GeGS-PCR: Effective and Robust 3D Point Cloud Registration with Two-Stage Color-Enhanced Geometric-3DGS Fusion

Jiayi Tian[1], Haiduo Huang[1], Tian Xia[1], Wenzhe Zhao[1], Pengju Ren[1,*]
[1] National Key Laboratory of Human-Machine Hybrid Augmented Intelligence,
National Engineering Research Center of Visual Information and Applications,
Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University,
Xi'an, Shaanxi, China
{tianreg, huanghd}@stu.xjtu.edu.cn
{tian_xia, wenzhe, pengjuren}@xjtu.edu.cn

Abstract

We address the challenge of point cloud registration using color information, where traditional methods relying solely on geometric features often struggle in low-overlap and incomplete scenarios. To overcome these limitations, we propose GeGS-PCR, a novel two-stage method that combines geometric, color, and Gaussian information for robust registration. Our approach incorporates a dedicated color encoder that enhances color features by extracting multi-level geometric and color data from the original point cloud. We introduce the **Ge**ometric-3D**GS** module, which encodes the local neighborhood information of colored superpoints to ensure a globally invariant geometric-color context. Leveraging LORA optimization, we maintain high performance while preserving the expressiveness of 3DGS. Additionally, fast differentiable rendering is utilized to refine the registration process, leading to improved convergence. To further enhance performance, we propose a joint photometric loss that exploits both geometric and color features. This enables strong performance in challenging conditions with extremely low point cloud overlap. We validate our method by colorizing the Kitti dataset as ColorKitti and testing on both Color3DMatch and Color3DLoMatch datasets. Our method achieves state-of-the-art performance with *Registration Recall* at 99.9%, *Relative Rotation Error* as low as 0.013, and *Relative Translation Error* as low as 0.024, improving precision by at least a factor of 2.

1 Introduction

Fast and stable point cloud registration is a crucial technology in computer vision [1] and embodied intelligence [2], serving as the foundation for various practical applications, such as 3D scene reconstruction [3], semantic scene segmentation [4], and large-scale perception and mapping [5]. In essence, point cloud registration involves aligning two overlapping 3D point clouds using a rigid transformation through a series of estimation steps.

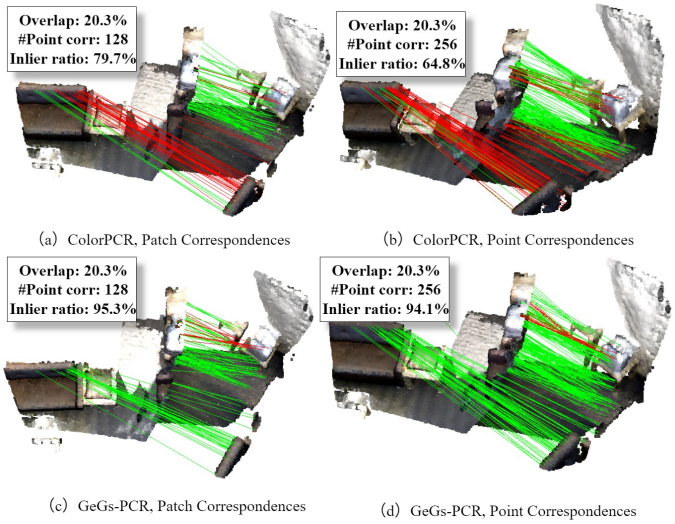
Recent advancements in deep learning have accelerated the development of 3D point cloud representation [6, 7] and differentiable optimization techniques [8, 9]. Previous works have focused on keypoints and correspondences, leveraging specialized neural networks to extract features from point clouds, and subsequently determining the rigid transformation using robust estimators like RANSAC [10, 11]. Inspired by image matching, recent research has highlighted the significance of

*Pengju Ren is the corresponding author

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

local neighborhood information [12, 13], by matching keypoints (superpoints) based on the detection of overlapping patches. This has led to advances in point cloud registration methods [14, 15], where downsampling is used to layer point clouds, and the Transformer architecture is employed to capture contextual information, adding informative constraints to the registration process. Furthermore, unsupervised correspondence-based point cloud registration methods that focus on optimizing point and Gaussian distribution correspondences [16, 17] have gained significant attention.

Despite rapid progress, point cloud registration remains challenging in real-world scenarios with low overlap between point clouds [11, 18], where registration often fails. This highlights the need for novel methods. Recent breakthroughs in colored point cloud registration [19, 20] have shown that integrating color information can reveal relationships that cannot be captured by geometric features alone, thereby improving registration performance. As shown in Fig. 1, color information plays a crucial role in scenarios with low overlap and subtle geometric features. When color differences are not distinct, simply incorporating color information still fails to establish the correct correspondences. However, when subfigures (c) and (d) specifically analyze color features, the registration process can successfully match the brown sofa with its armrest. The Gaussian distribution captures global shape variations and significantly suppresses noise. Combining color and geometric information, along with considering the relationships between point clouds and their Gaussian distributions, is crucial for faster and more robust registration.



(a) ColorPCR, Patch Correspondences (b) ColorPCR, Point Correspondences

(c) GeGs-PCR, Patch Correspondences (d) GeGs-PCR, Point Correspondences

Figure 1: In scenarios with minimal overlap, incomplete geometric features, and subtle color variations, methods that simply add color features perform moderately, whereas GeGS-PCR successfully identifies the brown sofa.

To address the challenges of point cloud registration in low-overlap real-world scenarios, we propose GeGS-PCR, a two-stage method that integrates Geometric-3DGS for colored point cloud registration. We designed a dedicated color encoder that enhances color features and extract multi-level geometric and color information from the original point cloud. The Geometric-3DGS module encodes local neighborhood information of colored superpoints, achieving globally invariant geographic color context. Using the parameterized multimodal local neighborhood information from 3DGS(geometric, Gaussian, and color information), we perform fast coarse registration. To reduce the computational complexity and parameter count introduced by 3DGS parameterization without sacrificing expressive power, we incorporate LORA optimization [21]. For better model convergence, we use 3DGS's fast differentiable rendering to refine the point cloud registration. Additionally, we introduce a joint photometric loss to improve the utilization of color information during the registration process. Through these modules, geometric and color data are tightly integrated, enabling GeGS-PCR to deliver strong performance even in challenging low-overlap scenarios.

Furthermore, due to the limited availability of publicly available colored datasets, currently only the publicly available COLOR3DMatch and Color3DLoMatch datasets [20] are available. Therefore, to validate the generalization ability of the model, we colorized the Kitti dataset to create ColorKitti. Evaluations on these datasets demonstrate that GeGS-PCR can achieve fast and stable registration under low overlap, proving its advanced effectiveness.

The main contributions of this paper are as follows:

- We tightly combine color and geometric information to achieve point cloud registration. Specifically, we design a color encoder for feature extraction, constructing a globally invariant geometric-color representation.

- We propose the Geometric-3DGS module to encode multimodal representations of superpoint neighborhood information. Using attention with 3DGS embeddings, we focus on global geometric distribution-color features and perform fast coarse registration by reducing computational complexity with LORA.
- We introduce a joint photometric loss. By performing fast differentiable rendering on the Geometric-3DGS module and calculating photometric loss during the rendering process, we refine the registration of point clouds.
- We colorize public point cloud datasets to generate ColorKitti. Experimental validation shows that GeGS-PCR performs excellently even in scenarios with extremely low overlap.

2 Related Work

2.1 Correspondence-Based Methods

Existing correspondence-based methods can be roughly divided into point-to-point and point-to-distribution registration approaches. Point-to-point methods (e.g., ICP) aim to estimate transformations through point coordinates or feature extraction [22], [23]. Using robust pose estimators (such as RANSAC or other RANSAC-free methods [13], [19], [24], [25]), registration is achieved through iterative optimization between correspondence search and transformation estimation. However, these methods are highly sensitive to noise and density variations and are typically supervised. The second category, point-to-distribution, maps points to probability distributions and estimates transformations through distribution alignment or clustering [20], [26], [27]. Although these methods are unsupervised, the iterative process can be time-consuming. Recent studies have adopted a coarse-to-fine approach [14], [28], achieving advanced performance. In this work, we follow this approach and focus on improving registration accuracy through a closer integration of color and geometric information (including color and distribution information).

2.2 Point Cloud Feature Extraction

Recently, due to the maturity of image matching methods, new paradigms for 3D point cloud feature extraction have emerged. PointNet performs deep learning-based feature extraction using graph convolution [29] and point convolution kernels [30]. Notably, feature extraction using KPConv-FPN [31], [32] has become a mainstay. To enhance the use of color information, PEAL [19] extracts RGB color from images, but it faces the problem of color information loss. Based on this, ColorPCR [20] integrates color information through the CEFE module and achieves better performance. However, these methods model color information separately. Inspired by the latest 3DGS technology in scene reconstruction, we consider tighter contextual relationships between color (opacity) and geometric information, enabling a Geometric-Color invariant representation. Therefore, we innovatively introduce 3DGS into point cloud registration and propose the Geometric-3DGS module. Through a color depth encoder, we extract deep color information and construct a global geographic relationship between point cloud color and geometric information to achieve fast registration in the coarse alignment stage.

2.3 Registration With Color Features

Some methods implicitly utilize color information to first detect keypoints or use 2D-3D multimodal learning [33], [34], [35], [36]. PEAL detects overlaps in 2D images and then transfers them to a 3D registration network. Additionally, ColorPCR uses multi-stage color processing in the global registration process to utilize color information. The ICP algorithm [37] and its variants (such as 4DICP) [38] increase the original point cloud feature dimensions to incorporate color information into geometric registration optimization. We follow this idea and use HSV and LAB to represent different color spaces [39], [40]. Due to the differentiable rendering properties of 3DGS [41], [42], [43], we adaptively fuse color information to achieve fast and stable fine registration of point clouds.

3 Method

Problem statement. Suppose we have two point clouds representing the target point cloud $P = \{p_i \in \mathbb{R}^3 | i = 1, ..., N\}$ and the source point cloud $Q = \{q_i \in \mathbb{R}^e | i = 1, ..., M\}$, and their

corresponding color information can be represented as $P_c = \{p_{c_i} \in [0,1]^3 | i = 1, ..., N\}$ and $Q_c = \{q_{c_i} \in [0,1]^3 | i = 1, ..., M\}$, respectively. The main objective of point cloud registration is to estimate a rigid transformation, represented by $T = \{R, t\}$, where $R \in SO(3)$ is the 3D rotation matrix and $t \in \mathbb{R}^3$ is the translation vector, such that the point clouds P and Q align through this transformation. The true correspondences between these point clouds are represented by the set C^*, which is initially unknown. Therefore, we can optimize the following objective to solve for the rigid transformation:

$$\min_{R,t} \sum_{(p^*_{x_i}, q^*_{y_i}) \in C^*} \| R \cdot p^*_{x_i} + t - q^*_{y_i} \|_2^2. \tag{1}$$

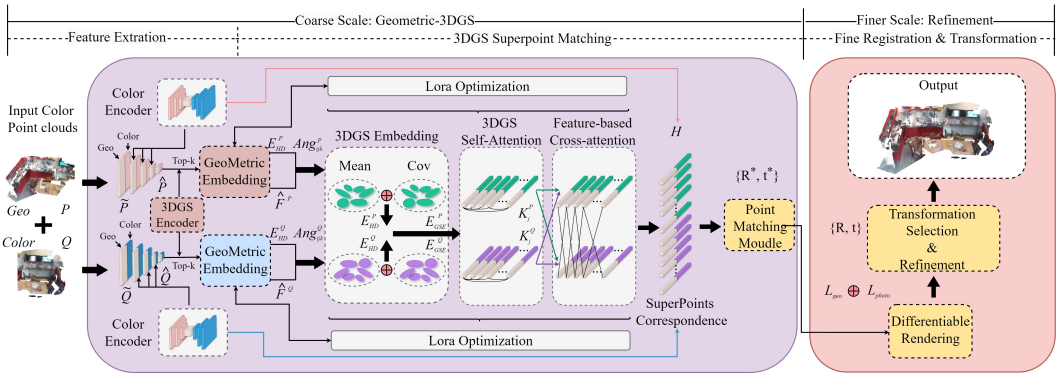


Figure 2: Pipeline. The entire network backbone is divided into coarse and fine scales. The feature extraction module extracts and integrates geometric and color information from the input point clouds P and Q using the color encoder and geometric encoder, producing superpoint representations $\hat{P}$ and $\hat{Q}$. The 3DGS Superpoint Matching Module identifies correspondences through 3DGS embeddings and self-attention mechanisms. LORA optimization is applied to reduce computational complexity and integrate both geometric and color information effectively. Finally, the Fine Registration & Transformation Module refines the registration by performing differentiable rendering after coarse registration, optimizing the rigid transformation parameters $\{R^*, t^*\}$.

