Computer-Aided Design 199 (2026) 104101

Contents lists available at ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad

PoseFusion: Fusing neural implicit surfaces for multi-view reconstruction from multi-pose captures☆

Guanli Hou [a], Yuanmu Xu [b], Tenglong Ren [a], Jiangbei Hu [c], Fei Hou [d,e], Peng Song [f], Ying He [a,*]

[a] Nanyang Technological University, Singapore, Singapore
[b] University of Science and Technology Beijing, Beijing, China
[c] Dalian University of Technology, Dalian, China
[d] Key Laboratory of System Software (CAS), Institute of Software, Chinese Academy of Sciences, Beijing, China
[e] University of Chinese Academy of Sciences, Beijing, China
[f] Singapore University of Technology and Design, Singapore, Singapore

ARTICLE INFO

Dataset link: Dataset of PoseFusion (Original data)

Keywords:
Neural implicit surfaces
Multi-view reconstruction
Multi-pose registration

ABSTRACT

Recent advances in multi-view 3D reconstruction, especially neural implicit surface methods, can recover high-quality geometry by representing shape with signed distance functions (SDFs). However, reconstructing complex objects with severe self-occlusion remains difficult when all images are captured under a single object pose, because the observed views often provide incomplete spatial coverage. We propose *PoseFusion*, a multi-pose reconstruction framework that fuses neural implicit surfaces independently learned from different object poses into a unified 3D representation. PoseFusion follows a two-stage registration-and-fusion pipeline. First, we extract an oriented bounding box (OBB) from the mesh derived from each pose-specific SDF and use the OBBs, SDF samples, and multi-view image features to estimate a coarse inter-pose alignment. Second, we refine the alignment using image- and SDF-guided correspondences: cross-pose image matches are lifted to 3D using the learned SDFs and used to iteratively optimize the relative transformations. To facilitate evaluation, we construct a dataset of synthetic and real-world objects with complex geometry and strong self-occlusion, captured across multiple poses with calibrated multi-view images. Experiments show that PoseFusion is robust under challenging capture conditions and consistently produces high-fidelity reconstructions from multi-pose, multi-view inputs. Code and dataset are publicly available at https://raining00.github.io/PoseFusion-page/.

1. Introduction

Neural radiance fields (NeRF) [1] have revolutionized novel view synthesis by producing photorealistic images from arbitrary viewpoints. NeRF represents scene geometry as a volumetric density field and models appearance through view-dependent radiance, rendering images via volume integration. While this density-based implicit representation enables high-quality novel view synthesis, its often ambiguous density estimates and lack of clear geometric regularization limit its use in geometry-focused tasks such as editing, simulation, or physical analysis.

Neural implicit surfaces extend NeRF to 3D surface reconstruction by incorporating signed distance functions (SDFs) as the underlying geometric representation, such as NeuS [2] and VolSDF [3]. Unlike NeRF's volumetric density field, SDFs define surfaces implicitly as the zero level set of a continuous function, providing a more precise and well-regularized geometric formulation. This representation allows for accurate surface normal computation, supports mesh extraction via techniques such as marching cubes, and facilitates gradient-based optimization with geometric priors. As a result, SDF-based methods enable high-fidelity reconstruction of complex surfaces with sharp features and accurate topology. Building on this foundation, follow-up work has further improved efficiency and robustness. For instance, Voxurf [4] and NeuS2 [5] reduce training time through voxel grids and multi-resolution hash encodings. Other methods improve reconstruction accuracy by integrating 3D geometric priors [6,7], while Sparse NeuS [8] enhances robustness under sparse view settings. Together, these advances have pushed the frontier of neural implicit surface reconstruction in a variety of real-world scenarios.

However, existing neural implicit surface methods remain limited by a practical constraint: they typically assume that all input views

☆ This article is part of a Special issue entitled: 'SPM 2026' published in Computer-Aided Design.

