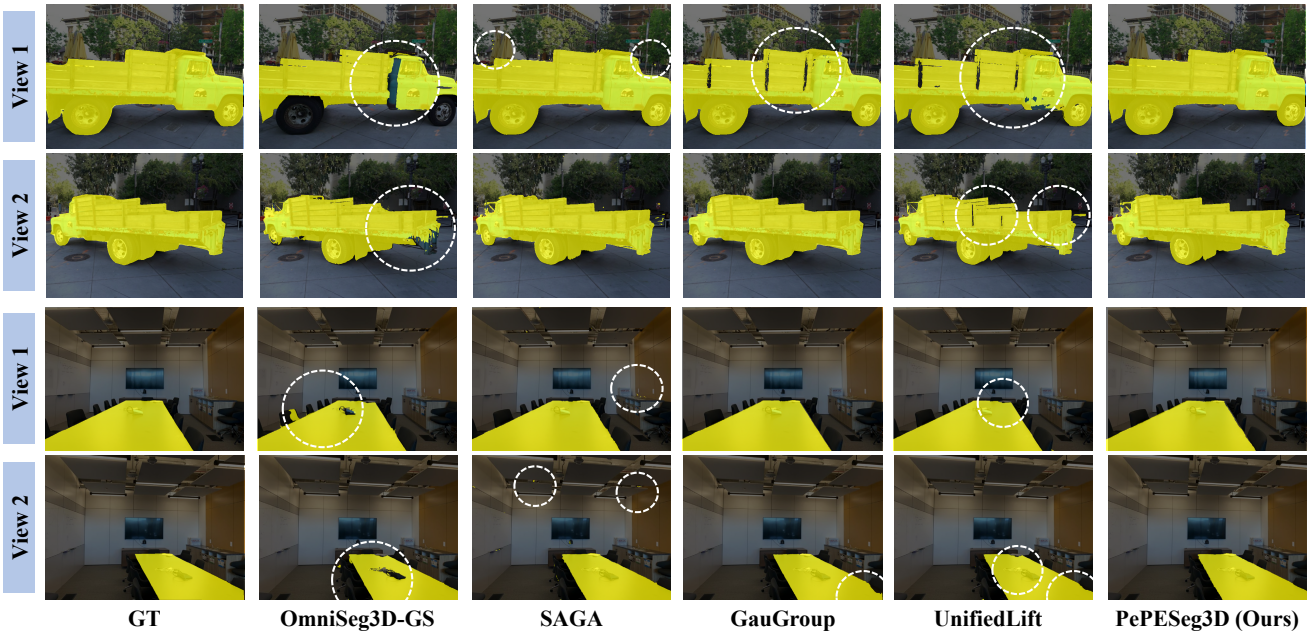

Figure 3. Qualitative comparison of segmentation on SPIn-NeRF. White circles highlight erroneous masks from baseline methods, while our method produces accurate and consistent segmentation across varying viewpoints.

$\mathcal{S}$ and pixel pairs $\mathcal{P}$:

$$\mathcal{L}_m = \mathop{\mathbb{E}}_{s\sim\mathcal{S}}\left[\mathop{\mathbb{E}}_{(p,q)\sim\mathcal{P}}\left[-G^s_{pq}\,C^s_{pq} + \left(1-G^s_{pq}\right)\,\max(C^s_{pq},0)\right]\right]. \tag{3}$$

2.2.1. Perception Contrastive Loss

We adjust feature similarity using photometric and geometric priors. For each sampled scale $s \in \mathcal{S}$, we consider pixel pairs (p, q) within a scale-adaptive window $\|p-q\|_2 \le r(s)$. From the local mean and standard deviation computed over the $r(s)$-neighborhood, we define adaptive thresholds for relative depth and color, denoted by θ_d and θ_c, using the relative depth map D and the training image I, respectively. Based on these thresholds, we define the similar-pair set $\mathcal{P}^s_{\text{sim}}$ and the dissimilar-pair set $\mathcal{P}^s_{\text{dis}}$. A pair (p, q) is assigned to $\mathcal{P}^s_{\text{sim}}$ when both the relative depth difference and the color difference are below θ_d and θ_c, respectively, and to $\mathcal{P}^s_{\text{dis}}$ when both exceed the corresponding thresholds. The perception-guided loss is then written as

