GaussianCross: Cross-modal Self-supervised 3D Representation Learning via Gaussian Splatting

Lei Yao
rayyoh.yao@connect.polyu.hk
Hong Kong Polytechnic University
Hong Kong, Hong Kong

Yi Wang*
yi-eie.wang@polyu.edu.hk
Hong Kong Polytechnic University
Hong Kong, Hong Kong

Yi Zhang
yi-eee.zhang@connect.polyu.hk
Hong Kong Polytechnic University
Hong Kong, Hong Kong

Moyun Liu
lmomoy@hust.edu.cn
Huazhong University of Science and Technology
Wuhan, China

Lap-Pui Chau*
lap-pui.chau@polyu.edu.hk
Hong Kong Polytechnic University
Hong Kong, Hong Kong

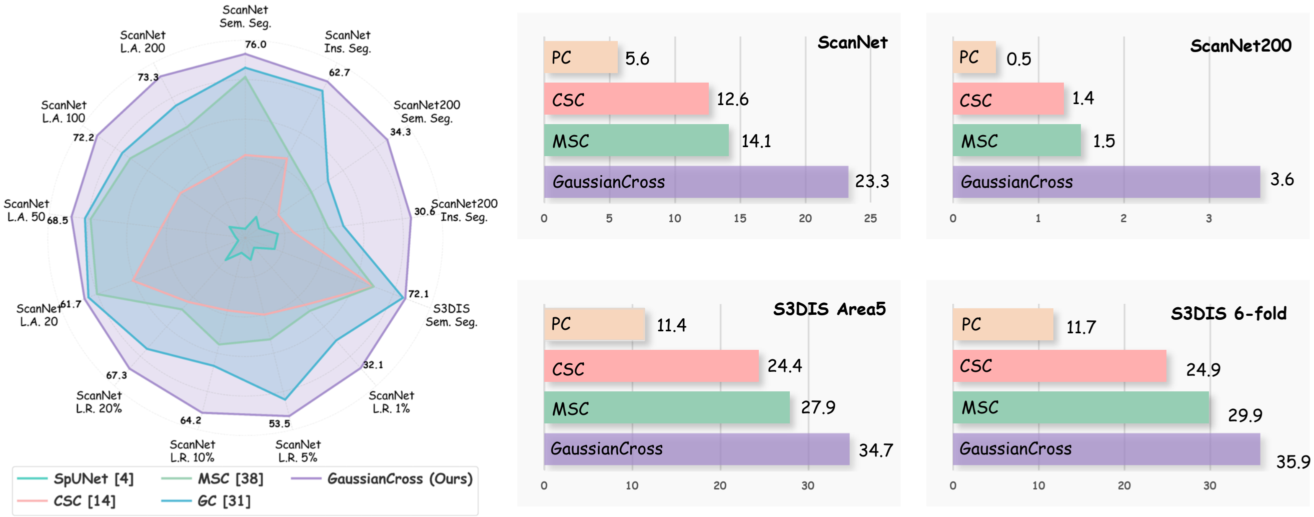

Figure 1: Performance comparison of GaussianCross on 3D scene understanding tasks. GaussianCross achieves superior performance across various tasks, including semantic segmentation (*Sem. Seg.*) [4], instance segmentation (*Ins. Seg.*) [17], and linear probing [37]. *Left*: full fine-tuning results on various downstream tasks. *Right*: linear probing accuracy.

Abstract

The significance of informative and robust point representations has been widely acknowledged for 3D scene understanding. Despite existing self-supervised pre-training counterparts demonstrating promising performance, the model collapse and structural information deficiency remain prevalent due to insufficient point discrimination difficulty, yielding unreliable expressions and suboptimal performance. In this paper, we present ***GaussianCross***, a novel cross-modal self-supervised 3D representation learning architecture integrating feed-forward 3D Gaussian Splatting (3DGS) techniques to address current challenges. GaussianCross seamlessly converts scale-inconsistent 3D point clouds into a unified cuboid-normalized Gaussian representation without missing details, enabling stable and generalizable pre-training. Subsequently, a tri-attribute adaptive distillation splatting module is incorporated to construct a 3D feature field, facilitating synergetic feature capturing of appearance, geometry, and semantic cues to maintain cross-modal consistency. To validate GaussianCross, we perform extensive evaluations on various benchmarks, including ScanNet, ScanNet200, and S3DIS. In particular, GaussianCross shows a prominent parameter and data efficiency, achieving superior performance through linear probing (<0.1% parameters) and limited data training (1% of scenes) compared to state-of-the-art methods. Furthermore, GaussianCross demonstrates strong generalization capabilities, improving the full fine-tuning accuracy by 9.3% mIoU and 6.1% AP_{50} on ScanNet200 semantic and instance segmentation tasks, respectively, supporting the effectiveness of our approach. The code, weights, and visualizations are publicly available at https://rayyoh.github.io/GaussianCross/.

1 Introduction

Self-supervised representation learning has emerged as a transformative training paradigm for capturing expressive features from large-scale unlabeled data. It has demonstrated promising potential

[*] Corresponding author

across diverse downstream applications, including scene understanding [22, 42], navigation [20], and embodied manipulation [49]. While the success of 2D visual foundation models (VFMs) such as MAE [11], MoCo [12], and DINOv2 [25] trained by self-supervised pre-training, the development of comparable 3D methodologies remains critical for comprehensive physical world understanding [37]. However, different from available web-scale images, 3D data, especially point clouds, are usually scarce and come with sophisticated spatial structures, hindering the design of effective self-supervised representation learning strategies. The sparse and irregular nature of the point cloud further complicates the learning process.

Although recent investigations [26, 27, 45, 46] have advanced object-level point cloud representation learning, these approaches face fundamental scale incompatibility when transitioning to scene-level scenarios. Concurrently, some frameworks [8, 14, 33, 40, 41] have attempted to explore contrastive learning-based algorithms for capturing compelling 3D scene features, which typically generate dual distinct views from the same scene and consider point-wise discrimination as their pretext tasks. Despite empirical improvements on downstream tasks, persistent challenges remain. For instance, PointContrast [41] suffers from model collapse stemming from inadequate diversity in view augmentation strategies, while GroupContrast [33] exhibits significant parameter sensitivity and depends on precomputed over-segmentations [9], thereby restricting its adaptability. On the other hand, the integration of neural rendering techniques introduces alternative pathways for self-supervised representation learning. Ponder [15] pioneers a Neural Radiance Field (NeRF) [24] based pre-training paradigm that leverages novel view synthesis as the supervisory signal, but its practical scalability is hampered by the inherent slow training and rendering speed. GS^3 [21] conducts a preliminary exploration of 3D Gaussian Splatting [18] (3DGS) for rendering-based pre-training strategy, which implements epipolar transformer [36] for cross-view pixel-wise alignment. However, this approach focuses exclusively on photometric reconstruction while neglecting critical geometric and semantic relationships, resulting in suboptimal performance on structurally complex downstream tasks. Additionally, the method starts from back-projected point clouds of sparse view RGB-D frames, which is inherently limited to global context modeling.

To address the aforementioned challenges, we propose ***GaussianCross***, a novel cross-modal self-supervised 3D representation learning framework with Gaussian Splatting to learn informative and robust point representations for scene understanding. Unlike the per-scene optimization paradigm of vanilla 3DGS [18], our method operates in a generalizable manner and is tailored to capture diverse intrinsic properties. Nevertheless, a potential challenge is scale uncertainty across different indoor scenes, which causes the model struggling to learn a unified representation as shown in Fig. 3 top (*w/o* Cuboid-Normalized). To this end, we propose *Cuboid-Normalized Gaussian Initialization*, a technique leveraged to transform scene point clouds into a cuboid structure and parameterize them as a collection of Gaussian primitives. The process enables the model to flexibly adapt to scale variations in different scenes, allowing seamless scene description conversion without compromising detail fidelity. Furthermore, we introduce a *Tri-Attribute Adaptive Distillation Splatting* module that utilizes the real-time rendering capability of rasterization splatting [18]. Apart from common Gaussian characteristics, we predict an offset to dynamically refine the mean position and integrate an opacity-driven pruning mechanism to control primitive density, which has proved crucial for accurate scene representation. In addition, we incorporate a 3D feature field to guide semantic map synthesis, aiming to pursue high-level semantic-aware details. The generated maps are then upsampled by a projection head to align with latent embeddings of a pre-trained 2D foundation model, facilitating cross-modal knowledge distillation. GaussianCross achieves simultaneous capture of complementary photometric appearance, geometric structure, and semantic context, prompting synergistic feature learning. The self-supervised training process is performed by reconstructing randomly sampled views to provide robust supervision, effectively mitigating model collapse risk. Our contributions comprise:

- We propose a novel cross-modal self-supervised 3D representation learning architecture for scene understanding with generalizable Gaussian Splatting, named GaussianCross.
- We introduce a cuboid-normalized Gaussian initialization technique to represent scenes as structured 3D Gaussians, adapting to inconsistent scales across different scenes.
- We design a tri-attribute adaptive distillation splatting module to jointly capture the appearance, geometry, and semantic properties of scenes, achieving cross-modal knowledge distillation from visual foundation models.
- Comprehensive experiments on various scene understanding tasks demonstrate the superior performance of GaussianCross over previous state-of-the-art methods.

