SPLAT IN PLACE: POINT PINNED GAUSSIANS FOR POINT CLOUD SEGMENTATION

Anonymous authors
Paper under double-blind review

ABSTRACT

3D Gaussian Splatting (3DGS) provides rich geometric and appearance cues for point cloud segmentation. However, existing Gaussian-based methods rely on rendering-oriented primitives and establish point-Gaussian correspondence only after reconstruction, causing misaligned semantic evidence and limiting the use of Gaussian anisotropic support for neighborhood reasoning. To address these limitations, we introduce Splat In Place (SIP), a segmentation-oriented reconstruction framework that establishes point-Gaussian correspondence during reconstruction. SIP proposes Point Pinned Reconstruction, which initializes one Gaussian per point and preserves its identity, enabling direct access to point-aligned Gaussian attributes. Based on this representation, Joint Support Aggregation (JSA) exploits Gaussian support as a structural metric for adaptive neighbor aggregation, while Addressable Gaussian Recurrent Field (AGRF) further refines point features with Gaussian-aware relations. Experiments on ScanNet, S3DIS, and KITTI-360 demonstrate that SIP improves segmentation performance over existing methods, especially under sparse input conditions, validating the effectiveness of point-indexed Gaussian representations for 3D semantic understanding.

1 INTRODUCTION

Point cloud segmentation assigns a semantic label to each 3D point and underpins a wide range of 3D applications, such as autonomous driving (Liao et al., 2023), indoor scene parsing (Dai et al., 2017; Armeni et al., 2016), and robotic perception (Deng et al., 2025). Despite the steady progress, a point cloud offers only sparse geometric evidence, recording where surfaces are sampled but saying little about the local structure and appearance. 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023) supplies precisely what a point lacks with a set of anisotropic Gaussian ellipsoids, each describing the spatial extent, orientation, and appearance of the region it covers. Therefore, recent studies increasingly adopt 3DGS as a source of evidence for point cloud segmentation (Song et al., 2026a;b; Li et al., 2026; Wang et al., 2025).

Applying 3DGS to segmentation requires connecting rendering primitives back to the points that receive labels. Photometric reconstruction can clone, split, prune, or displace Gaussians, leaving evidence indexed by primitives rather than by input points. Existing methods add 2D semantic cues to reconstructed primitives (Zhou et al., 2024; Qin et al., 2024; Ye et al., 2024; Zhai et al., 2025; Li et al., 2025a) or transfer Gaussian attributes to the original points (Song et al., 2026b;a). However, establishing the point-Gaussian relation outside reconstruction gives rise to two fundamental limitations, as illustrated in Figure 1.

1) Ownership is estimated rather than known. Rendering optimization leaves no record of which primitive arose from which point, so the relation between a point and its evidence can only be recovered after reconstruction. As shown in Figure 1(a), a single point can draw on primitives from two distinct structures at once.

2) Anisotropic support is treated as a feature. Once ownership has been assigned, the Gaussian attributes are combined with the point and enter the network as extra input channels. Specifically, a covariance describes the surface around a point rather than the point itself, so as a channel it changes what is computed for that point but never which neighbors it aggregates from. Nearby points from different surfaces may then contribute to the same aggregation, especially where sampling is sparse.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

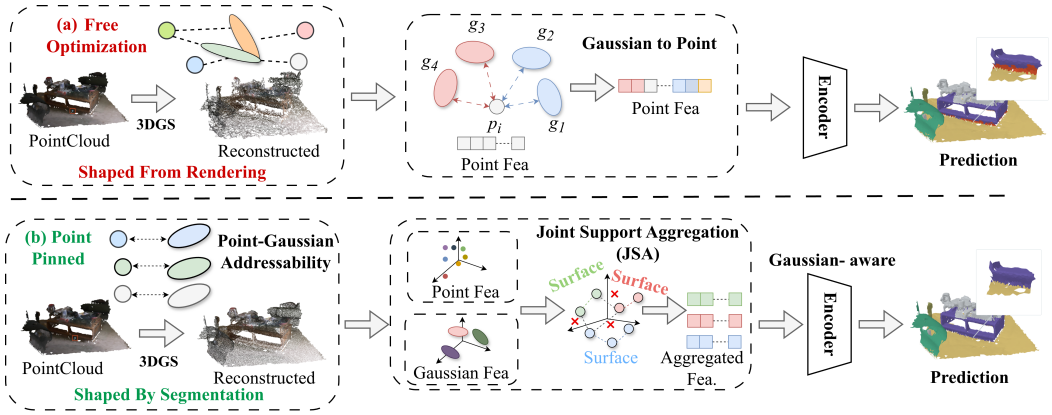

Figure 1: **From rendering-oriented reconstruction to point-pinned Gaussian representation for segmentation.** **(a)** Free Optimization shapes Gaussian primitives for rendering, while their correspondence to the original points must be recovered after reconstruction. **(b)** Point Pinned Reconstruction preserves the source identity of each Gaussian throughout reconstruction, thereby establishing Point Gaussian Addressability. The resulting point-indexed Gaussian representation provides directly accessible anisotropic support for subsequent segmentation reasoning.

We argue that *Gaussians optimized for rendering are not yet Gaussians prepared for segmentation*. **Splat In Place (SIP)** establishes the point Gaussian relation during reconstruction. Point Pinned Reconstruction (PPR) initializes one Gaussian at each input point and keeps its center and source index fixed while multiview images fit the remaining attributes. Every point can then retrieve its Gaussian by index, a property we call *Point Gaussian Addressability*. Point kNN supplies candidate neighbors. Joint Support Aggregation (JSA) compares the Gaussian supports at both ends of each pair to estimate whether the points lie on the same surface and weights their interaction accordingly. After the backbone forms scene features, the Addressable Gaussian Recurrent Field (AGRF) uses these relations to refine them. Point labels supervise the network and update the reconstructed Gaussian attributes.

The main contributions of this paper are as follows.

- ✧ We present **Splat In Place (SIP)**, which reconstructs Gaussians in place on the points to be labeled, so that every point owns exactly one indexed Gaussian by construction, a property we term Point Gaussian Addressability.
- ✧ We propose Joint Support Aggregation, which turns the anisotropic supports of two neighboring points into the metric that governs local aggregation, and the Addressable Gaussian Recurrent Field, consulting the same relation once contextual features have been formed.
- ✧ We show on ScanNet, S3DIS, and KITTI-360 that SIP, trained end to end under point supervision alone, improves indoor and outdoor segmentation and achieves the best performance even at the sparsest input setting in each dataset.

2 Related Work

Point Cloud Segmentation and 3D Representation Learning. Point cloud segmentation has advanced through local point modeling (Qi et al., 2017a;b), point convolutions (Choy et al., 2019; Thomas et al., 2019; Qian et al., 2022; Lin et al., 2023; Deng et al., 2023), and Transformer architectures (Lai et al., 2023; Liu et al., 2023; Wang, 2023; Yang et al., 2025; Wu et al., 2024a). Pretraining (Yan et al., 2024; Zheng et al., 2024; Wu et al., 2025), multimodal learning (Zhang et al., 2025; Wang et al., 2025), and shared encoders (Zhang et al., 2026) further improve point features. Image and Gaussian cues must reach the points being labeled.

Gaussian Representations for Point Segmentation. Beyond novel view synthesis, 3D Gaussian Splatting (Kerbl et al., 2023) has been used for semantic, instance, panoptic, and open vocabulary understanding, including point cloud segmentation (Song et al., 2026b;a). Methods that attach 2D features or masks to Gaussians predict in Gaussian space (Zhou et al., 2024; Qin et al., 2024; Ye

Table 1: Comparison of representative Gaussian based methods for point cloud understanding. Gaussian counts are measured on the randomly selected ScanNet scene `scene0062_00`. SceneSplat denotes methods that directly use pretrained Gaussian scene assets (Li et al., 2025b). The compared methods use 2D priors, a separate transfer of Gaussian information to points, or both. SIP pairs each point with its Gaussian during reconstruction.