$$\mathcal{L}_{\text{p}} = \mathop{\mathbb{E}}_{s\sim\mathcal{S}}\left[\mathop{\mathbb{E}}_{(p,q)\sim\mathcal{P}^s_{\text{dis}}}\left[\max(C^s_{pq},0)\right] - \mathop{\mathbb{E}}_{(p,q)\sim\mathcal{P}^s_{\text{sim}}}\left[C^s_{pq}\right]\right]. \tag{4}$$

2.2.2. View-Consistent Centroid Loss

We enforce scene-level consistency by constructing two centroid sets at extreme scales: the uppermost set $\mathcal{C}_U$ ($s = 1.0$) and the lowermost set $\mathcal{C}_L$ ($s = 0.0$). From the upper-scale gated features $\mathbf{F}^U$, we obtain candidate centroids via HDBSCAN [10] and merge those with cosine similarity above δ using EMA updates to form $\mathcal{C}_U$. Similarly, $\mathcal{C}_L$ is derived from $\mathbf{F}^L$. We denote $\mathcal{C}(p, c)$ as the cosine similarity between the feature at pixel p (at scale s) and centroid c. For each extreme scale $s \in U, L$, the nearest centroid is given by $c^s_* = \arg\max_{c\in\mathcal{C}_s}\mathcal{C}(p, c)$, and is considered only if $\mathcal{C}(p, c^s_*) > \tau$ to avoid mismatches with the current view. For each extreme scale $s \in U, L$, the nearest centroid is given by $c^s_* = \arg\max_{c\in\mathcal{C}_s}\mathcal{C}(p, c)$. The centroid loss is applied only when the similarity exceeds τ:

$$\mathcal{L}_{\text{c}} = \sum_{s\in\{U,L\}}\lambda^s_c \cdot \mathop{\mathbb{E}}_{p\sim I}\left[-\mathbf{1}[\mathcal{C}(p, c^s_*) > \tau]\cdot\mathcal{C}(p, c^s_*)\right], \tag{5}$$

where δ, τ, λ^U_c and λ^L_c are hyperparameters.

2.2.3. Final Objective

The final training objective is a weighted combination of proposed losses. We define the total loss $\mathcal{L}_{\text{total}}$ as follows:

$$\mathcal{L}_{cont} = \lambda_m\mathcal{L}_m + \lambda_p\mathcal{L}_p + \mathcal{L}_c, \tag{6}$$

where λ_m and λ_p balance the contribution of each term.

3. Experiments

3.1. Experimental Settings

3.1.1. Implementation Details

In the PePE reconstruction stage, each scene is trained for 30k iterations, with Gaussian refinement starting at 4k and lasting for twice the number of input views. We set $(\lambda_{close}, \lambda_{ph}, \lambda_d) = (0.1, 0.2, 0.05)$ and adopt the remaining settings from 3DGS. In the contrastive stage, geometry is

Model	mIoU↑ (%)	mAcc↑ (%)	Time
Single-Scale			
GauGroup [15]	84.9	97.5	32 min
UnifiedLift [19]	85.0	97.8	72 min
Multi-Scale			
OmniSeg3D-GS [16]	82.5	97.4	42 min
SAGA [2]	91.9	**98.9**	22 min
PePESeg3D (Ours)	**92.3**	**98.9**	33 min

Table 1. Segmentation results on SPIn-NeRF.

