Register Any Point: Scaling 3D Point Cloud Registration by Flow Matching

Yue Pan[1] Tao Sun[2] Liyuan Zhu[2] Lucas Nunes[3]
Iro Armeni[2] Jens Behley[1,4] Cyrill Stachniss[1,4]

[1] Center for Robotics, University of Bonn, Germany
[2] Stanford University, USA
[3] RWTH Aachen University, Germany
[4] Lamarr Institute for Machine Learning and Artificial Intelligence, Germany

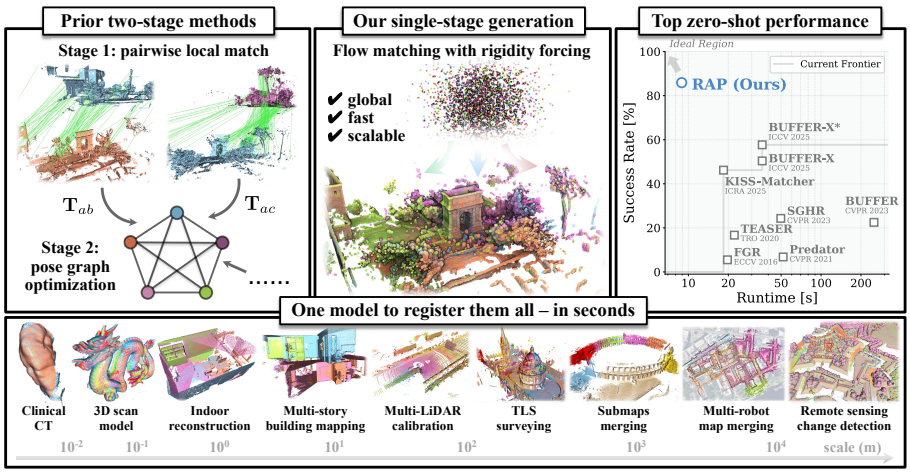


Fig. 1: Our method for multi-view point cloud registration. Prior works perform pairwise registration via correspondence matching and then conduct pose graph optimization (top-left). We use a single-stage flow matching model to generate the registered points (top-middle). Our model generalizes across various view counts, scales, and sensor modalities (bottom) and achieves superior zero-shot performance with the shortest runtime on the cross-domain multi-view registration benchmark (top-right).

Abstract. Point cloud registration aligns multiple unposed point clouds into a common reference frame and is a core step for 3D reconstruction and robot localization when no initial pose guess is available. In this work, we cast point cloud registration as conditional generation: a learned, continuous point-wise velocity field transports noisy points to a registered scene, from which the pose of each view is recovered. Unlike prior methods that perform correspondence matching to estimate pairwise transformations and then optimize a pose graph for multi-view registration, our model directly generates the registered point cloud, yielding both efficiency and point-level global consistency. By scaling the training data and conducting test-time rigidity enforcement, our approach achieves state-of-the-art average performance on existing pairwise regis-

tration benchmarks and on our proposed cross-domain multi-view registration benchmark. The superior zero-shot performance on this benchmark demonstrates that our method generalizes across view counts, scene scales, and sensor modalities even with low overlap.

Keywords: Point Cloud Registration · Flow Matching

1 Introduction

Point cloud registration is a cornerstone in 3D vision, robotics, and photogrammetry, with broad applications, from merging multiple partial 3D scans into a consistent 3D model to localizing sensors in an existing 3D map for downstream tasks, including simultaneous localization and mapping (SLAM) [65, 107], 3D reconstruction [24], and robotic manipulation [68]. Yet, obtaining reliable registration in the wild without an initial guess is a hard problem. Real-world data is sparse, noisy, and non-uniform in density. Sensors differ in modality and calibration, overlaps between point clouds can be small, and local matches can be ambiguous [2, 39, 57, 79].

Prevailing approaches for multi-view point cloud registration follow a two-stage pipeline: align all overlapping pairs of scans independently, then solve a global pose graph to enforce consistency [37, 75]. Pairwise alignment typically relies on matching local feature correspondences with a robust estimator [7, 28, 47, 53, 97]. While conceptually appealing, this has two limitations: (i) quadratic complexity, where cost scales quadratically with the number of scans due to exhaustive pairwise registration across all pairs, and (ii) limited global context, where the pairwise stage limits capturing global context, hurting performance with low overlap and incomplete observations. Although specialized modules can improve low-overlap pairwise registration [39, 98] and some works conduct hierarchical registration [25] or edge selection [84] to avoid the quadratic cost, these add complexity while remaining tied to iterative pose-graph refinement sensitive to pairwise alignment errors.

Recent 3D vision research departs from this two-stage pipeline by leveraging feed-forward and generative models. In image-based 3D reconstruction, feed-forward approaches [49, 89] encapsulate the entire structure-from-motion process into a single neural network, directly producing globally consistent poses and dense geometry from a set of images. VGGT [87] demonstrates that a large transformer can infer all key 3D attributes, including camera poses and depth maps, from one or many views in a single pass. In the point cloud domain, Rectified Point Flow (RPF) [80] pioneered a generative approach to pose estimation by learning a continuous flow field that moves points from random noise to their assembled target positions for multiple object-centric benchmarks. These findings suggest that a single feed-forward model can holistically reason about multiple partial observations and produce a consistent 3D alignment, given sufficient capacity and training data.

Scaling such single-stage models to large-scale, multi-view 3D registration, however, raises another key challenge: the sampling process does not always yield

stable, perfectly rigid predictions, especially in cluttered environments where geometry is more diverse than in object-centric settings. Even with an explicit projection step of the final prediction onto SE(3), as in RPF [80], the post-hoc correction cannot constrain the entire flow trajectory. Thus, the sampled flows can drift away from the flow distribution on which the model was trained, limiting performance.

This motivates our work, a scalable generative model that aligns multiple point clouds in a single stage while explicitly enforcing rigidity. Instead of exhaustive pairwise transformation estimation, the model learns to transform all input point clouds directly into a canonical coordinate frame, effectively fusing them into a coherent scene. To make generation robust and satisfy rigid constraints, we propose using rigidity as a guidance signal for flow sampling at test time. To train at scale, we curate over 100k multi-view registration instances from 17 diverse datasets spanning object-centric, indoor, and outdoor settings. Supervising in Euclidean space across this mixture of data provides strong scene priors that enable the model to complete partial views and generalize across scales and sensor modalities. To address the lack of a unified evaluation protocol for cross-domain generalization, we also introduce a multi-view registration benchmark spanning five scene categories across diverse scales, sensors, and overlap conditions. We will release our code, model, and proposed benchmark to facilitate reproducibility at: `https://github.com/PRBonn/RAP`.

In summary, our contributions are three-fold:

- We propose RAP, a generative flow-matching model for single-stage multi-view point cloud registration, together with a rigidity-enforcing sampling strategy that enforces per-scan rigid constraints at test time, achieving state-of-the-art performance on both pairwise and multi-view point cloud registration benchmarks.
- We develop a large-scale training recipe that aggregates over 100k multi-view registration instances from 17 heterogeneous datasets, enabling strong generalization across diverse scenarios, scales, and sensor modalities, including challenging low-overlap conditions.
- We introduce a challenging cross-domain multi-view registration benchmark spanning five scene categories, on which our method substantially outperforms existing approaches in a zero-shot manner.

2 Related Work

Pairwise point cloud registration has long relied on local feature matching with robust transformation estimators [28, 54, 97]. Early approaches rely on hand-crafted local descriptors [26, 35, 61, 72, 73, 83] to establish correspondences. Modern methods [4, 8, 9, 17, 18, 21, 32, 39, 51, 57, 67, 70, 77, 78, 85, 86, 94, 98, 101–103] replace or augment these with learned local features. Beyond explicit matches, correspondence-free and end-to-end approaches [5, 10, 91, 92, 100] directly supervise the transformation with differentiable assignment or iterative refinement. Our approach departs from both families: rather than seeking correspondences

or iteratively refining poses, we learn a conditional velocity field that transports noise to the merged scene, from which rigid transformations are recovered. Closer to our approach, DeepVCP [59] and DeepPRO [48] generate virtual corresponding points near the target, but remain limited to pairwise registration with a transformation initial guess or object-scale scenes.

Multi-view point cloud registration is typically handled by first estimating local pairwise relative poses and then synchronizing the transformations via pose graph optimization [20,33]. In practice, optimization is performed using a factor graph with one node per scan and edges that encode relative pose measurements and their uncertainties under various robust objectives [16, 24, 25, 31, 43, 82, 84]. By contrast, our single-stage approach aligns an arbitrary number of scans at once, enforcing multi-view consistency by construction. As a result, our method dispenses with the need for a separate pose graph optimization stage and avoids the quadratic costs from pairwise transformation estimation. Nonetheless, our predictions can still serve as strong and time-efficient initializations for downstream solvers that incorporate additional signals (e.g., from gravity, IMU, or GNSS) or task-specific constraints.

Generative modeling approaches for 3D data leverage diffusion- and flow-based models to generate geometric structures [12,27,34,41,50,63,71,74,80, 93, 96, 105, 106]. Both formulate generation as stochastic transport from source to target distributions: diffusion via iterative denoising, while flow-matching predicts velocity fields to iteratively transform data. These models have been applied to text-to-shape generation [74, 96, 105], 3D scene completion [27, 63, 106], and annotated data generation [12, 34, 64, 71].

More recent works leverage diffusion and flow-matching models to achieve point cloud registration [1, 41, 50, 80] as a way to overcome limitations of standard approaches. In DiffusionReg [41], point cloud registration is formulated as a diffusion process on the SE(3) manifold, generating the corresponding rigid-body transformation between the source and target point clouds. Closest to our approach, RPF [80] applies Euclidean-space conditional flow matching to pairwise registration and multi-part shape assembly, but is limited to object-scale scenes (e.g., furniture and tableware). Concurrently, FUSER [42] proposes a feed-forward transformer for multi-view registration with SE(3) diffusion refinement, though it focuses on indoor scenes. RAP extends Euclidean-space flow matching to large-scale, cross-domain multi-view registration and achieves superior performance by scaling the training data, performing canonicalized generation conditioned on local embeddings, and rigidifying the flow sampling at test time.

3 Generative Point Cloud Registration

In this section, we present our generative approach to multi-view point cloud registration. We formulate registration as conditional flow matching, where a transformer-based model learns to directly generate the aggregated registered point cloud from unposed inputs. We then describe our model architecture and training procedure, and how we enforce generation rigidity during inference.

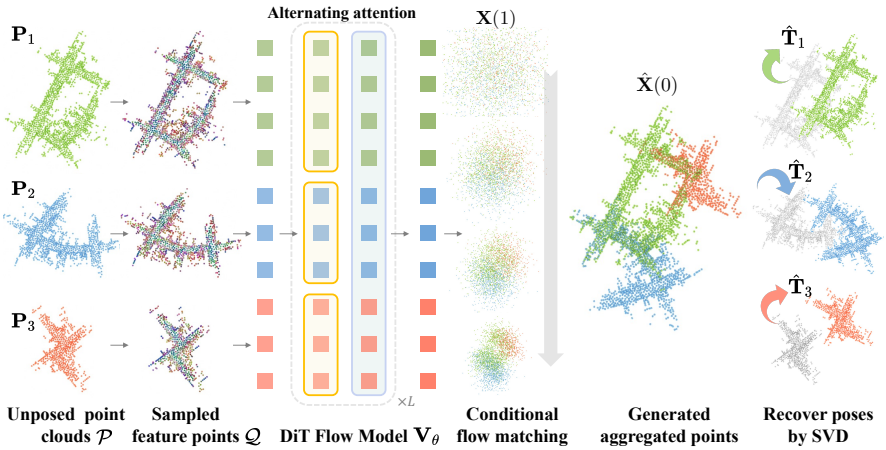


Fig. 2: Overview of our approach to multi-view point cloud registration. Starting with unposed point clouds $\mathcal{P}$, we sample keypoints $\mathcal{Q}$ with corresponding local features $\mathcal{F}$. We use a diffusion transformer (DiT) with alternating-attention blocks for conditional flow matching that generates the registered point cloud $\widehat{\mathbf{X}}(0)$ from Gaussian noise $\mathbf{X}(1)$. Finally, we recover the individual transformations $\widehat{\mathbf{T}}_i$ using SVD and apply them to the original unposed point clouds. The example illustrates submap registration on the KITTI [30] dataset.

3.1 Problem Definition

We consider the general multi-view point cloud registration problem. The input is a set of $N \geq 2$ unordered point clouds $\mathcal{P} = \{\mathbf{P}_i \in \mathbb{R}^{3\times M_i}\}_{i=1}^N$, where M_i is the point count of the i-th point cloud. These point clouds may come from individual LiDAR or depth-camera scans, or from accumulated point cloud maps built by SLAM or photogrammetry systems. We do *not* assume any initial guess of the transformation from the coordinate frame of each point cloud to a global frame, but we assume that the point clouds are observations of the same scene and can be registered into a single connected point cloud, even if the overlap is small.

The goal is to estimate registered point clouds $\mathcal{P}^r = \{\mathbf{P}_i^r \in \mathbb{R}^{3\times M_i}\}_{i=1}^N$ in a common global coordinate frame. In practice, there may be non-rigid deformations caused by effects such as motion distortion, dynamic objects, or SLAM drift. When these non-rigid effects are negligible, the registered point clouds can be transformed from the inputs by a set of rigid body transformations $\mathcal{T} = \{\mathbf{T}_i \in \mathrm{SE}(3)\}_{i=1}^N$.

3.2 Flow Matching for Multi-View Registration

Following RPF [80], we formulate multi-view registration as a conditional generation problem and directly generate the registered point cloud $\mathcal{P}^r$ given the unposed input $\mathcal{P}$. The rigid transformations $\mathcal{T}$ are then recovered as a by-product. We apply flow matching [58] directly to the 3D Euclidean coordinates of point clouds. The model transports a noised point cloud $\mathbf{X}(1) \in \mathbb{R}^{3\times M}$ sampled from

a standard Gaussian $\mathcal{N}(\mathbf{0}, \mathbf{I})$ to a target $\mathbf{X}(0) \in \mathbb{R}^{3\times M}$ via a learned velocity field $\nabla_t \mathbf{X}(t)$, parameterized by a neural network $\mathbf{V}_\theta(t, \mathbf{X}(t) \mid \mathbf{C})$ conditioned on $\mathbf{C}$, detailed in Sec. 3.4. The forward process is a linear interpolation in 3D space between noise and the target, *i.e.*,

$$\mathbf{X}(t) = (1-t)\mathbf{X}(0) + t\mathbf{X}(1), \quad t \in [0,1]. \tag{1}$$

Flow model $\mathbf{V}_\theta$ is trained by minimizing the conditional flow matching loss [55]:

$$\mathcal{L}_{\text{FM}} = \mathbb{E}_{t,\mathbf{X}} \left[\left\| \mathbf{V}_\theta(t, \mathbf{X}(t) \mid \mathbf{C}) - \nabla_t \mathbf{X}(t) \right\|^2 \right]. \tag{2}$$

For our registration task, the target $\mathbf{X}(0)$ is the aggregated registered point cloud $\mathbf{P}^r = \bigcup_{i=1}^{N} \mathbf{P}_i^r$.

At inference time, we reconstruct the registered point cloud by numerically integrating the predicted velocity field $\mathbf{V}_\theta(t, \mathbf{X}(t) \mid \mathbf{C})$ from $t = 1$ to $t = 0$. In practice, we use $\kappa = 10$ uniform Euler steps:

$$\widehat{\mathbf{X}}(t - \Delta t) = \widehat{\mathbf{X}}(t) - \mathbf{V}_\theta(t, \widehat{\mathbf{X}}(t) \mid \mathbf{C})\Delta t, \tag{3}$$

with $\Delta t = 1/\kappa$. After integration, the resulting $\widehat{\mathbf{X}}(0)$ approximates the registered point cloud. We then partition $\widehat{\mathbf{X}}(0)$ into per-view subsets and estimate the corresponding poses $\widehat{\mathbf{T}}_i$ via the Kabsch algorithm [6] using SVD.

3.3 Rigidity-Enforcing Inference

The flow model alone does not guarantee per-view rigidity. While allowing non-rigid point motions increases the model's expressiveness, it can also drive sampling trajectories away from the training distribution defined in Eq. (1). We therefore exploit the Euclidean nature of our flow formulation and introduce a *rigidity-enforcing* Euler integration that projects intermediate predictions onto per-view SE(3) orbits at each step. In addition, we empirically find that the resulting rigidity error provides an effective criterion for selecting among multiple generations.

We define the per-view projection operator Π of an estimate $\widehat{\mathbf{X}}_i(0)$ onto the rigid orbit of the input $\mathbf{P}_i$, as

$$\Pi_{\mathbf{P}_i} \left(\widehat{\mathbf{X}}_i(0) \right) := \widehat{\mathbf{R}}_i \, \mathbf{P}_i + \widehat{\mathbf{t}}_i, \tag{4}$$

where $(\widehat{\mathbf{R}}_i, \widehat{\mathbf{t}}_i)$ is the optimal rigid transformation between $\mathbf{P}_i$ and $\widehat{\mathbf{X}}_i(0)$, computed via the Kabsch algorithm [6]. Given the current state $\mathbf{X}(t)$ and the velocity prediction $\mathbf{V}_\theta(t, \mathbf{X}(t) \,|\, \mathbf{C})$, we extrapolate the registered point cloud estimate, as

$$\widehat{\mathbf{Y}}(0) := \mathbf{X}(t) - t \, \mathbf{V}_\theta(t, \mathbf{X}(t) \,|\, \mathbf{C}). \tag{5}$$

We now rigidify it to obtain the rigid projection of all views $\Pi_{\mathcal{P}}$ following Eq. (4) and compute the flow at the next step $t' \leftarrow t - \Delta t$ as

$$\mathbf{X}(t') := (1 - t') \, \Pi_{\mathcal{P}} \left(\widehat{\mathbf{Y}}(0) \right) + t' \, \mathbf{X}(1). \tag{6}$$

We repeat the above sampling step until t reaches 0.

3.4 Canonicalized Registration Pipeline

Our method is designed to scale across scenes with diverse point densities, arbitrary coordinate frames, and a metric scale ranging from object-level scans to large outdoor environments. For this, we introduce a *canonicalized keypoint-based* registration pipeline for our flow model. The pipeline has four steps: (i) sampling a compact keypoint representation with local descriptors, (ii) canonicalizing all views into a shared similarity-invariant frame, (iii) conditioning a flow network on this representation to generate a canonical registered point cloud, and (iv) lifting the canonical prediction back to the original dense point clouds.

(i) Keypoint selection with local descriptors. Directly generating millions of points is computationally inefficient and unstable. Instead, we construct a compact, geometry-aware representation by sampling keypoints in each view and attaching local descriptors. For each input point cloud $\mathbf{P}_i$, we first perform voxel downsampling with voxel size v_d, obtaining a reduced cloud $\mathbf{P}_i^v$. We then apply the farthest point sampling to $\mathbf{P}_i^v$ to select K_i keypoints, forming $\mathbf{Q}_i \in \mathbb{R}^{3\times K_i}$, with the total keypoint count $K = \sum_i K_i$ across all views. To obtain uniform coverage over the scene, we choose K_i in proportion to the metric scale of $\mathbf{P}_i$.