Method	Gaussian Construction	2D Prior	Point Transfer	# Gaussians
PanoGS (Zhai et al., 2025)	Free Optimization	✓	✗	51,610
InstanceGaussian (Li et al., 2025a)	Anchor Based	✓	✗	258,050
Chorus (Li et al., 2026)	SceneSplat (Li et al., 2025b)	✓	✗	1,499,999
PointGS (Song et al., 2026b)	Free Optimization	✓	✓	693,096
G2P (Song et al., 2026a)	SceneSplat (Li et al., 2025b)	✗	✓	1,499,999
Ours	**Point Pinned**	✗	✗	**51,610**

et al., 2024; Li et al., 2025a; Zhai et al., 2025; Li et al., 2026; Cen et al., 2025; Wu et al., 2024b; Li et al., 2025b), so disagreement across views can enter the 3D result. PointGS transfers reconstructed attributes to points by nearest neighbor association (Song et al., 2026b); G2P uses covariance based matching before distillation into a point backbone (Song et al., 2026a). Both form the point-Gaussian relation after reconstruction, when rendering has already shaped the primitives. SIP instead keeps the source point index during reconstruction and uses Gaussian spatial support to weigh local point interactions, without any 2D prior or a separate transfer step (Table 1).

3 Method

3.1 Overview

Given an input point cloud $\mathcal{P} = \{(p_i, c_i^P, n_i)\}_{i=1}^N$ and calibrated RGB views, Splat In Place (SIP) predicts a label for every input point. Here p_i, c_i^P, and n_i are its position, color, and normal. SIP has three stages (Figure 2). Point Pinned Reconstruction (PPR) fits one Gaussian at p_i while keeping its source index. Joint Support Aggregation (JSA) uses the fitted Gaussians to weight interactions between neighboring points. A point backbone processes the aggregated features. The Addressable Gaussian Recurrent Field (AGRF) then refines them before prediction.

3.2 Point Pinned Reconstruction

Segmentation labels belong to the input points, so Gaussian reconstruction keeps the same point indices. For each observation $x_i^P = [p_i, c_i^P, n_i]$, PPR initializes one Gaussian with its center at p_i and enforces

$$\begin{aligned} &\mu_i = p_i, \qquad \mathrm{id}(g_i) = \mathrm{id}(p_i), \qquad |\mathcal{G}| = N, \\ &g_i = (\mu_i, \bar{s}_i, \bar{q}_i, \bar{\alpha}_i, H_i). \end{aligned} \tag{1}$$

Here μ_i, $\bar{s}_i$, $\bar{q}_i$, $\bar{\alpha}_i$, and H_i denote the Gaussian center, scale, rotation, opacity, and spherical harmonic coefficients. The center and index remain fixed while multiview RGB reconstruction fits the other attributes. We denote the reconstructed attributes by $\bar{\gamma}_i = (H_i, \bar{s}_i, \bar{q}_i, \bar{\alpha}_i)$. The scale and rotation define the covariance $\bar{\Sigma}_i = R(\bar{q}_i)\,\mathrm{diag}(\bar{s}_i^2)R(\bar{q}_i)^\top$. Because p_i and g_i retain the same index, we can retrieve $\bar{\gamma}_i$ for p_i directly: $p_i \xrightarrow{\text{index } i} \bar{\gamma}_i$. We call this *Point Gaussian Addressability*.

RGB views fit each Gaussian for reconstruction, before any point label is used. During segmentation training, the point-level loss directly updates the reconstructed scale, rotation, opacity, and spherical harmonic coefficients. We write the current scale, rotation, and opacity as s_i, q_i, and α_i, while the center and point index remain fixed. The base DC color and channelwise spherical harmonic energy of H_i form a_i^G, summarizing fitted appearance across views. These attributes define $\Sigma_i = R(q_i)\,\mathrm{diag}(s_i^2)R(q_i)^\top$, $x_i^G = [a_i^G, s_i, \alpha_i]$, and $u_i = \phi_{\text{node}}([x_i^P; x_i^G])$. This gives each point an initial feature. JSA then uses Σ_i to weight the neighboring features. Gaussian attributes thus enter both the point feature and the relations between points.

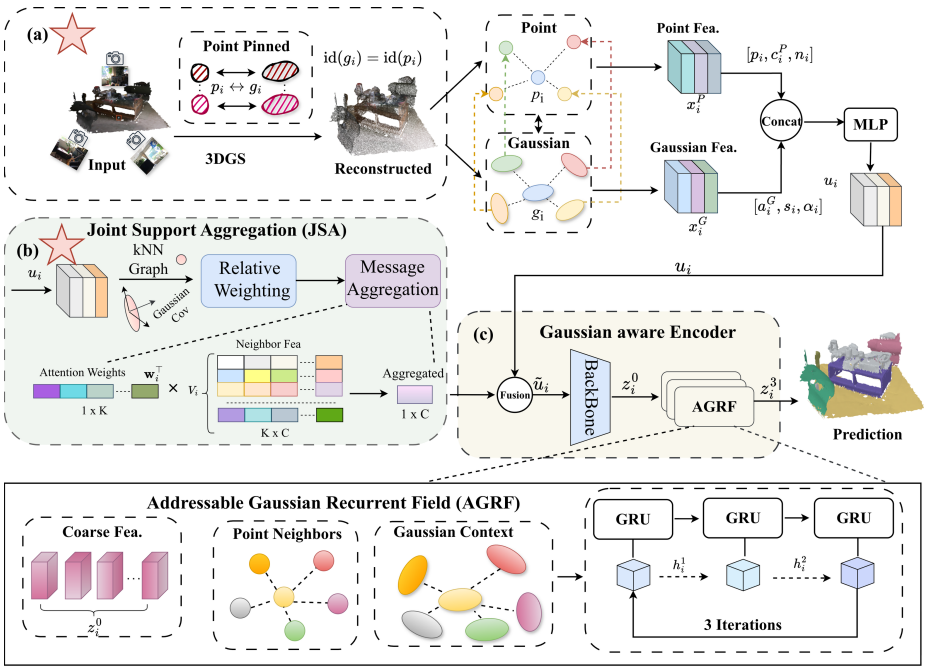

Figure 2: **Overview of Splat In Place (SIP). (a) Point Pinned Reconstruction** fits Gaussian g_i at point p_i and keeps their indices paired, so Gaussian attributes can be read directly at each point to form u_i. **(b) Joint Support Aggregation (JSA)** takes candidate neighbors from point kNN and uses the two Gaussians in each pair to weight their messages. **(c) Gaussian aware Encoder** combines these messages with u_i. A point backbone extracts scene features for three AGRF updates.

3.3 Joint Support Aggregation

JSA receives the point and Gaussian feature u_i and spatial support Σ_i. Nearby points can belong to different surfaces, so JSA estimates whether a neighbor lies on the same local surface as the center point before aggregating its feature. Point kNN provides K candidate neighbors for each point. For $j \in \mathcal{N}(i)$, let $d_{ij} = p_j - p_i$. The covariances of their Gaussians define

$$\Sigma_{ij}^J = \frac{\Sigma_i + \Sigma_j}{2} + \epsilon I, \qquad D_{ij}^G = d_{ij}^\top \left(\Sigma_{ij}^J\right)^{-1} d_{ij}. \tag{2}$$

The small ϵI term keeps the joint covariance invertible. At the same Euclidean distance, a neighbor along the broad direction of the paired Gaussians has a smaller D_{ij}^G than one across their narrow direction. Thus a neighbor across a narrow structure can receive less support than one farther along it, even when both are selected by Euclidean kNN. We combine this distance with d_{ij}, point distance relative to local spacing $\delta_{ij}^{\mathrm{rel}}$, relative Gaussian volume Δv_{ij}, differences in point and Gaussian appearance Δc_{ij}^P and Δc_{ij}^G, and opacity difference $\Delta \alpha_{ij}$:

$$e_{ij} = \left[d_{ij}, \delta_{ij}^{\mathrm{rel}}, D_{ij}^G, \Delta v_{ij}, \Delta c_{ij}^P, \Delta c_{ij}^G, \Delta \alpha_{ij}\right]. \tag{3}$$