2 Related work

2.1 Point Clouds Self-supervised Learning

The recent proliferation of self-supervised learning in 2D [13, 52] has inspired research efforts to adapt this paradigm to point cloud analysis. Pioneering works like Point-MAE [26] and Point-BERT [46] successfully transferred masked autoencoding [7] to object-level point clouds by transformer-based architectures [32]. However, scaling such object-centric approaches to scene tasks is non-trivial due to sparse geometric structures in real-world 3D scenes. To address this challenge, PointContrast (PC) [41] established an unsupervised framework for indoor scenes, which learns point-wise representation derived from RGB-D frames by maximizing the mutual information between augmented views. Building upon this foundation, Contrastive Scene Context (CSC) [14] introduced spatial contextual constraints to encode structural relationships beyond individual points correspondence. In [40], Masked Scene Contrast (MSC) unified color reconstruction and surfel normal prediction within a pipeline and proposed an efficient view generation strategy. In contrast, recent innovations highlight semantic-aware learning as a critical frontier. For example, GroupContrast (GC) [33] identified the semantic ambiguity problem and addressed it by a segment grouping strategy based on pre-computed superpoints [9]. It further proposed a group-aware contrastive loss to enhance the representation, while Point-GCC [8] incorporated deep clustering for object-level supervision. Despite these advancements, current contrastive methods remain susceptible to model collapse phenomena [33] and exhibit parametric sensitivity. Our approach diverges from them by leveraging a cross-modal pre-training paradigm, which enhances robustness and generalizability.

2.2 Cross-modal 3D Pre-training

There is another series of works aiming to pre-train 3D models with cross-modal data. MM-Point [45] enforced cross-modal consistency representations through point-to-pixel projection, aligning specific view images with point clouds. While effective, these methods critically rely on the availability of well-aligned 2D-3D pairs, which may not be feasible in many real-world applications. Instead, some recent works [15, 21, 51] consider differentiable rendering as a self-supervised signal by comparing arbitrary synthetic views with real images for 3D scenes. Ponder [15] employed the neural radiance fields-based [24] technique for SDF values and colors prediction from query points based on NeuS [35]. Subsequent work GS^3 [21] adopted 3D Gaussian Splatting [18] for photorealistic rendering starting from multi-view RGB-D frames, but this approach required input views to have overlapped regions and additional computational cost due to its epipolar transformer [36] for view alignment. PonderV2 [51] extended the prior version [15] to multi-source pre-training based on Point Prompt Training (PPT) [39] with language-guided alignment. Nevertheless, a potential limitation is its reliance on 2D ground-truth supervision, which hinders its scalability. Our work establishes another paradigm in this domain through semantic-aware knowledge distillation from VFMs to point clouds with feed-forward Gaussian splatting, enabling effective pre-training without any annotations.

2.3 Generalizable 3D Gaussian Splatting

Neural Radiance Fields (NeRF) [24] implicitly represent 3D scenes with shallow Multi-Layer Perceptrons (MLPs), learning continuous mappings from spatial coordinates to radiance fields. However, the necessity of dense point sampling imposes a significant computational burden during both the training and rendering phases. 3D Gaussian Splatting (3DGS) [18] revolutionized this paradigm by explicit scene parameterization using anisotropic Gaussian primitives, achieving real-time rendering via differentiable rasterization splatting. Although its high-quality rendering output, 3DGS is limited to scene-specific optimization and lacks the ability to generalize to unseen scenes [2]. To address this problem, anchor-based 3DGS methods [2, 3, 36] are proposed. Specifically, PixelSplat [2] incorporated epipolar transformers into the pipeline to enable a feed-forward training paradigm for generalizable 3DGS, while MVSplat [3] and FreeSplat [36] introduced additional techniques to construct cost volume for efficient training and free-viewpoint rendering. Parallel advancements focus on enhancing Gaussian representations through cross-modal fusion. GaussianGrouping [43] integrated priors for part-aware decomposition, Feature-3DGS [50] established dense 2D-3D feature correspondences, and FiT3D [47] adapted visual foundation models via 3D-aware fine-tuning. Inspired by these works, our GaussianCross introduces a novel knowledge distillation framework that transfers VFM-derived semantic features into geometrically grounded Gaussian embeddings, enabling label-efficient pre-training of point cloud encoders.

3 Methodology

This section begins with the preliminaries of 3DGS and presents the overall architecture of GaussianCross in Fig. 2. We subsequently detail our cuboid-normalized Gaussian initialization in Sec. 3.2 and introduce the tri-attribute adaptive distillation splatting in Sec. 3.3. Finally, we describe the loss functions in Sec. 3.4 that regularize our cross-modal self-supervised learning.

3.1 Preliminaries

3DGS [18] considers a cluster of translucent ellipsoids characterized by Gaussian primitives to represent scenes explicitly. Each of them is defined by a center $\boldsymbol{\mu} \in \mathbb{R}^3$ and covariance matrix $\Sigma \in \mathbb{R}^{3\times 3}$, expressed as:

$$G(\boldsymbol{x}) = e^{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})}. \tag{1}$$

To assure positive semi-definiteness during differentiable optimization, Σ is decomposed as $\Sigma = \boldsymbol{RSS}^T\boldsymbol{R}^T$, where $\boldsymbol{R} = \mathtt{q2r}(\boldsymbol{q})$ and $\boldsymbol{S} = \mathtt{diag}(\boldsymbol{s})$ are rotation and scaling matrices, respectively. The operators $\mathtt{q2r}(\cdot)$ and $\mathtt{diag}(\cdot)$ convert quaternions to rotation matrices and construct diagonal matrices from scaling vectors, respectively. Given an arbitrary view transformation matrix $\boldsymbol{W}$, the 3D Gaussians are splatted onto specific 2D camera plane with corresponding mean and covariance:

$$\boldsymbol{\mu}_{2D} = \boldsymbol{PW\mu}, \quad \Sigma_{2D} = \boldsymbol{JW}\Sigma\boldsymbol{W}^T\boldsymbol{J}^T, \tag{2}$$

where $\boldsymbol{P}$ denotes projective transformation and $\boldsymbol{J}$ the Jacobian. Final pixel color is computed by alpha-blending $\mathcal{N}$ ordered Gaussians:

$$\boldsymbol{C}(\boldsymbol{p}) = \sum_{i\in\mathcal{N}} \boldsymbol{c}_i\alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \tag{3}$$

where $\boldsymbol{c}_i$ represents view-dependent spherical harmonics color and α_i combines Σ_{2D} with opacity $\boldsymbol{\sigma}_i$.

3.2 Cuboid-Normalized Gaussian Initialization

This section investigates the integration of 3DGS into point cloud representation learning, motivated by its promise in complex scene modeling without requiring labor-intensive 3D annotations. However, conventional 3DGS methods face limitations in scale-variant scenes representation due to their scene-specific optimization. Inspired by [51], we propose cuboid-normalized Gaussian initialization aiming to alleviate scale variance effects while enabling generalizable feature learning directly from point input.

Given a raw scene point cloud $\mathrm{P}_r = \{\mathrm{C}_{r,i}, \mathrm{A}_{r,i}\}_{i=1}^n$, where $\mathrm{C}_{r,i} \in \mathbb{R}^3$ denotes spatial coordinates x_i, y_i, z_i and $\mathrm{A}_{r,i} \in \mathbb{R}^c$ represents associated c-dimensional attributes (*e.g.* RGB colors, surface normals) per point. Analogous to previous works [21, 40], we mask out a portion of the input by a ratio γ and apply a sampling pattern:

$$\mathcal{S} : \lfloor \gamma \mathrm{P}_r \rfloor \mapsto \mathrm{P}_g = \{\mathrm{C}_{g,i}, \mathrm{A}_{g,i}\}_{i=1}^m \tag{4}$$

with the size $\boldsymbol{g}$ to downsample the point cloud from n to m points. The subsampled point cloud P_g is subsequently processed by a 3D backbone $\mathcal{E}_{\boldsymbol{\phi}}$ with learnable parameters $\boldsymbol{\phi}$:

$$\mathrm{F}_s = \mathcal{E}_{\boldsymbol{\phi}}(\mathrm{P}_g) \in \mathbb{R}^{m\times d_s}, \tag{5}$$

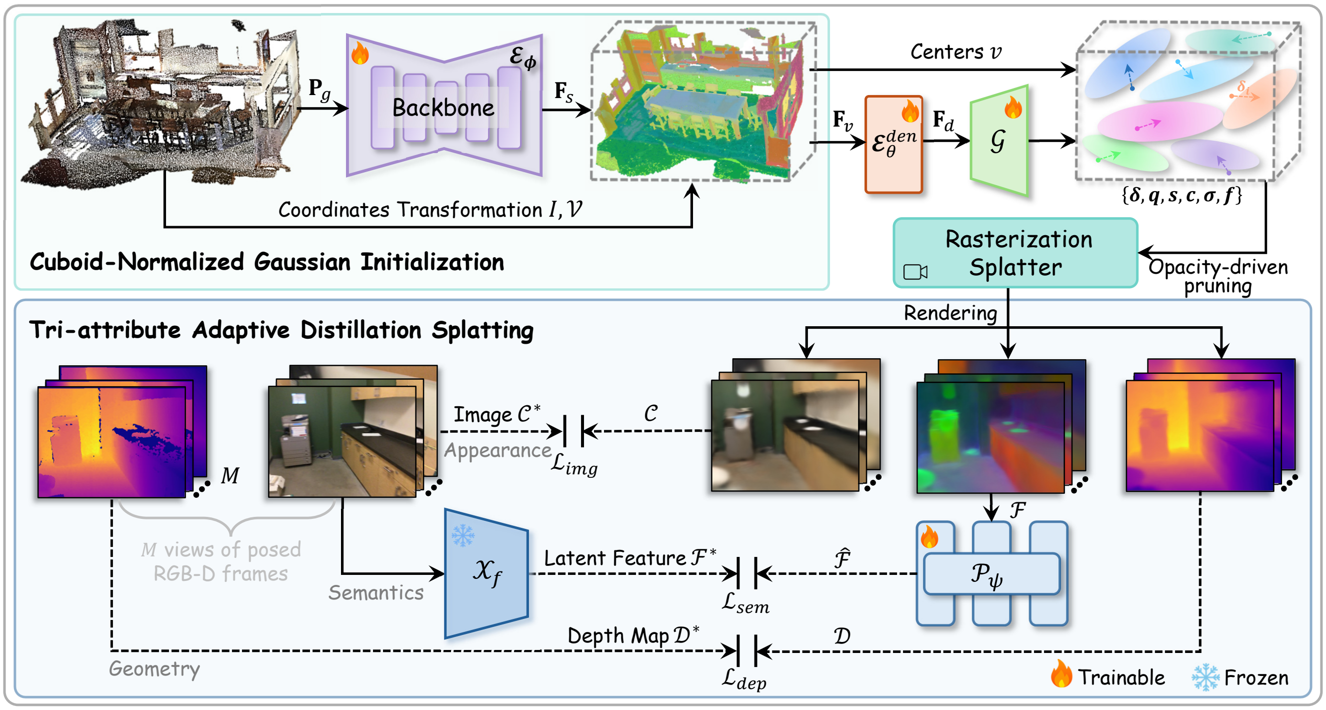

Figure 2: The overall architecture of GaussianCross. The pipeline commences with cuboid-normalized Gaussian initialization to establish coarse primitive means. Gaussian properties are subsequently decoded by $\mathcal{G}$ with a feature field. The tri-attribute adaptive distillation splatting is performed to ensure cross-modal consistency.

yielding sparse features where d_s is the channel dimension. Our objective centers on learning discriminative and reliable point-wise representations through $\mathcal{E}_\phi$ by leveraging cross-modal self-supervision signals.