To encode local geometry, for each sampled point in $\mathbf{Q}_i$ we define a local patch by a ball query of radius $r_s = 20v_d$ in the reduced point cloud $\mathbf{P}_i^v$ and extract a local descriptor from the normalized points within this patch. We use the lightweight, rotation-invariant MiniSpinNet [3,77] pretrained on 3DMatch [104], yielding $\mathbf{F}_i \in \mathbb{R}^{32\times K_i}$. Concatenating all views, we obtain sampled points $\mathcal{Q} = \{\mathbf{Q}_i \in \mathbb{R}^{3\times K_i}\}_{i=1}^N$ and their local features $\mathcal{F} = \{\mathbf{F}_i \in \mathbb{R}^{D\times K_i}\}_{i=1}^N$, which serve as a compressed yet informative representation of the dense input clouds $\mathcal{P}$.

(ii) Canonicalization of inputs and targets. To make training invariant to global pose and metric scale, we canonicalize both the conditioning representation and the flow target in a shared frame. For each view i, we first translate the sampled points $\mathbf{Q}_i$ so that its center of mass is at the origin. We then compute a global scale factor s as the longest edge length of the bounding box of the view with the most points, and scale all centered point sets by $1/s$, so that the entire scene fits in a unit cube. Finally, we apply a random 3D rotation to each centered, scaled cloud. This yields normalized unposed keypoints $\bar{\mathcal{Q}} = \{\bar{\mathbf{Q}}_i \in \mathbb{R}^{3\times K_i}\}_{i=1}^N$ and corresponding similarity transforms $\bar{\mathbf{T}}_i \in \mathrm{SIM}(3)$ that map the original $\mathbf{Q}_i$ to $\bar{\mathbf{Q}}_i$ in the canonical frame.

The training target is defined in the same canonical frame. We first transform the keypoints by the ground-truth poses $\widehat{\mathcal{T}}$ to obtain registered keypoints $\mathcal{Q}^r = \{\mathbf{Q}_i^r\}_{i=1}^N$ and merge them into $\mathbf{Q}^r = \bigcup_{i=1}^N \mathbf{Q}_i^r$. We then (i) recenter $\mathbf{Q}^r$ at the origin, (ii) apply the same random rotation used for the view with the most points to fix a reference orientation, and (iii) scale by the global factor s, resulting in the normalized registered point cloud $\bar{\mathbf{Q}}^r$, which is set as the target $\mathbf{X}(0)$.

(iii) Conditional flow model. We adopt the Diffusion Transformer [66] for $\mathbf{V}_\theta$, and, following VGGT [87], employ a transformer with alternating attention blocks (Fig. 2). Specifically, we alternate per-view self-attention within each point cloud, which consolidates view-specific structure, with global attention over all point tokens to fuse information across views. Our model comprises $L = 10$

alternating attention blocks with hidden dimension $d = 512$ and $h = 8$ attention heads, totaling 73 million parameters.

The model's condition $\mathbf{C} = f_{\text{emb}}(\bar{\mathcal{Q}}, \mathcal{F})$ is obtained via a linear feature embedder f_{emb} that maps the concatenation of 32-dimensional MiniSpinNet descriptors $\mathcal{F}$ and Fourier-encoded [62] 3D coordinates $\bar{\mathcal{Q}}$ (63 dimensions) to the hidden dimension d=512. The flow model $\mathbf{V}_\theta$ takes $\mathbf{X}(t)$ and $\mathbf{C}$ as input. Importantly, we do not condition the model $\mathbf{V}_\theta$ on view indices, making the architecture view-count-agnostic and allowing it to generalize to larger numbers of views at test time than seen during training.

(iv) Lifting to dense registered point clouds. At inference, the model generates a canonical registered keypoint cloud $\widehat{\mathbf{X}}(0)$. Using the rigidity-enforcing procedure in Sec. 3.3, we recover per-view rigid transformations $\widehat{\mathbf{T}}_i$ that align $\bar{\mathbf{Q}}_i$ to the corresponding subsets of $\widehat{\mathbf{X}}(0)$. The overall transformation from the original input frame of view i to the final registered frame is $\mathbf{T}_i = \widehat{\mathbf{T}}_i \bar{\mathbf{T}}_i$, which we apply to all points in the dense cloud $\mathbf{P}_i$. Finally, we undo the global scaling by s to obtain the registered point clouds at the metric scale.

4 Experimental Evaluation

We present our experiments on pairwise and multi-view point cloud registration to demonstrate the effectiveness of RAP.

4.1 Experimental Setup

Implementation details. We train our model for three days with about 120k iterations using 32 NVIDIA A100 GPUs with 80 GB VRAM each. We use Muon [56] as the optimizer with an initial learning rate of $2 \cdot 10^{-3}$ for matrix-like parameters and $2 \cdot 10^{-4}$ for vector-like parameters.

Unlike VGGT [87] or RPF [80], our training samples have varying numbers of views N and feature point (token) counts K. To efficiently train under this irregular data setup, we devise a dynamic batching strategy that allocates samples to each GPU with a suitable batch size. We set the maximum token count per batch on one GPU to 110,000.

Training data. Our training requires only a set of point clouds under the same reference frame without any annotations for keypoints or correspondences. This makes it straightforward to scale up training as any dataset providing point clouds and accurate sensor poses can be used.

For our model, we curate over 100k multi-view registration instances from 17 diverse datasets spanning outdoor, indoor, and object-centric settings. We train on 12 outdoor LiDAR datasets KITTI [29], KITTI360 [52], Apollo [40], nuScenes [15], MulRAN [45], Boreas [14], Oxford Spires [81], VBR [13], UrbanNav [36], WildPlace [46], HeLiPR [44], and KITTI-Carla [23]. We also use 4 indoor depth camera datasets 3DMatch [104], NSS [79], ScanNet [19], and ScanNet++ [99], and the object-centric dataset ModelNet [95]. Each sample consists

Table 1: Pairwise registration success rate (%) on six commonly used pairwise registration benchmarks for object-centric, indoor, and outdoor scenarios. The best result is in **bold**, and the second best is underscored.

Method	ModelNet	3DMatch	3DLoMatch	NSS	ETH	KITTI	Avg.
FPFH+FGR [108]	84.04	62.53	15.42	30.82	91.87	98.74	63.90
FPFH+TEASER [97]	86.10	52.00	13.25	25.78	93.69	98.92	61.62
KISS-Matcher [54]	87.12	67.22	20.44	53.69	96.77	**100.0**	70.87
FCGF [18]	16.51	88.18	40.09	42.86	55.53	98.92	58.68
Predator [39]	84.36	90.60	62.40	92.99	54.42	99.82	80.77
GeoTransformer [70]	86.26	92.00	75.00	55.59	71.53	99.82	80.03
BUFFER [3]	92.42	92.90	71.80	72.44	99.30	99.64	88.12
PARENet [98]	66.14	95.00	**80.50**	45.07	69.42	99.82	75.99
BUFFER-X [77]	**99.84**	95.58	74.18	85.60	**99.72**	99.82	92.46
RAP (Ours)	99.13	**95.90**	78.78	**96.27**	99.44	**100.0**	**94.92**

of N views of point clouds, where $2 \leq N \leq 16$. We split the samples into training and validation sets with an approximate ratio of 9:1 while also excluding sequences used for testing in common registration benchmarks from the training set. The datasets span diverse scenes across multiple continents, captured by LiDAR and depth cameras with varying resolutions and fields of view. We curate both single-frame and sequence-accumulated submap samples. We require all point clouds in a sample to form a connected overlap graph, but deliberately include hard samples with very low pairwise overlap to improve the model's robustness to low-overlap scenarios.

4.2 Pairwise Registration Evaluation

We first evaluate our model on pairwise registration across six widely used benchmarks: ModelNet [95] for object-centric scenes, 3DMatch [104], 3DLoMatch [39], and NSS [79] for indoor depth-camera scenarios, and ETH [69] and KITTI [29] for outdoor LiDAR scenarios. Following prior work, we evaluate performance using registration success rate (%), computed with thresholds on correspondence RMSE for 3DMatch and 3DLoMatch, and on translation and rotation error for the remaining datasets. We adopt the success thresholds used in previous studies [77, 79, 104]. Details are provided in the supplementary material. We compare against three conventional baselines based on hand-crafted features (FPFH+FGR [108], FPFH+TEASER [97], KISS-Matcher [54]) and six learning-based methods (FCGF [18], Predator [39], GeoTransformer [70], BUFFER [3], PARENet [98], BUFFER-X [77]).

Tab. 1 shows that our model ranks first on half of the benchmarks and achieves the best average performance across all six benchmarks compared to state-of-the-art methods. We note that our model is trained on more diverse data, while some learning-based baselines are trained solely on 3DMatch or KITTI. We provide additional results on pairwise registration under low overlap in the supplementary material.

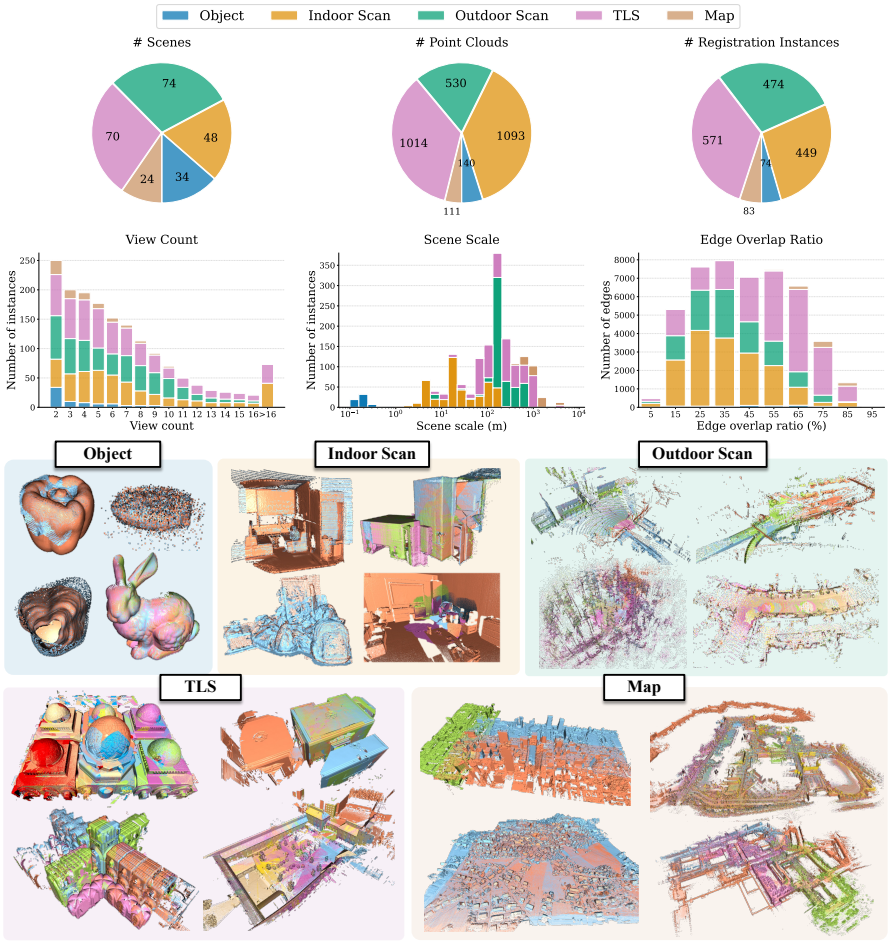


Fig. 3: Overview of our proposed cross-domain multi-view registration benchmark, showing key dataset statistics (top), and representative samples from each scenario category (bottom). Different colors of the point cloud denote different views.

4.3 Multi-View Registration Evaluation

Cross-domain multi-view registration benchmark. We identify a gap in the evaluation of point cloud registration methods on challenging real-world data. While pairwise registration success rates on established benchmarks such as ModelNet, ETH, and KITTI are nearly saturated (see Tab. 1), methods that perform well on these benchmarks often fail to generalize to practical applications. Furthermore, while some prior works use indoor depth-camera datasets [16, 19, 79, 104] for multi-view registration evaluation, no cross-domain benchmark exists covering diverse scales, scenarios, and sensor modalities.

To address this gap, we propose a challenging cross-domain multi-view registration benchmark, as shown in Fig. 3. It aggregates point cloud datasets from

diverse sources, most of which were not designed for registration evaluation, and spans five scene categories: object, indoor scan, outdoor scan, terrestrial laser scanning (TLS), and map. Crucially, none of these datasets overlap with our training mixture, making evaluation zero-shot. These categories reflect different applications. *Object* data support object pose estimation, medical and microscopic 3D imaging. *Indoor scans* support multi-story building mapping, depth camera SLAM, and indoor robot relocalization. *Outdoor scans* support urban reconstruction, autonomous vehicle LiDAR SLAM, and multi-LiDAR extrinsic calibration. *TLS* covers digital cultural heritage, forestry, and infrastructure. *Map*-scale data support multi-robot collaborative SLAM, multi-session map merging, and airborne laser scanning strip registration as well as change detection. Dataset sources and curation details are provided in the supplementary material.

Evaluation metrics and baselines. We adopt the edge success rate and graph success rate as our main evaluation metrics. The edge success rate is the mean over all valid edges (i.e., pairs with non-zero overlap) in a sample, while the graph success rate requires every valid edge in the multi-view registration graph to be successful. To handle diverse scene scales, we normalize translation error by the longest axis of the ground-truth bounding box. A registration succeeds if the normalized translation error is below 2.5% and rotation error below 15°. The 15° threshold is standard for coarse global registration methods, as such errors lie within ICP's convergence basin [11] and can be refined by downstream local alignment. Stricter thresholds (0.5%, 3°) are reported in the supplementary material. At these thresholds, Indoor and Object scenes are more challenging as they lack the large distinctive structures of Map and TLS scenes. The ECDF plots in the supplementary material further show that RAP consistently outperforms baselines across all error ranges.

We select the top-3 performing conventional and learning-based pairwise registration methods from Tab. 1 as baselines, each paired with a robust and efficient implementation of pose graph optimization (PGO) [16] to form a complete two-stage multi-view registration pipeline. For a fair comparison, we retrain BUFFER-X, the best-performing baseline in the pairwise evaluation, on the same extended training data as RAP, denoting this variant BUFFER-X*. We also include two dedicated multi-view methods: SGHR [84], which uses learned overlap scores and history-reweighting iterative optimization, and RPF [80], a flow-matching method targeting object-level shape assembly.

Experimental results. As shown in Tab. 2, RAP achieves the best performance across all five scenario categories by a large margin, improving the overall edge success rate from 57.7% to 85.9% and graph success rate from 36.5% to 77.3% compared to the strongest pairwise baseline BUFFER-X*. Training BUFFER-X with more diverse data yields a modest gain over the original model, yet still falls significantly short of RAP, indicating that data scaling alone is insufficient without a more capable model architecture and training target. Other learning-based methods (Predator [39], BUFFER [3], SGHR [84], RPF [80]) fail to generalize across domains. RPF in particular struggles as it operates on raw points without local descriptors, which does not scale to large scenes.

Table 2: Comparison of the zero-shot testing performance for multi-view registration on the cross-domain multi-view registration benchmark. We report the registration success rate (%) calculated for edges and graphs (with a threshold of 2.5% for normalized translation error and 15° for rotation error) in different scenarios as well as the average runtime. The best result is in **bold**, and the second best is underscored.

Method		Object		Indoor		Outdoor		TLS		Map		All		Runtime (s)
		edge	graph	edge	graph	edge	graph	edge	graph	edge	graph	edge	graph	
FPFH+FGR [108]	+PGO	14.4	6.8	2.8	1.3	4.3	1.1	5.6	2.0	9.0	7.7	5.5	2.2	19.7
FPFH+TEASER [97]	+PGO	18.3	13.5	8.7	5.3	13.9	9.6	17.3	12.4	30.2	19.4	16.7	11.4	22.2
KISS-Matcher [54]	+PGO	38.1	29.7	30.5	14.0	33.4	19.4	63.9	42.5	81.8	70.3	46.2	29.2	18.4
Predator [39]	+PGO	2.8	1.9	9.0	5.9	7.2	5.1	4.3	3.6	8.2	5.7	6.8	4.6	51.8
BUFFER [3]	+PGO	2.3	1.1	25.5	12.0	16.7	9.8	28.7	16.9	3.9	1.5	22.5	11.8	250.3
BUFFER-X [77]	+PGO	67.6	56.8	43.4	22.3	44.6	31.0	63.9	42.1	26.9	17.2	50.4	32.4	36.0
BUFFER-X* [77]	+PGO	70.3	59.2	47.3	25.1	54.0	37.1	69.2	46.8	33.6	21.5	57.7	36.5	36.0
SGHR [84]		12.6	8.1	28.9	22.0	19.8	11.0	24.6	17.6	15.1	10.9	24.3	17.2	49.7
RPF [80]		2.5	1.7	0.8	0.5	0.7	0.3	0.9	0.3	1.3	0.9	1.0	0.6	11.5
RAP (Ours)		**86.3**	**85.1**	**75.1**	**59.6**	**86.6**	**80.6**	**92.2**	**84.6**	**94.0**	**91.6**	**85.9**	**77.3**	**8.9**
RAP w/o rigidity enforcing		83.1	82.4	72.3	54.7	84.7	76.6	90.1	80.0	89.7	86.8	83.6	72.9	8.9

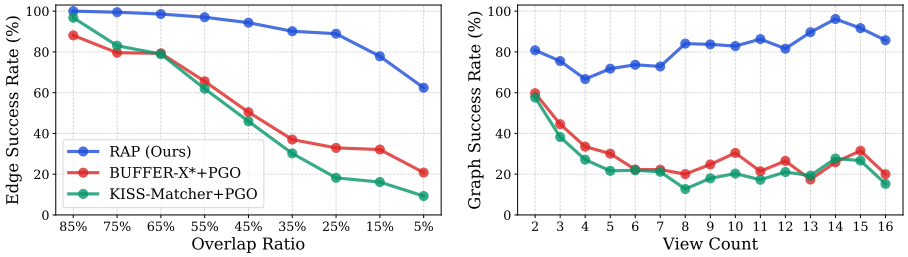


Fig. 4: Multi-view registration performance on the cross-domain point cloud registration benchmark. Left: Edge success rate vs. edge overlap ratio. Right: Graph success rate vs. number of views. RAP outperforms BUFFER-X* and KISS-Matcher across all overlap ratios and view counts.

KISS-Matcher [54] is relatively competitive (46.2%/29.2%) thanks to its scalable hand-crafted features and robust transformation estimator, but is still far behind RAP. RAP w/o rigidity enforcing already surpasses all baselines, with test-time rigidity enforcing integration providing further consistent improvement.