The volume term compares the extent of the two Gaussians; the color terms keep observed point color separate from appearance fitted across views. We use e_{ij} to score and aggregate the candidate neighbors:

$$\begin{aligned} r_{ij} &= \frac{Q(u_i)^\top K(u_j)}{\sqrt{d_f}} + \phi_{\mathrm{rel}}(e_{ij}) - \frac{1}{2}\log\left(1 + D_{ij}^G\right), \\ w_{ij} &= \mathrm{softmax}_{j \in \mathcal{N}(i)}(r_{ij}), \qquad m_i = \sum_{j \in \mathcal{N}(i)} w_{ij} V(u_j). \end{aligned} \tag{4}$$

d_f is the feature dimension. $Q(\cdot)$, $K(\cdot)$, and $V(\cdot)$ are learned maps of the point features, while ϕ_{rel} maps e_{ij} to a score. Feature similarity and the relative terms compare the two points; the Gaussian distance reduces the score for displacement across their spatial support. Together, these cues estimate whether j belongs to the same local surface as i. The softmax compares all candidate neighbors of point i, and their weighted features form m_i.

3.4 Gaussian Aware Encoder

JSA gives each point a weighted message from its neighbors. We add this message to the point feature through a learned residual: $\tilde{u}_i = u_i + \phi_{\text{fuse}}([u_i; m_i])$. A point backbone processes the fused features. By default, we use the first three encoder stages of Utonia (Zhang et al., 2026) and write the feature associated with p_i as $z_i^0 = \text{Backbone}(\{\tilde{u}_i\}_{i=1}^N)_i$. Other backbones are compared in Appendix A.2. The backbone gives each point information from a wider part of the scene. AGRF then scores the same point neighbors using this context, which can distinguish surfaces with similar local appearance or geometry. The learned scorer ϕ_{task} compares each pair through z_i^0, z_j^0, and the pair descriptor e_{ij}:

$$a_{ij} = \text{softmax}_{j \in \mathcal{N}(i)} \left[\phi_{\text{task}} \left(z_i^0, z_j^0, e_{ij} \right) \right]. \tag{5}$$

We compute a_{ij} once and use the same weights for three recurrent iterations. Starting from $h_i^0 = \phi_{\text{init}}(z_i^0)$, AGRF updates each point as follows:

$$\begin{aligned} m_i^t &= \sum_{j \in \mathcal{N}(i)} a_{ij} V_h(h_j^{t-1}), \\ h_i^t &= \text{GRU}\left(h_i^{t-1}, \phi_h(z_i^0, m_i^t)\right), \\ z_i^t &= z_i^0 + \lambda \tanh(W_o h_i^t), \qquad t = 1, 2, 3. \end{aligned} \tag{6}$$

At each iteration, a point receives a weighted message from its neighbors and updates its hidden state with a gated recurrent unit (GRU). A bounded correction is then added to z_i^0, with $\lambda \tanh(\cdot)$ controlling its magnitude. The backbone feature remains the reference as the hidden state accumulates neighborhood information. The semantic head predicts a label for each input point from z_i^3. The predictions from z_i^1 and z_i^2 are also supervised during training. Cross-entropy and Dice losses supervise all three iterations and update the Gaussian attributes, node encoder, JSA, backbone, and AGRF. Further implementation details are in Appendix A.

4 Experiments

Datasets and evaluation. We evaluate point segmentation on ScanNet v2 (Dai et al., 2017), S3DIS (Armeni et al., 2016), and KITTI-360 (Liao et al., 2023). On ScanNet, we train with all 1,201 training scenes and report validation mIoU on the 312 validation scenes and 20 semantic classes. Inspired by PointGS (Song et al., 2026b), we generate RGB views for S3DIS by projecting the colored point clouds. We train with ten randomly selected rooms outside Area 5 and evaluate on Area 5. For KITTI-360, we randomly select ten accumulated point-cloud windows from the official 3D training split and evaluate on the official validation split, using calibrated camera views. For S3DIS and KITTI-360, all compared methods are retrained on the same selected rooms or windows. To examine input density, we subsample the original points before reconstruction and fit Gaussians for each retained set. We rebuild the neighborhoods and evaluate on the retained points. For ScanNet and S3DIS, High, Medium, and Low denote 75%, 50%, and 25% point retention. These columns average IoU only over the selected challenging classes.

Implementation details. We reconstruct one Gaussian per input point, keeping its center and source index fixed while fitting its appearance and spatial support to RGB views. RGB fitting is based on the reconstruction setup used in SAGA (Cen et al., 2025). The segmentation model uses the first three Utonia encoder stages, initialized at random, and three AGRF updates. We train the segmentation network with AdamW at a learning rate of 10^{-5}. Gaussian scale, rotation, opacity, and spherical harmonic coefficients receive the point-level loss at a learning rate of 3×10^{-5}. At evaluation, each scene is reconstructed from its available views and processed without segmentation updates.

4.1 Segmentation Results

Semantic segmentation. ScanNet tests point segmentation when captured RGB views are available for Gaussian reconstruction. We compare SIP with 2D-to-point methods, point encoders, Gaussian segmentation methods, and G2P's transfer of Gaussian information to points. Alongside whole-scene mIoU, we measure classification near class boundaries and the alignment of predicted boundaries with ground truth. SIP reaches 78.7 mIoU on the full validation input, 0.8 above G2P (Table 2).

Table 2: **Semantic segmentation on ScanNet.** Full-input mIoU covers 20 classes; B-mIoU and B-F denote Boundary mIoU and Boundary F-score. High, Medium, and Low average six challenging classes at 75%, 50%, and 25% point retention, respectively.

Method	Full input			Challenging classes		
	mIoU	B-mIoU	B-F	High	Medium	Low
2D / Cross modal						
VMVF (Kundu et al., 2020)	72.2	32.2	31.1	43.4	42.8	40.1
BPNet (Hu et al., 2021)	70.7	34.2	32.1	46.2	41.4	45.2
UniPre3D (Wang et al., 2025)	75.6	36.8	39.2	52.9	54.8	51.3
Geometric						
MinkUNet (Choy et al., 2019)	68.9	32.5	29.5	40.2	45.9	47.8
Stratified Transformer (Lai et al., 2022)	72.2	31.8	29.9	50.9	52.5	54.2
PTv3 (Wu et al., 2024a)	75.1	36.2	38.1	47.7	52.0	49.1
Swin3D (Yang et al., 2025)	74.5	29.8	37.7	51.8	46.5	44.3
OctFormer (Wang, 2023)	70.6	34.7	36.9	45.1	49.3	42.1
Gaussian based						
PanoGS (Zhai et al., 2025)	72.4	35.3	37.8	53.2	53.6	44.5
PointGS (Song et al., 2026b)	69.8	30.2	36.4	45.5	50.6	50.7
Chorus (Decoder) (Li et al., 2026)	69.4	32.3	38.2	48.4	47.1	43.6
Chorus (FT) (Li et al., 2026)	74.1	35.9	40.2	**57.2**	51.3	53.7
Point Gaussian						
G2P (Song et al., 2026a)	77.9	36.8	39.9	52.8	53.5	48.8
Ours	**78.7**	**37.2**	**40.8**	51.7	**55.8**	**58.1**

The boundary columns of Table 2 test whether the overall gain extends to class interfaces. SIP's 37.2 Boundary mIoU, against 36.8 for the nearest baselines, indicates fewer classification errors around ground-truth class interfaces. Its Boundary F-score of 40.8, versus 40.2 for Chorus (FT), indicates that the predicted interfaces lie closer to ground truth. The challenging-class results expose a second difference: Chorus (FT) leads at 75% point retention, whereas SIP leads at 50% and 25%. At the sparsest level it scores 58.1 against the next best 54.2, consistent with joint Gaussian support helping when fewer points define a neighborhood.