Method	mIoU↑ (%)	mBIoU↑ (%)
Single-Scale		
Panoptic-Lifting-GS [13]	70.7	65.8
GauGroup [15]	72.8	67.6
Gaga [9]	74.7	72.2
UnifiedLift [19]	**80.9**	**77.1**
Multi-Scale		
OmniSeg3D-GS [16]	74.7	71.8
SAGA [2]	78.4	74.0
PePESeg3D (ours)	80.5	76.5

Table 2. Segmentation results on LERF-Mask.

frozen. We optimize 32-dimensional Gaussian features and a two-layer sigmoid scale gate using Adam with a learning rate of 0.0025. Training runs for 10k iterations with batches of 1k sampled pixels and $(\delta, \tau, \lambda_m, \lambda_p, \lambda_c^U, \lambda_c^L) = (0.9, 0.9, 1.0, 0.2, 0.3, 0.1)$. To stabilize training, λ_c^U and λ_c^L are activated after 7k iterations, and centroid sets are refreshed every 200 steps. All experiments are run on a single NVIDIA RTX 4090 GPU.

3.1.2. Datasets

We evaluate PePESeg3D on real-world datasets including SPIn-NeRF [11] and LERF-Mask [15]. For SPIn-NeRF, which provides ground-truth masks across most views, we hold out every eighth image for evaluating segmentation. For LERF-Mask, which provides text-paired masks from coarse objects to fine parts, we use the official splits with 150–250 training views and 3–4 test views to evaluate generalization. For both datasets, we follow the label propagation protocol [15] and report IoU, pixel accuracy (Acc), and Boundary IoU (BIoU). A reference centroid is extracted via HDBSCAN, and corresponding centroids across views are identified using a feature similarity threshold of 0.7. All baselines follow their original evaluation protocols and hyperparameters.

3.2. Segmentation Results

As presented in Table 1, PePESeg3D demonstrates the best performance on the SPIn-NeRF dataset by achieving mIoU of 92.3% and mAcc of 98.9%. This performance corresponds to an improvement of 0.4 percent point(%p) in mIoU over the previous SOTA multi-scale baseline SAGA and a substantial gain of 7.3%p compared to the leading single-scale method UnifiedLift. Qualitatively, as highlighted by the white circles in Figure 3, baseline methods produce erroneous segmentation masks for the desk, whereas our method accurately segments it. Furthermore, under the setting where every eighth image is excluded from training, other methods often exhibit view inconsistency when predicting masks on novel views. In contrast, our method maintains robust label consistency across these unseen viewpoints, enabled by the proposed view consistency centroid loss. This precision is driven by our model's robust understanding of view consistency and multi-granularity semantics, which enables it to effectively capture the entire desk region, including the objects on top.

Method	mIoU (%)	mBIoU (%)
Baseline	78.4	74.0
+ PePE Reconstruction	79.2	75.3
+ Perception Loss	80.2	76.0
+ Consistency (Full Model)	**80.5**	**76.5**

Table 3. Ablation of PePESeg3D on LERF-Mask.

The LERF-Mask benchmark results in Table 2 demonstrate that PePESeg3D establishes state-of-the-art performance among multi-scale methods, surpassing the SAGA by margins of 2.1%p in mIoU and 2.5%p in mBIoU. Notably, it demonstrates performance comparable to UnifiedLift, the state-of-the-art single-scale method. These results indicate that our method mitigates ambiguity in multi-granularity learning by injecting perception priors, enabling precise segmentation across varying granularities.

3.3. Ablation Studies

We conduct ablations to analyze the contribution of each component in PePESeg3D. As shown in Table 3, all components improve segmentation performance. The view-consistency loss improves mIoU by 0.3%p, promoting consistent predictions across views. The perception loss yields a gain of 1.0%p, demonstrating the effectiveness of perception priors for feature learning. PePE reconstruction improves performance by 0.8%p, indicating the importance of accurate geometric structure for multi-scale segmentation.

4. Conclusion

We present PePESeg3D, a multi-scale segmentation framework that bridges geometry and semantics by integrating perception priors, and effectively addresses the inconsistencies of 2D foundation model masks while achieving view-consistent and scale-robust segmentation. PePE Reconstruction aligns geometry using monocular depth and Gaussian refinement, and PePE Contrastive Learning enables effective multi-scale feature learning through scale-aware, perception-guided, and view-consistent objectives. Extensive experiments demonstrate that PePESeg3D achieves strong performance in segmentation, and that each components contributes meaningfully.