Notably, RAP achieves the shortest runtime at 8.9 s per sample, outpacing all baselines including KISS-Matcher+PGO (18.4 s), BUFFER-X*+PGO (36.0 s), and SGHR (49.7 s). This efficiency stems from RAP's single-stage design: unlike two-stage pipelines that first perform pairwise registration on all or a selected subset of likely-overlapping pairs (as in SGHR) and then run pose graph optimization, RAP directly generates the registered point cloud in one forward pass without any pairwise stage.

Fig. 4 provides an in-depth analysis of RAP's performance as a function of overlap ratio and view count, comparing against the two best-performing baselines. As shown on the left, RAP maintains a consistently higher edge success rate than BUFFER-X*+PGO and KISS-Matcher+PGO across all overlap ratios. The performance gap is modest at high overlap but widens substantially as overlap decreases. At an overlap ratio of 20%, RAP still achieves above 80%

Table 3: Ablation studies. We report the registration success rate (%) on datasets for both pairwise and multi-view registration tasks. Best results are shown in **bold**. On the right figure, we compare the edge success rate of [A]-[C] on the cross-domain multi-view registration benchmark for five different scenario categories.

Setting	Pairwise		Multi-view	
	3DMatch	3DLoMatch	Edge	Graph
[A] Ours	**95.90**	**78.78**	**85.92**	**77.31**
Inference				
[B] w/o rigidity enforcing	94.65	74.63	83.57	72.88
[C] w/ 1-step generation	90.35	61.05	72.87	50.32
Model				
[D] w/o local feature extraction	83.28	50.05	43.07	26.85
[E] w/ $L = 8$ transformer blocks	93.24	71.63	81.08	72.19
[F] w/ $L = 6$ transformer blocks	90.90	57.59	65.64	54.35

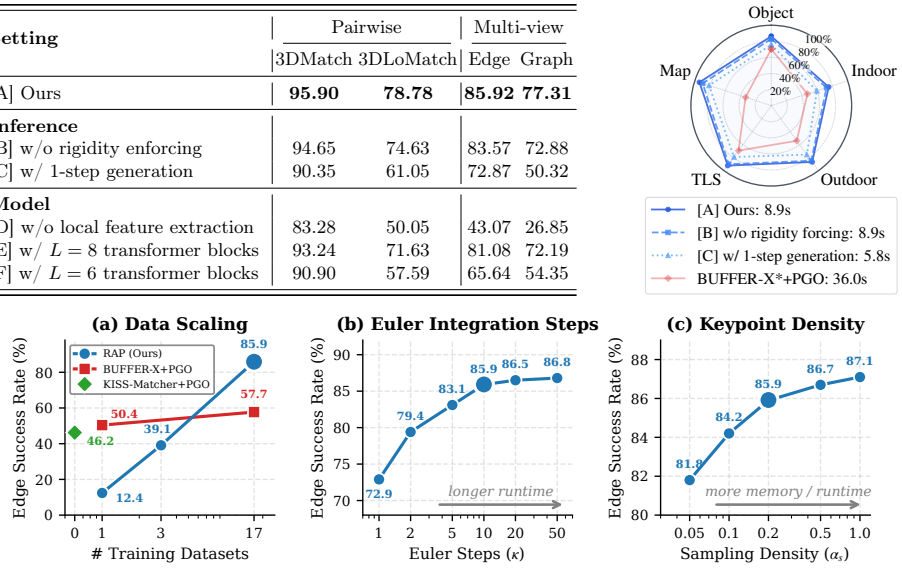


Fig. 5: Data scaling comparison and ablation studies on the cross-domain multi-view registration benchmark. (a) Edge success rate for RAP trained on $\{1, 3, 17\}$ dataset mixes versus BUFFER-X+PGO and KISS-Matcher+PGO. (b) Euler steps κ: performance saturates at κ=10. (c) Keypoint density α_s: performance saturates at α_s=0.2.

success rate, while the baselines drop below 40%, demonstrating the superior robustness of our method under low-overlap conditions. The right plot shows that the graph success rate of RAP remains stable and even improves with more views, while the baselines deteriorate more steeply. This highlights a key weakness of two-stage pipelines: erroneous pairwise registrations propagate as outliers into the pose graph, which PGO cannot reliably resolve. In contrast, RAP performs an implicit global adjustment at the point level, transporting all keypoints across all views simultaneously toward a globally consistent configuration. Additional views therefore provide complementary geometric evidence rather than additional sources of pairwise error. Additional multi-view results on 3DMatch and ScanNet (Supp. Tab. 4) confirm RAP outperforms SGHR [84], LMVR [31], and pairwise+PGO baselines.

4.4 Ablation Studies

We evaluate our model under different inference and architectural configurations on 3DMatch and 3DLoMatch for pairwise registration, and the cross-domain multi-view registration benchmark for multi-view registration. The results are summarized in Table 3.

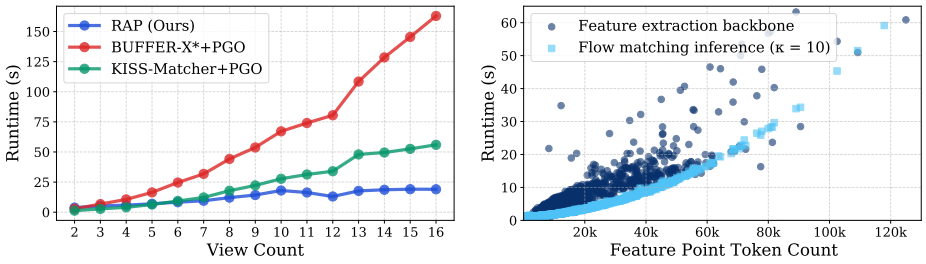


Fig. 6: Runtime analysis on the cross-domain benchmark. Left: Mean runtime per sample vs. view count for RAP, BUFFER-X*+PGO, and KISS-Matcher+PGO. Right: RAP runtime vs. total feature point token count, split into feature extraction and flow-matching inference. Runtimes on a single A5000 GPU.

Under the default setting [A] with test-time rigidity enforcing and κ=10 Euler steps, our model achieves the highest success rates across all datasets. Setting [B] removes rigidity enforcing and setting [C] reduces to 1-step generation. Both degrade performance, with [B] showing that rigidity enforcement during sampling is crucial and [C] confirming an accuracy–speed trade-off. Note that even with one-step generation, our model outperforms all baselines on the benchmark.

Settings [D]–[F] study the impact of model design. Setting [D] removes local descriptors, causing a sharp performance drop especially on the benchmark, highlighting the importance of local geometric features. Settings [E] and [F] evaluate smaller models with L decreasing from 10 to 8 and 6, respectively. As expected, reducing model capacity induces a clear performance drop.

Data scaling. Fig. 5(a) reports edge success rate on our cross-domain multi-view registration benchmark for RAP trained on $\{1, 3, 17\}$ dataset mixes versus BUFFER-X and KISS-Matcher. Here 1 dataset denotes 3DMatch only, 3 denotes 3DMatch + KITTI + ModelNet, and 17 denotes our full mix. With 1 and 3 datasets, RAP underperforms baselines because the generative formulation has weaker geometric inductive biases and no RANSAC, so it relies on data diversity. However, RAP's gain with more data is far steeper than that of BUFFER-X. This confirms that the approach-data combination unlocks cross-domain generalization that prior methods cannot reach by scaling data alone.

Hyperparameter sensitivity. Fig. 5(b–c) show that performance saturates at κ=10 Euler steps (85.9% vs. 86.8% at κ=50 with 5× runtime) and at keypoint sampling density α_s=0.2 (85.9% vs. 87.1% at α_s=1.0 with much higher memory), indicating favorable accuracy–efficiency trade-offs.

4.5 Runtime

We further evaluate the inference time of our model on a single NVIDIA A5000 GPU with 24 GB VRAM (Fig. 6). At two views, RAP is slightly slower than KISS-Matcher and BUFFER-X* due to the flow-matching inference cost. As the number of views grows, however, the baselines scale poorly because the number of pairwise registrations grows quadratically with the view count, followed by

pose graph optimization, while RAP's runtime remains nearly linear since it processes all views jointly in a single forward pass. Feature extraction scales linearly with the total number of tokens, while flow-matching inference grows more steeply due to quadratic attention over the token sequence. In practice, feature extraction dominates at lower token counts, and flow-matching inference becomes the bottleneck only at higher token counts.

5 Conclusion

We presented RAP, a generative flow-matching model for single-stage multi-view point cloud registration. By casting registration as conditional point flow generation with an alternating-attention transformer, RAP achieves holistic multi-view reasoning without pairwise registration or pose graph optimization, yielding strong robustness under low overlap. Trained on over 100k instances from 17 diverse datasets and evaluated zero-shot on our proposed cross-domain benchmark, RAP achieves state-of-the-art performance across object, indoor, outdoor, TLS, and map domains. A rigidity-enforcing sampler provides further test-time gains. We hope RAP serves as a step toward a universal registration foundation model for SLAM, 3D reconstruction, and robotic manipulation.

Limitations and future work. Several directions remain open. First, because RAP models flow in Euclidean space rather than on the transformation group, it could handle non-rigid deformations from dynamic objects, temporal change, or reconstruction drift, given suitable training data. This would enable temporal alignment of dynamic point clouds [38] and merging point maps from photogrammetry [76] and feed-forward 3D reconstruction [87, 89, 90] that lack a fixed metric scale. Second, like other feed-forward reconstruction methods [87, 90], our approach has no mechanism to detect whether input scans originate from the same environment and will always attempt to register all inputs jointly. Devising a mechanism to detect and reject out-of-scene inputs would improve robustness. Third, scaling RAP to longer sequences of streaming data is an open challenge. Drawing on recent advances in hierarchical and incremental feed-forward 3D reconstruction [22, 60, 88], a natural direction is to integrate RAP into a hierarchical SLAM system that progressively fuses local submaps, enabling generative registration at the scale of full mapping sessions.

Acknowledgements

This work has partially been funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy, EXC-2070-390732324-PhenoRob, and by the German Federal Ministry of Research, Technology and Space (BMFTR) under the Robotics Institute Germany (RIG). Tao Sun is funded by the Stanford Graduate Fellowship. Liyuan Zhu and Iro Armeni are partly supported by the NVIDIA Academic Grant. The authors thank the University of Bonn and the Lamarr Institute for providing the computational resources of the Marvin cluster and the Lamarr clusters.

References

1. An, L., Zhou, P., Zhou, M., Wang, Y., Geng, G.: Diffusion Transformer for point cloud registration: digital modeling of cultural heritage. Heritage Science **12**(1), 198 (2024),
2. An, L., Zhou, P., Zhou, M., Wang, Y., Zhang, Q.: PointTr: Low-overlap point cloud registration with transformer. IEEE Sensors Journal **24**(8), 12795–12805 (2024)
3. Ao, S., Hu, Q., Wang, H., Xu, K., Guo, Y.: BUFFER: Balancing Accuracy, Efficiency, and Generalizability in Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2023),
4. Ao, S., Hu, Q., Yang, B., Markham, A., Guo, Y.: SpinNet: Learning a General Surface Descriptor for 3D Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2021),
5. Aoki, Y., Goforth, H., Arun Srivatsan, R., Lucey, S.: PointNetLK: Robust and Efficient Point Cloud Registration Using PointNet. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2019),
6. Arun, K., Huang, T., Blostein, S.: Least-Squares Fitting of Two 3-D Point Sets. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI) **9**(5), 698–700 (1987),
7. Babin, P., Giguere, P., Pomerleau, F.: Analysis of robust functions for registration algorithms. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2019)
8. Bai, X., Luo, Z., Zhou, L., Chen, H., Li, L., Hu, Z., Fu, H., Tai, C.L.: PointDSC: Robust Point Cloud Registration using Deep Spatial Consistency. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2021),
9. Bai, X., Luo, Z., Zhou, L., Fu, H., Quan, L., Tai, C.L.: D3Feat: Joint Learning of Dense Detection and Description of 3D Local Features. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2020),
10. Bernreiter, L., Ott, L., Nieto, J.I., Siegwart, R., Cadena, C.: PHASER: a Robust and Correspondence-Free Global Pointcloud Registration. IEEE Robotics and Automation Letters (RA-L) **6**(2), 855–862 (2021),
11. Besl, P., McKay, N.: A Method for Registration of 3D Shapes. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI) **14**(2), 239–256 (1992),
12. Bian, H., Kong, L., Xie, H., Pan, L., Qiao, Y., Liu, Z.: DynamicCity: Large-Scale 4D Occupancy Generation from Dynamic Scenes. In: Proc. of the Intl. Conf. on Learning Representations (ICLR) (2025),
13. Brizi, L., Giacomini, E., Giammarino, L.D., Ferrari, S., Salem, O., Rebotti, L.D., Grisetti, G.: VBR: A Vision Benchmark in Rome. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2024),
14. Burnett, K., Yoon, D.J., Wu, Y., Li, A.Z., Zhang, H., Lu, S., Qian, J., Tseng, W.K., Lambert, A., Leung, K.Y.K., Schoellig, A.P., Barfoot, T.D.: Boreas: A Multi-Season Autonomous Driving Dataset. Intl. Journal of Robotics Research (IJRR) **42**(1-2), 33–42 (2023),
15. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuScenes: A Multimodal Dataset for Autonomous Driving. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2020)
16. Choi, S., Zhou, Q., Koltun, V.: Robust Reconstruction of Indoor Scenes. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2015),

17. Choy, C., Dong, W., Koltun, V.: Deep Global Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2020),
18. Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2019),
19. Dai, A., Chang, A., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2017),
20. Dellaert, F.: Factor graphs and GTSAM: A hands-on introduction. Tech. rep., Georgia Institute of Technology (2012),
21. Deng, H., Birdal, T., Ilic, S.: 3d local features for direct pairwise registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2019),
22. Deng, K., Ti, Z., Xu, J., Yang, J., Xie, J.: VGGT-Long: Chunk it, Loop it, Align it – Pushing VGGT's Limits on Kilometer-scale Long RGB Sequences. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2026),
23. Deschaud, J.E.: KITTI-CARLA: a KITTI-like dataset generated by CARLA Simulator. arXiv preprint **arXiv:2109.00892** (2021),
24. Dong, Z., Liang, F., Yang, B., Xu, Y., Zang, Y., Li, J., Wang, Y., Dai, W., Fan, H., Hyypä, J., et al.: Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **163**, 327–342 (2020),
25. Dong, Z., Yang, B., Liang, F., Huang, R., Scherer, S.: Hierarchical registration of unordered tls point clouds based on binary shape context descriptor. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **144**, 61–79 (2018),
26. Dong, Z., Yang, B., Liu, Y., Liang, F., Li, B., Zang, Y.: A novel binary shape context for 3d local surface description. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **130**, 431–452 (2017),
27. Du, Y., Zhao, Z., Su, S., Golluri, S., Zheng, H., Yao, R., Wang, C.: SuperPC: A single diffusion model for point cloud completion, upsampling, denoising, and colorization. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2025),
28. Fischler, M., Bolles, R.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM **24**(6), 381–395 (1981),
29. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2012),
30. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets Robotics: The KITTI Dataset. Intl. Journal of Robotics Research (IJRR) **32**(11), 1231–1237 (2013),
31. Gojcic, Z., Zhou, C., Wegner, J.D., Guibas, L.J., Birdal, T.: Learning Multiview 3D Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2020),
32. Gojcic, Z., Zhou, C., Wegner, J.D., Wieser, A.: The Perfect Match: 3D Point Cloud Matching with Smoothed Densities. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2019),
33. Grisetti, G., Kümmerle, R., Stachniss, C., Burgard, W.: A tutorial on graph-based SLAM. IEEE Trans. on Intelligent Transportation Systems Magazine **2**(4), 31–43 (2010)
34. Guo, X., Wu, Z., Xiong, K., Xu, Z., Zhou, L., Xu, G., Xu, S., Sun, H., Wang, B., Chen, G., et al.: Genesis: Multimodal Driving Scene Generation with Spatio-

Temporal and Cross-Modal Consistency. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2025),
35. He, L., Wang, X., Zhang, H.: M2DP: A Novel 3D Point Cloud Descriptor and Its Application in Loop Closure Detection. In: Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) (2016)
36. Hsu, L.T., Kubo, N., Wen, W., Chen, W., Liu, Z., Suzuki, T., Meguro, J.: UrbanNav: An Open-Sourced Multisensory Dataset for Benchmarking Positioning Algorithms Designed for Urban Areas. Navigation **70**(4), 226–256 (2023),
37. Huang, H., Sun, Y., Wu, J., Jiao, J., Hu, X., Zheng, L., Wang, L., Liu, M.: On bundle adjustment for multiview point cloud registration. IEEE Robotics and Automation Letters (RA-L) **6**(4), 8269–8276 (2021),
38. Huang, S., Gojcic, Z., Huang, J., Wieser, A., Schindler, K.: Dynamic 3D Scene Analysis by Point Cloud Accumulation. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2022),
39. Huang, S., Gojcic, Z., Usvyatsov, M., Wieser, A., Schindler, K.: PREDATOR: Registration of 3D Point Clouds with Low Overlap. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2021),
40. Huang, X., Cheng, X., Geng, Q., Cao, B., Zhou, D., Wang, P., Lin, Y., Yang, R.: The ApolloScape Dataset for Autonomous Driving. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (2018),
41. Jiang, H., Salzmann, M., Dang, Z., Xie, J., Yang, J.: SE(3) Diffusion Model-based Point Cloud Registration for Robust 6D Object Pose Estimation. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2023),
42. Jiang, H., Xie, J., Yang, J., Yu, L., Zheng, J.: FUSER: Feed-Forward Multiview 3D Registration Transformer and $SE(3)^N$ Diffusion Refinement. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2026),
43. Jin, S., Armeni, I., Pollefeys, M., Barath, D.: Multiway Point Cloud Mosaicking with Diffusion and Global Optimization. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
44. Jung, M., Yang, W., Lee, D., Gil, H., Kim, G., Kim, A.: HeLiPR: Heterogeneous LiDAR Dataset for inter-LiDAR Place Recognition under Spatiotemporal Variations. Intl. Journal of Robotics Research (IJRR) **12**(43), 1867—-1883 (2024),
45. Kim, G., Park, Y., Cho, Y., Jeong, J., Kim, A.: Mulran: Multimodal range dataset for urban place recognition. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2020),
46. Knights, J., Vidanapathirana, K., Ramezani, M., Sridharan, S., Fookes, C., Moghadam, P.: Wild-Places: A Large-Scale Dataset for Lidar Place Recognition in Unstructured Natural Environments. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2023),
47. Laserna, J., San Segundo, P., Álvarez, D.: CliReg: Clique-Based Robust Point Cloud Registration. IEEE Trans. on Robotics (TRO) **41**, 1898–1917 (2025)
48. Lee, D., Hamsici, O.C., Feng, S., Sharma, P., Gernoth, T.: DeepPRO: Deep Partial Point Cloud Registration of Objects. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2021),
49. Leroy, V., Cabon, Y., Revaud, J.: Grounding Image Matching in 3D with MASt3R. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2024),
50. Li, S., Jiang, Z., Chen, G., Xu, C., Tan, S., Wang, X., Fang, I., Zyskowski, K., McPherron, S.P., Iovita, R., Feng, C., Zhang, J.: GARF: Learning Generalizable 3D Reassembly for Real-World Fractures. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2025),