Our pipeline is shown in Fig. 2. We represent the dense points and colors of the target point cloud as $P \in \mathbb{R}^{|P| \times 3}$ and $P_c \in [0,1]^{|P| \times 3}$, respectively. The coarsest level of the dense points is represented as $\tilde{P} \in \mathbb{R}^{|\tilde{P}| \times 3}$ and $\tilde{P}_c \in [0,1]^{|\tilde{P}| \times 3}$, and the finest level of the point cloud (superpoints) is represented as $\hat{P} \in \mathbb{R}^{|\hat{P}| \times 3}$ and $\hat{P}_c \in [0,1]^{|\hat{P}| \times 3}$. Similarly, the source point cloud is represented in the same way.

3.1 Coarse Registration With Color Features

3.1.1 Color Encoder Module

We design a dedicated color encoder module to inject effective color information into point cloud features (see Fig. 2). The rationale for using this encoder is twofold: (1) typical geometry-color fusion is often performed by simple addition, which cannot effectively handle noise in color information; (2) direct color feature extraction across different levels of feature extraction leads to reliance on low-level features. The encoder first inputs the three-channel color vector $F_c \in \mathbb{R}^3$ into a multi-layer perceptron (MLP). This MLP processes the color information, fully encodes it, and extracts deep color features, which are then bounded in the [0, 1] range through a sigmoid function to produce normalized color features F'_C. The noise-robust color mapping is as follows:

$$F'_C = \delta(LN(W_3 \cdot \delta(LN(W_2 \cdot (\delta(LN(W_1 \delta)))))))), \tag{2}$$

where W_1, W_2, and $W_3 \in \mathbb{R}^{d_{\text{in}} \times d_{\text{out}}}$ are learnable weights, d_{in} and d_{out} are input and output dimensions, $LN(\cdot)$ denotes layer normalization, and δ is the activation function. This process maps the original color to a decoupled high-dimensional space, effectively removing noise interference.

Based on this, we introduce a learned scalar weight $\alpha = \delta(\omega)$, where ω represents the parameter, to adaptively fuse the geometric and color features. During training, this weight dynamically adjusts the fusion process by balancing the contribution of geometric features F_g and color information F_c'. The final enhanced feature F_{enh} is obtained as follows:

$$F_{\text{enh}}^l = \begin{cases} F_g^l \oplus (\alpha \cdot F_c^{\text{color}}), & l < L \\ F_g^l \oplus F_c^{\text{color}}, & l = L \end{cases}, \tag{3}$$

where F_g is the original geometric feature, $F_c^{\text{color}} \in \mathbb{R}^3$ is the color space vector (HSV, LAB) processed by the MLP, and $\oplus$ represents channel concatenation. We use this color encoder in feature extraction at different levels. At the final level, we only use feature concatenation without the color encoder. This design injects color information while preserving the original geometry and other feature information, ensuring sufficient multimodal representation for downstream tasks.

3.1.2 Geometric-3DGS Module

The Geometric-3DGS module mainly consists of three components: the 3DGS encoder, attention with 3DGS embeddings, and Gaussian superpoint registration, as shown in Fig. 2. To improve the robustness and accuracy of the coarse registration stage, this module deeply integrates geometric and color features. Directly parameterizing all point cloud information with 3DGS at the beginning would result in a large amount of redundant parameters and very high memory requirements. Therefore, the core of this module is to 3DGS encode the local neighborhood information during point cloud downsampling, and use 3DGS embeddings and attention mechanisms to focus on relevant features in the 3DGS encoding. This module implements feature extraction at different granularities and provides a global, transformation-invariant geometric-color representation, which is crucial for fine point cloud registration.

3DGS encoder. We propose a Dual-Modal Color Encoder (DMCE), which is responsible for transforming local neighborhood patches in the point cloud into more robust and rich representations, thus capturing both geometric and color features. This method uses Gaussian distribution-based representations to model the local neighborhood structure and the relationships between points. The specific steps of the 3DGS encoder are as follows: We calculate the covariance matrix of each local neighborhood in the point cloud, which can capture the local geometric structure. This matrix considers the relative distances between points within the neighborhood to enhance the representation of local surface directions. Specifically, for a key point P_i with a neighborhood N_i, we first compute the differentiable covariance matrix:

$$\sum\nolimits_i = Ro(r_i) \cdot diag(\exp(s_i)) \cdot Ro(r_i)^T + \lambda n_i n_i^T, \tag{4}$$

where $Ro(\cdot) \in SO(3)$ represents the rotation matrix, $r_i \in \mathbb{R}^3$ is the rotation quaternion, $s_i \in \mathbb{R}^3$ is the logarithmic scale vector, n_i is the estimated normal vector, and λ controls the strength of the normal vector, with $\lambda \in [0.01, 0.1]$. This covariance matrix captures the geometric features of the local region of the point cloud. Finally, we perform top-k ($k \in [2, 5]$) fusion between the color features from the color encoder and the 3DGS parameters. By combining position, covariance matrix, color, and transparency, we construct the 3DGS feature vector:

$$F_{3DGS}^i = Top_{k_i}(Concat[\mu_i, vec(\Sigma_i), \alpha_i, F_{enh}^i]) \tag{5}$$

where μ_i is the Gaussian position center, and α_i represents transparency.

Attention with 3DGS embeddings. In this part, we implement 3DGS position embeddings to obtain globally invariant geometric-color fusion encodings. Finally, based on the GeoTransformer, we use self-attention and cross-attention to focus on the color information in the point cloud structure and guide superpoint registration. In both indoor and outdoor scenes, uncertainties such as sensors, lighting, and natural weather conditions often introduce noise or missing data in color sampling. To mitigate the impact of these color noises on global features, we aim to differentiate the parts that are heavily affected by noise. For indoor scenes, we use the HSV color space, and for outdoor scenes, we use the LAB color space. This ensures the most stable color information embedding, thus enabling robust structural position encoding. The specific process for the 3DGS position embedding is in Appendix A.2

Superpoint registration with 3DGS. To quickly register the point cloud information from Geometric-3DGS during the coarse registration stage, we use the ICP algorithm to align the Gaussian distributions. By minimizing the 3DGS distance between the source and target point clouds p_i, p_j, we iteratively

optimize the rigid transformation between p_i, p_j using least squares until convergence. We define the generalized distance metric as $d_{ij} = \left\|\mu_i^s - \mu_j^t\right\|_2^2 + \lambda_d \left\|\Sigma_i^s - \Sigma_j^t\right\|_F$, where the parameter λ_d is adjustable within the range of [0.05, 0.2], μ_i^s, μ_j^t are the means of the Gaussians in the source and target point clouds, and Σ_i^s, Σ_j^t are the covariances. Based on this definition, we give the associated probability matrix $A \in R^{N \times M}$, which measures the confidence of the match between p_i, p_j:

$$A_{ij} = \frac{\exp(-\gamma d_{ij})}{\sum_{k=1}^{M} \exp(-\gamma d_{ik})}, \tag{6}$$

where γ is the sensitivity control parameter, adjustable within the range of [1, 3], and this probability matrix is normalized using softmax. By combining the generalized distance and the probability matrix, we construct the objective to be optimized based on Mahalanobis distance:

$$\min_{R \in \mathrm{SO}(3), t \in \mathbb{R}^3} \sum_{i=1}^{N} \sum_{j=1}^{M} A_{ij} \left\| R(\mu_i^s - \bar{\mu}^s) + t - (\mu_j^t - \bar{\mu}^t) \right\|_{\Sigma_j^{-1}}^2, \tag{7}$$

where μ_i^s, μ_j^t represent the weighted centroids of the source and target point clouds, respectively, $\bar{\mu}^s = \frac{1}{N} \sum_i A_{ij} \mu_i^s$, $\bar{\mu}^t = \frac{1}{N} \sum_i A_{ij} \mu_i^t$. $\|\cdot\|$ denotes Mahalanobis distance, with Σ_j^{-1} used to down-weight regions where the covariance computation has large errors. This objective can be solved using SVD decomposition, resulting in:

$$H = \sum_{ij} A_{ij} \Sigma_j^{-1} (\mu_i^s - \bar{\mu}^s)(\mu_j^t - \bar{\mu}^t)^T, \tag{8}$$

$$U, S, V = SVD(H), \quad R^* = VU^T, \quad t^* = \bar{\mu}^t - R^* \bar{\mu}^s. \tag{9}$$

Once the transformation values during training are smaller than a minimum tolerance, the 3DGS registration is achieved.

LORA optimization. In the Geometric-3DGS module, the 3DGS parameterization introduces a large amount of high-dimensional data, which may result in significant computational and storage burdens, especially in large-scale point cloud scenarios. Therefore, we introduce LORA optimization within the overall Transformer structure to reduce unnecessary computational overhead. By using the LORA module, we convert the high-dimensional 3DGS embeddings into a low-rank form, allowing the model to remain efficient and accurate without sacrificing its capacity.

3.2 Fine Registration With Photometric Optimization

To improve point cloud registration accuracy, we propose a fine registration method based on photometric optimization. After coarse registration, we optimize point cloud alignment by rendering the 3DGS of the target and source point clouds under the new pose and minimizing the weighted photometric loss. Traditional photometric optimization methods typically only consider the L1 distance, which fails to effectively handle noise and lighting variations. To address this, we introduce a weighted photometric loss, assigning different weights to each pixel to enhance robustness and reduce the effects of uneven lighting and noise. Our photometric loss is calculated as follows:

$$L_{photo} = \sum_i \omega_i \left\| \hat{f}(G_1, \bar{C}) - \hat{f}(G_2, \bar{C}) \right\|^2, \quad \omega_i = \exp(-\gamma_d \cdot d_i), \tag{10}$$

where $\hat{f}(G_1, \bar{C})$, $\hat{f}(G_2, \bar{C})$ represent the rendered results of the target and source point clouds under the new pose $\bar{C}$, G_1 and G_2represent the 3DGS from the coarse registration, ω_i represents the pixel weight factor, and d_i is the pixel distance between the source and target point clouds. The sensitivity control parameter γ_d adjusts the influence of distance on registration, adjustable within the range of [1, 3]. Using differentiable rendering, we backpropagate the loss to the transformation parameters R^*, t^* and update them with gradient descent. This approach enhances accuracy, particularly in key regions, by combining geometric loss with the weighted photometric loss for precise point cloud alignment, as shown:

$$L_{total} = L_{geo} + \lambda_p L_{photo}. \tag{11}$$

where L_{geo} is the geometric registration loss, and $\lambda_p \in [0.1, 1]$ is a weight hyperparameter that adjusts the balance between geometric and photometric losses. We provide the detailed proof of the convergence of the joint photometric optimization in the Appendix A.1.