References

[1] Jiazhong Cen, Zanwei Zhou, Jiemin Fang, Chen Yang, Wei Shen, Lingxi Xie, Dongsheng Jiang, Xiaopeng Zhang, and Qi Tian. Segment anything in 3d with nerfs. In *NeurIPS*, 2023. 1

[2] Jiazhong Cen, Jiemin Fang, Chen Yang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Segment any 3d gaussians. In *Proceedings of the Thirty-Ninth AAAI Conference on Artificial Intelligence and Thirty-Seventh Conference on Innovative Applications of Artificial Intelligence and Fifteenth Symposium on Educational Advances in Artificial Intelligence*. AAAI Press, 2025. 1, 2, 4

[3] Qiao Gu, Zhaoyang Lv, Duncan Frost, Simon Green, Julian Straub, and Chris Sweeney. Egolifter: Open-world 3d segmentation for egocentric perception. In *European conference on computer vision*, pages 382–400. Springer, 2024. 1

[4] Xu Hu, Yuxi Wang, Lue Fan, Junsong Fan, Junran Peng, Zhen Lei, Qing Li, and Zhaoxiang Zhang. Semantic anything in 3d gaussians. *arXiv preprint arXiv:2401.17857*, 2024. 2

[5] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42 (4), 2023. 2

[6] Chung Min* Kim, Mingxuan* Wu, Justin* Kerr, Matthew Tancik, Ken Goldberg, and Angjoo Kanazawa. Garfield: Group anything with radiance fields. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 2

[7] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv:2304.02643*, 2023. 1, 2

[8] Yichen Liu, Benran Hu, Chi-Keung Tang, and Yu-Wing Tai. Sanerf-hq: Segment anything for nerf in high quality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3216–3226, 2024. 2

[9] Weijie Lyu, Xueting Li, Abhijit Kundu, Yi-Hsuan Tsai, and Ming-Hsuan Yang. Gaga: Group any gaussians via 3d-aware memory bank, 2024. 4

[10] Leland McInnes, John Healy, and Steve Astels. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205, 2017. 3

[11] Ashkan Mirzaei, Tristan Aumentado-Armstrong, Konstantinos G. Derpanis, Jonathan Kelly, Marcus A. Brubaker, Igor Gilitschenski, and Alex Levinshtein. SPIn-NeRF: Multiview segmentation and perceptual inpainting with neural radiance fields. In *CVPR*, 2023. 4

[12] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3), 2022. 2

[13] Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulò, Norman Müller, Matthias Nießner, Angela Dai, and Peter Kontschieder. Panoptic lifting for 3d scene understanding with neural fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9043–9052, 2023. 4

[14] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024. 2

[15] Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3d scenes. In *ECCV*, 2024. 1, 4

[16] Haiyang Ying, Yixuan Yin, Jinzhi Zhang, Fan Wang, Tao Yu, Ruqi Huang, and Lu Fang. Omniseg3d: Omniversal 3d segmentation via hierarchical contrastive learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20612–20622, 2024. 1, 4

[17] Hongjia Zhai, Hai Li, Zhenzhe Li, Xiaokun Pan, Yijia He, and Guofeng Zhang. PanoGS: Gaussian-based panoptic segmentation for 3d open vocabulary scene understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025. 1

[18] Jiaxin Zhang, Junjun Jiang, Youyu Chen, Kui Jiang, and Xianming Liu. Cob-gs: Clear object boundaries in 3dgs segmentation based on boundary-adaptive gaussian splitting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19335–19344, 2025.