51. Li, Y., Harada, T.: Lepard: Learning Partial Point Cloud Matching in Rigid and Deformable Scenes. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2022),
52. Liao, Y., Xie, J., Geiger, A.: KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI) **45**(3), 3292–3310 (2022),
53. Lim, H., Kim, B., Kim, D., Lee, E.M., Myung, H.: Quatro++: Robust global registration exploiting ground segmentation for loop closing in lidar slam. Intl. Journal of Robotics Research (IJRR) **43**(5), 685–715 (2024),
54. Lim, H., Kim, D., Shin, G., Shi, J., Vizzo, I., Myung, H., Park, J., Carlone, L.: KISS-Matcher: Fast and Robust Point Cloud Registration Revisited. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2025),
55. Lipman, Y., Chen, R.T.Q., Ben-Hamu, H., Nickel, M., Lehtinen, M.: Flow Matching for Generative Modeling. In: Proc. of the Intl. Conf. on Learning Representations (ICLR) (2023),
56. Liu, J., Su, J., Yao, X., Jiang, Z., Lai, G., Du, Y., Qin, Y., Xu, W., Lu, E., Yan, J., Chen, Y., Zheng, H., Liu, Y., Liu, S., Yin, B., He, W., Zhu, H., Wang, Y., Wang, J., Dong, M., Zhang, Z., Kang, Y., Zhang, H., Xu, X., Zhang, Y., Wu, Y., Zhou, X., Yang, Z.: Muon is Scalable for LLM Training. arXiv preprint **arXiv:2502.16982** (2025),
57. Liu, Q., Zhu, H., Wang, Z., Zhou, Y., Chang, S., Guo, M.: Extend Your Own Correspondences: Unsupervised Distant Point Cloud Registration by Progressive Distance Extension. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
58. Liu, X., Gong, C., Liu, Q.: Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In: Proc. of the Intl. Conf. on Learning Representations (ICLR) (2023),
59. Lu, W., Wan, G., Zhou, Y., Fu, X., Yuan, P., Song, S.: DeepVCP: An End-to-End Deep Neural Network for Point Cloud Registration. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2019),
60. Maggio, D., Lim, H., Carlone, L.: VGGT-SLAM: Dense RGB SLAM Optimized on the SL(4) Manifold. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2025),
61. Mellado, N., Aiger, D., Mitra, N.J.: Super 4PCS: Fast Global Pointcloud Registration via Smart Indexing. Computer Graphics Forum **33**(5), 205–215 (2014),
62. Mildenhall, B., Srinivasan, P., Tancik, M., Barron, J., Ramamoorthi, R., Ng, R.: NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2020),
63. Nunes, L., Marcuzzi, R., Mersch, B., Behley, J., Stachniss, C.: Scaling Diffusion Models to Real-World 3D LiDAR Scene Completion. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
64. Nunes, L., Marcuzzi, R., Behley, J., Stachniss, C.: Towards Generating Realistic 3D Semantic Training Data for Autonomous Driving. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI) pp. 1–12 (2026),
65. Pan, Y., Xiao, P., He, Y., Shao, Z., Li, Z.: MULLS: Versatile LiDAR SLAM Via Multi-Metric Linear Least Square. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2021)
66. Peebles, W., Xie, S.: Scalable Diffusion Models with Transformers. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2023),

67. Poiesi, F., Boscaini, D.: Learning General and Distinctive 3D Local Deep Descriptors for Point Cloud Registration. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI) **45**(3), 3979–3985 (2023),
68. Pomerleau, F., Colas, F., Siegwart, R.: A Review of Point Cloud Registration Algorithms for Mobile Robotics. Foundations and Trends in Robotics **4**, 1–104 (2015)
69. Pomerleau, F., Liu, M., Colas, F., Siegwart, R.: Challenging Data Sets for Point Cloud Registration Algorithms. Intl. Journal of Robotics Research (IJRR) **31**(14), 1705–1711 (2012),
70. Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Xu, K.: Geometric Transformer for Fast and Robust Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2022),
71. Ren, X., Huang, J., Zeng, X., Museth, K., Fidler, S., Williams, F.: XCube: Large-Scale 3D Generative Modeling using Sparse Voxel Hierarchies. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
72. Rusu, R., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2009),
73. Salti, S., Tombari, F., Stefano, L.: SHOT: Unique Signatures of Histograms for Surface and Texture Description. Journal of Computer Vision and Image Understanding (CVIU) **125**, 251–264 (2014),
74. Sanghi, A., Fu, R., Liu, V., Willis, K.D., Shayani, H., Khasahmadi, A.H., Sridhar, S., Ritchie, D.: CLIP-Sculptor: Zero-Shot Generation of High-Fidelity and Diverse Shapes From Natural Language. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2023),
75. Schneider, J., Schindler, F., Läbe, T., Förstner, W.: Bundle adjustment for multi-camera systems with points at infinity. In: ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences (2012),
76. Schönberger, J.L., Frahm, J.M.: Structure-from-Motion Revisited. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2016),
77. Seo, M., Lim, H., Lee, K., Carlone, L., Park, J.: BUFFER-X: Towards Zero-Shot Point Cloud Registration in Diverse Scenes. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2025),
78. Shi, C., Chen, X., Huang, K., Xiao, J., Lu, H., Stachniss, C.: Keypoint Matching for Point Cloud Registration using Multiplex Dynamic Graph Attention Networks. IEEE Robotics and Automation Letters (RA-L) **6**(4), 8221–8228 (2021),
79. Sun, T., Hao, Y., Huang, S., Savarese, S., Schindler, K., Pollefeys, M., Armeni, I.: Nothing Stands Still: A Spatiotemporal Benchmark on 3D Point Cloud Registration Under Large Geometric and Temporal Change. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **220**, 799–823 (2025),
80. Sun, T., Zhu, L., Huang, S., Song, S., Armeni, I.: Rectified Point Flow: Generic Point Cloud Pose Estimation. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2025),
81. Tao, Y., Ángel Muñoz-Bañón, M., Zhang, L., Wang, J., Fu, L.F.T., Fallon, M.: The Oxford Spires Dataset: Benchmarking Large-Scale LiDAR-Visual Localisation, Reconstruction and Radiance Field Methods. International Journal of Robotics Research **44**(1), 1–14 (2025),
82. Theiler, P.W., Wegner, J.D., Schindler, K.: Globally Consistent Registration of Terrestrial Laser Scans via Graph Optimization. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **109**, 126–138 (2015),

83. Tombari, F., Salti, S., Di Stefano, L.: Unique Signatures of Histograms for Local Surface Description. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2010),
84. Wang, H., Liu, Y., Dong, Z., Guo, Y., Liu, Y.S., Wang, W., Yang, B.: Robust Multiview Point Cloud Registration with Reliable Pose Graph Initialization and History Reweighting. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2023),
85. Wang, H., Liu, Y., Dong, Z., Wang, W.: You Only Hypothesize Once: Point Cloud Registration with Rotation-equivariant Descriptors. In: Proc. of the ACM Intl. Conf. on Multimedia (2022),
86. Wang, H., Liu, Y., Hu, Q., Wang, B., Chen, J., Dong, Z., Guo, Y., Wang, W., Yang, B.: RoReg: Pairwise Point Cloud Registration With Oriented Descriptors and Local Rotations. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI) **45**(8), 10376–10393 (2023),
87. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: VGGT: Visual Geometry Grounded Transformer. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2025),
88. Wang, Q., Zhang, Y., Holynski, A., Efros, A.A., Kanazawa, A.: CUT3R: Continuous 3D Perception Model with Persistent State. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2025),
89. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: DUSt3R: Geometric 3D Vision Made Easy. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
90. Wang, Y., Zhou, J., Zhu, H., Chang, W., Zhou, Y., Li, Z., Chen, J., Pang, J., Shen, C., He, T.: π^3: Scalable Permutation-Equivariant Visual Geometry Learning. In: Proc. of the Intl. Conf. on Learning Representations (ICLR) (2026),
91. Wang, Y., Solomon, J.: Deep Closest Point: Learning Representations for Point Cloud Registration. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2019),
92. Wang, Y., Solomon, J.M.: PRNet: Self-Supervised Learning for Partial-to-Partial Registration. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2019),
93. Wang, Z., Chen, J., Furukawa, Y.: PuzzleFusion++: Auto-Agglomerative 3D Fracture Assembly by Denoise and Verify. In: Proc. of the Intl. Conf. on Learning Representations (ICLR) (2025),
94. Wiesmann, L., Guadagnino, T., Vizzo, I., Grisetti, G., Behley, J., Stachniss, C.: DCPCR: Deep Compressed Point Cloud Registration in Large-Scale Outdoor Environments. IEEE Robotics and Automation Letters (RA-L) **7**(3), 6327–6334 (2022)
95. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2015),
96. Xu, J., Wang, X., Cheng, W., Cao, Y.P., Shan, Y., Qie, X., Gao, S.: Dream3D: Zero-Shot Text-to-3D Synthesis Using 3D Shape Prior and Text-to-Image Diffusion Models. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2023),
97. Yang, H., Shi, J., Carlone, L.: TEASER: Fast and Certifiable Point Cloud Registration. IEEE Trans. on Robotics (TRO) **37**(2), 314–333 (2020),
98. Yao, R., Du, S., Cui, W., Tang, C., Yang, C.: PARE-Net: Position-Aware Rotation-Equivariant Networks for Robust Point Cloud Registration. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2024),

99. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: ScanNet++: A High-Fidelity Dataset of 3D Indoor Scenes. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2023),
100. Yew, Z.J., Lee, G.H.: RPM-Net: Robust Point Matching using Learned Features. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2020),
101. Yu, H., Li, F., Saleh, M., Busam, B., Ilic, S.: CoFiNet: Reliable Coarse-to-Fine Correspondences for Robust Point Cloud Registration. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2021),
102. Yu, H., Qin, Z., Hou, J., Saleh, M., Li, D., Busam, B., Ilic, S.: Rotation-Invariant Transformer for Point Cloud Matching. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2023),
103. Yu, J., Ren, L., Zhang, Y., Zhou, W., Lin, L., Dai, G.: PEAL: Prior-Embedded Explicit Attention Learning for Low-Overlap Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2023),
104. Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T.: 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2017),
105. Zeng, X., Chen, X., Qi, Z., Liu, W., Zhao, Z., Wang, Z., Fu, B., Liu, Y., Yu, G.: Paint3D: Paint Anything 3D with Lighting-Less Texture Diffusion Models. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
106. Zhang, S., Zhao, A., Yang, L., Li, Z., Meng, C., Xu, H., Chen, T., Wei, A., Gu, P.P., Sun, L.: Distilling Diffusion Models to Efficient 3D LiDAR Scene Completion. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2025),
107. Zhang, Y., Shi, P., Li, J.: 3D Lidar SLAM: A Survey. The Photogrammetric Record **39**(186), 457–517 (2024)
108. Zhou, Q., Park, J., Koltun, V.: Fast Global Registration. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2016),

Register Any Point: Scaling 3D Point Cloud Registration by Flow Matching (Supplementary Material)

Yue Pan[1] Tao Sun[2] Liyuan Zhu[2] Lucas Nunes[3]
Iro Armeni[2] Jens Behley[1,4] Cyrill Stachniss[1,4]

[1] Center for Robotics, University of Bonn, Germany
[2] Stanford University, USA
[3] RWTH Aachen University, Germany
[4] Lamarr Institute for Machine Learning and Artificial Intelligence, Germany

Overview

In the supplementary material, we provide the following:

- Implementation details including point cloud sampling and feature extraction, flow model architecture and training, training data curation, pairwise registration testing data, and evaluation metrics (Section A)
- Details on the selected baseline methods used in our experiments (Section B)
- Additional experimental results including pairwise registration with low overlap, indoor multi-view registration on 3DMatch and ScanNet, cross-domain multi-view registration benchmark, and offline SLAM pose estimation (Section C)
- Additional information of the cross-domain multi-view registration benchmark (Section D)
- Additional qualitative results and failure cases (Section E)

A Implementation Details

A.1 Point Cloud Sampling and Feature Extraction

For each point cloud $\mathbf{P}_i$, we first apply voxel downsampling with voxel size v_d to obtain $\mathbf{P}_i^v$. We then apply statistical outlier removal to $\mathbf{P}_i^v$ to remove outliers. To ensure uniform sampling density across input point clouds, we determine the sample count K_i proportionally to the spatial voxel coverage: we voxelize $\mathbf{P}_i^v$ with voxel size v_c and let V_i be the number of remaining points, then set $K_i = \lfloor \alpha_s V_i \rfloor$, where α_s is a hyperparameter controlling the FPS sampling density. We apply farthest point sampling (FPS) on $\mathbf{P}_i^v$ to sample K_i points as the feature points $\mathbf{Q}_i = \{\mathbf{q}_{i,k}\}_{k=1}^{K_i} \in \mathbb{R}^{3\times K_i}$. Typically, we obtain $K_i \in [200, 5000]$ for our training data. For each feature point $\mathbf{q}_{i,k} \in \mathbf{Q}_i$, we extract a local patch by a ball query of radius $r_s = 20\, v_d$ in $\mathbf{P}_i^v$ and normalize the patch points. We then use the lightweight MiniSpinNet [2, 59] pretrained on 3DMatch [84] as our feature extractor f_{desc} to compute a descriptor for each patch with maximum 512 points

Algorithm 1: Point cloud sampling and miniSpinNet feature extraction

Input: Set of input point clouds $\{\mathbf{P}_i\}$;
voxel sizes v_d (downsampling) and v_c (coverage);
FPS sampling ratio α_s; patch radius r_s;
miniSpinNet feature extractor f_{desc}.
Output: Sampled points $\{\mathbf{Q}_i\}$ and features $\{\mathbf{F}_i\}$.

foreach *input point cloud* $\mathbf{P}_i$ **do**
- Voxel-downsample $\mathbf{P}_i$ with voxel size v_d to obtain $\mathbf{P}_i^v$;
- Apply statistical outlier removal to $\mathbf{P}_i^v$ to remove outliers;
- Voxelize $\mathbf{P}_i^v$ with voxel size v_c and let V_i be the number of remaining points;
- Set $K_i \leftarrow \lfloor \alpha_s V_i \rfloor$;
- Apply FPS on $\mathbf{P}_i^v$ to sample K_i points as feature points $\mathbf{Q}_i = \{\mathbf{q}_{i,k}\}_{k=1}^{K_i}$;
- For each $\mathbf{q}_{i,k} \in \mathbf{Q}_i$, extract a local patch by a ball query of radius r_s in $\mathbf{P}_i^v$ and normalize the patch points;
- Use f_{desc} (miniSpinNet) to compute a descriptor for each patch and stack them into $\mathbf{F}_i \in \mathbb{R}^{D \times K_i}$;

return $\{\mathbf{Q}_i\}$ and $\{\mathbf{F}_i\}$;

and stack them into local features $\mathbf{F}_i \in \mathbb{R}^{D \times K_i}$, where $D = 32$. During inference, there are only two tunable hyperparameters: the sampling density ratio α_s and the downsampling voxel size v_d. For ease of use, we set $\alpha_s = 0.2$ by default and determine v_d adaptively based on the scene scale, requiring no scene-specific hyperparameter tuning.

The pseudocode for this step is shown in Algorithm 1.

To improve efficiency and reduce memory usage, we adopt the lightweight patch-wise network MiniSpinNet from BUFFER [2] as our local feature descriptor. MiniSpinNet is a compact SpinNet-style [3] network that encodes each input patch into a 32-dimensional feature descriptor by decreasing the voxelization hyperparameters and simplifying the 3D cylindrical convolution (3DCC) layers, making it nearly nine times faster than the vanilla SpinNet.

Currently, we use MiniSpinNet [2] as our feature extractor. However, our framework is compatible with other feature extractors for encoding local geometry, including task-specific ones trained for registration (e.g., FCGF [14] and YOHO [70]) as well as more general self-supervised backbones based on point transformers [76] (e.g., Sonata [75] and Utonia [85]). We leave the exploration of other feature extractors as our future work.

A.2 Flow Matching Model Architecture

Following RPF [64], we use a diffusion transformer [51] with alternating-attention blocks [71] for conditional flow matching.

Our transformer architecture comprises $L = 10$ alternating-attention blocks with hidden dimension $d = 512$ and $h = 8$ attention heads, totaling 73 million

parameters. The alternating-attention mechanism alternates between two types of attention layers: (i) per-view self-attention that operates within each point cloud to consolidate view-specific structure, and (ii) global attention over all point tokens across all views to fuse information and enable cross-view reasoning. This design allows the model to simultaneously capture local geometric structure within each view and global relationships across multiple views.

The model's input consists of the noisy point cloud $\mathbf{X}(t)$ at time step t and a conditioning signal $\mathbf{C} = f_{\text{emb}}(\bar{\mathcal{Q}}, \mathcal{F})$ obtained via a linear feature embedder f_{emb}. The conditioning $\mathbf{C}$ concatenates two components: (i) local geometric descriptors $\mathcal{F}$ extracted by MiniSpinNet for each sampled keypoint, (ii) positional encodings of the normalized point coordinates $\bar{\mathcal{Q}}$ using a multi-frequency Fourier feature mapping [44]. The flow matching network $\mathbf{V}_\theta$ takes $\mathbf{X}(t)$ and $\mathbf{C}$ as input and predicts the velocity field $\nabla_t \mathbf{X}(t)$ that transports the noisy points toward the target registered configuration.

Unlike RPF [64], we do not take the point-wise normals and the scalar view index as additional conditioning signals. We also do not rely on a PTv3-based encoder [76] pretrained for the overlapping prediction task. These design choices make our model simpler and can be applied to various cross-domain training datasets.

A.3 Flow Matching Model Training

We train our model using the Muon Optimizer [39], with an initial learning rate of $2 \cdot 10^{-3}$ for matrix-like parameters and $2 \cdot 10^{-4}$ for vector-like parameters. From our experiments we find that using Muon instead of AdamW [32] can achieve faster convergence and better performance. For the learning rate schedule, we halve the learning rate after 200, 240, 280, 320, 360, 400, and 500 epochs. During the flow matching model training, we sample the time steps from a U-shaped distribution [34].

We train the model for three days with about 120k iterations using 32 NVIDIA A100 GPUs with 80 GB VRAM each. The training data consists of registration instances with view counts ranging from 2 to 16.

A.4 Training Data Curation

We curate both the scan-level and submap-level training samples using the same script with different settings.