S3DIS tests the representation when RGB views are projected from the colored points. We retrain the point and Gaussian baselines on the same ten rooms. On Area 5 (Table 3), SIP reaches 60.8 mIoU, 3.2 above Chorus (FT). Its 38.9 Boundary mIoU exceeds the nearest baseline by 1.0 point, showing more correct labels near class interfaces; its 41.8 Boundary F-score exceeds Chorus (FT) by 1.3 points, showing closer alignment of those interfaces. SIP also leads on the challenging classes at every retention level. At 25% retention, it scores 65.4 against 60.2 for Chorus (FT), and the three-level mean is 65.5 against 59.1. Gaussian appearance and spatial support remain useful even when the reconstruction views are generated from the points themselves.

Table 3: **Semantic segmentation on S3DIS Area 5.** Full-input mIoU covers 13 classes; B-mIoU and B-F denote Boundary mIoU and Boundary F-score. High, Medium, and Low average seven challenging classes at 75%, 50%, and 25% point retention; Mean averages these three scores.

Method	Full input			Challenging classes			
	mIoU	B-mIoU	B-F	High	Medium	Low	Mean
PanoGS (Zhai et al., 2025)	47.7	36.2	35.4	42.0	42.0	40.8	41.6
PointGS (Song et al., 2026b)	49.6	35.9	36.7	47.6	49.7	46.2	47.9
UniPre3D (Wang et al., 2025)	52.7	37.9	39.5	57.3	55.0	51.3	54.5
G2P (Song et al., 2026a)	49.2	37.2	40.2	55.5	55.3	55.6	55.4
Chorus (Decoder) (Li et al., 2026)	42.4	32.1	37.2	51.1	51.4	50.9	51.1
Chorus (FT) (Li et al., 2026)	57.6	37.4	40.5	61.2	56.1	60.2	59.1
Ours	**60.8**	**38.9**	**41.8**	**65.5**	**65.6**	**65.4**	**65.5**

KITTI-360 tests whether sparse outdoor input changes this ranking. We retrain G2P and both Chorus variants on the same ten windows. On the official validation split in Table 4, SIP obtains 57.16 mIoU against 50.4 for Chorus (FT).

After 75% of points are removed before reconstruction, SIP retains 53.1, exceeding Chorus (FT) by 5.3 points and G2P by 12.7. The gain extends the sparse-input result to roads, vegetation, and distant structures with uneven point sampling.

Instance segmentation. To test whether the learned point features also support grouping, we feed them to a parallel instance mask branch. We compare with PanoGS, G2P, Chorus, and InstanceGaussian (Li et al., 2025a) under one matched-mask mIoU rule in Table 5. SIP ranks first on ScanNet at 56.1, a 1.7-point gain over InstanceGaussian. On S3DIS Area 5 it reaches 54.7, below InstanceGaussian's 56.6 but above the other baselines. This downstream comparison shows that the representation can support instance masks as well as semantic labels. Branch and metric details are given in Appendix B.2.

Table 4: **KITTI-360 semantic segmentation.** All-class mIoU; 50% and 75% denote points removed before reconstruction.

Method	Full	50%	75%
Chorus (Decoder)	41.7	40.1	38.4
G2P	43.2	41.8	40.4
Chorus (FT)	50.4	49.6	47.8
Ours	**57.16**	**55.2**	**53.1**

Table 5: **Indoor instance segmentation.** Matched-mask mIoU.

Method	ScanNet	S3DIS
PanoGS	45.8	47.7
G2P	47.8	49.4
Chorus (Decoder)	42.0	44.3
Chorus (FT)	53.1	51.8
InstanceGaussian	54.4	**56.6**
Ours	**56.1**	54.7

4.2 Feature and Scene Analysis

The KITTI-360 examples make the aggregate result visible at class interfaces. In the first row of Figure 3, both baselines assign other classes to the red-boxed vegetation patch, while SIP recovers the ground-truth class. The lower rows focus on road, sidewalk, and terrain regions: G2P scatters labels through them, whereas SIP follows the larger GT regions and their boundaries more closely.

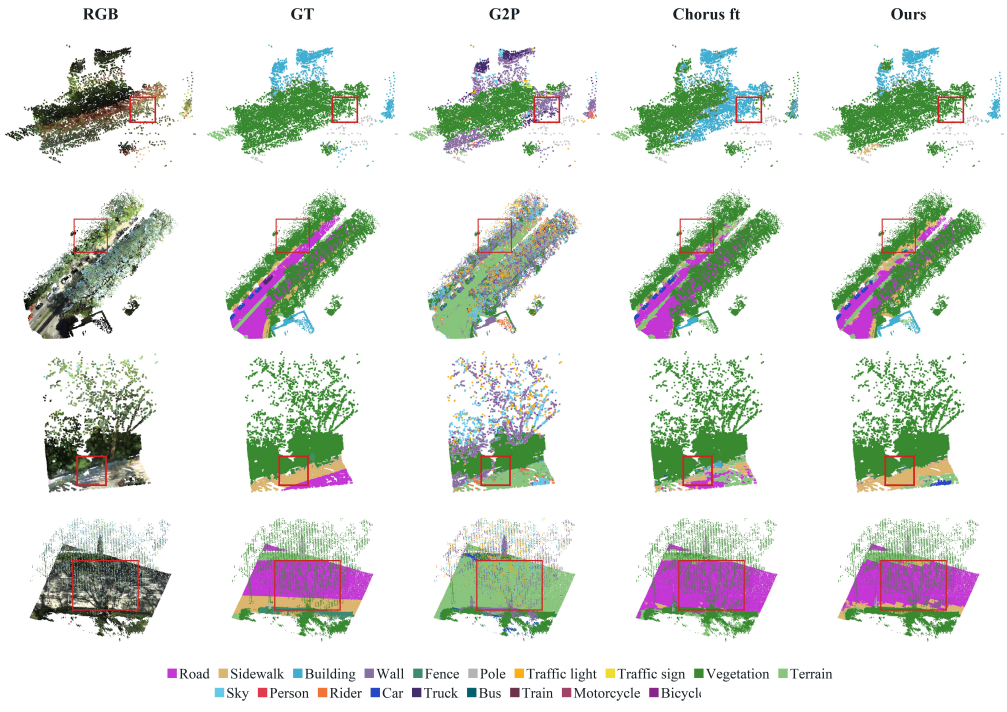

Figure 3: **KITTI-360 qualitative results.** Rows show four scenes; columns show RGB, ground truth, G2P (Song et al., 2026a), Chorus (FT) (Li et al., 2026), and SIP. Red boxes mark areas.

The ScanNet feature maps and confusion matrices trace this distinction from representation to prediction. Within each independently fitted PCA map in Figure 4, SIP's pre-classification features vary more coherently over the scene's structures than those of G2P and Chorus (FT). For the same points, its confusion matrix shows fewer door and window points assigned to wall. The spatial organization of the features is thus reflected in more accurate decisions between adjacent classes.

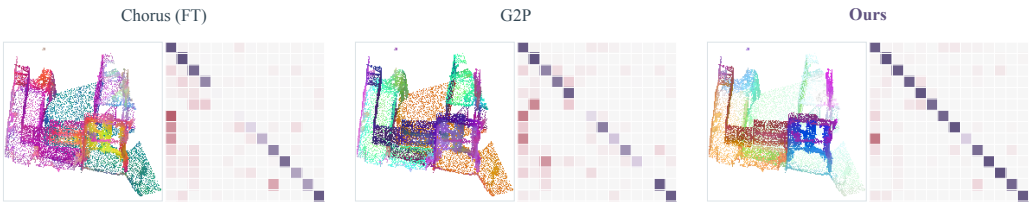

Figure 4: **Feature organization and class confusion on ScanNet.** Top: independently fitted PCA feature maps for each method. Bottom: row-normalized class-confusion matrices. Matrix rows denote ground truth and columns denote predictions in the class order in Appendix B.4; purple marks correct predictions and pink marks cross-class errors.