To construct scale-agnostic representations, we develop a normalized cuboid volumetric encoding scheme. This spatial normalization is essential for learning generalizable scene representations across varying scales. Specifically, we perform coordinate transformation $\mathcal{I}$ to map raw positions $\mathbf{C}_g$ into a unit cube, which guarantees all scenes occupy a canonical domain while preserving relative spatial relationships. We further apply a discretization operation $\mathcal{V}$, partitioning the cube into $X \times Y \times Z$ uniformly voxels. This process is described in Eq. 6 and voxel centers $\mathbf{C}_v$ are given by:

$$\mathbf{C}_v = \mathcal{V}\big(\mathcal{I}(\mathbf{C}_g), X, Y, Z\big). \tag{6}$$

Each point $\mathbf{C}_{g,i}$ is assigned a unique voxel index $id \in \{1, 2, \dots, X \times Y \times Z\}$ determined by spatial hashing and grid resolution, yielding an index set $ids = \{id_i\}_{i=1}^m$. The voxel-wise embeddings are then attained by scattering sparse features sharing identical indices:

$$\mathbf{F}_v = \mathtt{Scatter}\big(\mathbf{F}_s, \mathcal{I}(\mathbf{C}_g), ids, \mathbf{C}_v\big) \in \mathbb{R}^{X \times Y \times Z \times d_s} \tag{7}$$

where unoccupied voxels are filled with zeros. The features $\mathbf{F}_v$ are then processed by a 3D convolutional neural network $\mathcal{E}_\theta^{den}$ to establish a dense feature volume:

$$\mathbf{F}_d = \mathcal{E}_\theta^{den}(\mathbf{F}_v) \in \mathbb{R}^{X \times Y \times Z \times d_o}, \tag{8}$$

where d_o denotes the output dimension. With the structured scene representation, we consider each voxel as an anchor and directly serve its center $\mathbf{C}_{v,i}$ as coarse mean $\boldsymbol{\nu}_i$ of the Gaussian. The voxel features $\mathbf{F}_{d,i}$ are also assigned to the i-th Gaussian. Our experiments demonstrate this cuboid-normalized initialization empirically outperforms traditional SfM-based 3DGS methods [18, 31] in representation consistency (see Fig. 3), effectively enabling direct Gaussian initialization from raw point clouds.

3.3 Tri-attribute Adaptive Distillation Splatting

To achieve self-supervised 3D representation learning, we consider novel view synthesis as a pretext task, eliminating dependency on 3D supervision while maximally utilizing available 2D data. Building upon the dense features $\mathbf{F}_d$ obtained in Sec. 3.2, we parameterize Gaussian attributes via dedicated Multi-Layer Perceptrons (MLPs) decoders with associated activations:

$$\boldsymbol{q}_i = Normalize\big(\mathcal{G}_q(\mathbf{F}_{d,i})\big), \quad \boldsymbol{s}_i = Softplus\big(\mathcal{G}_s(\mathbf{F}_{d,i})\big), \tag{9}$$

where $\mathcal{G}_q$ and $\mathcal{G}_s$ are quaternion and scaling prediction heads. Color $\boldsymbol{c}_i$ and opacity $\boldsymbol{\sigma}_i$ are similarly decoded by:

$$\boldsymbol{c}_i = Sigmoid\big(\mathcal{G}_c(\mathbf{F}_{d,i})\big), \quad \boldsymbol{\sigma}_i = Sigmoid\big(\mathcal{G}_\sigma(\mathbf{F}_{d,i})\big). \tag{10}$$

To address inaccuracy of coarse mean $\boldsymbol{\nu}_i$ initialization in representing the actual scene, we introduce a predicted offset $\boldsymbol{\delta}_i$ by:

$$\boldsymbol{\delta}_i = tanh\big(\mathcal{G}_\delta(\mathbf{F}_{d,i})\big) \cdot \Delta. \tag{11}$$

Here, Δ controls the maximum displacement magnitude. The learned offset $\boldsymbol{\delta}_i$ is then added to $\boldsymbol{\nu}_i$ yielding the refined mean $\boldsymbol{\mu}_i = \boldsymbol{\nu}_i + \boldsymbol{\delta}_i$. Concurrently, we establish a feature field to capture potential semantic cues of each anchor by projecting the dense features $\mathbf{F}_{d,i}$ into a semantic-aware embedding $\boldsymbol{q}_i$ with the dimension of d_q:

$$f_i = \mathcal{G}_f(\mathrm{F}_{d,i}), \tag{12}$$

These attributes enable modeling the scene from different perspectives and capturing comprehensive information. Although directly initializing Gaussians from voxels ensures training efficiency, inherent redundancy may compromise rendering fidelity and computational efficiency. We therefore introduce an opacity-driven pruning mechanism with a threshold τ to determine whether reserving the anchor. Finally, we can explicitly represent the 3D scene by a series of Gaussian primitives characterized by predicted properties:

$$\{\boldsymbol{\mu}_i, \boldsymbol{q}_i, \boldsymbol{s}_i, \boldsymbol{c}_i, \boldsymbol{\sigma}_i, \boldsymbol{f}_i \mid \boldsymbol{\sigma}_i > \tau\}_{i=1}^{X\times Y\times Z}. \tag{13}$$

Then, we propose tri-attribute adaptive distillation splatting to render multi-view images, depth, and feature maps, enabling the model to pursue underlying photometric appearance, geometric structure, and semantic information. The splatting is performed by projecting 3D Gaussian primitives onto M camera planes with different poses. Instead of picking specific views like [34], we randomly sample M views from the training dataset for each scene to enhance generalization ability. Color outputs $\{\boldsymbol{C}_m\}_{m=1}^{M}$ are synthesized following Eq. 3, where $\boldsymbol{C}_m \in \mathbb{R}^{H\times W\times 3}$, H and W are height and width. Subsequently, geometric regularization is established by depth map $\boldsymbol{\mathcal{D}}_m \in \mathbb{R}^{H\times W}$ generation:

$$\boldsymbol{\mathcal{D}}_m(\boldsymbol{p}) = \sum_{i\in\mathcal{N}} d_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \tag{14}$$

where d_i is the camera space z-depth of the i-th Gaussian. Our framework further integrates feature field rendering into the procedure to distill semantic-aware knowledge from a 2D visual foundation model. Unlike PonderV2 [51] that directly predicts 2D semantic labels, we consider feature correlations as intermediate supervision to guide feature learning, eliminating the requirement of ground-truth labels. The rendered feature map $\boldsymbol{\mathcal{F}}_m \in \mathbb{R}^{H\times W\times d_f}$ is denoted as:

$$\boldsymbol{\mathcal{F}}_m(\boldsymbol{p}) = \sum_{i\in\mathcal{N}} f_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j). \tag{15}$$

We employ the latent features from a pre-trained VFM $\mathcal{X}_f$ as the prior: $\boldsymbol{\mathcal{F}}_m^* = \mathcal{X}_f(\boldsymbol{C}_m^*) \in \mathbb{R}^{H\times W\times d^*}$, where $\mathcal{X}_f$ is an arbitrary 2D foundation model and $\boldsymbol{C}_m^*$ is the corresponding real color image. Nevertheless, a potential challenge lies in that the dimension d^* of $\boldsymbol{\mathcal{F}}_m^*$ is usually large, making it time-consuming to render such high-dimensional feature maps. Therefore, we tend to render a low-dimensional map ($d_f \ll d^*$). To address the dimension disparity, we implement a lightweight projection head $\mathcal{G}_{\boldsymbol{\psi}}$ to upsample $\boldsymbol{\mathcal{F}}_m$ to align with the dimension of $\boldsymbol{\mathcal{F}}_m^*$:

$$\hat{\boldsymbol{\mathcal{F}}}_m = \mathcal{P}_{\boldsymbol{\psi}}(\boldsymbol{\mathcal{F}}_m), \tag{16}$$

where $\hat{\boldsymbol{\mathcal{F}}}_m \in \mathbb{R}^{H\times W\times d^*}$. This design strategically balances computational efficiency with semantic fidelity, enabling effective distillation of 2D priors into 3D representations without compromising rendering performance.