For each dataset, given the per-frame poses, we first select keyframes based on temporal and spatial thresholds, which removes redundant frames when the sensor is stationary or moving slowly. For datasets lacking accurate and globally consistent reference poses, we use a state-of-the-art SLAM system [49] to estimate the poses for data curation. For most LiDAR-based datasets (e.g. KITTI is already deskewed), we additionally apply deskewing (motion undistortion) to the keyframe point clouds when point-wise timestamps are available. For each sequence with M keyframes, we aim to generate $N_{\text{target}} = \beta M$ training samples, where β controls the number of samples per keyframe. For every sample, we

randomly select $N \in [N_{\min}, N_{\max}]$ point clouds. Each point cloud is constructed by accumulating points from $F \in [F_{\min}, F_{\max}]$ consecutive keyframes, and the resulting point clouds do not share frames. We then transform the point clouds to the world frame using the corresponding keyframe poses. For each sample, we allow at most $T_{\max}$ attempts to find a valid configuration. A sample is considered valid only if (i) all point clouds are spatially close to each other, i.e. the pairwise distances between their centers are below a threshold $d_{\max}$, and (ii) the point clouds are not isolated from each other, i.e. they form a connected graph under a minimum overlap-ratio threshold $\epsilon_{\text{overlap}}$. The overlap ratio between two point clouds is computed as the ratio of the number of occupied voxels in their intersection to the number of occupied voxels in their union, evaluated on a voxel grid with an adaptively set voxel size v_{overlap}. We set a very small overlapping ratio threshold $\epsilon_{\text{overlap}}$ (0.5%-2%) to add some hard samples that allow the model to learn to register low-overlapped point clouds. Whenever we find a valid set of point clouds in an attempt, we save it as a training sample. We set $N_{\min} = 2$ and $F_{\min} = 1$ for all datasets. For scan-based samples, we set $F_{\max} = 1$, and for submap-based samples, we use $F_{\max} > 1$.

The pseudocode for generating the training samples is shown in Algorithm 2.

The details of the training datasets and used parameters for data curation ($N_{\max}$, $F_{\max}$, $d_{\max}$, $\epsilon_{\text{overlap}}$) and point cloud preprocessing (α_s, v_d) are shown in Tab. 1. In total, we curate 141k samples containing 520k point clouds, resulting in more than 10 billion points. The data covers a diverse range of scenes across 9 countries on 4 continents, captured by 9 types of LiDAR and 6 types of depth cameras with varying resolutions and scales.

We split the data into training and validation sets. To ensure fair evaluation, we exclude sequences used for testing in commonly used benchmarks from the training set and designate them as validation sequences. For example, sequences 08–10 from the KITTI dataset are excluded from training. For some datasets not used in the testing benchmark (such as Nuscenes, Boreas, and WildPlace), we use all the sequences for training but keep 10% randomly selected samples for the in-sequence validation.

A.5 Pairwise Registration Testing Data Details

The details of the six adopted testing datasets for pairwise registration evaluation and the evaluation success criteria are shown in Tab. 2.

A.6 Evaluation Metrics

We evaluate pairwise registration performance using the registration success rate (%), computed with thresholds on correspondence RMSE for 3DMatch and on translation and rotation errors for all other datasets. The exact thresholds are summarized in Tab. 2 and follow the settings of prior works [59, 63, 84].

Given the ground-truth transformation $\mathbf{T}_{\text{gt}} = [\mathbf{R}_{\text{gt}} \mid \mathbf{t}_{\text{gt}}]$ and the estimated transformation $\mathbf{T}_{\text{est}} = [\mathbf{R}_{\text{est}} \mid \mathbf{t}_{\text{est}}]$, the translation error (TE) and rotation

Algorithm 2: Generate training samples from a sequence

Input: Per-frame poses $\{\mathbf{T}_i\}$, point cloud scans $\{\mathbf{S}_i\}$;
Keyframe thresholds $(\tau_{\text{time}}, \tau_{\text{space}})$;
Sampling parameters: β, $T_{\max}$, $N_{\min}, N_{\max}$, $F_{\min}, F_{\max}$;
Spatial and overlap thresholds: $d_{\max}$, $\epsilon_{\text{overlap}}$, v_{overlap};
Output: Generated training samples for training.

Select keyframe indices $\mathcal{K} \leftarrow \textsc{SelectKeyframes}(\{\mathbf{T}_i\}, \tau_{\text{time}}, \tau_{\text{space}})$;
if *deskewing enabled* **then**
  foreach $k \in \mathcal{K}$ **do**
    Apply Deskew to $\mathbf{S}_k$ if pointwise timestamps are available;
Let $M \leftarrow |\mathcal{K}|$, $N_{\text{target}} \leftarrow \beta M$;
for $n = 1$ **to** N_{target} **do**
  for $t = 1$ **to** $T_{\max}$ **do**
    Sample $N \sim \mathcal{U}\{N_{\min}, N_{\max}\}$;
    Sample N disjoint keyframe intervals $\{I_j\}_{j=1}^{N}$ from $\mathcal{K}$ with lengths $F_j \sim \mathcal{U}\{F_{\min}, F_{\max}\}$;
    Initialize $\mathcal{M} \leftarrow [\,]$, $\mathcal{C} \leftarrow [\,]$;
    for $j = 1$ **to** N **do**
      Accumulate a point cloud $\mathbf{M}_j \leftarrow \textsc{AccumulateFrames}(\{\mathbf{S}_i\}_{i \in I_j})$;
      Transform $\mathbf{M}_j$ to the world frame using $\{\mathbf{T}_i\}_{i \in I_j}$;
      Compute point cloud center $\mathbf{c}_j$ from $\mathbf{M}_j$ and append to $\mathcal{C}$;
      Append $\mathbf{M}_j$ to $\mathcal{M}$;
    if $\exists (a, b)$ *such that* $\|\mathbf{c}_a - \mathbf{c}_b\|_2 > d_{\max}$ **then**
      continue to next attempt;
    Build a graph G over nodes $\{1, \ldots, N\}$ with edge (a, b) if $\textsc{OverlapRatio}(\mathbf{M}_a, \mathbf{M}_b, v_{\text{overlap}}) \geq \epsilon_{\text{overlap}}$;
    if G *is connected* **then**
      $\textsc{SaveTrainingSample}(\mathcal{M})$;
      break;

error (RE) are defined as:

$$\text{TE} = \|\mathbf{t}_{\text{gt}} - \mathbf{t}_{\text{est}}\|_2, \tag{1}$$

$$\text{RE} = \arccos\left(\frac{\text{tr}(\mathbf{R}_{\text{gt}}^{\top}\mathbf{R}_{\text{est}}) - 1}{2}\right) \cdot \frac{180}{\pi}, \tag{2}$$

where TE is in meters and RE is in degrees. For multi-view registration, we report the mean TE and RE across all scans in a sample.

For multi-view registration on the cross-domain benchmark, we adopt the *edge success rate* and *graph success rate* as the primary evaluation metrics. The edge success rate is the mean registration success rate over all valid edges (i.e., overlapping pairs) in a sample. The graph success rate is stricter: it requires

Table 1: Summary of the training datasets with parameter settings for data curation. Sampling parameters: $N_{\max}$, $F_{\max}$; Spatial and overlap thresholds: $d_{\max}$, $\epsilon_{\text{overlap}}$; Point cloud preprocessing parameters: α_s, v_d.

Dataset	Scenario	Sensor	# Scenes	Type	# Samples	# P. Clouds	Sampling Parameters		Spatial Thre.		Preprocess	
							$N_{\max}$	$F_{\max}$	$d_{\max}$	$\epsilon_{\text{overlap}}$	α_s	v_d
Outdoor LiDAR												
KITTI [21]	Germany; urban & highway	Velodyne-64	22	Scan	1,226	2,852	8	1	100.0	1%	0.2	0.25
				Submap	3,810	16,453	10	600	400.0	0.5%	0.05	0.25
KITTI360 [35]	Germany; urban	Velodyne-64	9	Scan	3,002	6,223	8	1	100.0	1%	0.2	0.25
				Submap	7,530	21,255	10	600	400.0	0.5%	0.05	0.25
Apollo [29]	USA; urban & highway	Velodyne-64	11	Scan	3,343	7,874	8	1	100.0	1%	0.2	0.25
				Submap	6,660	25,036	10	600	400.0	0.5%	0.05	0.25
MulRAN [31]	South Korea; urban & campus	Ouster-64	4	Scan	898	1900	8	1	100.0	2%	0.2	0.25
				Submap	1,388	4,477	10	600	400.0	0.5%	0.05	0.25
Oxford Spires [65]	UK; campus	Hesai-64	6	Scan	1,356	5,781	10	1	60.0	2%	0.2	0.25
				Submap	541	2,763	10	200	150.0	1%	0.1	0.25
VBR [5]	Italy; urban & campus	Ouster-64	5	Scan	3,371	8,647	8	1	80.0	1%	0.2	0.25
				Submap	1,906	8,511	10	500	300.0	0.5%	0.05	0.25
UrbanNav [25]	China; urban	Velodyne-32	4	Scan	1,912	4,228	8	1	100.0	1%	0.2	0.25
				Submap	979	3,540	10	600	400.0	0.5%	0.05	0.25
HeLiPR [30]	South Korea; urban	Ouster-128, Avia, Aeva	3	Scan	1,808	3,691	8	1	100.0	1%	0.2	0.25
				Submap	3,624	10,882	10	600	400.0	0.5%	0.05	0.25
Boreas [6]	Canada; urban	Velodyne-128	2	Scan	1,131	2,311	5	1	100.0	1%	0.2	0.25
				Submap	1,429	3,613	10	600	400.0	0.5%	0.05	0.25
WildPlace [33]	Australia; forest	Velodyne-16	2	Submap	1,167	2,613	5	600	300.0	1%	0.1	0.25
NuScenes [7]	USA & Singapore; urban	Velodyne-32	642	Scan	12,160	24,320	2	1	80.0	2%	0.5	0.25
KITTI-Carla [18]	Synthetic; urban	Simulated-64	7	Submap	1,733	7,729	10	600	400.0	0.5%	0.05	0.25
Indoor Depth Camera												
3DMatch [84]	USA; office & apartment	Kinect, RealSense, etc.	82	Scan	7,044	14,088	2	1	10.0	2%	1.0	0.02
				Submap	3,904	29,314	16	10	10.0	1%	0.5	0.02
ScanNet [17]	USA; office & apartment	Structure sensor	661	Submap	10,840	75,299	24	50	15.0	1%	0.2	0.02
ScanNet++ [82]	Germany; office & apartment	Faro, DSLR, iPhone	220	Scan	9,306	101,743	24	1	15.0	1%	0.5	0.02
NSS [63]	USA; office & construction site	Matterport camera	6	Scan	17,275	72,866	20	1	30.0	0.1%	0.2	0.05
Object-centric												
ModelNet-40 [77]	Synthetic; CAD	-	(12,308)	Scan	24,616	49,232	2	1	-	-	-	0.01
Training Set			**1,621**		**118,143**	**457,195**						
Total			**1,685**		**141,002**	**520,315**						

Table 2: Summary of the testing datasets used for pairwise registration evaluation.

Dataset	Scenario	Sensor	Type	# Samples	Scale [m]	Success Criteria
ModelNet [77]	Synthetic; object	CAD	Object	1,266	1	TE 0.1m, RE 5°
3DMatch [84]	USA; office & apartment	Kinect, RealSense, etc.	Scan	1,623	5	Pointwise RMSE 0.2m
3DLoMatch [27]	USA; office & apartment	Kinect, RealSense, etc.	Scan	1,781	5	Pointwise RMSE 0.2m
NSS [63]	USA; office & construction site	Matterport camera	Scan	1,125	10	TE 0.2m, RE 10°
ETH [52]	Switzerland; park	Hokuyo	Scan	713	100	TE 2m, RE 5°
KITTI [21]	Germany; urban & highway	Velodyne-64	Scan	555	160	TE 2m, RE 5°

every valid edge in the multi-view registration graph to be successful, thus measuring the overall consistency of the registration. To handle diverse scene scales, we normalize the translation error by dividing by the longest axis of the bounding box of the ground-truth registered point cloud. A registration is considered successful if the normalized translation error is below 2.5% and the rotation error is below 15°. We also report results under stricter thresholds (0.5%, 3°) in this supplementary material.

For the additional multi-view registration experiments in this supplementary material, we adopt the following metrics: Chamfer distance (CD), normalized global RMSE, and the empirical cumulative distribution function (ECDF) of error. The CD measures the bi-directional root-mean-squared distance between the registered point cloud and the ground-truth aggregated point cloud. Given the registered point cloud $\mathbf{P}_{\text{reg}}$ and the ground-truth point cloud $\mathbf{P}_{\text{gt}}$, the CD is computed as:

$$\text{CD} = \sqrt{\frac{1}{2}\left(\frac{1}{|\mathbf{P}_{\text{reg}}|}\sum_{\mathbf{p}\in\mathbf{P}_{\text{reg}}}\min_{\mathbf{q}\in\mathbf{P}_{\text{gt}}}\|\mathbf{p}-\mathbf{q}\|_2^2 + \frac{1}{|\mathbf{P}_{\text{gt}}|}\sum_{\mathbf{q}\in\mathbf{P}_{\text{gt}}}\min_{\mathbf{p}\in\mathbf{P}_{\text{reg}}}\|\mathbf{q}-\mathbf{p}\|_2^2\right)}. \quad (3)$$

The global RMSE is computed as the root-mean-squared error between the generated point cloud and the ground-truth registered point cloud using the known point-to-point correspondences, and is normalized by the longest axis of the ground-truth registered point cloud's bounding box to account for diverse scene scales.

The ECDF of error reports the fraction of samples whose error falls below a given threshold. Plotting the ECDF across thresholds summarizes the full error distribution and allows comparison of methods at different accuracy levels.

B Details on the Selected Baselines

In the main paper, we did not include detailed descriptions of the baseline methods due to space limitations. Here we provide comprehensive descriptions and used training datasets of all baselines used in our experiments in Tab. 3.

All pairwise registration baselines follow the standard correspondence matching and transformation estimation pipeline. All multi-view registration baselines except RPF [64] follow a two-stage pipeline: first performing pairwise registration along dense or sparse graph edges, then applying pose graph optimization to enforce global consistency. In contrast, our method is single-stage and directly generates the registered point cloud via flow matching, eliminating the need for exhaustive pairwise correspondence matching, transformation estimation, and pose graph optimization.

When evaluating on the cross-domain benchmark, we select models for learning-based baselines as follows. For methods with multiple models trained on different datasets, we choose based on scene scale: ModelNet for object-centric, 3DMatch for indoor small-scale, and KITTI for outdoor large-scale scenes. For methods with only a single model, we rescale the input to match the training scale at test time and then scale back for evaluation.

Table 3: Description of baseline methods used in our experiments, organized by pairwise and multi-view registration tasks.

Method	Category	Venue	Description
Pairwise Registration Baselines			
FPFH [57] + FGR [87]	Conventional	ECCV'16	Handcrafted FPFH features combined with fast global registration via a robust cost function
FPFH [57] + TEASER [80]	Conventional	TRO'20	Handcrafted FPFH features with certifiably optimal pose estimation via truncated least squares and semidefinite relaxation, robust to high outlier rates
KISS-Matcher [36]	Conventional	ICRA'25	Fast and robust registration combining efficient hand-crafted feature extraction with robust correspondence estimation, designed for scalability and generalization across diverse scenes
FCGF [14]	Deep learning	ICCV'19	Sparse 3D convolutional network for dense geometric feature extraction, enabling dense correspondence matching for robust registration. Trained on 3DMatch or KITTI.
Predator [27]	Deep learning	CVPR'21	Overlap-aware network that predicts overlap scores and uses attention-weighted features to focus on overlapping regions, particularly effective in low-overlap scenarios. Trained on 3DMatch, KITTI or ModelNet.
GeoTransformer [54]	Deep learning	CVPR'22	Keypoint-free method matching superpoints via transformation-invariant geometric features; uses optimal transport for dense correspondences without RANSAC. Trained on 3DMatch, KITTI or ModelNet.
BUFFER [2]	Deep learning	CVPR'23	Balances accuracy and efficiency via learned keypoint detection, patch feature embedding, and inlier correspondence generation. Trained on 3DMatch or KITTI.
PARENet [81]	Deep learning	ECCV'24	Rotation-equivariant, position-aware network for robust registration in low-overlap scenarios. Trained on 3DMatch or KITTI.
BUFFER-X [59]	Deep learning	ICCV'25	Extends BUFFER with improved zero-shot generalization via adaptive voxel sizing, farthest-point sampling, and patch-wise scale normalization. Originally trained on 3DMatch or KITTI; we additionally retrain a variant, denoted BUFFER-X*, on our training data.
Multi-view Registration Baselines			
Pairwise registration + PGO [11]	-	CVPR'15	All-pair pairwise registration followed by a grow-based initialization and pose graph optimization, implemented by Open3D [88].
SGHR [69]	Deep learning	CVPR'23	Constructs a sparse pose graph using NetVLAD-based overlap scores and conducts pairwise registration with YOHO [70] on the edges from the sparse pose graph. Then it applies history-reweighted iterative reweighted least squares (IRLS) for stable, outlier-robust convergence. Trained on 3DMatch.
RPF [64]	Deep learning	NeurIPS'25	Rectified Point Flow, a flow-matching framework in Euclidean space that formulates multi-part shape assembly as a conditional generative problem by learning a continuous point-wise velocity field that transports points to their assembled configuration. Trained on ModelNet and several object-centric shape assembly datasets.

C Additional Experimental Results

C.1 Pairwise Registration with Low Overlap

To further demonstrate our model's robustness to low overlap between point clouds, especially in the outdoor LiDAR scenarios, we follow EYOC [40] to curate testing data on both the KITTI [21] and Waymo [62] with increasing spatial distance from 10 m to 50 m between point clouds (thus decreasing overlap). Note that KITTI at 10 m is the setting used in the standard pairwise registration benchmark. As shown in Fig. 1, our model shows superior performance over state-of-the-art methods such as BUFFER-X [59], Predator [27], FCGF [14], and EYOC [40] with increasing scan distance.

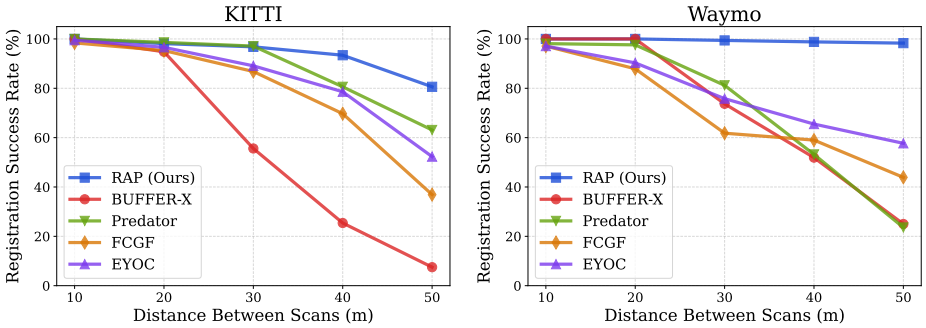


Fig. 1: Comparison of the pairwise point cloud registration: Registration success rate with increasing spatial distance between the two point clouds (thus decreasing overlap ratio) on KITTI and Waymo datasets.