4 Experiments

To validate the performance of the GeGS-PCR model, we evaluate it on the indoor benchmarks Color3DMatch (C3DM) and Color3DLoMatch (C3DLM), as well as our colorized outdoor ColorKitti (The specific data construction process, including the detailed steps and settings for preparing the datasets, can be found in Appendix A.4 and A.5.) odometry benchmark. Each point cloud in these datasets includes an RGB color value.

Table 1: Evaluation results on C3DM and C3DLM. #Samples in the table represents the number of correspondences selected by RANSAC.

	C3DM					C3DLM				
#Samples	5000	2500	1000	500	250	5000	2500	1000	500	250
Feature Matching Recall (%) ↑										
CoFiNet [44]	98.1	98.3	98.1	98.2	98.3	83.1	83.5	83.3	83.1	82.6
GeoTransformer [15]	97.9	97.9	97.9	97.9	97.6	88.3	88.6	88.8	88.6	88.3
PEAL [19]	99.0	99.0	99.1	99.1	98.8	91.7	92.4	92.5	92.9	92.7
ColorPCR [20]	99.5	99.5	99.5	99.5	99.5	96.5	96.5	97.0	97.0	96.7
GeGS-PCR (ours)	**99.5**	**99.6**	**99.5**	**99.7**	**99.6**	**97.6**	**97.4**	**97.1**	**97.2**	**97.0**
Inlier Ratio (%) ↑										
CoFiNet [44]	49.8	51.2	51.9	52.2	52.2	24.4	25.9	26.7	26.8	26.9
GeoTransformer [15]	71.9	75.2	76.0	82.2	85.1	43.5	45.3	46.2	52.9	57.7
PEAL [19]	72.4	79.1	84.1	86.1	87.3	45.0	50.9	57.4	60.3	62.2
ColorPCR [20]	75.0	80.5	84.7	86.5	87.8	51.2	56.6	63.1	66.0	68.0
GeGS-PCR (ours)	**76.3**	**82.4**	**86.3**	**86.6**	**89.1**	**53.4**	**58.7**	**66.9**	**69.7**	**70.3**
Registration Recall (%) ↑										
CoFiNet [44]	89.3	88.9	88.4	87.4	87.0	67.5	66.2	64.2	63.1	61.0
GeoTransformer [15]	92.0	91.8	91.8	91.4	91.2	75.0	74.8	74.2	74.1	73.5
PEAL [19]	94.6	93.7	93.7	93.9	93.4	81.7	81.2	80.8	80.4	80.1
GeoTransformer+MAC [15]	95.7	95.7	95.2	95.3	94.6	78.9	78.7	78.2	77.7	76.6
ColorPCR [20]	97.5	96.5	97.0	96.4	96.5	88.9	88.5	88.1	86.5	85.0
GeGS-PCR (ours)	**97.9**	**97.6**	**97.5**	**96.7**	**97.6**	**90.7**	**90.2**	**90.4**	**90.0**	**89.8**

4.1 Indoor Benchmarks: Color3DMatch & Color 3DLoMatch

Table 2: Registration results w/o RANSAC on C3DM and C3DLM. The model is the time for feature extraction, while the pose time is for transformation estimation

Model	Estimator	#Sample	RR (%)↑		Time (s)↓		
			C3DM	C3DLM	Model	Pose	Total
FCGF[8]	RANSAC-50k	5000	85.1	40.1	0.052	3.326	3.378
D3Feat[9]	RANSAC-50k	5000	81.6	37.2	**0.024**	3.088	3.112
SpinNet[7]	RANSAC-50k	5000	88.6	59.8	60.248	**0.388**	60.636
Predator[28]	RANSAC-50k	5000	89.0	59.8	<u>0.032</u>	5.120	5.152
CoFiNet[44]	RANSAC-50k	5000	89.3	67.5	0.115	1.807	1.922
GeoTransformer[15]	RANSAC-50k	5000	92.0	75.0	0.075	<u>1.558</u>	**1.633**
PEAL[19]	RANSAC-50k	5000	94.6	81.7	0.089	1.776	1.865
ColorPCR[20]	RANSAC-50k	5000	97.5	88.2	0.083	1.629	1.712
GeGS-PCR(ours)	RANSAC-50k	5000	**97.9**	**90.7**	0.082	1.618	<u>1.700</u>
CoFiNet[44]	RANSAC-free	all	87.6	64.8	0.115	<u>0.028</u>	0.143
GeoTransformer[15]	RANSAC-free	all	91.5	74.0	<u>0.075</u>	**0.013**	**0.088**
PEAL[19]	RANSAC-free	all	94.3	78.8	0.089	0.034	0.133
ColorPCR[20]	RANSAC-free	all	97.3	88.3	0.083	0.046	0.129
GeGS-PCR	RANSAC-free	all	**97.5**	**89.1**	**0.082**	0.032	<u>0.124</u>

Correspondence results. We compared GeGS-PCR with several SOTA methods (metrics in Appendix A.3). Most methods use RANSAC, and we followed the same approach for consistency. Key results are shown in Table 1 (additional results in the Appendix A.3). GeGS-PCR outperforms

ColorPCR with 99.5% FMR on C3DM and 97.6% on C3DLM. For IR, it reaches 89.1% on C3DM and 70.3% on C3DLM, surpassing ColorPCR by 1.3% on C3DM and 2.3% on C3DLM. In RR, GeGS-PCR achieves 97.9% on C3DM and 90.7% on C3DLM, outperforming ColorPCR by 0.4% on C3DM and 4.2% on C3DLM. These results demonstrate significant improvement over baseline methods, with GeGS-PCR excelling, especially in low-overlap scenarios.

Table 3: Performance of ablation experiments

Overlap	C3DM				C3DLM			
	PIR(%)	FMR(%)	IR(%)	RR(%)	PIR(%)	FMR(%)	IR(%)	RR(%)
(a) w/o differentiable rendering	71.7	98.0	59.1	90.7	45.6	85.8	40.1	72.1
(b) w/o 3DGS	75.9	97.9	65.6	91.0	50.6	87.1	42.5	73.1
(c) w/o Geometric-3DGS	80.1	97.9	70.4	91.3	51.5	88.2	42.7	73.5
(d) w/o color encoder	82.1	97.7	70.3	91.5	53.9	88.1	43.3	74.0
(e) w/o color	83.8	98.0	70.7	92.7	54.1	88.4	43.5	74.2
(f) w/o LoRA	**92.0**	99.3	86.3	97.5	**63.8**	97.4	59.9	90.5
(g) Geometric-3DGS(Full)	92.0	99.6	82.4	97.6	63.9	97.4	58.7	90.2

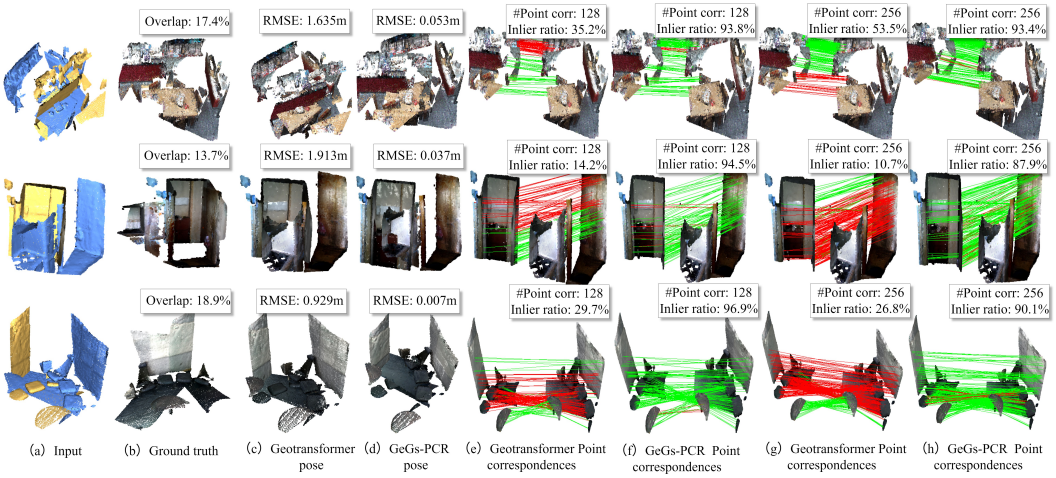


Figure 3: Registration performance with GeGS-PCR and Geometric self-attention.

Table 4: Ablation results based on ColorPCR baseline

Method	C3DM				C3DLM			
	PIR(%)	FMR(%)	IR(%)	RR(%)	PIR(%)	FMR(%)	IR(%)	RR(%)
(a) ColorPCR (baseline)	89.2	99.5	80.5	96.5	62.7	96.5	56.6	88.5
(b) w/o ColorEncoder	89.4	99.5	80.6	96.6	62.8	96.6	56.8	88.6
(c) w/o 3DGS	89.5	99.6	80.7	96.7	63.0	96.7	56.9	88.8
(d) w/o differentiable rendering	89.6	99.5	80.8	96.8	63.1	96.8	57.0	88.9
(e) w/o color	86.1	97.9	77.3	92.7	55.2	89.8	46.3	77.9
(f) w/o Geometric PE	88.8	99.3	80.2	96.3	62.4	96.0	56.4	88.0
(g) w/o LoRA	88.9	99.3	80.1	96.2	62.4	96.1	56.3	88.4
(h) Geometric-3DGS (all)	**90.0**	**99.6**	**82.4**	**97.6**	**63.9**	**97.4**	**58.7**	**90.2**

Registration results. As shown in Table 2, GeGS-PCR outperforms both RANSAC and RANSAC-free methods. For RANSAC-based methods, GeGS-PCR achieves 97.9% RR on C3DM and 90.7% on C3DLM, surpassing ColorPCR, with a total processing time of 1.703s, second only to D3Feat (1.712s). GeGS-PCR also achieves the best pose estimation time of 0.072s. For RANSAC-free methods, GeGS-PCR reaches 96.9% RR on C3DM and 89.1% on C3DLM, outperforming ColorPCR, with a total time of 0.124s, second only to GeoTransformer (0.088s). However, GeGS-PCR surpasses GeoTransformer in both RR and pose time, demonstrating its superior precision and speed.

Ablation experiments. Table 3 presents the ablation experiment results, analyzing the impact of each module on registration performance. Removing the color component (row e) reduces performance, particularly on C3DLM. Without the color encoder (row d), performance drops slightly, especially in FMR. Excluding the Geometric-3DGS module (row c) decreases PIR and IR, and removing 3DGS (row b) further lowers registration recall (RR). The removal of differentiable rendering (row a) significantly affects IR and PIR, with decreases of 7.1% and 9.5% in PIR and 6.5% and 8.1% in IR for C3DM and C3DLM, respectively. This highlights the importance of differentiable rendering for improving precision and inlier ratio. It helps the model utilize color information, improving point cloud alignment and boosting IR and PIR. In addition, removing LoRA optimization (row f) leads to a slight drop in registration performance, particularly in IR and RR, indicating that LoRA mainly accelerates convergence and provides a modest yet consistent improvement in accuracy while preserving efficiency. More detailed ablation analysis is shown in Appendix A.5.