3.4 Training Loss Functions

The principle of our design is to adhere the model to capture multi-faceted properties from raw 3D scenes and incorporate available priors from VFMs into 3D feature space. We introduce a l_1 loss denoted as $\mathcal{L}_{img}$ to measure the discrepancy of exported photo-realistic images $\boldsymbol{C}_m$ and the ground truth $\boldsymbol{C}_m^*$ aiming to capture adequate appearance details:

$$\mathcal{L}_{img} = \frac{1}{M}\sum_{m=1}^{M}\|\boldsymbol{C}_m - \boldsymbol{C}_m^*\|. \tag{17}$$

For splatted depth maps $\boldsymbol{\mathcal{D}}_m$, we also use the l_1 loss $\mathcal{L}_{dep}$ within valid pixels to regularize geometric features alignment with concomitant real depth maps $\boldsymbol{\mathcal{D}}_m^*$:

$$\mathcal{L}_{dep} = \frac{1}{M\cdot HW}\sum_{m=1}^{M}\sum_{h=1}^{H}\sum_{w=1}^{W}\mathbb{I}_{\{\boldsymbol{\mathcal{D}}_{m,h,w}^*\}}\|\boldsymbol{\mathcal{D}}_{m,h,w} - \boldsymbol{\mathcal{D}}_{m,h,w}^*\|. \tag{18}$$

where $\mathbb{I}_{\{\cdot\}}$ denotes the indicator function. Furthermore, in terms of the yielded feature maps $\hat{\boldsymbol{\mathcal{F}}}_m$ from our semantic feature field, we integrate a similarity loss $\mathcal{L}_{sem}$ to distill 2D knowledge priors by aligning with $\boldsymbol{\mathcal{F}}_m^*$ from VFMs:

$$\mathcal{L}_{sem} = \frac{1}{M}\sum_{m=1}^{M}\left[1 - \frac{\hat{\boldsymbol{\mathcal{F}}}_m \cdot \boldsymbol{\mathcal{F}}_m^*}{\|\hat{\boldsymbol{\mathcal{F}}}_m\|\|\boldsymbol{\mathcal{F}}_m^*\|}\right]. \tag{19}$$

Therefore, our cross-modal pre-training framework can work in a self-supervised manner without the requirement of human annotations, and the total loss is defined as:

$$\mathcal{L} = \lambda_{img}\mathcal{L}_{img} + \lambda_{dep}\mathcal{L}_{dep} + \lambda_{sem}\mathcal{L}_{sem}, \tag{20}$$

where λ_{img}, λ_{dep}, and λ_{sem} are weights to balance different losses.

4 Experiments

4.1 Experimental Settings

Backbone and Data. We implement our GaussianCross by Pointcept [5]. Following established practice [33, 51], we adopt a Submanifold Sparse Convolution UNet [10] (SparseUNet) as the 3D backbone $\mathcal{E}_{\boldsymbol{\phi}}$ and consider 6-dimensional attributes as input features, comprising RGB values and normal vectors. We pre-train GaussianCross on ScanNet [6] and evaluate downstream scene understanding performance on ScanNet, ScanNet200 [30], and S3DIS [1] benchmarks, respectively. *ScanNet* [6] provides 1601 3D scenes with corresponding RGB-D frames, including 20 semantic classes for semantic segmentation and 18 object categories for instance recognition. The extended challenging version, *ScanNet200* [30], shares the same data yet contains more fine-grained annotations, expanding the labels to 200 semantic categories and 198 instance types. *S3DIS* complements our evaluation with 271 indoor scans across 6 large-scale areas, annotated with 13 distinct classes. We evaluate the performance on Area5 and 6-fold cross-validation settings.

Training Details. We train GaussianCross on ScanNet [6] for 1200 epochs using 8 NVIDIA RTX 4090 GPUs with a batch size of 32. The learning rate is initialized as $2e^{-3}$ with the AdamW optimizer, modulated by a OneCycle learning rate scheduling policy. Input point clouds undergo standard geometric augmentations, including random rotation, anisotropic scaling, and flipping. Our view synthesis configuration uses 5 rendering views, each with a resolution of 480

Table 1: Parameter efficiency via linear probing. *SpUNet* means SparseUNet [10] as the backbone.

Linear Prob.	*ScanNet*		*ScanNet200*		*S3DIS Area5*		*S3DIS 6-fold*	
Methods	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc	mIoU	mAcc
○ SpUNet [4]	72.2	80.2	25.0	32.9	66.3	72.5	72.4	80.9
● PC [41]	5.6	9.7	0.5	0.9	11.4	18.6	11.7	19.0
● CSC [14]	12.6	18.1	1.3	2.1	24.4	32.0	24.9	32.5
● MSC [40]	14.1	20.3	1.5	2.5	27.9	35.5	29.9	37.9
● **Ours**	**23.3**	**30.9**	**3.6**	**5.3**	**34.7**	**44.1**	**35.9**	**45.5**

× 640. The mask ratio γ is set to 50%, and the opacity threshold τ is set to 0.3 to trade-off between rendering fidelity and computational efficiency. For semantic feature alignment, we integrate pre-trained weight from RADIOv2.5 [13] as the frozen visual encoder $\mathcal{X}_f$.

4.2 Comparison with State-of-the-Art Methods

In this section, we conduct comprehensive benchmarking of GaussianCross against existing approaches across various tasks. We start by assessing parameter efficiency by linear probing following the protocol established in Sonata [37] and data efficiency with limited scene reconstruction and point annotation data settings. We then evaluate the transfer learning performance through full fine-tuning on 3D semantic and instance segmentation tasks. In our tables, we denote ○, ●, and ● as training from scratch, self-supervised pre-training, and supervised pre-training, respectively. For more details, please refer to the supplementary materials.

4.2.1 ***Linear Probing.*** To quantify the intrinsic quality of learned representations, we implement a linear evaluation protocol where only the classification layer undergoes training while the backbone remains frozen. This parameter-efficient paradigm directly measures feature separability in the pre-trained embedding space. Results in Tab. 1 demonstrate GaussianCross's superiority, achieving 23.3%, 3.6%, 34.7%, 35.9% mIoU on ScanNet, ScanNet200, S3DIS Area5 and 6-fold, respectively. Although GaussianCross outperforms other methods, the performance discrepancy between linear probing and full training reveals that current self-supervised objectives remain to be further optimized. This suggests that while GaussianCross excels in learning transferable representations, there is still room for improvement in the pre-training process itself.

4.2.2 ***Data Efficiency.*** In Tab. 2, we systematically evaluate the data efficiency by fine-tuning on ScanNet Data Efficient benchmark [14] with limited scenes and point annotations. The results on both configurations exhibit impressive improvements compared to learning from scratch baselines (*cf*. ○). In the case of extreme data scarcity and limited point annotations, GaussianCross also obtains the best performance among all other counterparts, with 32.1% and 61.7% mIoU on 1% scenes and 20 points per scene scenarios. Notably, GaussianCross can even outperform the supervised pre-training model (*e.g.* ● PPT [39]), providing empirical validation that our cross-modal self-supervised objectives learn more transferable structural priors than manually curated supervision.

Table 2: Data efficiency on ScanNet Data Efficient benchmark [14] by limited scenes and point annotations.

Data Eff.	*Limited Scenes (Pct.)*				*Limited Annotations (Pts.)*			
Methods	1%	5%	10%	20%	20	50	100	200
○ SpUNet [4]	26.0	47.8	56.7	62.9	41.9	53.9	62.2	65.5
● CSC [14]	28.9	49.8	59.4	64.6	55.5	60.5	65.9	68.2
● MSC [40]	29.2	50.7	61.0	64.9	60.1	66.8	69.7	70.7
● GC [33]	30.7	52.9	62.0	66.5	61.2	67.3	70.3	71.8
● PPT [39]	31.3	52.3	62.8	66.4	60.6	67.5	70.8	72.2
● **Ours**	**32.1**	**53.5**	**64.2**	**67.3**	**61.7**	**68.5**	**72.2**	**73.3**
Δ	**+6.1**	**+5.7**	**+7.5**	**+4.4**	**+19.8**	**+14.6**	**+10.0**	**+7.8**

This evidence positions GaussianCross as a theoretically grounded framework for label-efficient 3D scene understanding.

4.2.3 ***3D Semantic Segmentation.*** In Tab. 3, we present mIoU (%) results for 3D semantic segmentation on ScanNet [6], ScanNet200 [30], and S3DIS [1] benchmarks. Under the self-supervised pre-training setting (*cf*. ●), GaussianCross attains the best performance across all datasets, demonstrating a 76.0% mIoU on ScanNet validation set - a 2.5% absolute improvement over prior neural rendering approaches such as GS^3 [21] and Ponder [15]. Moreover, our method outperforms multi-datasets pre-training strategies MSC [40] and PPT Unsup. [39] by 4.8% and 3.2% on ScanNet200, respectively. Although supervised pre-training baselines (*cf*. ●) maintain marginal advantages on ScanNet (≤1%), our method establishes new state-of-the-art on ScanNet200 by enhanced semantic discriminability. This demonstrates the generalization of our method in learning transferable 3D representations and the potential of processing semantically complex scenarios. Consistent performance gains are observed on S3DIS under both Area5 (72.1%) and 6-fold cross-validation (76.8%) settings, confirming its robustness.

4.2.4 ***3D Instance Segmentation.*** In Tab. 4, we compare the results of instance segmentation on ScanNet [6] and ScanNet200 [30] validation splits with PointGroup [17] as the baseline model. We report AP$_{25}$, AP$_{50}$, and mAP for comprehensive evaluation, following the common practice [17, 42]. On ScanNet, the achieved 62.7% AP$_{50}$ represents a 6.2% improvement over the baseline without pre-training, significantly outperforming previous contrastive learning methods that typically struggle with instance boundary discrimination. The performance gap is more pronounced on ScanNet200, where GaussianCross attains 30.6% mAP. The consistent superiority suggests that our method provides complementary benefits beyond pure color rendering (GS^3), underscoring the effectiveness of our designs in instance-level understanding.

4.3 Ablation Studies and Analysis

We perform systematic ablation studies to investigate the efficacy of our core designs and analyze the effect of different parameter choices. We utilize 3D semantic segmentation and assess the performance on both ScanNet [6] and ScanNet200 [30] validation splits for a comprehensive evaluation.

Table 3: 3D semantic segmentation results. The best results are highlighted in bold, and the second-best results are in underlined.