C.2 Results on Indoor Multi-View Registration Benchmark

For multi-view registration evaluation, we also follow prior work [22,69] and use the common multi-view registration benchmarks on 3DMatch [84] and ScanNet [17]. They are both indoor depth camera datasets. We evaluate on sparse-view subsets (with $3 \leq N \leq 12$) of 3DMatch and ScanNet and report the translation error (TE), rotation error (RE), and Chamfer distance (CD). Comparisons with the baseline methods are shown in Tab. 4, demonstrating superior performance. The sparse-view setting makes the pairwise registration more challenging, thus most of the two-stage multi-view registration baselines fail to achieve good performance.

C.3 Additional Results on Cross-Domain Multi-View Registration Benchmark

We show the comparison results on the cross-domain multi-view registration benchmark in Tab. 5, where we use the strict registration success rate (with a

Table 4: Quantitative comparison of multi-view point cloud registration on 3DMatch and ScanNet dataset under the sparse view setting (with view count $3 \leq N \leq 12$). We report the mean rotation error (RE) in degrees, mean translation error (TE) in meters, and Chamfer distance (CD) in meters. Best results are shown in **bold**.

	3DMatch			ScanNet		
Method	RE ↓	TE ↓	CD ↓	RE ↓	TE ↓	CD ↓
FGR [87] + PGO	52.81	0.71	0.49	68.80	1.43	0.76
BUFFER-X [59] + PGO	48.16	0.74	0.48	47.63	1.31	0.52
LMVR [22]	15.46	0.44	0.16	46.10	0.87	0.50
SGHR [69]	50.28	0.78	0.53	23.59	0.64	0.34
RAP (Ours)	**7.27**	**0.23**	**0.11**	**13.85**	**0.34**	**0.12**

Table 5: Comparison of the zero-shot testing performance for multi-view registration on the cross-domain multi-view registration benchmark. We report the *strict* registration success rate (%) calculated for edges and graphs (with a threshold of 0.5% for the normalized translation error and 3° for the rotation error) in different scenarios as well as the average runtime. The best result is in **bold**, and the second best is underscored.

Method		Object		Indoor		Outdoor		TLS		Map		All		Runtime (s)
		edge	graph	edge	graph	edge	graph	edge	graph	edge	graph	edge	graph	
FPFH+FGR [87]	+PGO	9.7	2.1	1.2	0.4	2.2	0.5	2.7	0.6	4.3	1.6	2.5	0.7	19.7
FPFH+TEASER [80]	+PGO	14.8	10.2	5.1	2.3	9.5	3.0	13.8	7.9	26.1	9.8	10.6	5.9	22.2
KISS-Matcher [36]	+PGO	16.4	13.6	15.9	5.7	21.8	7.4	51.6	25.1	75.1	63.5	34.4	17.3	18.4
Predator [27]	+PGO	1.7	0.8	4.4	2.1	3.8	1.4	2.5	1.1	5.0	2.2	3.4	1.5	51.8
BUFFER [2]	+PGO	1.3	0.5	15.3	6.7	8.2	3.7	16.3	6.3	1.4	0.3	12.0	5.4	250.3
BUFFER-X [59]	+PGO	46.1	36.5	14.2	3.4	28.3	15.0	42.2	20.4	11.5	2.9	28.9	13.7	36.0
BUFFER-X* [59]	+PGO	52.8	40.1	21.4	7.7	36.9	21.2	49.1	25.8	17.3	6.6	36.1	20.8	36.0
SGHR [69]		10.3	5.8	17.3	8.6	14.5	7.6	19.8	12.0	12.8	5.8	16.2	9.1	49.7
RPF [64]		1.8	1.1	0.2	0.1	0.3	0.1	0.4	0.1	1.0	0.6	0.5	0.2	11.5
RAP (Ours)		**74.7**	**58.1**	**60.2**	**36.5**	**84.0**	**76.6**	**89.6**	**75.9**	**92.7**	**90.4**	**79.8**	**65.8**	**8.9**
RAP w/o rigidity enforcing		68.8	51.4	53.0	29.9	81.3	70.5	84.6	68.3	89.6	86.8	74.9	59.0	8.9

threshold of 0.5% for the normalized translation error and 3° for the rotation error) instead of the standard threshold (2.5% for the normalized translation error and 15° for the rotation error) used in the main paper.

We additionally provide the ECDF plots for the rotation error, normalized translation error, and normalized global RMSE on the cross-domain multi-view registration benchmark in Fig. 2 (overall) and Fig. 3 (by scenario category). We compare RAP with the two best-performing baselines: BUFFER-X*+PGO and KISS-Matcher+PGO. The results show that RAP outperforms the baselines across all metrics and scenarios consistently.

C.4 Pose Estimation Results for Offline SLAM

One potential application of our model is offline SLAM, where we use it to estimate poses of LiDAR scans in a batch. We evaluate pose estimation accuracy using the absolute trajectory error (ATE) metric on the FusionPortablev2 dataset [73], which is an unseen dataset for our model. We compare our model with state-of-the-art SLAM systems: R3LIVE [37] (LiDAR + camera + IMU),

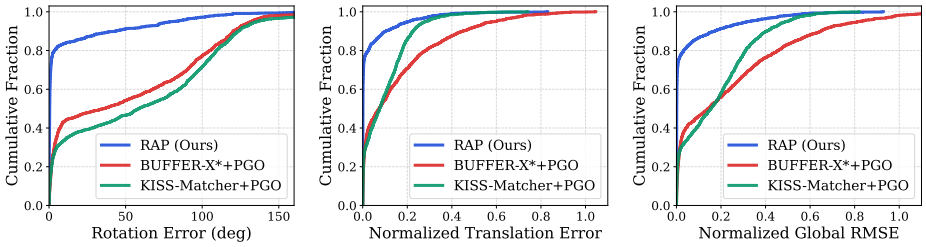


Fig. 2: Comparison of the ECDF of the rotation error, normalized translation error, and normalized global RMSE on the cross-domain multi-view registration benchmark.

FAST-LIO2 [79] (LiDAR + IMU), VINS-Fusion [53] (camera + IMU), DROID-SLAM [66] (camera only), and PIN-SLAM [49] (LiDAR only) in Tab. 6. We additionally show estimated trajectories of two sequences in Fig. 5 and merged point cloud maps on FusionPortablev2 and the Newer College dataset (NCD) [55] in Fig. 4.

Although our model is trained with at most 16 views, it can handle an arbitrary number of views during inference since we do not condition on view indices. However, feeding thousands of LiDAR scans at once is computationally expensive and may cause out-of-memory issues. We therefore sample LiDAR scans at 2 s (20 frames) and 10 s (100 frames) for evaluation, resulting in hundreds or tens of frames per sequence. Note that our model does not assume sequential order of the LiDAR scans and does not rely on any initial pose guess, which are typically required by conventional SLAM systems.

As shown in Tab. 6, our model achieves comparable or better accuracy than the SLAM baselines (R3LIVE, FAST-LIO2, VINS-Fusion) that fuse LiDAR, camera, and IMU data, especially on scenes with geometric degenerations (e.g., ugv_parking00). Compared to PIN-SLAM [49], which is a LiDAR-only SLAM system with sequential scan-to-map registration, RAP achieves competitive or better results despite not relying on sequential processing or any initial pose guess. Intuitively, our model accomplishes an implicit LiDAR bundle adjustment, leading to better performance than approaches relying on incremental scan-to-map registration. The generation process takes about 30 s for more than 100 sampled frames (and more than 2000 frames in the original sequence) on an NVIDIA A5000 GPU, which is comparable to or faster than the compared baselines that process at about 30 ms per frame.

D Additional Information of the Cross-Domain Multi-View Registration Benchmark

Tab. 7 provides an overview of the datasets used to compose the cross-domain multi-view registration benchmark separated by scenario category. Most of these datasets are originally not designed for registration evaluation. The detailed statistics of the datasets can be found in the overview figure (Fig. 3) in the main paper.

Table 6: Offline SLAM localization accuracy comparison with the state-of-the-art SLAM systems on the FusionPortablev2 dataset [73]. We calculate mean translation ATE [m] for each sequence. The best result is shown in **bold**, and the second best result is underlined. Our model works zero-shot on this dataset and takes the LiDAR point clouds per 2 s (20 frames) and 10 s (100 frames) in batches for evaluation.

Method	LiDAR	Camera	IMU	handheld room00	handheld escalator00	legged grass00	legged room00	ugv parking00
R3LIVE [37]	✓	✓	✓	0.057	0.093	**0.069**	0.068	0.424
FAST-LIO2 [79]	✓	×	✓	0.058	**0.085**	0.327	0.093	0.271
VINS-Fusion w/ LC [53]	×	✓	✓	0.063	0.258	1.801	0.149	2.400
DROID-SLAM [66]	×	✓	×	0.118	4.427	7.011	0.135	2.019
PIN-SLAM [49]	✓	×	×	0.061	0.241	0.378	0.073	0.253
RAP (Ours) per 2 s	✓	×	×	**0.050**	0.164	0.105	0.082	**0.250**
RAP (Ours) per 10 s	✓	×	×	0.052	0.140	0.108	**0.064**	0.252

E Additional Qualitative Results

Registration results. We provide additional qualitative results of RAP on both pairwise and multi-view point cloud registration across a variety of datasets. Fig. 6 shows pairwise registration results under challenging conditions such as large scan intervals and low overlap ratios. Fig. 8 illustrates object-centric pairwise registration on ModelNet. For multi-view registration, Fig. 7 presents results on various indoor and outdoor scenarios, Fig. 9 on terrestrial laser scanning data, and Fig. 10 on our cross-domain benchmark spanning object-centric to map-level scenes.

Comparison with VGGT. We show a qualitative comparison between RAP and VGGT [71] on two RGB-D sequences in Fig. 11. The two methods operate on fundamentally different input modalities: RAP takes only geometric point clouds without color, while VGGT takes RGB images without depth. As a result, RAP produces registered point clouds at metric scale (as measured by the depth camera), whereas VGGT's reconstruction is up-to-scale. On the other hand, VGGT's reconstructions tend to be smoother than ours, likely benefiting from the dense photometric information in RGB images.

Failure cases. We provide some failure cases of our model in Fig. 12. Our model sometimes fails to correctly register point clouds when overlap is very low or absent, *e.g.* due to thin walls. For the NSS dataset, the model can still estimate a reasonable global layout of the scene even in these ambiguous cases.

Table 7: Overview of the datasets used to compose the cross-domain multi-view registration benchmark separated by scenario category.

Dataset	Geography	Sensor	License
Object			
C3VD [4]	-	Clinical colonoscope	CC BY-NC-SA 3.0
dSTORM [38]	-	dSTORM microscope	-
Fruit completion [41]	Germany	Hand-held scanner, realsense RGB-D	-
Stanford 3D Models [16]	USA	Cyberware scanner	Custom research license
Indoor Scan			
3DCSR [28]	USA	Kinect RGB-D camera, 16-beam LiDAR, SfM	CC BY-NC-SA 3.0
Bonn RGB-D [48]	Germany	ASUS RGB-D camera	-
IILABS3D [56]	Portugal	Livox Mid-360, RoboSense Helios	CC BY-SA
Leg-KILO [47]	China	Livox Mid-360	CC BY 4.0
Matterport3D [9]	USA	Matterport RGB-D camera	Custom research license
TIERS [60]	Finland	Velodyne 16-beam, Ouster 64/128-beam, Livox	MIT
TUM RGB-D [61]	Germany	Kinect RGB-D camera	CC BY 4.0
Outdoor Scan			
Argoverse2 [74]	USA	2 x LiDARs	CC BY-NC-SA 4.0
Digiforest [42]	Switzerland	Hesai 32/64-beam LiDAR	-
MUST-C [12]	Germany	Ouster 128-beam LiDAR, RIEGL scanner	CC BY 4.0
NCLT [8]	USA	Velodyne 32-beam LiDAR	ODbL
Nuscenes [7]	USA, Singapore	Velodyne 32-beam LiDAR	CC BY-NC 4.0
PandaSet [78]	USA	Hesai 64-beam and forward-facing LiDAR	CC BY-NC-SA 4.0
SGAB [13]	Singapore	3 x Velodyne 16-beam LiDARs	Custom research license
SubT [86]	USA	Velodyne 16-beam LiDAR	-
Truckscenes [20]	Germany	6 x LiDARs	CC BY-NC-SA 4.0
Waymo [62]	USA	5 x LiDARs	Custom research license
ZOD [1]	14 European countries	Velodyne 128-beam LiDAR	CC BY-SA 4.0
TLS			
ETH3D [58]	Switzerland	Leica TLS	CC BY-NC-SA 4.0
ETH-TLS [67]	Switzerland	TLS	-
IndoorLRS [50]	-	Faro TLS and Asus RGB-D camera	-
MCD [45]	Sweden, Germany, Singapore	Leica and Faro TLS	CC BY-NC-SA 4.0
Oxford-TLS [55, 65]	UK	Leica TLS	-
R3DS [46]	Germany, Croatia	Riegl TLS	-
RESSO [10]	Netherlands, Saudi Arabia	Leica and Faro TLS	-
Semantic3D [23]	Switzerland	TLS	CC BY-NC-SA 3.0
Tongji-Trees [72]	China	Z+F TLS	-
VMML [43]	Switzerland	Faro TLS	-
WHU-TLS [19]	China	RIEGL TLS	-
Map			
Abenberg [24]	Germany	Airborne laser scanning	CC BY-NC-SA 4.0
AHN [15]	Netherlands	Airborne laser scanning	CC0
Alita [83]	USA	Map built by Velodyne 16-beam LiDAR	BSD 3-Clause
Kimera-Multi [68]	USA	Map built by Velodyne 16-beam LiDAR	MIT
MS-HKUSTGZ [26]	China	Map built by HESAI 32-beam LiDAR	-

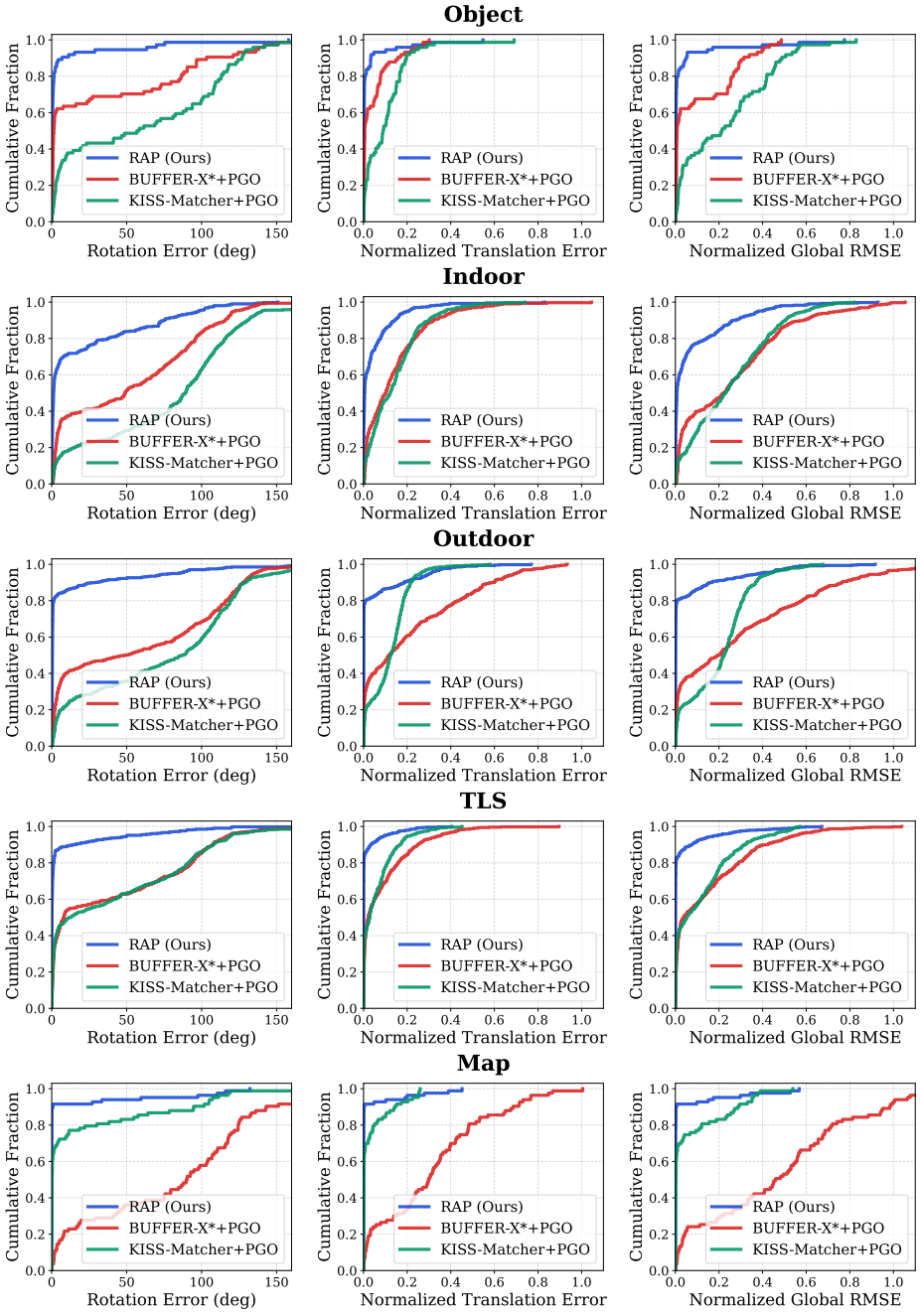


Fig. 3: Comparison of the ECDF of the rotation error, normalized translation error, and normalized global RMSE on the five scenario categories of the cross-domain multi-view registration benchmark.