Table 5: Registration results w/o RANSAC on Kitti

Model	RTE (cm)	RRE (°)	RR (%)
3DFeat-Net [45]	25.9	0.25	96.0
FCGF [8]	9.5	0.30	96.6
D3Feat [9]	7.2	0.30	99.8
SpinNet [7]	9.9	0.47	99.1
Predator [28]	6.8	0.27	99.8
CoFiNet [44]	8.2	0.41	99.8
GeoTransformer [15]	7.4	0.27	99.8
GeGS-PCR (ours, RANSAC-50k)	**6.3**	**0.16**	**99.9**
FMR [11]	∼66	1.49	90.6
DGR [18]	∼32	0.37	98.7
HRegNet [46]	∼12	0.29	99.7
GeoTransformer (LGR)	6.8	0.24	99.8
GeGS-PCR (ours, LGR)	**5.7**	**0.13**	**99.9**

Baselines ablation experiments Table 4 reports the ablation results based on the ColorPCR baseline. Overall, each module contributes to performance improvements to varying degrees. Removing color information (row e) causes the most significant degradation, with PIR, IR, and RR dropping notably on both C3DM and C3DLM, highlighting the critical role of color features in low-overlap scenarios. In contrast, removing the color encoder (row b) or geometric positional encoding (row f) only leads to minor fluctuations, suggesting these modules play supportive roles. Excluding 3DGS (row c) or differentiable rendering (row d) results in moderate decreases, particularly in IR and RR, indicating their importance for fine-grained alignment and convergence. Removing LoRA (row g) slightly reduces performance, confirming its role in accelerating convergence and providing consistent accuracy gains while maintaining efficiency. Finally, the full Geometric-3DGS model (row h) achieves the best results across all metrics, with PIR reaching 90.0% and IR improving to 82.4% on C3DM, and PIR reaching 63.9% and IR improving to 58.7% on C3DLM, demonstrating the effectiveness of the proposed modules in synergy. To clarify metric repetition, we ran independent ablations on ColorPCR (Table 4) controlling module order and data synergy. Results show consistent gains even on weaker baselines, proving our module design is sound, generalizable, and optimizable.

Qualitative Results. Fig. 3 shows the comparison of registration results using geometric attention and our 3DGS self-attention in scenes with low overlap and geometric feature overlap. Geometric attention produces a large dispersive effect, while 3DGS self-attention is able to find consistent correspondences for low-overlap features, significantly improving the inlier ratio and resulting in more accurate registration. The visualization results show that 3DGS self-attention can accurately identify correspondences in low-overlap areas and reject incorrect matches.

4.2 Outdoor Benchmarks: ColorKitti

Registration results. Based on previous research settings [7, 15, 44], we introduced the Registration Recall ($RR_{Outdoor}$) metric (defined in Appendix A.3). In Table 5, we first compare our GeGS-PCR model with RANSAC-based methods. Additionally, we compare it with methods that do not use RANSAC. The results show that GeGS-PCR performs comparably, with significant improvements in some metrics. Using RANSAC, GeGS-PCR achieves 99.9% Registration Recall (RR), demonstrating excellent performance. Even without RANSAC, GeGS-PCR maintains a 99.9% RR, and achieves the best RTE and RRE values.

Qualitative results. Fig. 4 shows the comparison of registration results using geometric attention and our 3DGS self-attention in low-overlap scenes. The red boxes in the ground truth highlight the colored regions. The results demonstrate that GeGS-PCR achieves good registration in these regions under both high and low point correspondence conditions. Notably, GeGS-PCR produces almost

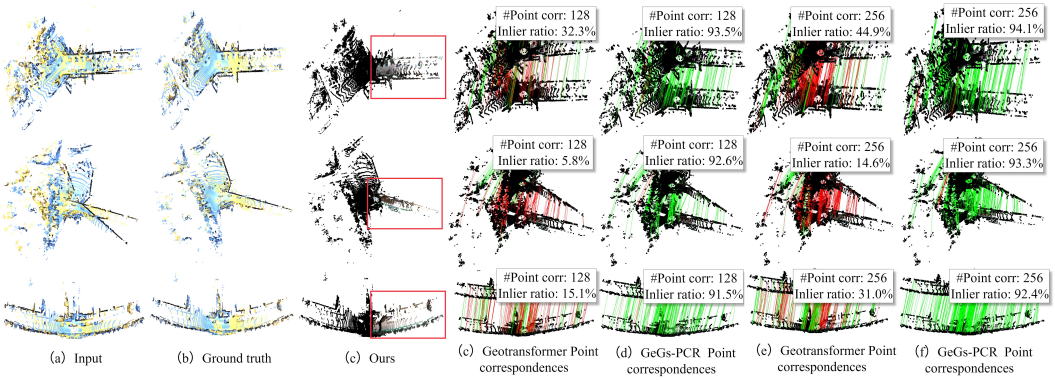


Figure 4: Registration performance with GeGS-PCR and Geometric self-attention.

no incorrect correspondences in the colored areas. While geometric attention causes significant dispersive effects, 3DGS self-attention consistently finds correct correspondences in low-overlap features, significantly improving the inlier ratio, especially in such regions. Further limitations and a comprehensive performance analysis can be found in Appendix A.5 and Appendix A.6.

5 Conclusion

In this paper, we present GeGS-PCR, an innovative two-stage point cloud registration method that enhances registration accuracy and robustness by integrating geometric, color, and Gaussian distribution information. In the coarse registration phase, GeGS-PCR effectively extracts reliable features in low-overlap scenarios with incomplete geometric features by introducing the Geometric-3DGS module and a color encoder. Additionally, LoRA optimization is applied to reduce the complexity introduced by feature encoding. In the fine registration phase, differentiable rendering combined with photometric optimization loss further improves performance. The advantages of GeGS-PCR lie in the collaborative optimization of global and local structures. Through local Gaussian feature extraction, GeGS-PCR effectively suppresses noise interference and robustly fuses geometric and color features. GeGS-PCR is not only suitable for high-overlap point cloud registration but also offers an efficient solution for low-overlap tasks, such as autonomous driving and large-scale scene reconstruction.

6 Acknowledgments

This work was supported in part by National Natural Science Foundation of China No. 62088102 and No.62302381. The Authors are with the National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center of Visual Information and Applications, and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, Shaanxi, China.

References

[1] Yu Chen and Gim Hee Lee. Dreg-nerf: Deep registration for neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22703–22713, 2023.

[2] Corentin Sautier, Gilles Puy, Spyros Gidaris, Alexandre Boulch, Andrei Bursuc, and Renaud Marlet. Image-to-lidar self-supervised distillation for autonomous driving data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9891–9901, 2022.

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

[4] Nur Muhammad Mahi Shafiullah, Chris Paxton, Lerrel Pinto, Soumith Chintala, and Arthur Szlam. Clip-fields: Weakly supervised semantic fields for robotic memory. *arXiv preprint arXiv:2210.05663*, 2022.

[5] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 12786–12796, 2022.

[6] Zi Jian Yew and Gim Hee Lee. Regtr: End-to-end point cloud correspondences with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 6677–6686, 2022.

[7] Sheng Ao, Qingyong Hu, Bo Yang, Andrew Markham, and Yulan Guo. Spinnet: Learning a general surface descriptor for 3d point cloud registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 11753–11762, 2021.

[8] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV)*, pages 8958–8966, 2019.

[9] Xuyang Bai, Zixin Luo, Lei Zhou, Hongbo Fu, Long Quan, and Chiew-Lan Tai. D3feat: Joint learning of dense detection and description of 3d local features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 6359–6367, 2020.

[10] Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppfnet: Global context aware local features for robust 3d point matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 195–205, 2018.

[11] Xiaoshui Huang, Guofeng Mei, and Jian Zhang. Feature-metric registration: A fast semi-supervised approach for robust point cloud registration without correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 11366–11374, 2020.

[12] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 8922–8931, 2021.

[13] Qunjie Zhou, Torsten Sattler, and Laura Leal-Taixe. Patch2pix: Epipolar-guided pixel-level correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 4669–4678, 2021.

[14] Xiyu Zhang, Jiaqi Yang, Shikun Zhang, and Yanning Zhang. 3d registration with maximal cliques. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 17745–17754, 2023.

[15] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. Geometric transformer for fast and robust point cloud registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 11143–11152, 2022.

[16] Kexue Fu, Shaolei Liu, Xiaoyuan Luo, and Manning Wang. Robust point cloud registration framework based on deep graph matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 8893–8902, 2021.

[17] Wentao Yuan, Benjamin Eckart, Kihwan Kim, Varun Jampani, Dieter Fox, and Jan Kautz. Deepgmr: Learning latent gaussian mixture models for registration. In *European conference on computer vision (ECCV)*, pages 733–750. Springer, 2020.

[18] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 2514–2523, 2020.

[19] Junle Yu, Luwei Ren, Yu Zhang, Wenhui Zhou, Lili Lin, and Guojun Dai. Peal: Prior-embedded explicit attention learning for low-overlap point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17702–17711, 2023.

[20] Juncheng Mu, Lin Bie, Shaoyi Du, and Yue Gao. Colorpcr: Color point cloud registration with multi-stage geometric-color fusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21061–21070, 2024.

[21] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3, 2022.

[22] P.J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[23] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8957–8965, 2019.

[24] Xuyang Bai, Zixin Luo, Lei Zhou, Hongkai Chen, Lei Li, Zeyu Hu, Hongbo Fu, and Chiew-Lan Tai. Pointdsc: Robust point cloud registration using deep spatial consistency. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 15859–15869, 2021.

[25] Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 5545–5554, 2019.

[26] Georgios Dimitrios Evangelidis and Radu Horaud. Joint alignment of multiple point sets with batch and incremental expectation-maximization. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1397–1410, 2017.

[27] Xiaoshui Huang, Sheng Li, Yifan Zuo, Yuming Fang, Jian Zhang, and Xiaowei Zhao. Unsupervised point cloud registration by learning unified gaussian mixture models. *IEEE Robotics and Automation Letters*, 7(3):7028–7035, 2022.

[28] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition (CVPR)*, pages 4267–4276, 2021.

[29] 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)*, pages 652–660, 2017.

[30] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[31] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 2117–2125, 2017.

[32] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François 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)*, pages 6411–6420, 2019.

[33] Mohamed El Banani and Justin Johnson. Bootstrap your own correspondences. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (CVPR)*, pages 6433–6442, 2021.

[34] Ziming Wang, Xiaoliang Huo, Zhenghao Chen, Jing Zhang, Lu Sheng, and Dong Xu. Improving rgb-d point cloud registration by learning multi-scale local linear transformation. In *European Conference on Computer Vision (ECCV)*, pages 175–191. Springer, 2022.

[35] Mingzhi Yuan, Kexue Fu, Zhihao Li, Yucong Meng, and Manning Wang. Pointmbf: A multi-scale bidirectional fusion network for unsupervised rgb-d point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 17694–17705, 2023.

[36] Yu Zhang, Junle Yu, Xiaolin Huang, Wenhui Zhou, and Ji Hou. Pcr-cg: Point cloud registration via deep explicit color and geometry. In *European conference on computer vision (ECCV)*, pages 443–459. Springer, 2022.

[37] Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Colored point cloud registration revisited. In *Proceedings of the IEEE international conference on computer vision (ICCV)*, pages 143–152, 2017.

[38] Hao Men, Biruk Gebre, and Kishore Pochiraju. Color point cloud registration with 4d icp algorithm. In *2011 IEEE International Conference on Robotics and Automation*, pages 1511–1516. IEEE, 2011.