Semantic Segmentation				ScanNet	ScanNet200	S3DIS	
Methods	Venue	Pre-training Datasets	Type	Val mIoU	Val mIoU	Area5	6-fold
Supervised Learning from Scratch							
○ PointNeXt [28]	NeurIPS 2022	✗	✗	71.5	-	70.5	74.9
○ StFormer [19]	CVPR 2022	✗	✗	74.3	-	72.0	-
○ PTv1 [48]	ICCV 2021	✗	✗	70.6	27.8	70.4	65.4
○ PTv2 [38]	NeurIPS 2022	✗	✗	75.4	30.2	71.6	75.1
○ SpUNet [4]	CVPR 2019	✗	✗	72.2	25.0	66.3	72.4
Self-supervised Pre-training							
● GS^3 [21]	arXiv 2024	ScanNet	Rendering	73.4$_{+1.2}$	-	70.1$_{+3.8}$	-
● Ponder [15]	CVPR 2023	ScanNet	Rendering	73.5$_{+1.3}$	-	-	-
● CSC [14]	CVPR 2021	ScanNet	Contrast	73.8$_{+1.6}$	26.4$_{+1.4}$	70.7$_{+4.4}$	75.5$_{+3.1}$
● PC [41]	ECCV 2020	ScanNet	Contrast	74.1$_{+1.9}$	26.2$_{+1.2}$	70.3$_{+4.0}$	74.7$_{+2.3}$
● MSC [40]	CVPR 2023	ScanNet, ArkitScenes	Contrast	75.5$_{+3.3}$	32.0$_{+7.0}$	70.7$_{+4.4}$	-
● GC [33]	CVPR 2024	ScanNet	Contrast	75.7$_{+3.5}$	30.0$_{+5.0}$	72.0$_{+5.7}$	-
● PPT Unsup. [39]	CVPR 2024	ScanNet, Structure3D, S3DIS	Contrast	75.8$_{+3.6}$	30.4$_{+5.4}$	71.9$_{+5.6}$	-
● **GaussianCross**	-	ScanNet	Rendering	**76.0**$_{+3.8}$	**34.3**$_{+9.3}$	**72.1**$_{+5.8}$	**76.8**$_{+4.4}$
Supervised Pre-training							
● PPT Sup. [39]	CVPR 2024	ScanNet, Structure3D, S3DIS	3D Sup.	76.4$_{+4.2}$	31.9$_{+6.9}$	72.7$_{+6.4}$	78.1$_{+5.7}$
● PonderV2 [51]	arXiv 2024	ScanNet, Structure3D, S3DIS	2D Sup.	77.0$_{+4.8}$	32.3$_{+7.3}$	73.2$_{+6.9}$	79.9$_{+7.4}$
● ARKit LM [16]	CVPR 2025	ALS200, ScanNet/ScanNet200	3D Sup.	77.0$_{+4.8}$	30.6$_{+5.6}$	-	-

Table 4: 3D instance segmentation performance on ScanNet [6] and ScanNet200 [30]. *PG* indicates PointGroup [17].

Ins. Seg.	ScanNet			ScanNet200		
Methods	AP$_{25}$	AP$_{50}$	mAP	AP$_{25}$	AP$_{50}$	mAP
○ PG [17]	72.8	56.9	36.0	32.2	24.5	15.8
● PC [41]	-	58.0	-	-	24.9	-
● GS^3 [21]	-	59.2	37.0	-	-	-
● CSC [14]	-	59.4	-	-	25.2	-
● MSC [40]	74.7	59.6	39.3	34.3	26.8	17.3
● GC [33]	-	62.3	-	-	27.5	-
● **Ours**	**77.0**$_{+4.2}$	**62.7**$_{+6.2}$	**40.8**$_{+4.8}$	**38.4**$_{+5.8}$	**30.6**$_{+6.1}$	**20.6**$_{+4.8}$

Table 5: Ablation study of rendering targets. *img., dep., sem.* denote RGB image, depth, and semantic feature maps.

w/ img.	*w/ dep.*	*w/ sem.*	ScanNet		ScanNet200	
			mIoU	mAcc	mIoU	mAcc
✓	✗	✗	75.0	82.9	32.8	42.1
✓	✓	✗	75.3	83.0	33.0	42.4
✓	✗	✓	75.5	83.0	33.7	42.5
✓	✓	✓	**76.0**	**83.5**	**34.3**	**43.1**

Core Designs. In Fig. 3 top, we analyze the impact of our core designs by recording the PSNR of rendered images during pre-training. We observe that using traditional Gaussian mean initialization leads to a significant drop (14.9 *v.s.* 18.2), indicating that the model struggles to learn meaningful representations. The variant without Gaussian mean refinement achieves a PSNR of 17.6, suggesting that the learned offset can help with accurate scene representation. Different rendering targets specialize in distinct attributes of 3D scenes, thus impacting the representations. Therefore, we explore the synergistic effects of multi-target rendering in Tab. 5. The baseline using only photometric reconstruction achieves 75.0% mIoU on ScanNet and 32.8% on ScanNet200, establishing a performance floor that highlights the limitation of pure appearance modeling. Incorporating geometric consistency by depth supervision yields a slight improvement, revealing that explicit spatial cues enhance 3D structure understanding. The performance is elevated to 75.5% and 33.7% when bridging semantic alignment via knowledge distillation. The optimal configuration combining photometric, geometric, and semantic targets achieves 76.0% and 34.1% mIoU, respectively, proving the complementary nature of tripartite rendering.

Masking Ratio γ. We adopt a stochastic masking strategy governed by parameter γ to occlude a portion of input regions during pre-training. To test its impact, we vary γ from 10% to 90% in 20% increments. As evidenced in Fig. 3, the results show that better

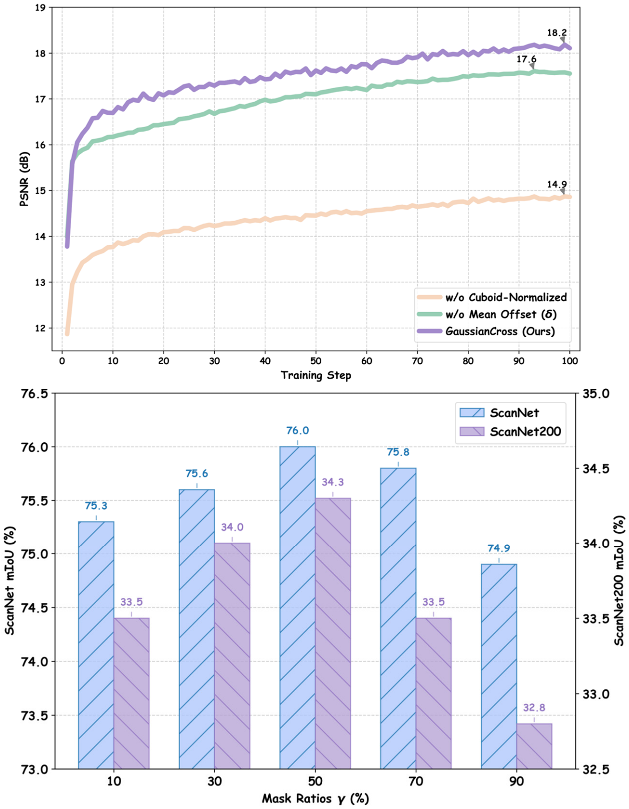

Figure 3: Ablation study of core designs and masking ratio γ.

Table 6: Impact of opacity threshold τ.

τ	Sc. mIoU	Sc.200 mIoU	Average mIoU	PSNR↑ (dB)	Memory↓ (MB)	Time↓ (s/scene)
0.1	75.4	33.3	54.3	18.16	5614	0.265
0.3	**76.0**	**34.3**	**55.1**	**18.18**	5296	0.255
0.5	75.6	33.6	54.6	17.94	5251	0.251
0.7	74.8	33.0	53.9	17.62	**5204**	**0.241**

performance can be achieved when γ equals 50%, with perturbations within ±20% causing statistically insignificant performance deviations. However, extreme values of 10% or 90% induce significant performance degradation, revealing the model's sensitivity to excessive occlusion or exposure. This suggests the importance of balanced masking in self-supervised learning.

Opacity Threshold τ. We introduce an opacity-driven pruning strategy to determine the visibility of each anchor Gaussian and optimize the rendering quality. In Tab. 6, we examine τ from 0.1 to 0.7. We also report memory consumption and training time for each scene. When increasing the threshold from 0.1 to 0.3, the performance is also improved, while further raising the value to 0.5 or 0.7 will lead to a drop. This is because a higher threshold will filter out more anchors. Therefore, we set τ to 0.3 in our experiments to balance rendering quality and amount of information.

Table 7: Effectiveness of M and $\mathcal{X}_f$.

Datasets & Metrics		Rendering Views M			VFMs $\mathcal{X}_f$		
		2	5	8	CLIP	DINOv2	RADIO
ScanNet	mIoU	75.8	**76.0**	75.6	75.4	74.9	**76.0**
	mAcc	**83.7**	83.5	83.5	83.5	83.1	83.5
ScanNet200	mIoU	33.9	**34.3**	34.0	33.9	33.6	**34.3**
	mAcc	42.8	**43.1**	42.9	42.9	43.0	**43.1**
Average	mIoU	54.3	**55.1**	54.6	54.6	54.2	**55.1**
	mAcc	62.4	**63.3**	62.1	63.2	63.0	**63.3**

Visual Foundation Models $\mathcal{X}_f$. GaussianCross's architectural flexibility allows for seamless integration with diverse visual foundation models. However, different models excel at distinctive properties that affect scene understanding. Results in Tab. 6 indicate notable performance variance across foundation models, with CLIP [29] and DINOv2 [25] yielding suboptimal results. Because of the agglomerative multi-domain training strategy, RADIO [13] achieves optimal 76.0% mIoU on ScanNet and 34.1% on ScanNet200.

Number of Rendering Views M. Theoretically, more views could offer broader supervision for pre-training, but it also introduce extra computational costs and increase training time. Thus, we investigate the impact of M in Tab. 7. We set M to 5 in our experiments to balance the performance and efficiency.