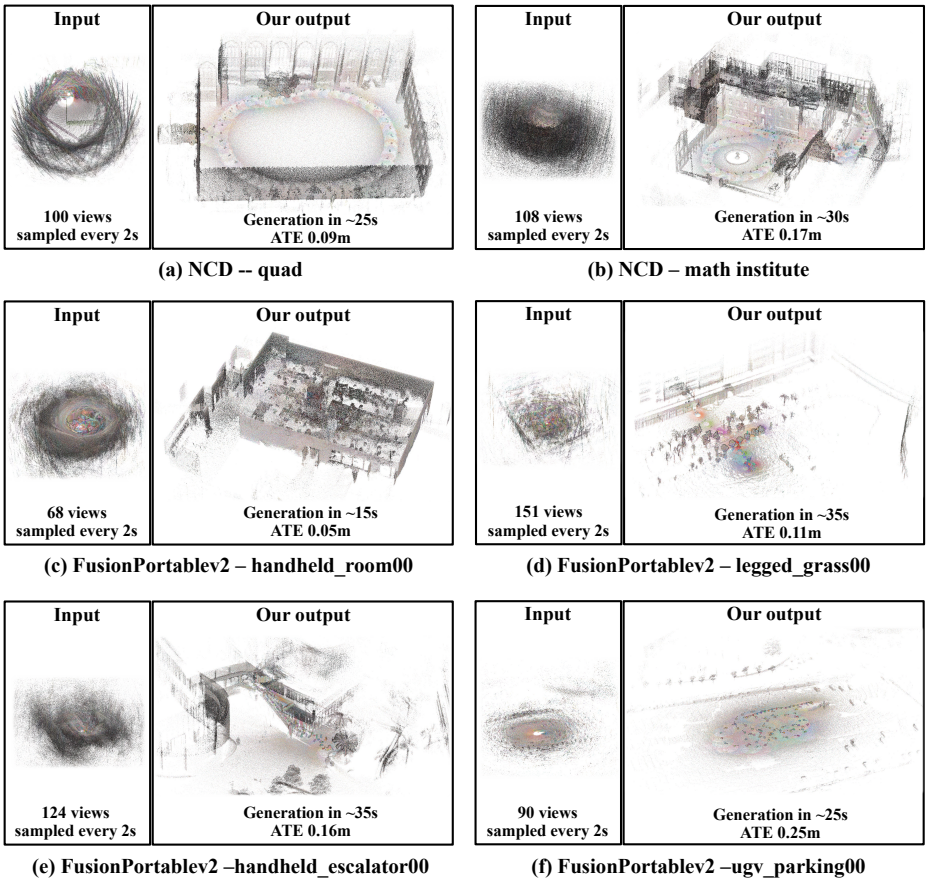
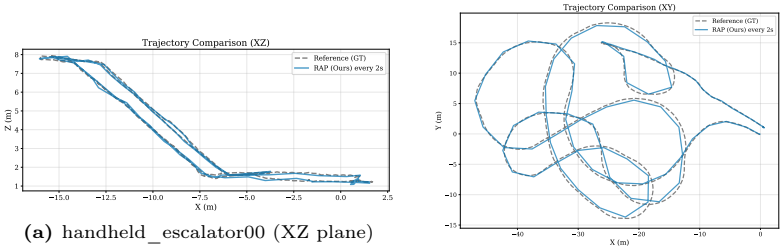


Fig. 4: Offline SLAM results of our model on the FusionPortablev2 [73] and the Newer College dataset [55]. Our model works zero-shot on these datasets, taking LiDAR point clouds per 2 s (20 frames) in batches for evaluation. Different colors in the merged point cloud represent different LiDAR scans. We report the view count, generation runtime, and the localization ATE [m] for each sequence.



(a) handheld_escalator00 (XZ plane)

(b) ugv_parking00 (XY plane)

Fig. 5: Estimated trajectories of RAP compared with the ground truth trajectories on the FusionPortablev2 dataset [73]. Left: handheld_escalator00 sequence shown in the XZ plane. Right: ugv_parking00 sequence shown in the XY plane.

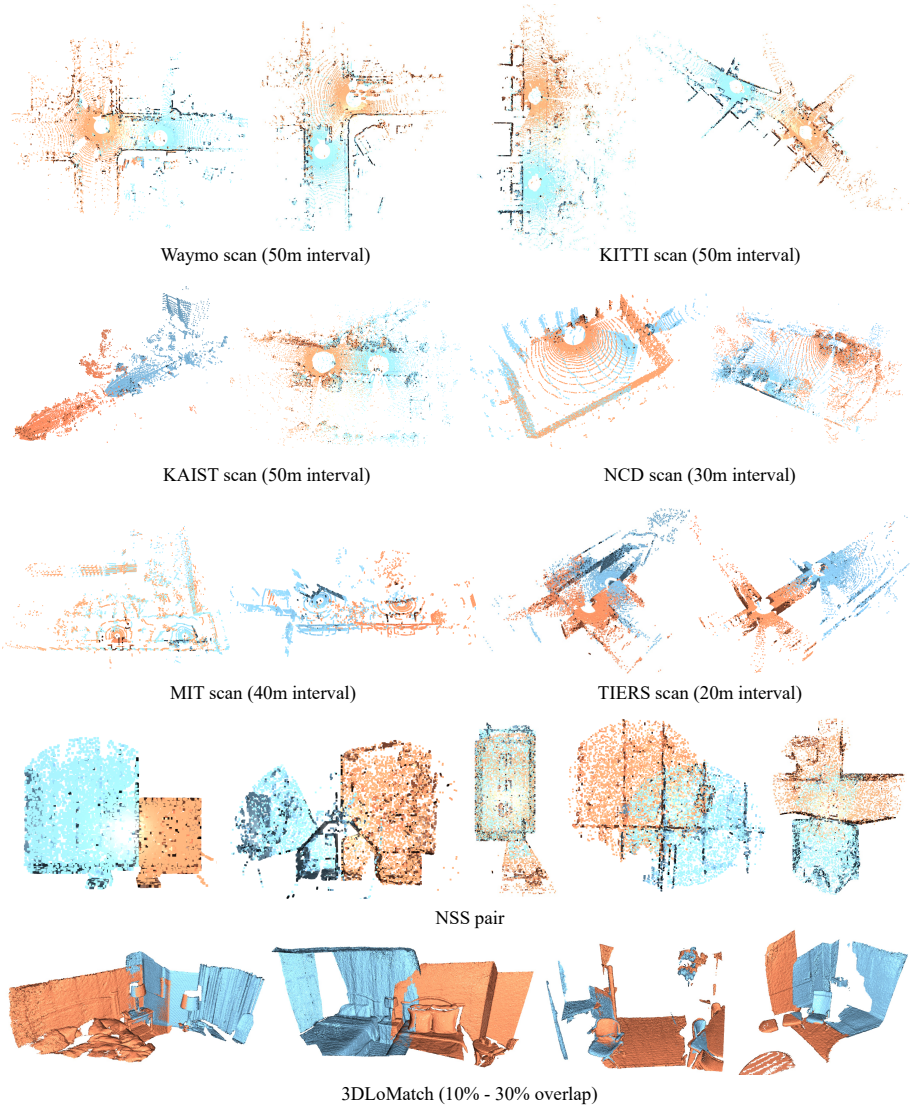


Fig. 6: Qualitative results of our model on pairwise registration with large scan interval (Waymo, KITTI, KAIST, NCD, MIT, TIERS) or low overlap ratio (NSS, 3DLoMatch).

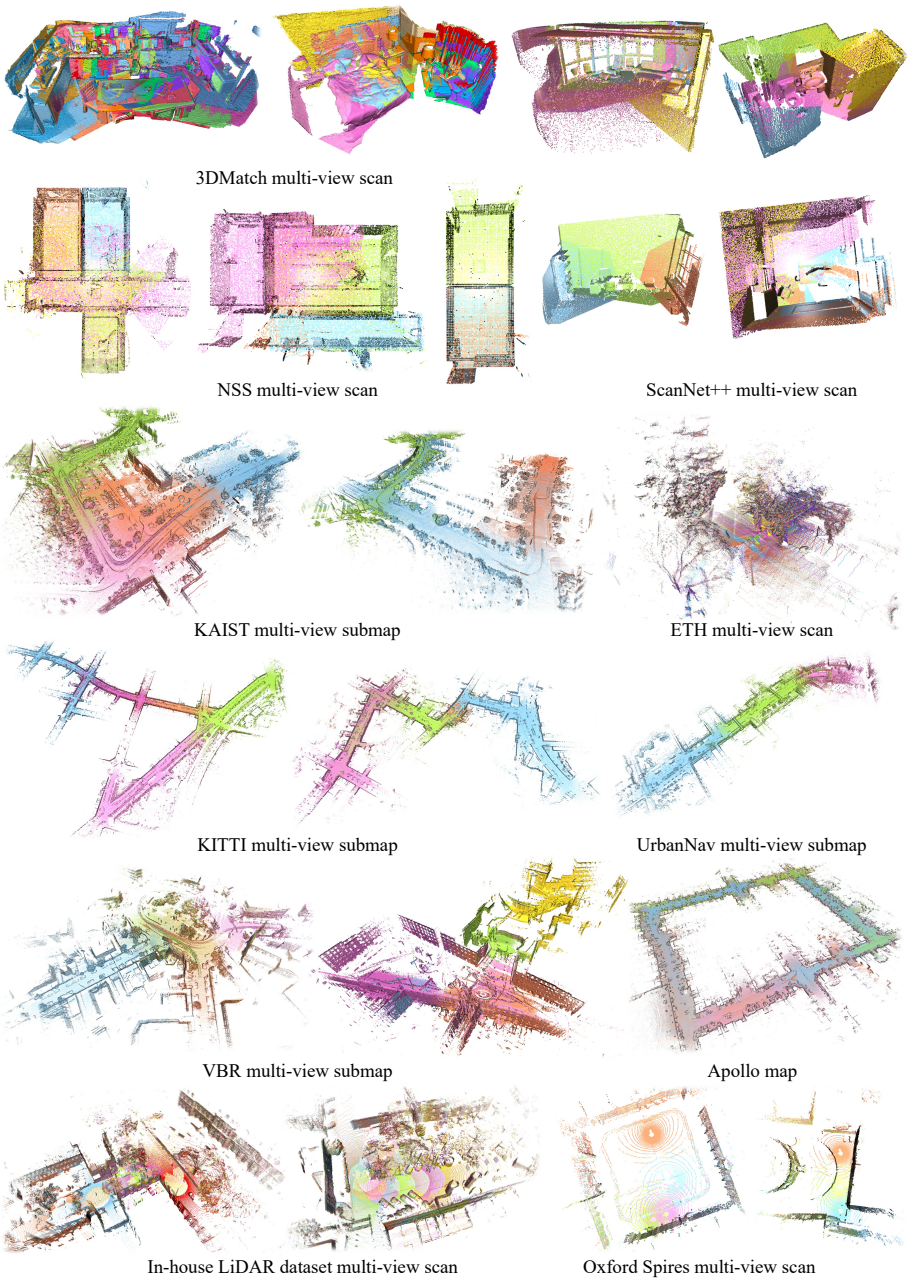


Fig. 7: Qualitative results of our model on multi-view point cloud registration.

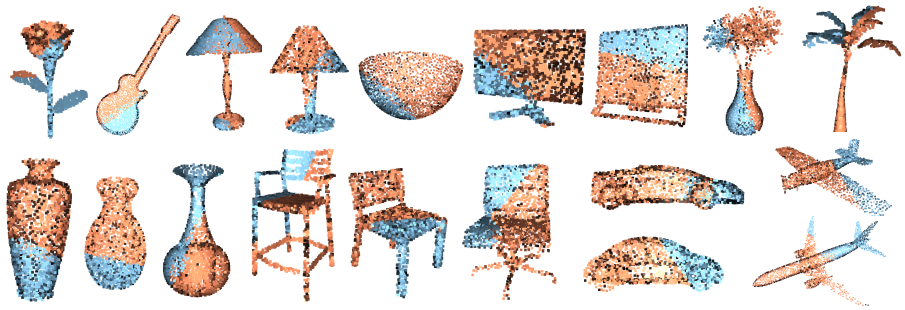


Fig. 8: Qualitative results of our model on ModelNet for object-centric pairwise registration.

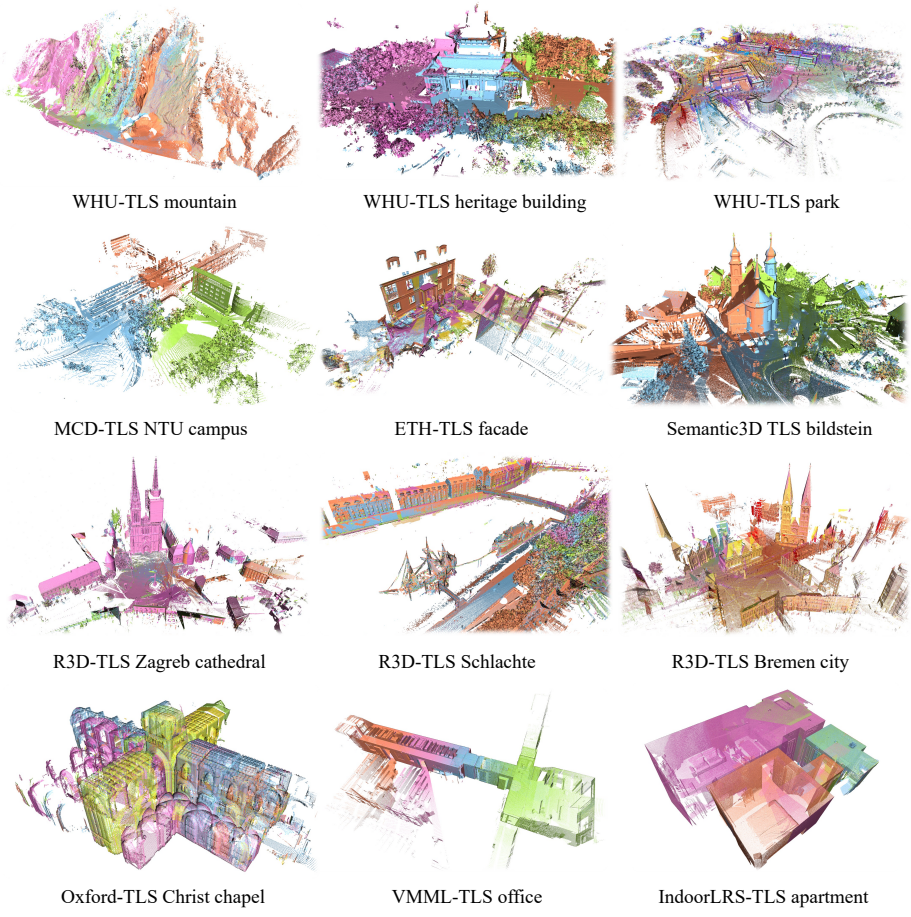


Fig. 9: Qualitative results of our model on multi-view TLS point cloud registration.

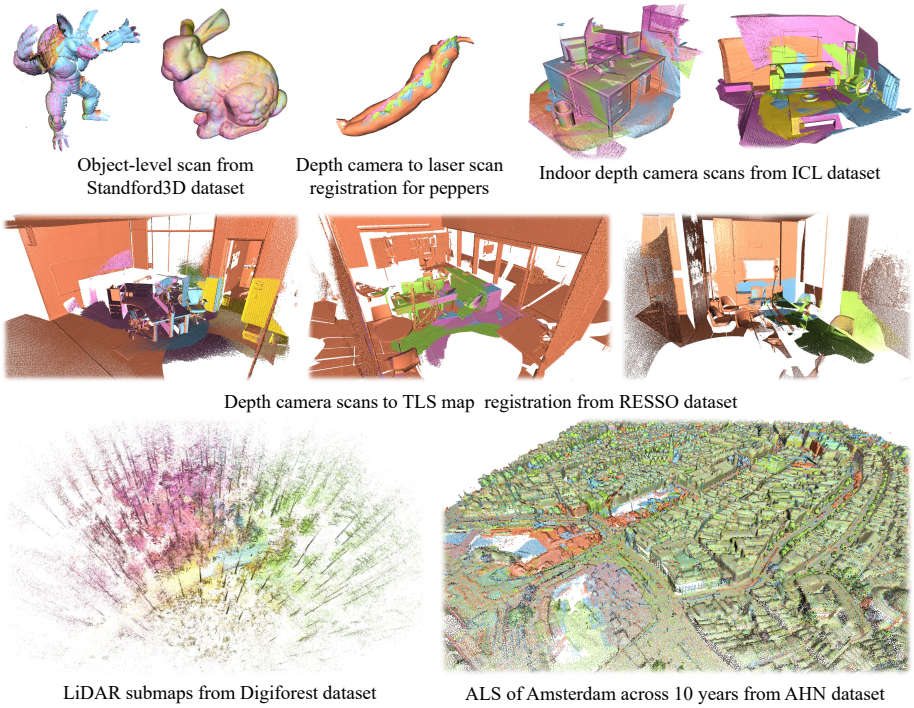


Fig. 10: Qualitative results of our model on cross-domain multi-view registration benchmark, from object-centric to map-level scenes.

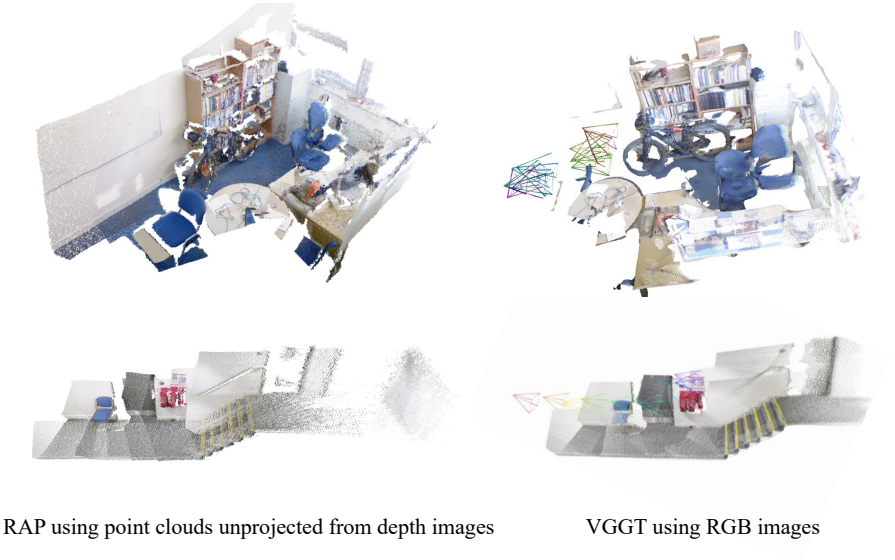


Fig. 11: Qualitative comparison between RAP and VGGT [71] on two RGB-D data sequences. RAP takes colorless point clouds as input and produces metric-scale registered point clouds, while VGGT takes RGB images and produces up-to-scale reconstructions.

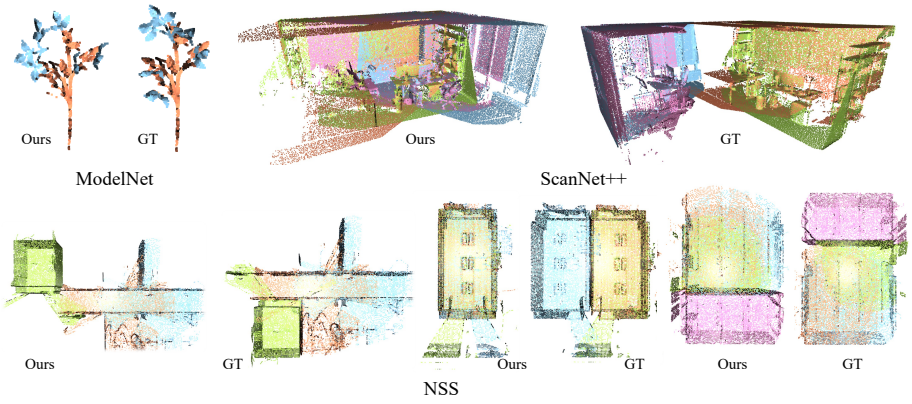


Fig. 12: Failure cases of our model. We provide the prediction of our model and the ground truth for comparison.