[39] Zhexin Liang, Zhaochen Li, Shangchen Zhou, Chongyi Li, and Chen Change Loy. Control color: Multimodal diffusion-based interactive image colorization: Z. liang et al. *International Journal of Computer Vision*, pages 1–27, 2025.

[40] Wei Zhou, Qi Yang, Wu Chen, Qiuping Jiang, Guangtao Zhai, and Weisi Lin. Blind quality assessment of dense 3d point clouds with structure guided resampling. *ACM Transactions on Multimedia Computing, Communications and Applications*, 20(8):1–21, 2024.

[41] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 20310–20320, 2024.

[42] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 19447–19456, 2024.

[43] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*, pages 800–809. IEEE, 2024.

[44] Hao Yu, Fu Li, Mahdi Saleh, Benjamin Busam, and Slobodan Ilic. Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. *Advances in Neural Information Processing Systems*, 34:23872–23884, 2021.

[45] Zi Jian Yew and Gim Hee Lee. 3dfeat-net: Weakly supervised local 3d features for point cloud registration. In *Proceedings of the European conference on computer vision (ECCV)*, pages 607–623, 2018.

[46] Fan Lu, Guang Chen, Yinlong Liu, Lijun Zhang, Sanqing Qu, Shu Liu, and Rongqi Gu. Hregnet: A hierarchical network for large-scale outdoor lidar point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 16014–16023, 2021.

[47] Haiping Wang, Yuan Liu, Zhen Dong, and Wenping Wang. You only hypothesize once: Point cloud registration with rotation-equivariant descriptors. In *Proceedings of the 30th ACM International Conference on Multimedia*, pages 1630–1641, 2022.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction clearly outline the core contributions of the paper, including the introduction of a novel point cloud registration method, GeGS-PCR, and a detailed explanation of its innovative use of both geometric and color information. The abstract clearly states the method's experimental results and its outstanding performance in low-overlap scenarios, with comparisons made to existing techniques to validate its superiority. The introduction also mentions the method's limitations and potential future improvements, ensuring consistency with the actual results and objectives.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper clearly discusses the limitations of the method. First, GeGS-PCR relies on superpoints (patches) extracted through downsampling during the registration process. In regions with high point cloud overlap, the large number of superpoints, combined with 3DGS parameterization, can lead to significant memory usage and computational overhead. Since patch decomposition shares similarities with semantic scene understanding, the authors plan to leverage semantic scene understanding for point cloud pair registration in future research. Additionally, the current registration method involves pairing after 3DGS parameterization of domain superpoints. In future work, the authors aim to explore scene-level registration of 3DGS for more realistic environmental registration.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.

- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The paper provides numbered theorems, formulas, and proofs, all of which are cross-referenced in the main text. All assumptions are clearly stated or referenced in the statement of each theorem. The proofs are either included in the main paper or the supplemental material. If provided in the supplemental material, a short proof sketch is included in the main paper to provide intuition. Additionally, all theorems and lemmas that the proof relies upon are properly referenced, ensuring logical completeness.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides a comprehensive description of the experimental setup, including dataset selection, experimental design, training and evaluation methods, and the metrics used for evaluation. In addition, the supplementary materials include the code, datasets, and pretrained model weights, enabling other researchers to efficiently replicate the experimental results. All experimental settings, hyperparameters, and methodologies are clearly outlined, ensuring the reproducibility of the experiments. The paper also offers a clear explanation of the proposed algorithms and model architecture, providing sufficient theoretical foundation and implementation details, which assist readers in understanding and reproducing the results. Through these efforts, the paper ensures the verifiability and reliability of the experimental outcomes, allowing readers to reconstruct and validate the experiments and conclusions based on the information provided.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.

- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper provides full access to the experimental data, code, and pretrained models through links in the supplemental material. Additionally, the authors list the experimental environment configuration in detail, including the operating system version, Python version, dependencies, and specific command-line scripts, ensuring that other researchers can accurately reproduce all the experimental results. Detailed instructions on how to access and prepare the raw data, as well as preprocess it, are also provided, ensuring the reproducibility of the experiments. The paper fully meets the NeurIPS requirements for open access to code and data, ensuring that other researchers can accurately reproduce the experimental results. Open source code link: `https://anonymous.4open.science/r/GeGS-PCR-81F5/`

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper provides comprehensive details of the experimental settings, including the data splitting method, chosen hyperparameters and the rationale behind their selection, and the type of optimizer used along with its specific configuration. Additionally, the full experimental setup, including values for hyperparameters and data processing steps, is provided in the appendix and supplemental material. This ensures that the readers can fully understand and reproduce the experiments. The paper clearly presents the necessary experimental details, allowing readers to fully comprehend and reproduce the results.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: The paper does not include experiments requiring statistical significance tests or error bars.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The paper provides adequate details on the computational resources required to reproduce the experiments. It clearly specifies the type of compute workers used (GPU), along with the memory and storage requirements. The paper also provides information on the time of execution for each experiment. Additionally, it discloses whether cloud services or internal clusters were used and includes estimates of the total compute required for the experiments. This transparency ensures that the experiments can be accurately reproduced and provides clarity on the computational demands involved.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: The research conducted in this paper fully conforms to the NeurIPS Code of Ethics. All necessary ethical considerations have been taken into account, including the respectful use of data and adherence to ethical guidelines in the treatment of human subjects, if applicable. The authors have ensured compliance with all relevant laws and regulations. There are no special circumstances that require deviation from the established code.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: The paper discusses both the potential positive and negative societal impacts of the work performed. On the positive side, the proposed method for point cloud registration could be beneficial in fields such as autonomous navigation, medical imaging, and urban planning by enabling more accurate and efficient processing of 3D data. This could lead to advances in technologies that improve safety, efficiency, and urban infrastructure development. On the negative side, there are potential privacy and security concerns when using advanced 3D registration techniques in sensitive applications, such as surveillance or data collection in private spaces. There could also be unintended consequences if such technologies are used to manipulate or distort 3D data for malicious purposes, such as creating fake 3D models. The authors discuss these risks and suggest mitigation strategies such as gated releases and incorporating safety measures into deployment to avoid misuse.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: The paper does not specifically mention safeguards for the release of data or models that pose a high risk of misuse. While the proposed model and datasets are not inherently high-risk for misuse, as they focus on point cloud registration and related tasks. so the authors have not outlined any specific safety measures or restrictions to control potential misuse.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: The authors have properly credited the original creators and owners of the assets used in the paper. The relevant licenses and terms of use are clearly mentioned, and the authors respect the licenses associated with the used code, data, and models. For example, the dataset used for training and evaluation has its licensing information included, and the used models are cited with the corresponding license details. All assets are used in accordance with their specified terms, ensuring compliance with intellectual property rights.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.

- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The new assets introduced in the paper are well documented, with detailed descriptions about the dataset/code/models included in the submission. The documentation provides key information about the training process, licenses, and limitations. Additionally, any necessary consent from individuals whose assets are used has been obtained and is discussed in the paper. Proper anonymization of the assets has been conducted where required, ensuring compliance with privacy and ethical standards.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing or research with human subjects. Therefore, there are no instructions, screenshots, or compensation details required for inclusion in the paper.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: The paper does not involve research with human subjects or crowdsourcing, so there are no risks, disclosures, or IRB approvals to report.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Technical Appendices and Supplementary Material

A.1 Proof of photometric optimization

To rigorously analyze the convergence of our joint photometric-geometric optimization, we formulate the camera pose optimization on the Special Euclidean group SE(3) manifold. Let the camera pose be parameterized as $T \in \mathrm{SE}(3)$, which can be expressed via the exponential map from its Lie algebra $\xi \in \mathfrak{se}(3)$:

$$T = \exp(\xi^{\wedge}), \quad \xi = [\omega, v]^T \in \mathbb{R}^6, \tag{12}$$

where $\omega \in \mathbb{R}^3$ and $v \in \mathbb{R}^3$ represent the angular and linear velocities, respectively. They can be derived from the rotation matrix R and the translation matrix T. Under the assumptions of local convexity of the photometric loss L_{photo} near the optimal pose T^* and bounded gradient noise, the proposed joint loss L_{total} converges to a local minimum when optimized via Riemannian gradient descent on SE(3). The gradient of L_{total} on SE(3) is computed through the retraction map R_T:

$$\text{grad}\, L_{total} = R_T^{-1} \left(\frac{\partial L_{geo}}{\partial \xi} + \lambda \frac{\partial L_{photo}}{\partial \xi} \right). \tag{13}$$

where $\xi \in \mathfrak{se}(3)$ is the Lie algebra of the transformation and R_T^{-1} is the retraction map that ensures the gradient is mapped correctly from the Lie algebra back to SE(3). Using the chain rule, the photometric gradient with respect to ξ is derived as:

$$\frac{\partial L_{photo}}{\partial \xi} = \sum_i \omega_i \cdot \frac{\partial \|f(G_1) - f(G_2)\|^2}{\partial \Delta I} \cdot \frac{\partial \Delta I}{\partial T} \cdot \frac{\partial T}{\partial \xi}, \tag{14}$$

where $f(G_1)$ and $f(G_2)$ represent the rendered results of the target and source point clouds under the new pose C, ω_i is the weight factor for each pixel, and ΔI is the difference between the rendered and target images. The image gradient is:

$$\frac{\partial \|f(G_1) - f(G_2)\|^2}{\partial \Delta I} = 2 \cdot (f(G_1) - f(G_2)), \tag{15}$$

We also calculate the rendering Jacobian and Lie Algebra Jacobian as:

$$\frac{\partial \Delta I}{\partial T}, \quad \frac{\partial T}{\partial \xi}, \tag{16}$$

The Lie algebra Jacobian follows the left perturbation model:

$$\frac{\partial T}{\partial \xi} = \lim_{\delta\xi \to 0} \exp((\delta\xi)^{\wedge}) T - T_{\delta}, \tag{17}$$

Using the Łojasiewicz inequality on matrix Lie groups, we show that with learning rate $\eta < \frac{1}{L}$ (where L is the Lipschitz constant), the gradient norm $\|\text{grad} L_{\text{total}}^{(k)}\|_2^2$ decreases monotonically until reaching a stationary point:

$$\|\text{grad}\, L_{total}^{(k)}\|_2^2 \leq C(L_{total}^{(k)} - L_{total}^{(k+1)}). \tag{18}$$

This guarantees that the loss function will decrease monotonically and converge to a local minimum.

A.2 Structure Embedding

3DGS embedding. In this part, we implement 3DGS position embeddings to obtain globally invariant geometric-color fusion encodings. Finally, based on the GeoTransformer, we use self-attention and cross-attention to focus on the color information in the point cloud structure and guide superpoint registration. In both indoor and outdoor scenes, uncertainties such as sensors, lighting, and natural weather conditions often introduce noise or missing data in color sampling. To mitigate the impact of these color noises on global features, we aim to differentiate the parts that are heavily affected by noise. For indoor scenes, we use the HSV color space, and for outdoor scenes, we use the LAB color space. This ensures the most stable color information embedding, thus enabling robust structural position encoding. The specific process for the 3DGS position embedding is as follows:

Gaussian embedding. We add Gaussian embedding to capture geometric distribution information that is closely related to color information. This is particularly useful for invariance between point clouds, where the distribution of local regions is represented using the mean μ and covariance Σ of 3DGS. The difference in the mean is computed as $\delta_{\mu_{ij}} = \|\mu_i - \mu_j\|_2$, where μ_i, μ_j represent the local means in the source and target point clouds, respectively. The Frobenius norm difference in covariance is $\delta_{\Sigma_{ij}} = \|\Sigma_i - \Sigma_j\|_F$, where Σ_i, Σ_j are the covariance matrices of the local neighborhoods in the two point clouds. The final Gaussian embedding is computed as:

$$E_{GS} = PE\left(\frac{(\delta_{\Sigma_{ij}} + \delta_{\Sigma_{ij}}) \cdot \delta_{\mu_{ij}}}{\sigma_{GS}} \right) W_{GS}, \tag{19}$$

where σ_{GS} is the sensitivity hyperparameter for the Gaussian embedding, adjustable within the range of [0.01, 0.1], PE is the sine position embedding function, and $W_{GS} \in \mathbb{R}^{d_t \times d_t}$ is the matrix used for projection, with d_t representing the output dimension. Then, we fuse all embeddings (including distance, color, angle, and Gaussian embeddings, where the structure of the color distance and angle embeddings is shown in A.1) to obtain the final 3DGS embedding E_{GSE}. This embedding method ensures that the point cloud information in each layer is adequately represented.