5 Conclusion

In this paper, we present GaussianCross, an innovative framework leveraging 3DGS for cross-modal self-supervised point cloud representation learning. Our cuboid-normalized Gaussian initialization establishes scale-consistent scene representations by transforming raw point clouds into a structured collection of Gaussian primitives within a canonical space. The proposed tri-attribute adaptive distillation splatting jointly optimizes photometric appearance, geometric structure, and semantic consistency by differentiable rendering with a feature field while effectively distilling the 2D visual foundation model for enhanced semantic awareness. Extensive experiments demonstrate state-of-the-art performance across multiple benchmarks, including linear probing and transfer learning. Comprehensive ablation studies further validate the effectiveness by systematically analyzing core design components. For future work, we will explore scalable backbone architectures to enhance representation capability and investigate the potential of scaling up GaussianCross to large-scale multi-source datasets, aiming to advance the development of 3D foundation models.

Acknowledgments

The research work was conducted in the JC STEM Lab of Machine Learning and Computer Vision funded by The Hong Kong Jockey Club Charities Trust.

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 2016. 3d semantic parsing of large-scale indoor spaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[2] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. 2024. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 19457–19467.

[3] Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. 2024. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *European Conference on Computer Vision (ECCV)*. Springer, 370–386.

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

[5] Pointcept Contributors. 2023. Pointcept: A Codebase for Point Cloud Perception Research. https://github.com/Pointcept/Pointcept.

[6] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. 2017. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*. Association for Computational Linguistics, 4171–4186.

[8] Guofan Fan, Zekun Qi, Wenkai Shi, and Kaisheng Ma. 2024. Point-gcc: Universal self-supervised 3d scene pre-training via geometry-color contrast. In *Proceedings of the 32nd ACM International Conference on Multimedia*. 4709–4718.

[9] Pedro F Felzenszwalb and Daniel P Huttenlocher. 2004. Efficient graph-based image segmentation. *International Journal of Computer Vision* 59 (2004), 167–181.

[10] Benjamin Graham and Laurens Van der Maaten. 2017. Submanifold sparse convolutional networks. *arXiv preprint arXiv:1706.01307* (2017).

[11] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. 2022. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 16000–16009.

[12] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 9729–9738.

[13] Greg Heinrich, Mike Ranzinger, Hongxu, Yin, Yao Lu, Jan Kautz, Andrew Tao, Bryan Catanzaro, and Pavlo Molchanov. 2025. RADIOv2.5: Improved Baselines for Agglomerative Vision Foundation Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[14] Ji Hou, Benjamin Graham, Matthias Nießner, and Saining Xie. 2021. Exploring data-efficient 3d scene understanding with contrastive scene contexts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 15587–15597.

[15] Di Huang, Sida Peng, Tong He, Honghui Yang, Xiaowei Zhou, and Wanli Ouyang. 2023. Ponder: Point cloud pre-training via neural rendering. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, 16089–16098.

[16] Guangda Ji, Silvan Weder, Francis Engelmann, Marc Pollefeys, and Hermann Blum. 2024. ARKit LabelMaker: A New Scale for Indoor 3D Scene Understanding. *arXiv preprint arXiv:2410.13924* (2024).

[17] Li Jiang, Hengshuang Zhao, Shaoshuai Shi, Shu Liu, Chi-Wing Fu, and Jiaya Jia. 2020. Pointgroup: Dual-set point grouping for 3d instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 4867–4876.

[18] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 2023. 3d gaussian splatting for real-time radiance field rendering. *ACM TOG* 42, 4 (2023), 139–1.

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

[20] Oliver Lemke, Zuria Bauer, René Zurbrügg, Marc Pollefeys, Francis Engelmann, and Hermann Blum. 2024. Spot-Compose: A Framework for Open-Vocabulary Object Retrieval and Drawer Manipulation in Point Clouds. *Internationl Conference on Robotics and Automation Workshops (ICRAW)* (2024).

[21] Hao Liu, Minglin Chen, Yanni Ma, Haihong Xiao, and Ying He. 2024. Point Cloud Unsupervised Pre-training via 3D Gaussian Splatting. *arXiv preprint arXiv:2411.18667* (2024).

[22] Moyun Liu, Youping Chen, Jingming Xie, Yijie Zhu, Yang Zhang, Lei Yao, Zhenshan Bing, Genghang Zhuang, Kai Huang, and Joey Tianyi Zhou. 2024. MENet: Multi-modal mapping enhancement network for 3D object detection in autonomous driving. *IEEE Transactions on Intelligent Transportation Systems* 25, 8 (2024), 9397–9410.

[23] Leland McInnes, John Healy, and James Melville. 2018. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426* (2018).

[24] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. 2021. Nerf: Representing scenes as neural radiance fields for view synthesis. *Commun. ACM* 65, 1 (2021), 99–106.

[25] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193* (2023).

[26] Yatian Pang, Wenxiao Wang, Francis EH Tay, Wei Liu, Yonghong Tian, and Li Yuan. 2022. Masked autoencoders for point cloud self-supervised learning. In *European Conference on Computer Vision (ECCV)*, Vol. 13662. Springer, 604–621.

[27] Zekun Qi, Runpei Dong, Guofan Fan, Zheng Ge, Xiangyu Zhang, Kaisheng Ma, and Li Yi. 2023. Contrast with reconstruct: Contrastive 3d representation learning guided by generative pretraining. In *International Conference on Machine Learning*. PMLR, 28223–28243.

[28] Guocheng Qian, Yuchen Li, Houwen Peng, Jinjie Mai, Hasan Hammoud, Mohamed Elhoseiny, and Bernard Ghanem. 2022. Pointnext: Revisiting pointnet++ with improved training and scaling strategies. *Advances in Neural Information Processing Systems (NeurIPS)* (2022).

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*. 8748–8763.

[30] David Rozenberszki, Or Litany, and Angela Dai. 2022. Language-grounded indoor 3d semantic segmentation in the wild. In *European Conference on Computer Vision (ECCV)*. Springer, 125–141.

[31] Noah Snavely, Steven M Seitz, and Richard Szeliski. 2006. Photo tourism: exploring photo collections in 3D. In *ACM siggraph 2006 papers*. Vol. 25. 835–846.

[32] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*. 5998–6008.

[33] Chengyao Wang, Li Jiang, Xiaoyang Wu, Zhuotao Tian, Bohao Peng, Hengshuang Zhao, and Jiaya Jia. 2024. Groupcontrast: Semantic-aware self-supervised representation learning for 3d understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 4917–4928.

[34] Jiaxu Wang, Ziyi Zhang, Junhao He, and Renjing Xu. 2024. PFGS: High Fidelity Point Cloud Rendering via Feature Splatting. In *European Conference on Computer Vision*. Springer, 193–209.

[35] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. 2021. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689* (2021).

[36] Yunsong Wang, Tianxin Huang, Hanlin Chen, and Gim Hee Lee. 2024. FreeSplat: Generalizable 3D Gaussian Splatting Towards Free-View Synthesis of Indoor Scenes. *Advances in Neural Information Processing Systems (NeurIPS)* (2024).

[37] Xiaoyang Wu, Daniel DeTone, Duncan Frost, Tianwei Shen, Chris Xie, Nan Yang, Jakob Engel, Richard Newcombe, Hengshuang Zhao, and Julian Straub. 2025. Sonata: Self-Supervised Learning of Reliable Point Representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

[38] Xiaoyang Wu, Yixing Lao, Li Jiang, Xihui Liu, and Hengshuang Zhao. 2022. Point transformer v2: Grouped vector attention and partition-based pooling. *Advances in Neural Information Processing Systems (NeurIPS)* (2022).

[39] Xiaoyang Wu, Zhuotao Tian, Xin Wen, Bohao Peng, Xihui Liu, Kaicheng Yu, and Hengshuang Zhao. 2024. Towards large-scale 3d representation learning with multi-dataset point prompt training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 19551–19562.

[40] Xiaoyang Wu, Xin Wen, Xihui Liu, and Hengshuang Zhao. 2023. Masked scene contrast: A scalable framework for unsupervised 3d representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

[41] Saining Xie, Jiatao Gu, Demi Guo, Charles R Qi, Leonidas Guibas, and Or Litany. 2020. Pointcontrast: Unsupervised pre-training for 3d point cloud understanding. In *European Conference on Computer Vision (ECCV)*, Vol. 12348. Springer, 574–591.

[42] Lei Yao, Yi Wang, Moyun Liu, and Lap-Pui Chau. 2024. SGIFormer: Semantic-guided and geometric-enhanced interleaving transformer for 3D instance segmentation. *IEEE Transactions on Circuits and Systems for Video Technology* (2024).

[43] Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. 2024. Gaussian Grouping: Segment and Edit Anything in 3D Scenes. In *European Conference on Computer Vision (ECCV)*, Vol. 15087. Springer, 162–179.

[44] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. 2023. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE.

[45] Hai-Tao Yu and Mofei Song. 2024. Mm-point: Multi-view information-enhanced multi-modal self-supervised 3d point cloud understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 6773–6781.

[46] Xumin Yu, Lulu Tang, Yongming Rao, Tiejun Huang, Jie Zhou, and Jiwen Lu. 2022. Point-bert: Pre-training 3d point cloud transformers with masked point modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and*

Pattern Recognition (CVPR). IEEE, 19291–19300.

[47] Yuanwen Yue, Anurag Das, Francis Engelmann, Siyu Tang, and Jan Eric Lenssen. 2024. Improving 2d feature representations by 3d-aware fine-tuning. In *European Conference on Computer Vision (ECCV)*, Vol. 15060. Springer, 57–74.

[48] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. 2021. Point transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE.

[49] Ying Zheng, Lei Yao, Yuejiao Su, Yi Zhang, Yi Wang, Sicheng Zhao, Yiyi Zhang, and Lap-Pui Chau. 2025. A survey of embodied learning for object-centric robotic manipulation. *Machine Intelligence Research* (2025), 1–39.

[50] Shijie Zhou, Haoran Chang, Sicheng Jiang, Zhiwen Fan, Zehao Zhu, Dejia Xu, Pradyumna Chari, Suya You, Zhangyang Wang, and Achuta Kadambi. 2024. 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)*. IEEE, 21676–21685.