The t-SNE plots in Figure 5 offer a second view of the learned features on the same ScanNet points. Class colors overlap more for G2P, Chorus, and PointGS than for SIP, in line with the PCA maps and the class-confusion results. SIP also has the smallest model and lowest latency among the six methods shown: 27.3M parameters and 134 ms per scene. Relative to G2P, it uses 41% fewer parameters and takes 40% less inference time under the same protocol. Embedding settings appear in Appendix B.4.

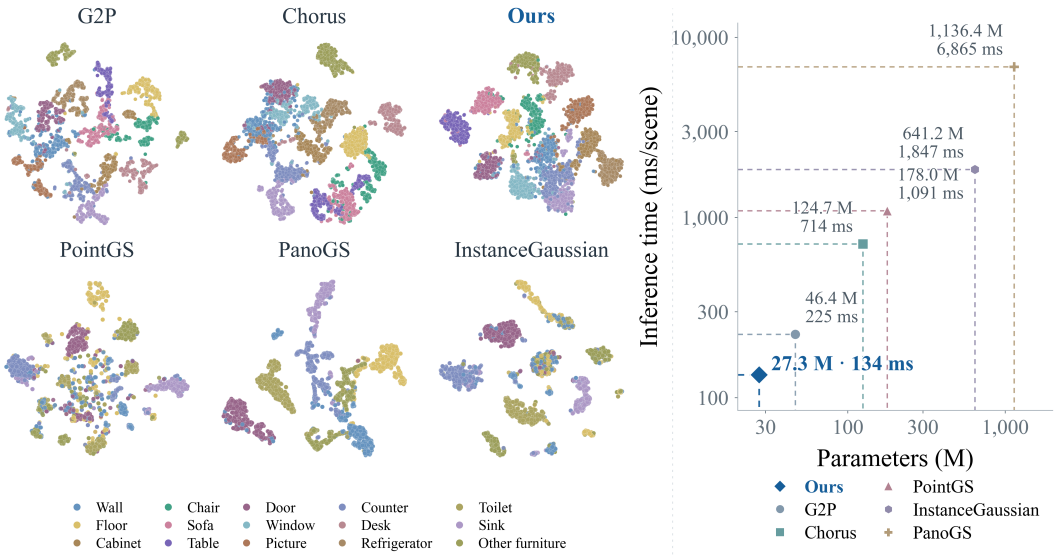

Figure 5: **Feature separability and efficiency.** Left: t-SNE features colored by ground-truth semantic class. Right: model size versus inference latency; lower is better.

4.3 Point Ownership and Gaussian Support

To locate where Gaussians help, we compare three ways to use them on S3DIS Area 5. Direct concatenation adds their attributes to point features. Shuffled pairing retains JSA and the same attributes but breaks their source-point assignment. The support comparison keeps that assignment fixed and changes the covariance used to weight neighbors. Together these variants separate attribute presence, point ownership, and pairwise geometry.

Table 6: **Addressability ablation.** G: Gaussian attributes; P: correct source-point pairing.

Variant	G	P	JSA	mIoU
Point only	–	–	–	47.9
Direct concat	✓	✓	–	42.9
Shuffled pairing	✓	✗	✓	56.6
Full	✓	✓	✓	**60.8**

Table 7: **JSA ablation.** Rows progressively introduce relative reasoning and Gaussian support.

Aggregation	Support	mIoU
Direct concat	–	42.9
Feature affinity	–	52.1
+ Relative relation	–	52.8
+ Single support	Σ_i	53.5
Joint support (JSA)	Σ_{ij}^{J}	**60.8**

Direct concatenation scores 42.9 mIoU, below the point-only model's 47.9 in the addressability comparison of Table 6. Extra Gaussian channels alone do not improve segmentation. Shuffling the point-Gaussian assignments reduces the JSA result from 60.8 to 56.6, showing why the correct owner matters. The support comparison in Table 7 keeps ownership fixed: the center covariance gives 53.5 mIoU, while joint support reaches 60.8. JSA benefits from the neighbor's spatial extent as well as the center's. Backbone variants appear in Appendix A.2.

AGRF revisits these relations after the point backbone has formed scene features. JSA has already combined nearby evidence; AGRF can now score the same point pairs using features that contain wider scene context. The iteration comparison in Table 9 of Appendix B.3 measures its effect on S3DIS Area 5: three updates raise mIoU from 57.73 to 60.8, Boundary mIoU from 34.45 to 38.91, and Boundary F-score from 38.75 to 41.82 relative to removing AGRF. The first boundary gain means fewer wrong labels near class interfaces; the second means those interfaces are placed closer to ground truth. The refinement therefore improves both local class decisions and boundary placement. Three updates perform best; two score below the version without AGRF.

5 Conclusion

This paper introduces Splat In Place (SIP), a segmentation-oriented reconstruction scheme that establishes the point-Gaussian relation during reconstruction rather than recovering it afterwards. By pinning every Gaussian to the point that spawned it, SIP grants each point direct access to its own optimized Gaussian state, a property we term Point Gaussian Addressability. Building on this access, Joint Support Aggregation turns the anisotropic supports of two neighboring points into the metric that governs local aggregation, while the Addressable Gaussian Recurrent Field consults the same relation once contextual features have been formed, refining both local class decisions and boundary placement. Across ScanNet, S3DIS, and KITTI-360, SIP improves indoor and outdoor segmentation over strong 2D-guided, geometric, and Gaussian-based baselines, and leads each dataset's sparsest-input comparison.

Limitations and Future Work.

While SIP performs well here, its evaluation follows established benchmarks, whose scene scale and class vocabulary are fixed in advance and may not reveal the full behavior of a point-indexed Gaussian representation. Assessing SIP on larger and more diverse scene collections is therefore an important direction for future work, together with extending the addressable representation to panoptic and open-vocabulary settings. These directions can make point-pinned Gaussians a general interface between reconstruction and 3D understanding.

AI Use Statement

We used generative AI tools to organize the paper's argument and revise its wording, including how the method and experiments are explained. We also used them to search and summarize related literature, discuss the interpretation of reported comparisons, and refine figure and table layouts and visual comparisons. The authors designed and ran the experiments; AI tools did not generate the reported measurements. The authors reviewed and revised the AI-assisted text, citations, numerical results, and figures against the underlying work and take responsibility for the final manuscript.

Ethics Statement

All experiments use previously released ScanNet, S3DIS, and KITTI-360 data; we collect no new real-world observations. These datasets contain captured indoor and street scenes, so reuse of their RGB images should follow the dataset terms and applicable privacy safeguards. Errors on rare classes and sparsely observed areas may matter in downstream driving systems. Benchmark accuracy alone does not establish safety for deployment.

Reproducibility Statement

Section 3 gives the reconstruction constraint, neighborhood weights, and recurrent updates used by SIP. Section 4 reports the datasets, evaluation protocols, training settings, comparisons, and ablations. Appendices A and B provide further architecture, optimization, and data details; Appendix C contains additional visualizations.

References

Iro Armeni, Ozan Sener, Amir R. Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3D semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1534–1543, 2016.

Jiazhong Cen, Jiemin Fang, Chen Yang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Segment any 3D gaussians. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 1971–1979, 2025. doi: 10.1609/aaai.v39i2.32193.

Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4D spatio-temporal convnets: Minkowski convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3075–3084, 2019.

Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. ScanNet: Richly-annotated 3D reconstructions of indoor scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5828–5839, 2017.

Tianchen Deng, Yue Pan, Shenghai Yuan, Dong Li, Chen Wang, Mingrui Li, Long Chen, Lihua Xie, Danwei Wang, Jingchuan Wang, et al. What is the best 3d scene representation for robotics? from geometric to foundation models. *arXiv preprint arXiv:2512.03422*, 2025.

Xin Deng, Wenyu Zhang, Qing Ding, and Xinming Zhang. PointVector: A vector representation in point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9455–9465, 2023.