3DGS self-Attention. To enhance the feature extraction from this part, we introduce the 3DGS self-attention mechanism. First, based on the 3DGS embeddings, we generate the query keys and values as follows:

$$Q_i = E^i_{GSE} W_\Omega, \quad K_j = E^j_{GSE} W_k + E^j_{GSE} W_K, \quad V_j = E^j_{GSE} W_v, \tag{20}$$

where $E^i_{GSE} \in \mathbb{R}^{d_t}$ represents the feature of the iii-th joint embedding, F represents the corresponding feature set (as shown in the Fig. 2), and $E^i_{Geo} \in \mathbb{R}^{d_t}$ represents the collection embedding, which is a combination of color-distance embedding and angle embedding (specifics can be found in A.2). $W_Q, W_K, W_V \in R^{d_t \times d_t}$ are learnable projection matrices used to generate the query, key, and value. Based on this, we obtain a new attention calculation method (the specific structure shown in the Fig. 5 below) :

$$a_{ij} = Softmax\left(\frac{Q_i K_j^T}{\sqrt{d_t}} + \lambda_g \log(1 + E^{ij}_{geo})\right) \cdot V_j. \tag{21}$$

where, $\lambda_g \in [0.01, 1]$ controls the weight of the geometric information in the attention computation by adjusting the influence of the $\log(1 + E^{ij}_{geo})$ term. This term is designed to emphasize the relationship between geometric features, such as color and angle, thus improving the feature fusion process.

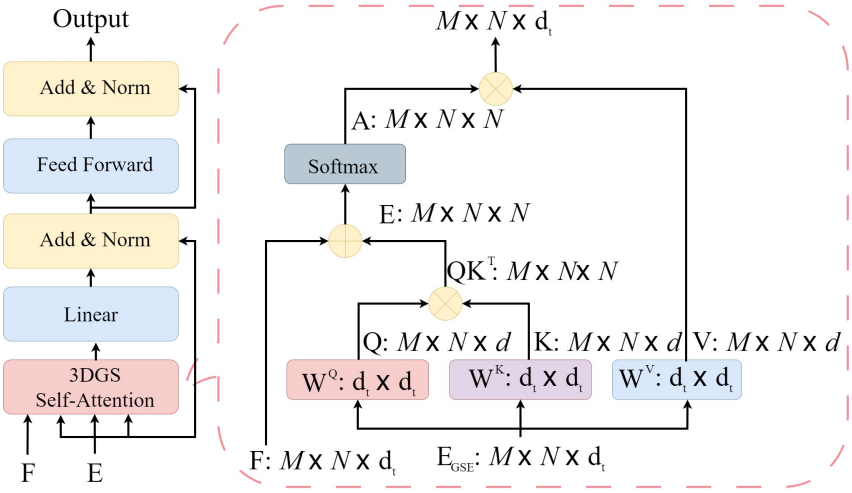


Figure 5: Left: The structure of 3DGS self-attention module. Right: The computation graph of 3DGS self-attention.

Color Distance embedding. Colorized position embedding with color is achieved by calculating the Euclidean distance between each pair of 3DGS in the point cloud and normalizing the distance before embedding:

$$d_{\text{indices}} = \frac{\|p_i - p_j\|_2}{\sigma_d}, \tag{22}$$

$$E_{\text{HD}} = PE(\Delta H \cdot d) W_{\text{HD}}, \tag{23}$$

Where, σ_d is the sensitivity parameter for distance control, $\Delta H = \left\|hp_i - hp_j\right\|_2$ represents the hue difference containing color information, PE is the sine position embedding function, and is the projection matrix.

Angle embedding. Angle embedding is based on the cross product of three points in the point cloud. For a given triplet of points, p_i, p_j and their neighbor p_k, we calculate the angle between the three points:

$$Ang_{ijk} = \max_{k \in K} PE\left(\frac{a \tan 2(\|v_i \times v_j\|, v_i \cdot v_j)}{\sigma_a}\right) W_a. \tag{24}$$

where $v_i = p_i - p_k$, $v_j = p_j - p_k$ are the vectors from p_k, and σ_a is the temperature parameter for angle embedding, controlling the sensitivity of angle embedding. $W_a$is the projection matrix.

A.3 Metrics

We evaluate the performance of GeGS-PCR using standard metrics, including the Registration Recall Rate (RR), which is the proportion of point clouds with a transformation error below a threshold (e.g., RMSE < 0.2), the Feature Matching Recall Rate (FMR), representing the proportion of point cloud pairs with an inlier ratio above a threshold (e.g., 5%), and the Inlier Ratio (IR), which measures the proportion of matching points with residuals below a threshold (e.g., 0.1m). Additionally, we assess the model's performance using the Relative Rotation Error (RRE) and Relative Translation Error (RTE).

Inlier Ratio (IR) is used to measure the proportion of correspondences with geometric consistency under the true transformation. For example, given a point cloud pair (P, Q) and its correspondence M, the definition is:

$$IR(P,Q) = \frac{1}{|M|} \sum_{(p_i,q_j)\in M} I(\|T_{gt}(p_i) - q_j\|_2 < \delta_{corr}), \tag{25}$$

where T_{gt} is the ground truth transformation matrix, and $\delta_{corr} < 0.1$ is the inlier threshold function, and $I(\cdot)$ is the Iversion bracket.

Feature Matching Recall (FMR) is used to evaluate whether the correspondence geometry contains enough inliers to support registration. For N point cloud pairs, the formula is:

$$FMR = \frac{1}{N} \sum_{k=1}^{N} I(IR(P_k, Q_k) \geq \eta), \tag{26}$$

where $\eta \geq 0.05$ is the minimum inlier ratio threshold required for robust registration (e.g., RANSAC).

Registration Recall (RR_{Indoor}) measures the proportion of successfully registered point cloud pairs that meet the geometric error tolerance. If the RMSE of the true correspondences C^* of a point cloud pair (P, Q) after registration satisfies the following. Then, for a threshold $\gamma \leq 0.2m$, we compute:

$$RMSE = \sqrt{\frac{1}{C^*} \sum_{(p_i,q_j)\in C^*} \|T_{\text{est}}(p_i) - q_j\|_2^2}, \tag{27}$$

$$RR_{\text{Indoor}} = \frac{1}{N} \sum_{k=1}^{N} I(RMSE_k < \gamma). \tag{28}$$

Patch Inlier Ratio (PIR) represents the proportion of actual overlap for superpoints (patches) under true transformations in the scene, which measures the registration of our 3DGS parameterized domain:

$$PIR = \frac{1}{\hat{C}} \sum_{(sp,sq)\in C} I\left(\exists (p_i, q_j) \in S_p \times S_q, \|T_{gt}(p_i) - q_j\|_2 < \zeta\right), \tag{29}$$

where $\zeta \leq 0.05m$ is the inlier distance threshold.

Pose Deviation Error indicates the error between the estimated rigid transformation $T_{est} = \{R, t\}$ and the true transformation $T_{gt} = \{R_{gt}, t_{gt}\}$. We use two metrics: Relative Rotation Error (RRE) and Relative Translation Error (RTE):

$$RRE = \arccos\left(\frac{trace(R^T R_{gt}) - 1}{2}\right) \quad [radians], \tag{30}$$

$$RTE = \|t - t_{gt}\|_2 \quad [meters]. \tag{31}$$

Registration Recall ($RR_{Outdoor}$) measures the error in point cloud registration in outdoor scenes, considering both RRE and RTE while ensuring both satisfy the minimum threshold requirements:

$$RR_{Outdoor} = \frac{1}{N} \sum_{k=1}^{N} I\left(RRE_k < 3^\circ \wedge RTE_k < 1.5m\right). \tag{32}$$

A.4 Color Data

Color3DMatch/Color3DLoMatch: Color3DMatch and Color3DLoMatch are pre-processed training datasets, which are the results of coloring the 3DMatch and 3DLoMatch point cloud data. 3DMatch contains 62 scenes, with 46 used for training, 8 for testing, and 8 for validation. The entire experimental evaluation strictly follows the dataset protocol. In 3DMatch, the overlap between point cloud pairs is greater than 30%. In Color3DLoMatch, the overlap is relatively low (10%-30%).

ColorKitti: The Kitti odometry dataset contains outdoor driving scenes with 11 sequences, all obtained through LIDAR scanning. We use the corresponding RGB images for each frame to color the point clouds, thus constructing ColorKitti. However, due to the limited viewpoint of the RGB image capturing device, we only color the point clouds visible from the image perspective. Sequences 0-5 are used for training, sequences 6 and 7 for validation, and sequences 8-10 for testing. As with 3DMatch/3DLoMatch, the ground truth poses are refined

using ICP, and the entire evaluation process strictly follows the dataset protocol. Additionally, only point clouds with a distance of at least 10 meters are used for evaluation. Unlike the 3DMatch/3DLoMatch scenes, the RGB images in the Kitti dataset are not captured in a panoramic manner, as the camera's viewpoint is limited. Initially, we attempted to use neighboring frames to color the point clouds of the current frame. However, visual analysis revealed that the coloring effect was unrealistic and the color distribution was not smooth, introducing excessive color noise, which negatively impacted point cloud registration. Therefore, we only color the point clouds visible from the image perspective, and normalize the regions that are not colored. During the data preprocessing stage, we filter out colorless data and retain only the geometric point clouds. In the 3DGS registration process, only the colored portion of the point clouds is registered, and the rigid transformation parameters (R, t) are applied to restore the point clouds into the global scene. Fig. 8 provides an example comparing the colored ground truth with the original ground truth.

A.5 Additional Experiments

C3DM consists of 62 scenes, with 46 scenes used for training, 8 for validation, and 8 for testing. The entire process follows the C3DM/C3DLM protocol and includes registration analysis for low-overlap scenarios. Additionally, ColorKitti Odometry [7] consists of 11 outdoor driving sequences scanned by LiDAR. We follow the setup in [15, 44], using sequences 0-5 for training, sequences 6-7 for validation, and sequences 8-10 for testing. As in [4, 8, 16], the ground-truth poses are refined with ICP, and we only use point cloud pairs that are at least 10m apart for evaluation.

Implementation details. We implemented and evaluated our GeGS-PCR using PyTorch [15] on an AMD 610M CPU and an NVIDIA RTX 4070 GPU. During the entire training process, GeGS-PCR was trained for 40 epochs on the 3DMatch dataset and 80 epochs on the KITTI dataset. The batch size was set to 1, with an initial learning rate of 10^{-4}, decaying by 0.05 every epoch. The Adam optimizer was used throughout the training process.