[51] Haoyi Zhu, Honghui Yang, Xiaoyang Wu, Di Huang, Sha Zhang, Xianglong He, Tong He, Hengshuang Zhao, Chunhua Shen, Yu Qiao, et al. 2023. Ponderv2: Pave the way for 3d foundataion model with a universal pre-training paradigm. *arXiv preprint arXiv:2310.08586* (2023).

[52] Yijie Zhu, Yibo Lyu, Zitong Yu, Rui Shao, Kaiyang Zhou, and Liqiang Nie. 2025. EmoSym: A Symbiotic Framework for Unified Emotional Understanding and Generation via Latent Reasoning. In *Proceedings of the 33nd ACM International Conference on Multimedia*.

Table S.2: Semantic segmentation settings of parameter efficiency [37], data efficiency [14], and full fine-tuning on ScanNet [6], ScanNet200 [14], and S3DIS [1].

Config	Value		
	ScanNet	ScanNet200	S3DIS
Optimizer	AdamW		
Betas	(0.9, 0.95)		
Weight Decay	0.05		
Learning Rate	0.005	0.005	0.003
Learning Rate Scheduler	Cosine		
Batch Size	32	32	24
Data Efficiency Batch Size	24	-	-
Epochs	800	800	3000
Warmup Epochs	40	40	150
Crop Size	102400	102400	204800
Grid Sampling	0.02m		

Table S.3: Instance segmentation settings on ScanNet [6] and ScanNet200 [14].

Config	Value	
	ScanNet	ScanNet200
Optimizer	AdamW	
Betas	(0.9, 0.95)	
Weight Decay	0.05	
Learning Rate	0.005	
Learning Rate Scheduler	Cosine	
Batch Size	12	24
Epochs	800	
Warmup Epochs	40	
Crop Size	Sample rate 0.8	
Grid Sampling	0.02m	

Table S.1: Implementation details of GaussianCross.

Config	Value
Training Details	
Optimizer	AdamW
Betas	(0.9, 0.95)
Weight Decay	0.05
Learning Rate	0.002
Learning Rate Scheduler	Cosine
Batch Size	32
Epochs	1200
Warmup Epochs	60
Mask Ratio	50%
Masking Strategy	Random
Data Augmentation	
Random Rotation	z, $[-\pi, \pi]$, p: 1.0 x, $[-\pi/64, \pi/64]$, p: 1.0 y, $[-\pi/64, \pi/64]$, p: 1.0
Random Scaling	$[0.9, 1.1]$, p: 1.0
Random Flip	p : 0.5
Shuffle Point	p: 1.0

A Appendix Overview

In this supplementary material, we provide more details about our proposed GaussianCross. Specifically, we demonstrate more qualitative results, including visualization of learned representations, rendered images, depth maps, and semantic-aware feature maps. We also visualize the zero-shot representation of GaussianCross on S3DIS [1] and ScanNet++ [44]. In addition, we include implementation details for self-supervised representation learning and fine-tuning on downstream tasks.

B Qualitative Results

B.1 Visualization of Learned Representations

In Fig. S.1, we visualize input point clouds, UMAP [23] results of learned representations, and corresponding synthetic RGB images, depth maps, and semantic-aware feature maps. From the results, we can observe that the learned point cloud representations are well clustered by UMAP on ScanNet [6], indicating that our GaussianCross can effectively learn meaningful and expressive representations. For example, as shown in the second row, our learned representations are able to distinguish chairs and tables, proving that our model can reveal potential spatial relationships from the input point clouds by self-supervised learning.

The color images, depth maps, and feature maps are rendered by our tri-attribute adaptive distillation splatting module during the pre-training process. Benefiting from the cuboid-normalized Gaussian initialization, our model can be generalizable to scale-variant point clouds. For instance, both the classroom (second row) and apartment (third row) scenes are well rendered with the correct colors and depth information. As for the semantic-aware feature maps, they can clearly recognize the semantic categories of the objects across different scenes, which is attributed to the incorporation of knowledge from 2D visual foundation models.

B.2 Spatial Matching

B.2.1 In Domain Representation. To further validate the quality of the learned representations by GaussianCross, we visualize the dense spatial matching [37] results by some examples. Specifically, we select one query point from each scene and calculate the cosine similarity between the query point and others in the scene. We demonstrate the activation maps of the cosine similarity scores, where the brighter regions indicate higher similarity. The results on ScanNet [6] are shown in Fig. S.2 with red cross marks highlighting the query points. We can observe that the learned representations are able to match the query points with their corresponding categories. For example, GaussianCross can successfully match the query points of sofa, monitor, bed, table, and wall across scenes. This indicates that the model can learn discriminative representations, which is beneficial for downstream tasks such as semantic segmentation and instance segmentation.

B.2.2 Zero-shot Representation. In Fig. S.3 and Fig. S.4, we visualize the zero-shot representation of GaussianCross on S3DIS [1] and ScanNet++ [44]. We directly apply the pre-trained weight on ScanNet to these two unseen datasets without any fine-tuning and

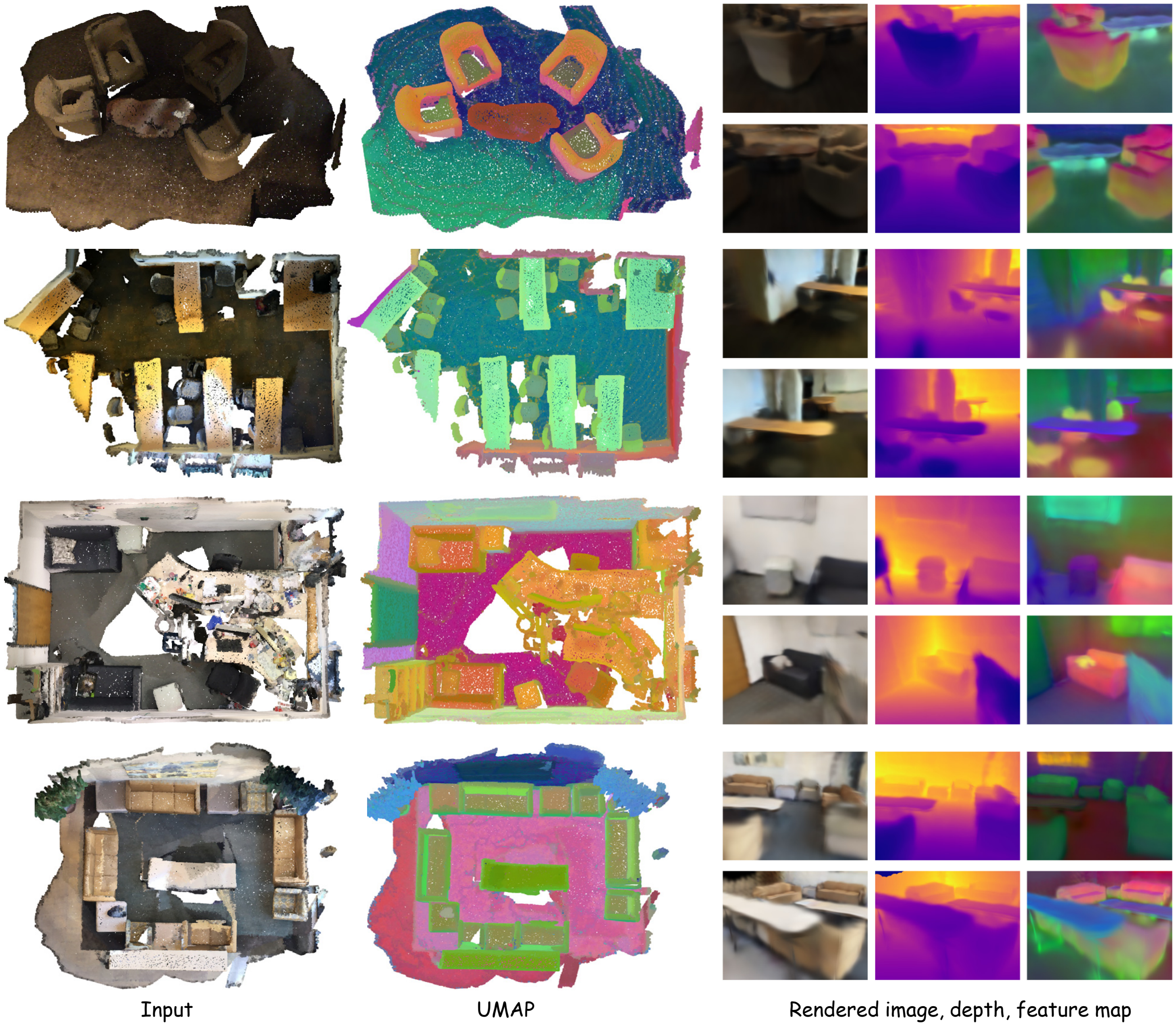

Figure S.1: Qualitative results of GaussianCross on ScanNet [6]. We visualize the input point cloud and learned point representations using UMAP [23]. We also present the corresponding rendered images, depth maps, and semantic-aware feature maps.

then visualize the results similar to Fig. S.2. From the figures, we find that GaussianCross demonstrates generalization ability to out-of-domain datasets.

B.3 Comparison with Ground Truth

In Fig. S.5, we provide a qualitative comparison of GaussianCross rendered images and depth maps with ground truth. We also show the synthesized semantic-aware feature maps. We can observe that the rendered images and depth maps are visually similar to the ground truth. Although there are some artifacts in the rendered images, the overall quality is still acceptable, and the rendered feature maps can help to alleviate this issue to some extent. Meanwhile, the depth information is also well-preserved to guarantee spatial consistency. This indicates that our tri-attribute adaptive distillation splatting can efficiently learn photometric appearance, geometrical structure, and semantic information simultaneously.