Wenbo Hu, Hengshuang Zhao, Li Jiang, Jiaya Jia, and Tien-Tsin Wong. Bidirectional projection network for cross dimension scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14373–14382, 2021.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4):1–14, 2023. doi: 10.1145/3592433.

Abhijit Kundu, Xiaoqi Yin, Alireza Fathi, David Ross, Brian Brewington, Thomas Funkhouser, and Caroline Pantofaru. Virtual multi-view fusion for 3D semantic segmentation. In *European Conference on Computer Vision (ECCV)*, pp. 518–535, 2020.

Xin Lai, Jianhui Liu, Li Jiang, Liwei Wang, Hengshuang Zhao, Shu Liu, Xiaojuan Qi, and Jiaya Jia. Stratified transformer for 3D point cloud segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8500–8509, 2022.

Xin Lai, Yukang Chen, Fanbin Lu, Jianhui Liu, and Jiaya Jia. Spherical transformer for LiDAR-based 3D recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17545–17555, 2023.

Haijie Li, Yanmin Wu, Jiarui Meng, Qiankun Gao, Zhiyao Zhang, Ronggang Wang, and Jian Zhang. InstanceGaussian: Appearance-semantic joint gaussian representation for 3D instance-level perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14078–14088, 2025a.

Yue Li, Qi Ma, Runyi Yang, Huapeng Li, Mengjiao Ma, Bin Ren, Nikola Popovic, Nicu Sebe, Ender Konukoglu, Theo Gevers, Luc Van Gool, Martin R. Oswald, and Danda Pani Paudel. SceneSplat: Gaussian splatting-based scene understanding with vision-language pretraining. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4961–4972, 2025b.

Yue Li, Qi Ma, Runyi Yang, Mengjiao Ma, Bin Ren, Nikola Popovic, Nicu Sebe, Theo Gevers, Luc Van Gool, Danda Pani Paudel, and Martin R. Oswald. Chorus: Multi-teacher pretraining for holistic 3D gaussian scene encoding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21431–21442, 2026.

Yiyi Liao, Jun Xie, and Andreas Geiger. KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2D and 3D. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3292–3310, 2023. doi: 10.1109/TPAMI.2022.3179507.

Haojia Lin, Xiawu Zheng, Lijiang Li, Fei Chao, Shanshan Wang, Yan Wang, Yonghong Tian, and Rongrong Ji. Meta architecture for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17682–17691, 2023.

Zhijian Liu, Xinyu Yang, Haotian Tang, Shang Yang, and Song Han. FlatFormer: Flattened window attention for efficient point cloud transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1200–1211, 2023.

Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. PointNet: Deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 652–660, 2017a.

Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017b.

Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Abed Al Kader Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. PointNeXt: Revisiting PointNet++ with improved training and scaling strategies. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 23192–23204, 2022.

Minghan Qin, Wanhua Li, Jiawei Zhou, Haoqian Wang, and Hanspeter Pfister. LangSplat: 3D language gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20051–20060, 2024.

Hojun Song, Chae-yeong Song, Jeong-hun Hong, Chaewon Moon, Soo Ye Kim, Yiyi Liao, Jaehyup Lee, and Sang-hyo Park. G2P: Gaussian-to-point attribute alignment for boundary-aware 3D segmentation. In *European Conference on Computer Vision (ECCV)*, 2026a.

Yixiao Song, Qingyong Li, Wen Wang, and Zhicheng Yan. PointGS: Semantic-consistent unsupervised 3D point cloud segmentation with 3D gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 33343–33352, 2026b.

Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, and Leonidas J. Guibas. KPConv: Flexible and deformable convolution for point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6411–6420, 2019.

Peng-Shuai Wang. OctFormer: Octree-based transformers for 3D point clouds. *ACM Transactions on Graphics*, 42(4):1–11, 2023.

Ziyi Wang, Yanran Zhang, Jie Zhou, and Jiwen Lu. UniPre3D: Unified pre-training of 3D point cloud models with cross-modal gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1319–1329, 2025.

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer V3: Simpler faster stronger. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4840–4851, 2024a.

Xiaoyang Wu, Daniel DeTone, Duncan Frost, Tianwei Shen, Chris Xie, Nan Yang, Jakob Engel, Richard Newcombe, Hengshuang Zhao, and Julian Straub. Sonata: Self-supervised learning of reliable point representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22193–22204, 2025.

Yanmin Wu, Jiarui Meng, Haijie Li, Chenming Wu, Yahao Shi, Xinhua Cheng, Chen Zhao, Haocheng Feng, Errui Ding, Jingdong Wang, and Jian Zhang. OpenGaussian: Towards point-level 3D gaussian-based open vocabulary understanding. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 19114–19138, 2024b.

Siming Yan, Yuqi Yang, Yu-Xiao Guo, Hao Pan, Peng-Shuai Wang, Xin Tong, Yang Liu, and Qixing Huang. 3D feature prediction for masked-autoencoder-based point cloud pretraining. In *International Conference on Learning Representations (ICLR)*, 2024.

Yu-Qi Yang, Yu-Xiao Guo, Jian-Yu Xiong, Yang Liu, Hao Pan, Peng-Shuai Wang, Xin Tong, and Baining Guo. Swin3D: A pretrained transformer backbone for 3D indoor scene understanding. *Computational Visual Media*, 11(1):83–101, 2025.

Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3D scenes. In *European Conference on Computer Vision (ECCV)*, pp. 162–179, 2024. doi: 10.1007/978-3-031-73397-0_10.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14114–14124, 2025.

Yujia Zhang, Xiaoyang Wu, Yixing Lao, Chengyao Wang, Zhuotao Tian, Naiyan Wang, and Hengshuang Zhao. Concerto: Joint 2D-3D self-supervised learning emerges spatial representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 38, pp. 69498–69522, 2025.

Yujia Zhang, Xiaoyang Wu, Yunhan Yang, Xianzhe Fan, Han Li, Yuechen Zhang, Zehao Huang, Naiyan Wang, and Hengshuang Zhao. Utonia: Toward one encoder for all point clouds. In *International Conference on Machine Learning (ICML)*, 2026.

Xiao Zheng, Xiaoshui Huang, Guofeng Mei, Yuenan Hou, Zhaoyang Lyu, Bo Dai, Wanli Ouyang, and Yongshun Gong. Point cloud pre-training with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22935–22945, 2024.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21676–21685, 2024.

A Implementation Details

A.1 Point Pinned Reconstruction

We start with input points and calibrated RGB views. Each point initializes one Gaussian. Image reconstruction fits its scale, rotation, opacity, and spherical harmonic coefficients while keeping the center, primitive count, and point index fixed. The reconstructed Gaussians therefore have the same order as the input points. We base the RGB fitting settings on SAGA (Cen et al., 2025); the segmentation loss uses point labels.

Point i retrieves its Gaussian attributes through this index. Selecting a subset of points also selects the corresponding Gaussians. We summarize appearance with DC color and channelwise spherical harmonic energy. Scale and opacity enter the node feature; scale and rotation define the covariance used by JSA.

A.2 Network Architecture

Joint attribute learning. The node encoder maps point coordinates, color, and normals together with Gaussian appearance, scale, and opacity to an initial feature. Point kNN selects candidate neighbors. For each pair, JSA measures point displacement and differences in appearance, Gaussian volume, and opacity. The covariances of both Gaussians give the joint support. JSA combines these quantities with feature similarity to weight the neighbor's message. A residual update adds the aggregated message to the initial feature before the point backbone.

Point backbone. We evaluate Utonia (Zhang et al., 2026), Point Transformer V3 (PTv3) (Wu et al., 2024a), and Stratified Transformer (Lai et al., 2022), using the first three encoder stages of each backbone. Utonia is the default configuration. The backbone comparison on S3DIS Area 5 is reported in Table 10.

Recurrent refinement. AGRF scores neighbors using the backbone features and e_{ij}. It reuses these weights for three updates with shared GRU parameters. Each update adds a residual to the backbone feature z_i^0, and the final prediction uses z_i^3. We also supervise the predictions after the first two updates during training.