Additional baselines. Table 6 is a continuation of Table 1, presenting the comparison between GeGS-PCR and other baselines. Most methods use RANSAC, and we have adopted the same approach for consistency. The results in the table show that GeGS-PCR performs excellently across all three key metrics: Feature Matching Recall (FMR), Inlier Ratio (IR), and Registration Recall (RR). In Feature Matching Recall (FMR), GeGS-PCR achieves 99.5% on C3DM and 97.6% on C3DLM, outperforming other methods. Particularly on C3DM, GeGS-PCR improves by 0.7% over YOHO and significantly surpasses methods like Predator and SpinNet. In Inlier Ratio (IR), GeGS-PCR stands out further with IR values of 89.1% on C3DM and 70.3% on C3DLM, greatly exceeding YOHO, Predator, and other methods, especially improving by nearly 15% on C3DM. In Registration Recall (RR), GeGS-PCR performs at 97.9% on C3DM and 90.7% on C3DLM, leading other methods, with a particularly large improvement of 5.2% over YOHO on C3DLM. Overall, GeGS-PCR demonstrates outstanding performance across multiple datasets, excelling in low-overlap scenarios and surpassing all existing methods.

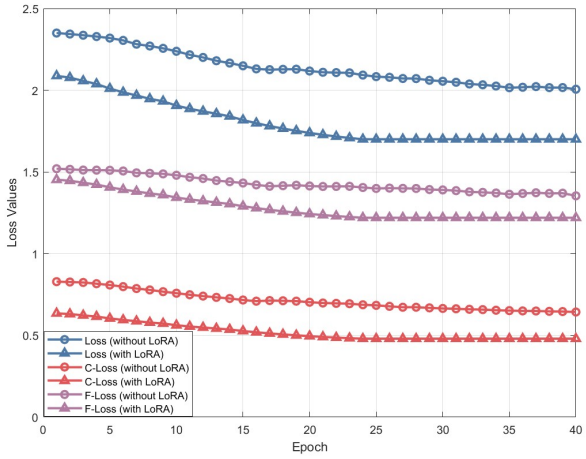


Figure 6: Comparison of Training Loss with and without LoRA Optimization (Color3DMatch Dataset).

3DGS self-attention. In Table 7, we present the performance of different self-attention models in low-overlap scenarios, including Vanilla Self-attention, Geometric Self-attention, and 3DGS Self-attention. GeGS-PCR performs exceptionally well, particularly in low-overlap regions. Compared to other methods, 3DGS Self-attention significantly improves PIR (Precision), IR (Inlier Ratio), and RR (Registration Recall). In the high-overlap (0.9-1.0) range, the 3DGS Self-attention model achieves a PIR of 0.997, an IR of 0.905, and an RR of 1.000, demonstrating the best performance. As the overlap decreases, GeGS-PCR maintains strong performance even in the 0.5-0.6 overlap range, with a PIR of 0.938, an IR of 0.872, and an RR of 0.963, highlighting its robustness when handling low-overlap data. Specifically, compared to Vanilla Self-attention, 3DGS Self-attention shows stronger robustness across the entire overlap range, with its advantages becoming more pronounced in complex environments.

Table 6: Evaluation results on C3DM and C3DLM. #Samples in the table represents the number of correspondences selected by RANSAC.

	C3DM					C3DLM				
#Samples	5000	2500	1000	500	250	5000	2500	1000	500	250
Feature Matching Recall (%) ↑										
PerfectMatch [25]	95.0	94.3	92.9	90.1	82.9	63.6	61.7	53.6	45.2	34.2
FCGF [8]	97.4	97.3	97.0	96.7	96.6	76.6	75.4	74.2	71.7	67.3
D3Feat [9]	95.6	95.4	94.5	94.1	93.1	67.3	66.7	67.0	66.7	66.5
SpinNet [7]	97.6	97.2	96.8	95.5	94.3	75.3	74.9	72.5	70.0	63.6
Predator [28]	96.6	96.6	96.5	96.3	96.5	78.6	77.4	76.3	75.7	75.3
YOHO [47]	98.2	97.6	97.5	97.7	96.0	79.4	78.1	76.3	73.8	69.1
GeGS-PCR (ours)	**99.5**	**99.6**	**99.5**	**99.7**	**99.6**	**97.6**	**97.4**	**97.1**	**97.2**	**97.0**
Inlier Ratio (%) ↑										
PerfectMatch [25]	36.0	32.5	26.4	21.5	16.4	11.4	10.1	8.0	6.4	4.8
FCGF [8]	56.8	54.1	48.7	42.5	34.1	21.4	20.0	17.2	14.8	11.6
D3Feat [9]	39.0	38.8	40.4	41.5	41.8	13.2	13.1	14.0	14.6	15.0
SpinNet [7]	47.5	44.7	39.4	33.9	27.6	20.5	19.0	16.3	13.8	11.1
Predator [28]	58.0	58.4	57.1	54.1	49.3	26.7	28.1	28.3	27.5	25.8
YOHO [47]	64.4	60.7	55.7	46.4	41.2	25.9	23.3	22.6	18.2	15.0
GeGS-PCR (ours)	**76.3**	**82.4**	**86.3**	**86.6**	**89.1**	**53.4**	**58.7**	**66.9**	**69.7**	**70.3**
Registration Recall (%) ↑										
PerfectMatch [25]	78.4	76.2	71.4	67.6	50.8	33.0	29.0	23.3	17.0	11.0
FCGF [8]	85.1	84.7	83.3	81.6	71.4	40.1	41.7	38.2	35.4	26.8
D3Feat [9]	81.6	84.5	83.4	82.4	77.9	37.2	42.7	46.9	43.8	39.1
SpinNet [7]	88.6	86.6	85.5	83.5	70.2	59.8	54.9	48.3	39.8	26.8
Predator [28]	89.0	89.9	90.6	88.5	86.6	59.8	61.2	62.4	60.8	58.1
YOHO [47]	90.8	90.3	89.1	88.6	84.5	65.2	65.5	63.2	56.5	48.0
GeGS-PCR (ours)	**97.9**	**97.6**	**97.5**	**96.7**	**97.6**	**90.7**	**90.2**	**90.4**	**90.0**	**89.8**

Table 7: Performance of Self-attention Models

	Vanilla Self-attention			Geometric Self-attention			3DGS Self-attention		
Overlap	PIR(%)	IR(%)	RR(%)	PIR(%)	IR(%)	RR(%)	PIR(%)	IR(%)	RR(%)
0.9-1.0	0.974	0.829	1.000	0.989	0.894	1.000	**0.990**	0.905	1.000
0.8-0.9	0.948	0.787	1.000	0.969	0.859	1.000	**0.969**	0.873	1.000
0.7-0.8	0.902	0.731	0.931	0.935	0.815	0.931	**0.938**	0.853	0.980
0.6-0.7	0.884	0.686	0.933	0.939	0.783	0.946	**0.947**	0.905	0.968
0.5-0.6	0.843	0.644	0.957	0.913	0.750	0.970	**0.920**	0.872	0.963
0.4-0.5	0.787	0.579	0.935	0.867	0.689	0.944	**0.872**	0.824	0.953
0.3-0.4	0.716	0.523	0.917	0.818	0.644	0.940	**0.825**	0.791	0.944
0.2-0.3	0.560	0.406	0.781	0.666	0.518	0.839	**0.669**	0.762	0.913
0.1-0.2	0.377	0.274	0.639	0.466	0.372	0.705	**0.512**	0.669	0.866

Ablation experiment. Additionally, in Table 8, we conduct a detailed analysis of the performance of recent techniques in terms of Relative Rotation Error (RRE) (the distance between the predicted rotation matrix and the true rotation matrix) and Relative Translation Error (RTE) (the Euclidean distance between the predicted translation vector and the true translation vector). From the comparison of results, it is evident that our GeGS-PCR consistently achieves better performance.

In Table 9, we compare the performance of five loss functions in point cloud registration: (a) Cross-entropy loss, (b) Weighted cross-entropy loss, (c) Circle loss, (d) Overlap-aware circle loss, and (e) Photometric optimization loss. As the loss function improves from cross-entropy to photometric optimization, the performance on all key metrics (PIR, FMR, IR, RR) also improves. The photometric optimization loss achieves the highest performance with 87.6% PIR, 98.2% FMR, 71.6% IR, and 91.9% RR on C3DM, and 56.1% PIR, 89.3% FMR, 44.2% IR, and 75.7% RR on C3DLM, outperforming all other methods.

Fig. 6 shows the training loss curves for both the standard model (without LoRA) and the model with LoRA applied on the Color3DMatch dataset. It can be observed that for all loss types (Loss, C-Loss, and F-Loss),

Table 8: Performance of RRE and RTE on C3DM and C3DLM.

Model	Estimator	C3DM		C3DLM	
		RRE(°)	RTE(m)	RRE(°)	RTE(m)
Predator [28]	RANSAC-50k	2.029	0.064	3.048	0.093
CoFiNet [44]	RANSAC-50k	2.002	0.064	3.271	0.090
GeoTransformer [15]	RANSAC-free	1.772	0.061	2.849	0.088
REGTR [6]	RANSAC-free	1.567	0.049	2.827	0.077
PEAL [19]	RANSAC-free	1.748	0.062	2.788	0.087
ColorPCR [20]	RANSAC-free	1.492	0.048	2.581	0.075
GeGS-PCR	RANSAC-free	**1.112**	**0.024**	**2.293**	**0.051**

the model with LoRA consistently outperforms the one without LoRA, as indicated by the lower loss values throughout the training process. Specifically, the losses for the LoRA-enhanced model decrease more steadily and reach a lower final value compared to the model without LoRA, suggesting that the LoRA optimization aids in faster convergence and better performance during training. This indicates that LoRA contributes effectively to improving the model's training efficiency and overall performance in point cloud registration tasks.

A.6 Limitations

GeGs-PCR relies on superpoints (patches) extracted through downsampling during the registration process. In regions with high point cloud overlap, the large number of superpoints, combined with 3DGS parameterization, can result in significant memory usage and computational overhead. Since patch decomposition shares similarities with semantic scene understanding, we plan to leverage semantic scene understanding for point cloud pair registration in future research. Furthermore, our current registration method involves pairing after 3DGS parameterization of domain superpoints. In future work, we aim to explore scene-level registration of 3DGS for more realistic environmental registration.

Table 9: Performance of ablation experiments

Overlap	C3DM				C3DLM			
	PIR(%)	FMR(%)	IR(%)	RR(%)	PIR(%)	FMR(%)	IR(%)	RR(%)
(a) Cross-entropy loss	80.0	97.7	65.7	90.0	45.9	85.1	37.4	68.4
(b) Weighted cross-entropy loss	83.2	**98.0**	67.4	90.0	49.0	86.2	38.6	70.7
(c) Circle loss	85.1	97.8	69.5	90.4	51.5	86.1	41.3	71.5
(d) Overlap-aware circle loss	86.1	97.7	70.3	91.5	54.9	88.1	43.3	74.0
(e) Photometric optimization loss	**87.6**	97.9	**71.6**	**91.9**	**56.1**	**89.3**	**44.2**	**75.7**

A.7 Qualitative Results

Fig. 7 and Fig. 8 show the registration results of GeGS-PCR on the Color3DMatch (C3DM), Color3DLoMatch (C3DLM), and ColorKitti datasets. GeGS-PCR achieves precise registration in scenarios with low overlap and subtle geometric features, demonstrating its exceptional performance in handling complex environments. From the results, it can be observed that even with smaller overlapping regions between point clouds, GeGS-PCR is still able to accurately align different point clouds. In particular, in several scenes from Fig. 6, GeGS-PCR effectively handles point cloud pairs with minimal overlap and is able to precisely reconstruct the spatial structure of the point clouds. In the ColorKitti dataset shown in Fig. 8, GeGS-PCR demonstrates its robustness, providing accurate registration results even in complex scenarios.