References

1. Alibeigi, M., Ljungbergh, W., Tonderski, A., Hess, G., Lilja, A., Motorniuk, D., Fu, J., Widahl, J., Petersson, C.: Zenseact Open Dataset: A large-scale and diverse multimodal dataset for autonomous driving. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2023),
2. Ao, S., Hu, Q., Wang, H., Xu, K., Guo, Y.: BUFFER: Balancing Accuracy, Efficiency, and Generalizability in Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2023),
3. Ao, S., Hu, Q., Yang, B., Markham, A., Guo, Y.: SpinNet: Learning a General Surface Descriptor for 3D Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2021),
4. Bobrow, T.L., Golhar, M., Vijayan, R., Akshintala, V.S., Garcia, J.R., Durr, N.J.: Colonoscopy 3d video dataset with paired depth from 2d-3d registration. Medical Image Analysis **90**, 102956 (2023)
5. Brizi, L., Giacomini, E., Giammarino, L.D., Ferrari, S., Salem, O., Rebotti, L.D., Grisetti, G.: VBR: A Vision Benchmark in Rome. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2024),
6. Burnett, K., Yoon, D.J., Wu, Y., Li, A.Z., Zhang, H., Lu, S., Qian, J., Tseng, W.K., Lambert, A., Leung, K.Y.K., Schoellig, A.P., Barfoot, T.D.: Boreas: A Multi-Season Autonomous Driving Dataset. Intl. Journal of Robotics Research (IJRR) **42**(1-2), 33–42 (2023),
7. Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O.: nuScenes: A Multimodal Dataset for Autonomous Driving. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2020)
8. Carlevaris-Bianco, N., Ushani, A., Eustice, R.: University of Michigan North Campus long-term vision and lidar dataset. Intl. Journal of Robotics Research (IJRR) **35**(9), 1023–1035 (2016),
9. Chang, A., Dai, A., Funkhouser, T., Halber, M., Nießner, M., Savva, M., Song, S., Zeng, A., Zhang, Y.: Matterport3d: Learning from rgb-d data in indoor environments. In: Proc. of the Intl. Conf. on 3D Vision (3DV) (2017),
10. Chen, S., Nan, L., Xia, R., Zhao, J., Wonka, P.: PLADE: A Plane-based Descriptor for Point Cloud Registration with Small Overlap. IEEE Trans. on Geoscience and Remote Sensing **58**(4), 2530–2540 (2020),
11. Choi, S., Zhou, Q., Koltun, V.: Robust Reconstruction of Indoor Scenes. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2015),
12. Chong, Y.L., Krämer, J., Chakhvashvili, E., Marks, E., Esser, F., Dreier, A., Rosu, R.A., Warstat, K., Pude, R., Behnke, S., Müller, O., Rascher, U., Kuhlmann, H., Stachniss, C., Behley, J., Klingbeil, L.: The Multi-Sensor and Multi-Temporal Dataset of Multiple Crops for In-Field Phenotyping and Monitoring. Scientific Data **13** (2026),
13. Chong, Y.L., Lee, C.D.W., Chen, L., Shen, C., Chan, K.K.H., Ang, M.H.J.: Online Obstacle Trajectory Prediction for Autonomous Buses. Machines **10**(3), 202 (2022),
14. Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2019),
15. Cserep, M., Lindenbergh, R.: Distributed processing of Dutch AHN laser altimetry changes of the built-up area. International Journal of Applied Earth Observation and Geoinformation **116**, 222–236 (2023),

16. Curless, B., Levoy, M.: A Volumetric Method for Building Complex Models from Range Images. In: Proc. of the Intl. Conf. on Computer Graphics and Interactive Techniques (SIGGRAPH) (1996),
17. Dai, A., Chang, A., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2017),
18. Deschaud, J.E.: KITTI-CARLA: a KITTI-like dataset generated by CARLA Simulator. arXiv preprint **arXiv:2109.00892** (2021),
19. Dong, Z., Liang, F., Yang, B., Xu, Y., Zang, Y., Li, J., Wang, Y., Dai, W., Fan, H., Hyypä, J., et al.: Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **163**, 327–342 (2020),
20. Fent, F., Kuttenreich, F., Ruch, F., Rizwin, F., Juergens, S., Lechermann, L., Nissler, C., Perl, A., Voll, U., Yan, M., Lienkamp, M.: MAN TruckScenes: A multimodal dataset for autonomous trucking in diverse conditions. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2024),
21. Geiger, A., Lenz, P., Urtasun, R.: Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2012),
22. Gojcic, Z., Zhou, C., Wegner, J.D., Guibas, L.J., Birdal, T.: Learning Multiview 3D Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2020),
23. Hackel, T., Savinov, N., Ladicky, L., Wegner, J.D., Schindler, K., Pollefeys, M.: SEMANTIC3D.NET: A new large-scale point cloud classification benchmark. In: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences (2017)
24. Hebel, M., Arens, M., Stilla, U.: Change Detection in Urban Areas by Object-Based Analysis and On-the-Fly Comparison of Multi-View ALS Data. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **86**, 52–64 (2013),
25. Hsu, L.T., Kubo, N., Wen, W., Chen, W., Liu, Z., Suzuki, T., Meguro, J.: UrbanNav: An Open-Sourced Multisensory Dataset for Benchmarking Positioning Algorithms Designed for Urban Areas. Navigation **70**(4), 226–256 (2023),
26. Hu, X., Wu, J., Jiao, J., Jiang, B., Zhang, W., Wang, W., Tan, P.: MS-Mapping: An Uncertainty-Aware Large-Scale Multi-Session LiDAR Mapping System. arXiv preprint **arXiv:2408.03723** (2024),
27. Huang, S., Gojcic, Z., Usvyatsov, M., Wieser, A., Schindler, K.: PREDATOR: Registration of 3D Point Clouds with Low Overlap. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2021),
28. Huang, X., Mei, G., Zhang, J., Abbas, R.: A comprehensive survey on point cloud registration. In: arXiv preprint (2021),
29. Huang, X., Cheng, X., Geng, Q., Cao, B., Zhou, D., Wang, P., Lin, Y., Yang, R.: The ApolloScape Dataset for Autonomous Driving. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (2018),
30. Jung, M., Yang, W., Lee, D., Gil, H., Kim, G., Kim, A.: HeLiPR: Heterogeneous LiDAR Dataset for inter-LiDAR Place Recognition under Spatiotemporal Variations. Intl. Journal of Robotics Research (IJRR) **12**(43), 1867—-1883 (2024),
31. Kim, G., Park, Y., Cho, Y., Jeong, J., Kim, A.: Mulran: Multimodal range dataset for urban place recognition. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2020),
32. Kingma, D., Ba, J.: Adam: A Method for Stochastic Optimization. In: Proc. of the Intl. Conf. on Learning Representations (ICLR) (2015),

33. Knights, J., Vidanapathirana, K., Ramezani, M., Sridharan, S., Fookes, C., Moghadam, P.: Wild-Places: A Large-Scale Dataset for Lidar Place Recognition in Unstructured Natural Environments. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2023),
34. Lee, S., Lin, Z., Fanti, G.: Improving the Training of Rectified Flows. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2024),
35. Liao, Y., Xie, J., Geiger, A.: KITTI-360: A Novel Dataset and Benchmarks for Urban Scene Understanding in 2D and 3D. IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI) **45**(3), 3292–3310 (2022),
36. Lim, H., Kim, D., Shin, G., Shi, J., Vizzo, I., Myung, H., Park, J., Carlone, L.: KISS-Matcher: Fast and Robust Point Cloud Registration Revisited. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2025),
37. Lin, J., Zhang, F.: R3LIVE: A Robust, Real-time, RGB-colored, LiDAR-Inertial-Visual tightly-coupled state Estimation and mapping package. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2022)
38. van de Linde, S., Löschberger, A., Klein, T., Heidbreder, M., Wolter, S., Heilemann, M., Sauer, M.: Direct stochastic optical reconstruction microscopy with standard fluorescent probes. Nature Protocols **6**(7), 991–1009 (2011),
39. Liu, J., Su, J., Yao, X., Jiang, Z., Lai, G., Du, Y., Qin, Y., Xu, W., Lu, E., Yan, J., Chen, Y., Zheng, H., Liu, Y., Liu, S., Yin, B., He, W., Zhu, H., Wang, Y., Wang, J., Dong, M., Zhang, Z., Kang, Y., Zhang, H., Xu, X., Zhang, Y., Wu, Y., Zhou, X., Yang, Z.: Muon is Scalable for LLM Training. arXiv preprint **arXiv:2502.16982** (2025),
40. Liu, Q., Zhu, H., Wang, Z., Zhou, Y., Chang, S., Guo, M.: Extend Your Own Correspondences: Unsupervised Distant Point Cloud Registration by Progressive Distance Extension. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
41. Magistri, F., Läbe, T., Marks, E., Nagulavancha, S., Pan, Y., Smitt, C., Klingbeil, L., Halstead, M., Kuhlmann, H., McCool, C., Behley, J., Stachniss, C.: A Dataset and Benchmark for Shape Completion of Fruits for Agricultural Robotics. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2025),
42. Malladi, M., Chebrolu, N., Scacchetti, I., Lobefaro, L., Guadagnino, T., Casseau, B., Oh, H., Freissmuth, L., Karppinen, M., Schweier, J., Leutenegger, S., Behley, J., Stachniss, C., Fallon, M.: DigiForests: A Longitudinal LiDAR Dataset for Forestry Robotics. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2025),
43. Michailidis, G.T., Pajarola, R.: ASPIRE: Automatic scanner position reconstruction. The Visual Computer (VC) **35**(9), 1209–1221 (2019),
44. Mildenhall, B., Srinivasan, P., Tancik, M., Barron, J., Ramamoorthi, R., Ng, R.: NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2020),
45. Nguyen, T.M., Yuan, S., Nguyen, T.H., Yin, P., Cao, H., Xie, L., Wozniak, M., Jensfelt, P., Thiel, M., Ziegenbein, J., Blunder, N.: MCD: Diverse Large-Scale Multi-Campus Dataset for Robot Perception. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
46. Nüchter, A., Borrmann, D., Lingemann, K., Elseberg, J.: Robotic 3D Scan Repository (2009), , collection of 3D laser scans for robotics and SLAM research
47. Ou, G., Li, D., Li, H.: Leg-KILO: Robust Kinematic-Inertial-Lidar Odometry for Dynamic Legged Robots. IEEE Robotics and Automation Letters (RA-L) **9**(10), 8194–8201 (2024),

48. Palazzolo, E., Behley, J., Lottes, P., Giguere, P., Stachniss, C.: ReFusion: 3D Reconstruction in Dynamic Environments for RGB-D Cameras Exploiting Residuals. In: Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) (2019),
49. Pan, Y., Zhong, X., Wiesmann, L., Posewsky, T., Behley, J., Stachniss, C.: PIN-SLAM: LiDAR SLAM Using a Point-Based Implicit Neural Representation for Achieving Global Map Consistency. IEEE Trans. on Robotics (TRO) **40**, 4045–4064 (2024),
50. Park, J., Zhou, Q., Koltun, V.: Colored Point Cloud Registration Revisited. In: Proc. of the IEEE Intl. Conf. on Computer Vision (ICCV) (2017),
51. Peebles, W., Xie, S.: Scalable Diffusion Models with Transformers. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2023),
52. Pomerleau, F., Liu, M., Colas, F., Siegwart, R.: Challenging Data Sets for Point Cloud Registration Algorithms. Intl. Journal of Robotics Research (IJRR) **31**(14), 1705–1711 (2012),
53. Qin, T., Li, P., Shen, S.: Vins-mono: A robust and versatile monocular visual-inertial state estimator. IEEE Trans. on Robotics (TRO) **34**(4), 1004–1020 (2018),
54. Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Xu, K.: Geometric Transformer for Fast and Robust Point Cloud Registration. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2022),
55. Ramezani, M., Wang, Y., Camurri, M., Wisth, D., Mattamala, M., Fallon, M.: The newer college dataset: Handheld lidar, inertial and vision with ground truth. In: Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) (2020)
56. Ribeiro, J., Sousa, R., Martins, J., Aguiar, A., Santos, F., Sobreira, H.: Indoor Benchmark of 3D LiDAR SLAM at iilab - Industry and Innovation Laboratory **13**(1), 212421–212442 (2025),
57. Rusu, R., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: Proc. of the IEEE Intl. Conf. on Robotics & Automation (ICRA) (2009),
58. Schöps, T., Schönberger, J.L., Galliani, S., Sattler, T., Schindler, K., Pollefeys, M., Geiger, A.: A Multi-View Stereo Benchmark with High-Resolution Images and Multi-Camera Videos. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2017),
59. Seo, M., Lim, H., Lee, K., Carlone, L., Park, J.: BUFFER-X: Towards Zero-Shot Point Cloud Registration in Diverse Scenes. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2025),
60. Sier, H., Li, Q., Yu, X., Queralta, J.P., Zou, Z., Westerlund, T.: A Benchmark for Multi-Modal LiDAR SLAM with Ground Truth in GNSS-Denied Environments. Remote Sensing **15**(13), 3314 (2023),
61. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A Benchmark for the Evaluation of RGB-D SLAM Systems. In: Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) (2012),
62. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Scalability in perception for autonomous driving: Waymo open dataset. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2020),
63. Sun, T., Hao, Y., Huang, S., Savarese, S., Schindler, K., Pollefeys, M., Armeni, I.: Nothing Stands Still: A Spatiotemporal Benchmark on 3D Point Cloud Registration Under Large Geometric and Temporal Change. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **220**, 799–823 (2025),

64. Sun, T., Zhu, L., Huang, S., Song, S., Armeni, I.: Rectified Point Flow: Generic Point Cloud Pose Estimation. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2025),
65. Tao, Y., Ángel Muñoz-Bañón, M., Zhang, L., Wang, J., Fu, L.F.T., Fallon, M.: The Oxford Spires Dataset: Benchmarking Large-Scale LiDAR-Visual Localisation, Reconstruction and Radiance Field Methods. International Journal of Robotics Research **44**(1), 1–14 (2025),
66. Teed, Z., Deng, J.: Droid-slam: Deep visual slam for monocular, stereo, and rgb-d cameras. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2021),
67. Theiler, P.W., Wegner, J.D., Schindler, K.: Globally Consistent Registration of Terrestrial Laser Scans via Graph Optimization. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **109**, 126–138 (2015),
68. Tian, Y., Chang, Y., Quang, L., Schang, A., Nieto-Granda, C., How, J.P., Carlone, L.: Resilient and Distributed Multi-Robot Visual SLAM: Datasets, Experiments, and Lessons Learned. In: Proc. of the IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS) (2023),
69. Wang, H., Liu, Y., Dong, Z., Guo, Y., Liu, Y.S., Wang, W., Yang, B.: Robust Multiview Point Cloud Registration with Reliable Pose Graph Initialization and History Reweighting. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2023),
70. Wang, H., Liu, Y., Dong, Z., Wang, W.: You Only Hypothesize Once: Point Cloud Registration with Rotation-equivariant Descriptors. In: Proc. of the ACM Intl. Conf. on Multimedia (2022),
71. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: VGGT: Visual Geometry Grounded Transformer. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2025),
72. Wang, X., Yang, Z., Cheng, X., Stoter, J.E., Xu, W., Wu, Z., Nan, L.: Globalmatch: Registration of forest terrestrial point clouds by global matching of relative stem positions. ISPRS Journal of Photogrammetry and Remote Sensing (JPRS) **197**, 71–86 (2023)
73. Wei, H., Jiao, J., Hu, X., Yu, J., Xie, X., Wu, J., Zhu, Y., Liu, Y., Wang, L., Liu, M.: FusionPortableV2: A Unified Multi-Sensor Dataset for Generalized SLAM Across Diverse Platforms and Scalable Environments. Intl. Journal of Robotics Research (IJRR) **44**(7), 1093–1116 (2025),
74. Wilson, B., Qi, W., Agarwal, T., Lambert, J., Singh, J., Khandelwal, S., Pan, B., Kumar, R., Hartnett, A., Pontes, J.K., Ramanan, D., Carr, P., Hays, J.: Argoverse 2: Next Generation Datasets for Self-driving Perception and Forecasting. In: Proc. of the Conf. on Neural Information Processing Systems (NeurIPS) (2021),
75. Wu, X., DeTone, D., Frost, D., Shen, T., Xie, C., Yang, N., Engel, J., Newcombe, R., Zhao, H., Straub, J.: Sonata: Self-Supervised Learning of Reliable Point Representations. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2025),
76. Wu, X., Jiang, L., Wang, P.S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H.: Point Transformer V3: Simpler, Faster, Stronger. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
77. Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., Xiao, J.: 3d shapenets: A deep representation for volumetric shapes. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2015),

78. Xiao, P., Shao, Z., Hao, S., Zhang, Z., Chai, X., Jiao, J., Li, Z., Wu, J., Sun, K., Jiang, K., Wang, Y., Yang, D.: PandaSet: Advanced Sensor Suite Dataset for Autonomous Driving. In: Proc. of the IEEE Intl. Conf. on Intelligent Transportation Systems (ITSC) (2021),
79. Xu, W., Cai, Y., He, D., Lin, J., Zhang, F.: FAST-LIO2: Fast Direct LiDAR-Inertial Odometry. IEEE Trans. on Robotics (TRO) **38**(4), 2053–2073 (2022)
80. Yang, H., Shi, J., Carlone, L.: TEASER: Fast and Certifiable Point Cloud Registration. IEEE Trans. on Robotics (TRO) **37**(2), 314–333 (2020),
81. Yao, R., Du, S., Cui, W., Tang, C., Yang, C.: PARE-Net: Position-Aware Rotation-Equivariant Networks for Robust Point Cloud Registration. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2024),
82. Yeshwanth, C., Liu, Y.C., Nießner, M., Dai, A.: ScanNet++: A High-Fidelity Dataset of 3D Indoor Scenes. In: Proc. of the IEEE/CVF Intl. Conf. on Computer Vision (ICCV) (2023),
83. Yin, P., Zhao, S., Ge, R., Cisneros, I., Fu, R., Zhang, J., Choset, H., Scherer, S.: ALITA: A Large-scale Incremental Dataset for Long-term Autonomy. arXiv preprint **arXiv:2205.10737** (2022),
84. Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T.: 3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions. In: Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2017),
85. Zhang, Y., Wu, X., Yang, Y., Fan, X., Li, H., Zhang, Y., Huang, Z., Wang, N., Zhao, H.: Utonia: Toward One Encoder for All Point Clouds. arXiv preprint **arXiv:2603.03283** (2026),
86. Zhao, S., Gao, Y., Wu, T., Singh, D., Jiang, R., Sun, H., Sarawata, M., Whittaker, W.C., Higgins, I., Su, S., Du, Y., Xu, C., Keller, J., Karhade, J., Nogueira, L., Saha, S., Qiu, Y., Zhang, J., Wang, W., Wang, C., Scherer, S.: SubT-MRS Dataset: Pushing SLAM Towards All-weather Environments. In: Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR) (2024),
87. Zhou, Q., Park, J., Koltun, V.: Fast Global Registration. In: Proc. of the Europ. Conf. on Computer Vision (ECCV) (2016),
88. Zhou, Q., Park, J., Koltun, V.: Open3D: A modern library for 3D data processing. arXiv preprint **arXiv:1801.09847** (2018),