Table 8: **SIP architecture.** Gaussian attributes retain their source-point indices.

Component	Configuration
Gaussian representation	One primitive per input point; fixed centers and identities
Point descriptor	Coordinates, color, and normals
Gaussian node descriptor	DC color, channelwise SH energy, scale, and opacity
JSA	Point kNN selects neighbors; point and Gaussian differences and joint covariance set their weights
Point backbone	First three encoder stages of Utonia, PTv3, or Stratified Transformer; Utonia by default
AGRF	Three iterations with shared GRU parameters
Semantic head input	z_i^3 for the final prediction; z_i^1 and z_i^2 are also supervised

A.3 Optimization

RGB reconstruction provides the initial Gaussian attributes. During segmentation training, the point-level loss directly updates scale, rotation, opacity, and spherical harmonic coefficients, along with the node encoder, JSA, point backbone, AGRF, and semantic head. Centers and source indices stay fixed. We initialize the first three Utonia encoder stages at random and use AdamW with a learning rate of 10^{-5} for the segmentation network. We apply cross-entropy and Dice losses to the final semantic prediction and to the outputs of the intermediate AGRF iterations. For the Gaussian attributes, we use a learning rate of 3×10^{-5} for scale, rotation, opacity, and spherical harmonic coefficients. The current covariance and opacity enter the encoder. At evaluation, each scene is reconstructed from its RGB views and processed without segmentation gradient updates.

B Experimental Settings

B.1 Datasets and Evaluation

ScanNet v2 (Dai et al., 2017) provides RGB-D observations and point annotations for 20 semantic classes. We use all 1,201 training scenes and report semantic mIoU on the 312 validation scenes. S3DIS (Armeni et al., 2016) contains annotated indoor rooms from six areas and has 13 semantic classes; Area 5 is held out for evaluation. KITTI-360 (Liao et al., 2023) provides calibrated camera images and accumulated outdoor point clouds. Semantic segmentation is evaluated with mean intersection over union (mIoU) on labeled points, reported in percentage points. Boundary mIoU computes class IoU within a band around ground-truth class boundaries. Boundary F-score matches predicted and ground-truth boundary points within a spatial tolerance and takes the harmonic mean of boundary precision and recall.

S3DIS protocol. Inspired by PointGS (Song et al., 2026b), we render RGB reconstruction views by projecting the colored S3DIS points. The views contain no semantic predictions or masks. We randomly select 10 rooms outside Area 5 for supervised segmentation training and evaluate on Area 5. Reconstruction views are generated from the input point cloud rather than captured by a camera. We retrain all compared methods using the same ten rooms.

Input point density. We select challenging classes following G2P (Song et al., 2026a). Random point dropout in PointNet++ (Qi et al., 2017b) motivates our density test. Before reconstruction, we shuffle the original point IDs with a fixed random seed and retain the specified fraction. We fit Gaussians to the retained points and rebuild their neighborhoods before inference. High, Medium, and Low retain 75%, 50%, and 25% of the original points, respectively. All methods use the same retained point IDs, and IoU is computed on those points. Each score averages IoU over the selected classes. For ScanNet, these classes are door, window, picture, curtain, refrigerator, and shower curtain. For S3DIS, they are column, window, door, sofa, bookcase, board, and clutter. Mean in the S3DIS table averages its three challenging-class scores. Overall in both indoor tables covers all semantic classes on the full input.

KITTI-360 protocol. For KITTI-360, we follow the official 3D train and validation window lists. We randomly select ten accumulated point-cloud windows from the training list and evaluate on the labeled validation windows, using their calibrated perspective RGB images. All compared methods are retrained on the same ten windows. We compare the full input with 50% and 75% point removal, reconstructing Gaussians for each retained point set.

B.2 Downstream Instance Segmentation

For instance segmentation, we add a mask branch parallel to the semantic head. It predicts query masks from z_i^3, with intermediate supervision at z_i^1 and z_i^2. We train on the semantic and instance annotations of ScanNet and S3DIS. The latter uses the same ten training rooms and Area 5 evaluation as semantic segmentation. Hungarian matching pairs queries with ground-truth instances. The instance loss comprises class cross-entropy, mask binary cross-entropy and Dice loss over the original points, and supervision for mask quality, instance centers, and point offsets. This downstream training also applies semantic cross-entropy, soft-IoU, and a constraint on Gaussian attribute updates. Ground truth enters matching and loss computation only. At inference, the branch predicts class labels, confidence scores, and instance masks on the original points; thresholding and same-class non-maximum suppression produce the final instances. We report matched-mask mIoU on the original points. Each unmatched ground-truth instance receives zero IoU, and all compared methods are reevaluated with the same metric.

B.3 Ablation Configurations

Both ablations use Utonia on S3DIS Area 5. In Table 6, the point-only variant uses the original observations, and direct concatenation adds the paired Gaussian attributes without JSA. Shuffled pairing retains JSA but assigns each point another Gaussian's attributes. The set of attributes stays the same; only their point assignments change. The full model restores the original pairing.

The variants in Table 7 start with direct concatenation, then add feature affinity and relative relations. The last two rows compare support from the center Gaussian with joint support from both Gaussians.

Table 9: **AGRF iterations on S3DIS Area 5.** Each update uses backbone features and the JSA pair descriptor. Zero updates remove AGRF; three updates are used by SIP and score highest on all three metrics.

AGRF updates	mIoU	Boundary mIoU	Boundary F-score
w/o AGRF	57.73	34.45	38.75
2	56.30	32.34	37.88
3 (SIP)	**60.8**	**38.91**	**41.82**
4	60.1	36.88	<u>39.11</u>
5	<u>60.3</u>	<u>37.10</u>	38.22

Point backbone comparison. We also evaluate SIP with the first three encoder stages of Utonia, PTv3, and Stratified Transformer under the S3DIS protocol described above. Table 10 shows 60.8 mIoU with Utonia, compared with 57.3 for PTv3 and 59.6 for Stratified Transformer.

Table 10: **Backbone comparison on S3DIS Area 5.** Each model uses its first three encoder stages.

Point backbone	mIoU (%)
PTv3 (Wu et al., 2024a)	57.3
Stratified Transformer (Lai et al., 2022)	<u>59.6</u>
Utonia (Zhang et al., 2026)	**60.8**

B.4 Feature Visualization and Efficiency Evaluation

Feature extraction. The t-SNE comparison uses features from the same set of ScanNet scenes (Figure 5). G2P and the decoder-only Chorus checkpoint provide 64-dimensional features before their semantic classifiers. SIP uses a saved 64-dimensional AGRF state. PointGS supplies 32-dimensional contrastive features, PanoGS supplies 512-dimensional features, and InstanceGaussian supplies six-dimensional instance features. PointGS and PanoGS features are transferred to the query points by nearest-neighbor association; InstanceGaussian anchors match the query coordinates.

Embedding settings. We normalize each feature to unit length and fit t-SNE independently for each method, using perplexity 30, an automatic learning rate, early exaggeration 12, and 1,500 iterations with random initialization and seed 0. Ground-truth labels determine the displayed colors.

PCA feature maps and class confusion. For ScanNet validation scene `scene0231_01`, we use all 1,386 channels of SIP's classifier input, which combines features from multiple scales. G2P and Chorus (FT) each use 64 channels. All features are taken before the semantic classifiers (Figure 4). We fit PCA independently to each method's unnormalized features on the same 50,000 original point indices, using six components, five power iterations, and seed 37. The RGB mapping combines components 1–3 and 4–6 with weights 0.6 and 0.4, respectively, with min–max scaling and a common brightness factor of 1.2. All columns share the camera and displayed geometry, including removal of the roof and camera-facing outer wall portions to expose the interior. The same normal-based shading rule is applied to the PCA colors. Semantic labels are excluded from PCA fitting, and features are displayed without smoothing or semantic recoloring. The independent projections assign their own colors, so the comparison concerns spatial variation within each feature map. The matrices below the PCA maps use point predictions for the same scene. They display the 14 classes present in the ground truth, ordered as wall, floor, cabinet, chair, sofa, table, door, window, picture, counter, desk, refrigerator, sink, and other furniture on both axes. Each row is normalized over predictions from all 20 ScanNet classes. Purple diagonal cells show correct predictions, and pink off-diagonal cells show cross-class errors.