Fig. 9 shows the registration performance comparison between GeGS-PCR and Geotransformer across various overlap conditions. The results demonstrate that in high overlap scenarios, GeGS-PCR exhibits superior registration accuracy compared to Geotransformer, particularly in terms of point correspondence and inlier ratio. For example, with a 79.7% overlap, GeGS-PCR achieves an inlier ratio of 93.1%, while Geotransformer only reaches 40.6%. In low overlap conditions, GeGS-PCR's advantage becomes even more evident. For instance, with a 41.8% overlap, GeGS-PCR's RMSE is 0.006m, much lower than Geotransformer's 0.877m. GeGS-PCR is able to effectively capture the correct correspondences between point clouds in low-overlap scenarios, maintaining high registration accuracy. Overall, GeGS-PCR demonstrates higher stability and precision across various overlap conditions, proving its robustness in complex scenarios.

Advantages. The advantages of GeGS-PCR lie in the collaborative optimization of global and local structures. Through local Gaussian feature extraction, GeGS-PCR effectively suppresses noise interference and robustly

fuses geometric and color features. In low-overlap or noisy point cloud data, GeGS-PCR dynamically adjusts local geometric distribution through covariance modeling, significantly improving registration accuracy. Additionally, photometric loss based on differentiable rendering optimizes global pose consistency, ensuring stable registration. GeGS-PCR is not only effective for high-overlap registration but also offers an efficient, scalable solution for low-overlap tasks like autonomous driving and large-scale scene reconstruction.

Additionally, by leveraging both geometric and color information, GeGS-PCR is able to find consistent feature correspondences in low-overlap regions, improving registration accuracy. The changes between point clouds before and after registration in the qualitative results clearly illustrate how our model handles challenging regions. Notably, in areas with similar geometric features such as floors and walls or appliances and furniture, GeGS-PCR maintains high accuracy. Overall, GeGS-PCR showcases superior registration capability and robustness when dealing with low-overlap scenarios, complex structures, and environments with rich color information.

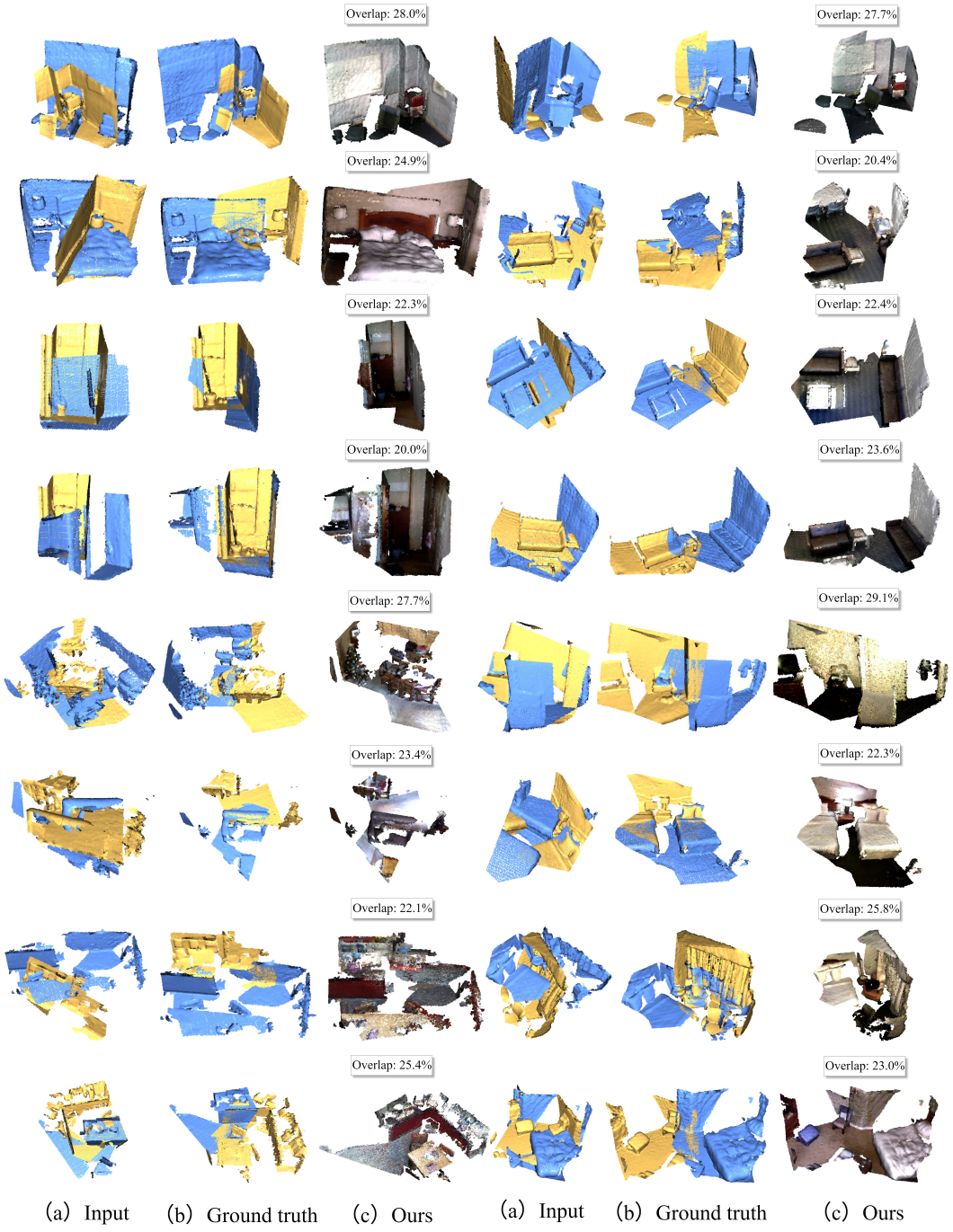


Figure 7: Registration results on Color3DMatch and Color3DLoMatch.

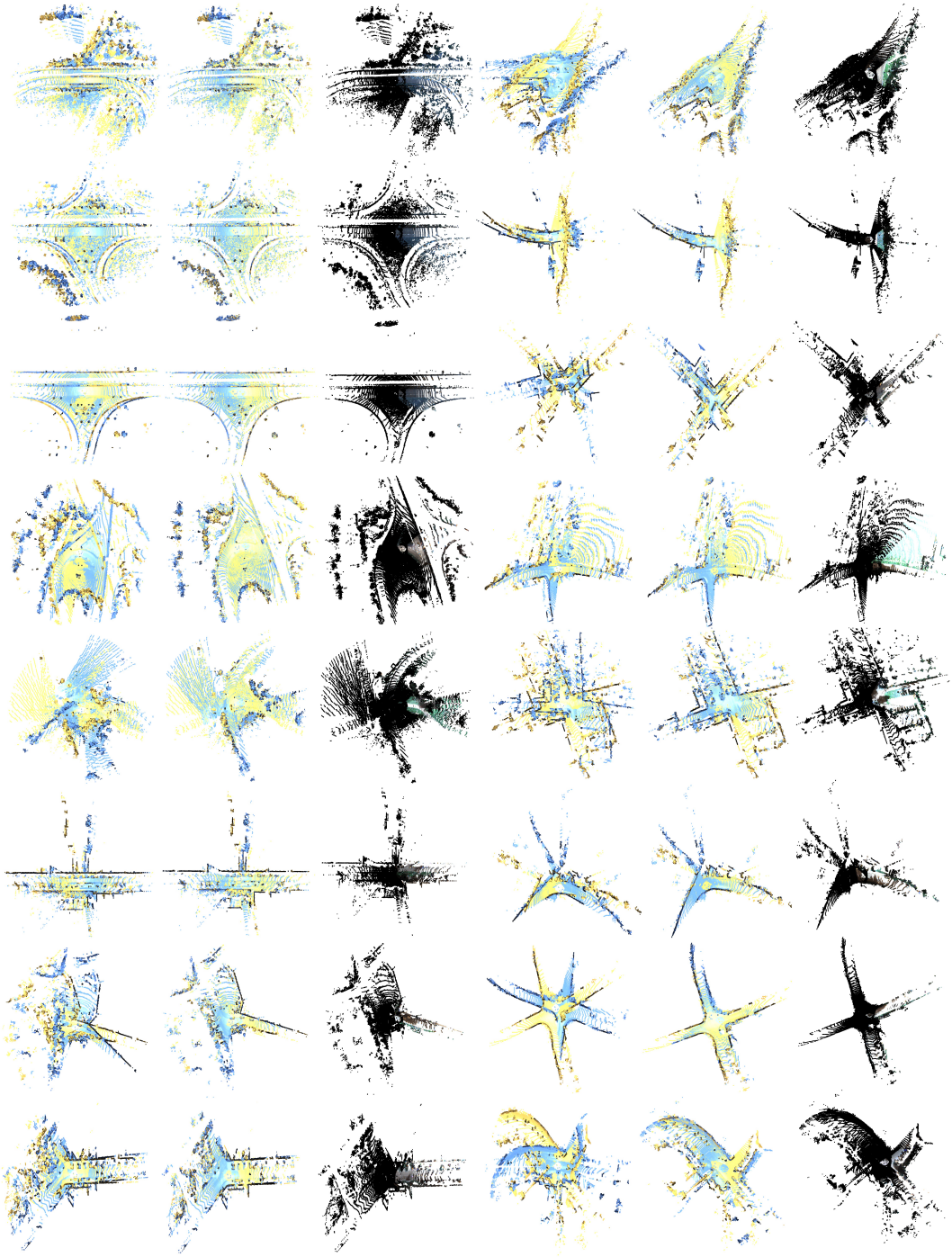


Figure 8: Registration results on ColorKitti.

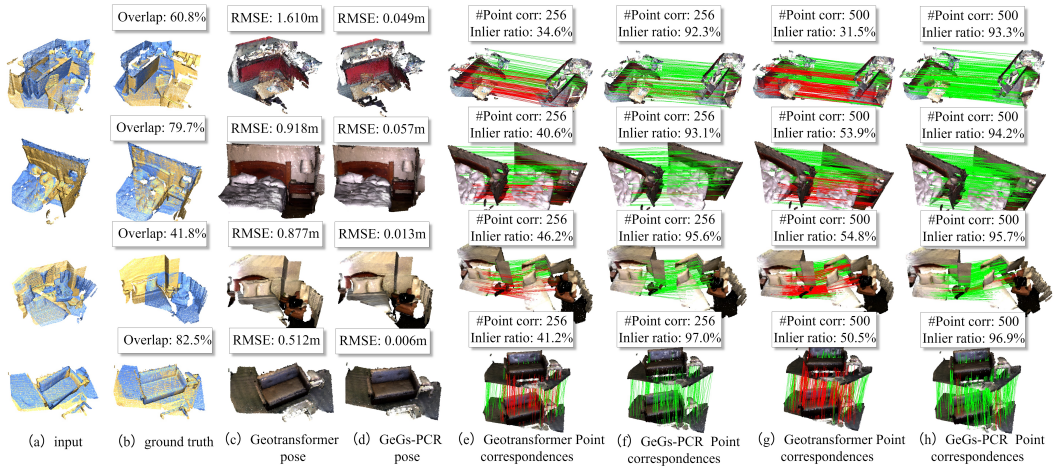


Figure 9: Registration performance with GeGS-PCR and Geometric Self-Attention across various overlap conditions.