C Experimental Details

C.1 Pre-training

We implement our GaussianCross using Pointcept [5] based on PyTorch. The self-supervised pre-training is conducted on ScanNet [6]. The training details and data augmentations for the pre-training process are summarized in Tab. S.1. We adopt a 5-layer submanifold sparse convolutional U-Net [4] (SparseUNet34C) as the point cloud

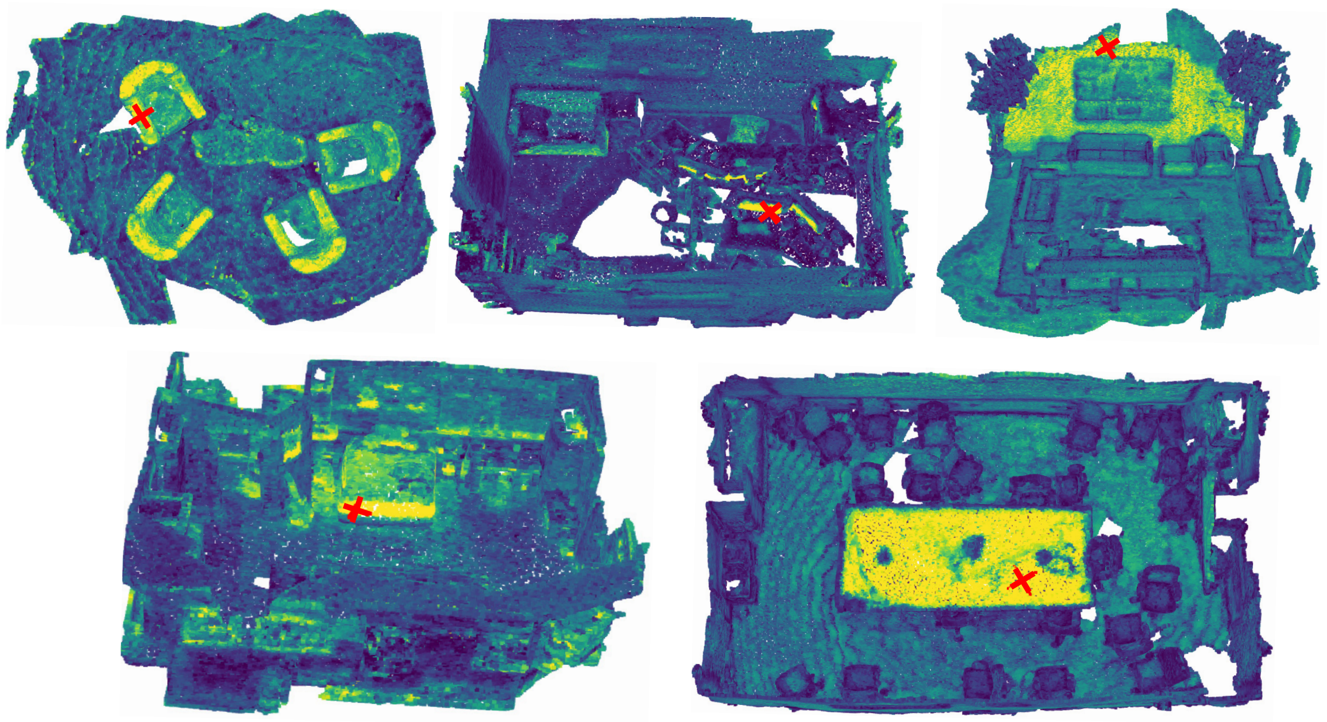

Figure S.2: Visualization of activation maps of cosine similarity scores on ScanNet [6]. The query points are highlighted with red cross marks.

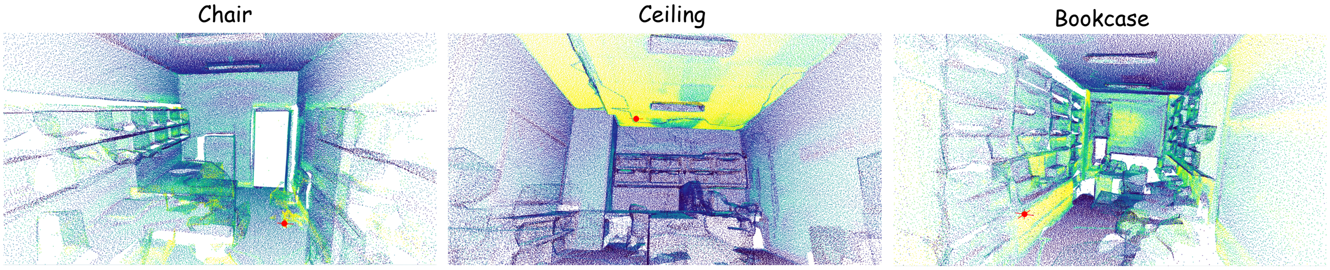

Figure S.3: Zero-shot representation of GaussianCross on S3DIS [1]. The query points are highlighted with red circles.

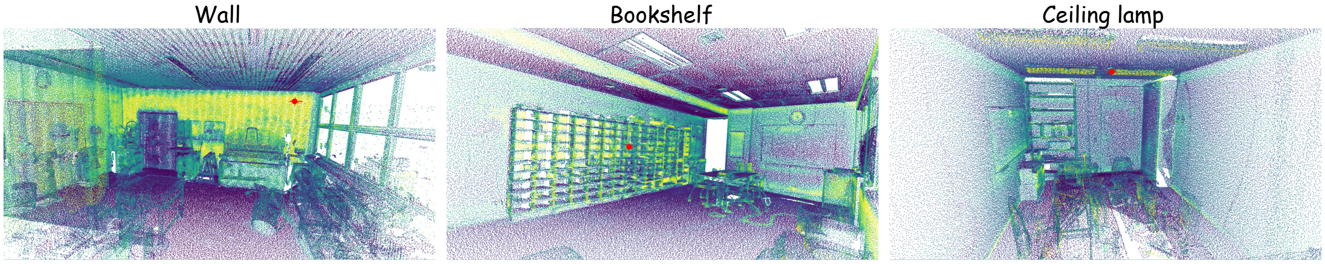

Figure S.4: Zero-shot representation of GaussianCross on ScanNet++ [44]. The query points are highlighted with red circles.

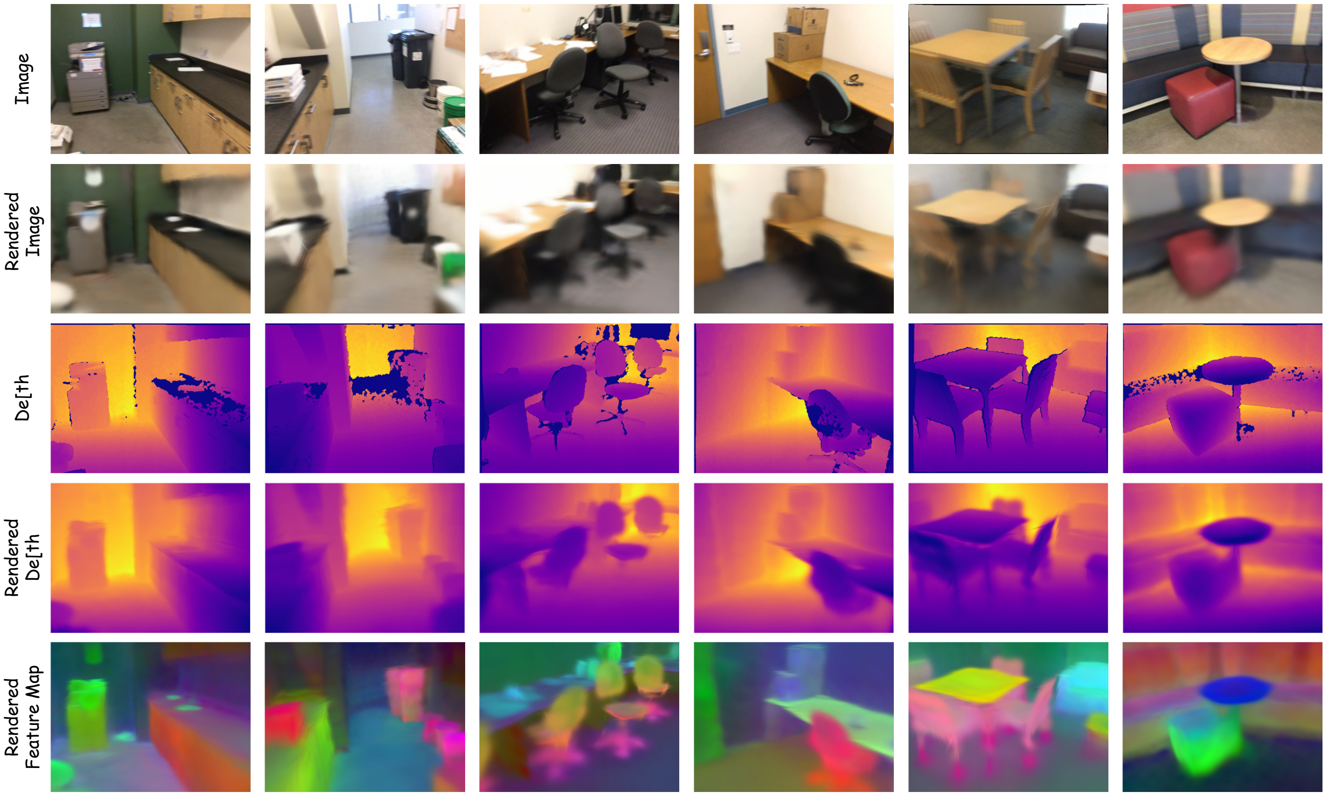

Figure S.5: Qualitative comparison of GaussianCross rendered images, depth, and semantic-aware feature maps with ground truth.

backbone for performance comparison and ablation studies similar to MSC [40], PPT [39], and GC [33].

C.2 Downsteam Tasks

We use the same backbone architecture as the pre-training process for downstream tasks. The training details for semantic segmentation and instance segmentation are demonstrated in Tab. S.2 and Tab. S.3, respectively. For parameter efficiency, data efficiency, and full fine-tuning, we follow the same settings. All downstream tasks are trained on 4 NVIDIA 4090 GPUs.