Efficiency evaluation. All six methods are evaluated on the same set of scenes under the same inference protocol. Latency is measured in milliseconds per scene and shown on a logarithmic scale in Figure 5.

C Additional Qualitative Results

The S3DIS Area 5 examples in Figure 6 compare G2P, PanoGS, Chorus, and SIP for semantic segmentation, and InstanceGaussian and SIP for instance segmentation.

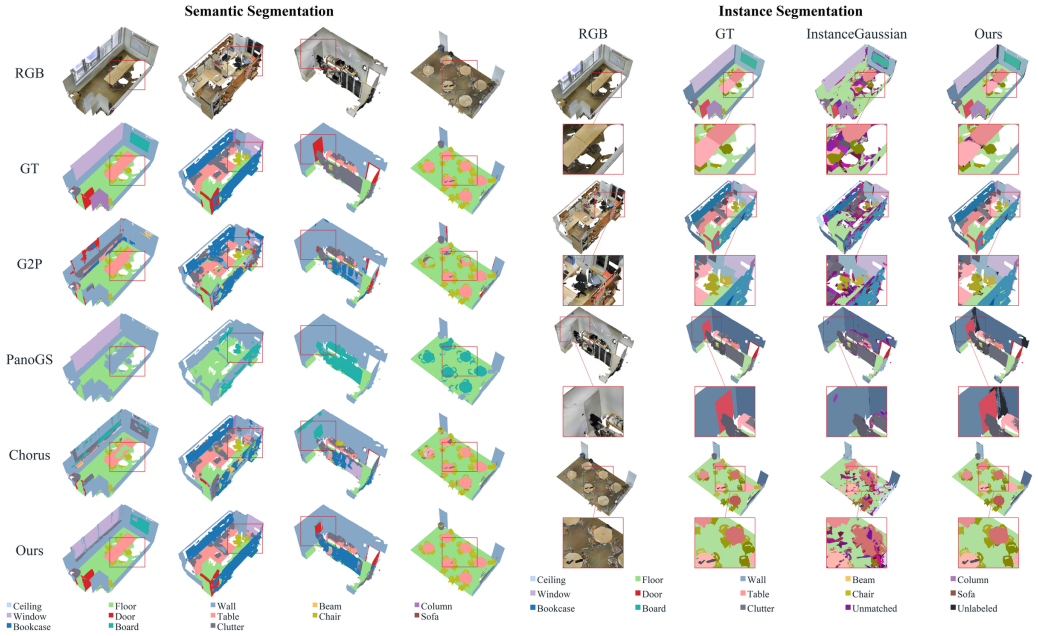

Figure 6: **S3DIS Area 5 qualitative results.** Left: semantic segmentation with scenes arranged by column and methods by row. Right: instance visualization with scenes arranged by row and methods by column. Red boxes highlight comparison regions. The full ScanNet semantic and instance panels appear in Figures 8 and 9.

Additional PCA feature maps. Two further ScanNet scenes use the settings above (Figure 7).

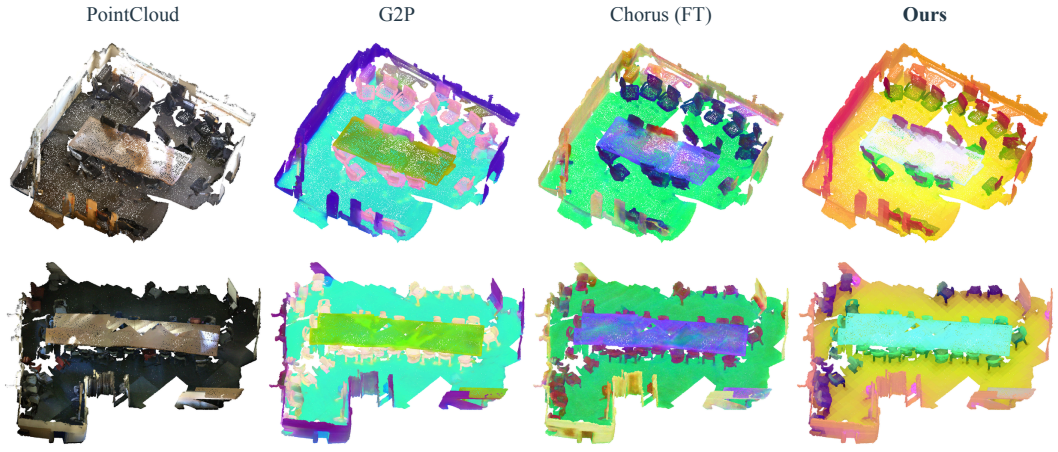

Figure 7: **Additional PCA feature maps on ScanNet.** Top: `scene0088_02`. Bottom: `scene0249_00`. Columns show the RGB point cloud, G2P, Chorus (FT), and SIP. Each row uses the same geometry and viewpoint for all methods. PCA colors are fitted independently for each method, following Figure 4.

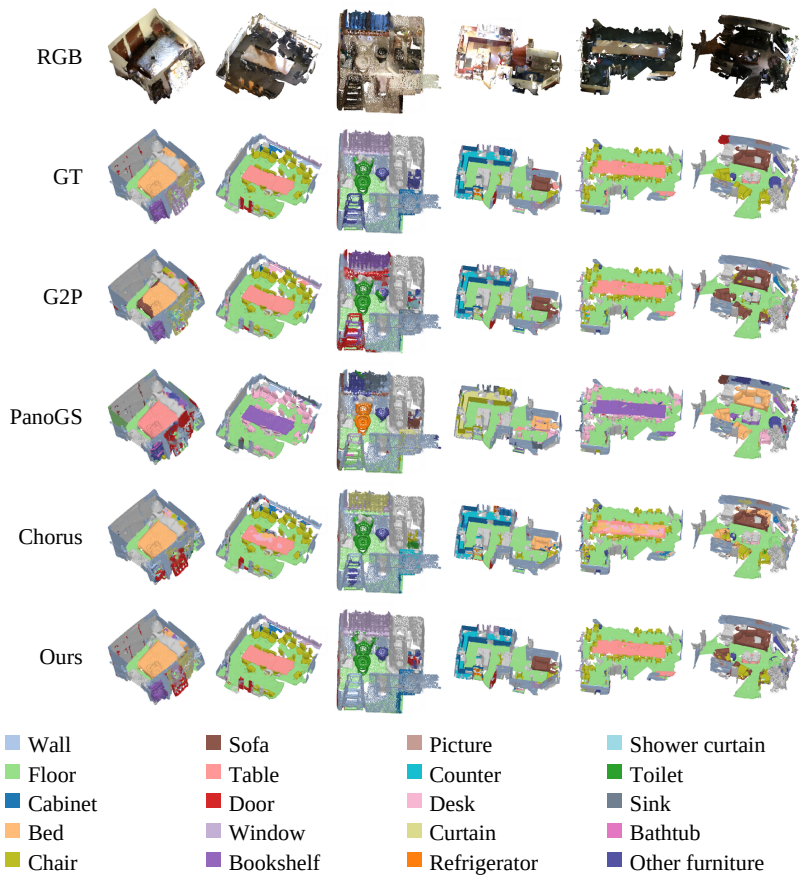

Figure 8: **ScanNet semantic segmentation.** Columns show six scenes; rows show RGB, ground truth, G2P (Song et al., 2026a), PanoGS (Zhai et al., 2025), Chorus (Li et al., 2026), and SIP.



Figure 9: **Instance visualization on ScanNet.** Rows show four scenes; columns show RGB, ground truth, InstanceGaussian (Li et al., 2025a), and SIP.