[19] Runsong Zhu, Shi Qiu, Zhengzhe Liu, Ka-Hei Hui, Qianyi Wu, Pheng-Ann Heng, and Chi-Wing Fu. Rethinking end-to-end 2d to 3d scene segmentation in gaussian splatting. *arXiv preprint arXiv:2503.14029*, 2025. 1, 4

PePESeg3D: Perception Prior Enhances Multi-Scale Segmentation for 3D Gaussian Splatting

Supplementary Material

5. Additional Implementation Details for PePE Contrastive Learning

5.1. Sampling Strategy

While we primarily employ uniform for query scales and pixel pairs, we introduce a hard example mining strategy to enhance discriminability in ambiguous regions. We identify hard sample pairs as those that satisfy the grouping condition solely based on the active label at scale s, without being co-located in any finer-grained mask. Formally, a hard sample pair (p, q) satisfies $\Lambda(s, p) = \Lambda(s, q)$ and does not belong to any common finer mask ($\nexists k : p, q \in M_k \wedge s_{M_k} < s$). These pairs represent cases where the grouping changes according to the scale, making them more challenging to learn and crucial for capturing multi-scale semantics. To address this, we augment the standard uniform sampling by explicitly oversampling these hard pairs within each training batch. Regarding the scale-aware contrastive loss (Equation (3)), we exclude pixels that are not covered by any mask instance from the sampling process. This constraint ensures that the model focuses on learning valid mask correspondences and prevents erroneous signals arising from ambiguous regions where no mask definitions exist.

5.2. Reweighting Strategy

To appropriately balance the contribution of masks during contrastive learning, we adopt the reweighting strategy from [1]. Since pixels belonging to large masks are more frequently sampled, they tend to dominate the optimization process. To mitigate this, we define a pixel-wise weight $\omega(p)$ as the inverse of the average area of masks containing pixel p:

$$\omega(p) = \Big(\frac{1}{|\mathcal{M}_p|} \sum_{M \in \mathcal{M}_p} |M|\Big)^{-1}, \tag{7}$$

where $\mathcal{M}_p = \{M \in \mathcal{M} \mid M(p) = 1\}$ denotes the set of masks covering pixel p, and $|M|$ represents the mask area defined as the number of pixels. For a sampled pixel pair (p, q), the final balancing weight is given by $\omega(p) \cdot \omega(q)$. To stabilize training, all weights are min-max normalized to the range $[1, 10]$ in each training iteration. This weighting scheme ensures that pixel pairs from small masks are not underrepresented during contrastive optimization. By applying this weighting term, the scale-aware contrastive loss is formally defined as:

$$\mathcal{L}_M = \mathop{\mathbb{E}}_{s \sim \mathcal{S}} \left[\mathop{\mathbb{E}}_{(p,q) \sim \mathcal{P}} w_{pq} \Big(- G^{(s)}_{pq}\, C^{(s)}_{pq} + \big(1 - G^{(s)}_{pq}\big)\, [C^{(s)}_{pq}]_+ \Big) \right], \tag{8}$$

where $[x]_+ = \max(x, 0)$.

5.3. Regularization

To promote local spatial consistency and mitigate feature noise in 3D space, we first apply feature smoothing by averaging the features of the 16 nearest-neighbor Gaussians before rendering. Subsequently, we address the stability of the rendered 2D representations. Without proper control, a few 3D features with excessively large norms can dominate the rendering regardless of spatial relevance, biasing the 2D representation. For stable contrastive learning, all rendered features should reside in a similar hypersphere. We regularize rendered features toward unit norm,

$$\mathcal{L}_{\text{2D}} = \frac{1}{N_p} \sum_{p \in N_p} (\|\mathbf{F}(p)\|_2 - 1)^2. \tag{9}$$

This regularization term $\mathcal{L}_{\text{2D}}$ is added to the objective function (Equation (6)).

6. More Qualitative Results

We provide additional qualitative comparisons to further validate the effectiveness of PePESeg3D. Figure 4 presents automatic scene decomposition results across three distinct scale levels. For the automatic scene decomposition task, we apply HDBSCAN clustering to the 2D rendered features and assign a random color to each label. The results demonstrate appropriate scale-dependent granularity while maintaining robust view consistency across different perspectives.

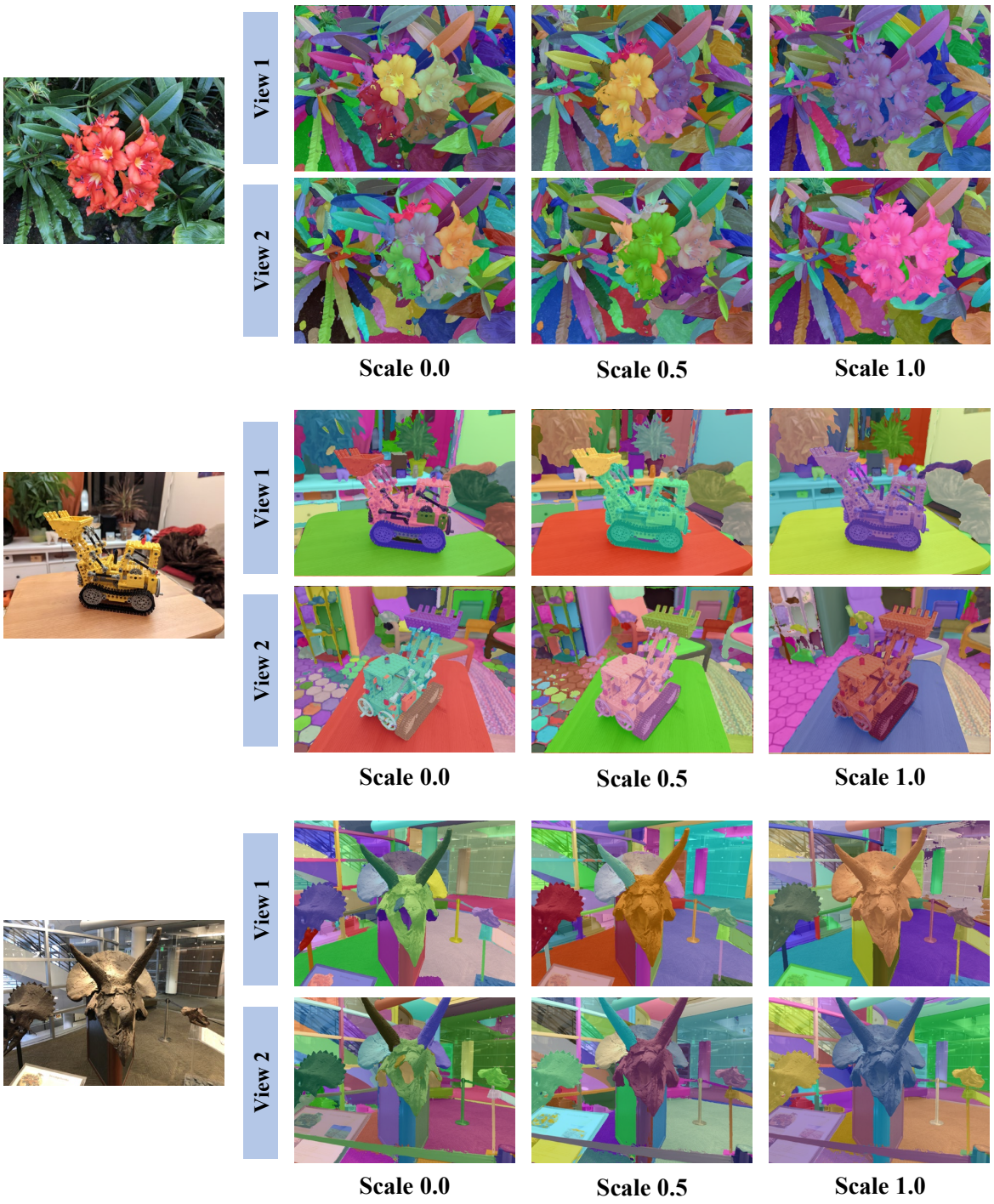

Figure 4. Automatic Decomposition Visualization on SPIn-NeRF generated by PePESeg3D. Random colors are assigned to semantic clusters derived via HDBSCAN.