* Corresponding author.
E-mail address: yhe@ntu.edu.sg (Y. He).

https://doi.org/10.1016/j.cad.2026.104101
Received 30 April 2026; Accepted 19 May 2026
Available online 30 May 2026

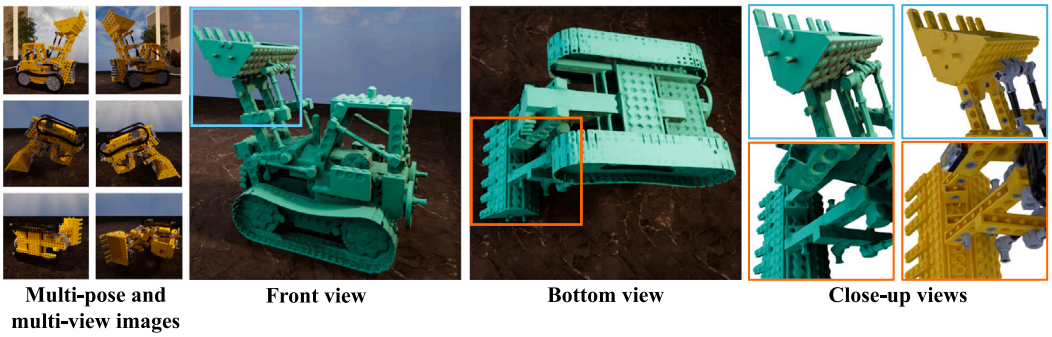

Fig. 1. Given multi-view images captured under multiple object poses, PoseFusion automatically registers and fuses the geometry and appearance learned from each pose into a unified neural implicit surface. The resulting representation supports both mesh extraction and novel-view rendering, recovering geometry in regions that are poorly observed from any single pose.

are captured under a single object pose. This assumption limits their ability to recover complete geometry, especially for objects with complex structures or severe self-occlusion. In such cases, large portions of the object may remain unobserved, including regions in contact with a supporting surface or occluded by other parts of the geometry, resulting in incomplete or distorted reconstructions. To overcome the limitations of single-pose reconstruction, we introduce *PoseFusion*, a novel framework for multi-view 3D reconstruction from multi-pose captures. Instead of relying on a single viewing configuration, PoseFusion leverages multiple sets of multi-view images captured from different poses of the same object, each yielding an independently trained neural implicit surface. Our key idea is to register and fuse these independently learned implicit surfaces into a unified 3D representation that captures the full geometry of the object.

PoseFusion is a two-stage registration-and-fusion pipeline. In the first stage, we extract a mesh from each implicit neural surface and compute its oriented bounding box (OBB). These OBBs, along with corresponding SDF values and multi-view image features, are used to estimate an initial coarse alignment between poses. Although this coarse registration resolves global misalignment, residual local inconsistencies can still hinder high-precision fusion. The second stage introduces a hybrid strategy that augments geometric cues with image-based correspondences to refine the alignment. By identifying cross-pose correspondences in the multi-view images and lifting them into 3D space using the underlying SDFs, we iteratively optimize the relative pose transformations. This refinement significantly improves registration accuracy and leads to high-quality surface fusion, even in the presence of severe occlusions and missing regions in individual captures.

To demonstrate the effectiveness of PoseFusion, we construct a new dataset comprising both synthetic and real-world objects with challenging geometry and appearance. Each object is captured from multiple poses, and for each pose, we provide calibrated multi-view images. Extensive experiments show that PoseFusion produces more complete and accurate 3D reconstructions compared to existing single-pose methods, especially for objects with complex geometry and severe self-occlusion. Fig. 1 shows the high-quality geometry and appearance reconstructed by PoseFusion on a challenging object. Our dataset offers a standardized benchmark for registration and is intended to stimulate further research in this area.

2. Related work

2.1. Neural radiance fields

NeRF [1] introduced a differentiable volumetric rendering framework that represents 3D scenes with a continuous density field and a view-dependent radiance field, enabling high-quality novel-view synthesis through volume rendering. To improve geometric accuracy, neural implicit surface methods such as NeuS [2], VolSDF [3], and their successors incorporate SDFs into differentiable rendering, allowing surfaces to be represented as zero-level sets. Follow-up methods further improve efficiency and reconstruction quality. For example, NeuS2 [5] and Voxurf [4] accelerate training using multi-resolution hash encodings and voxel grids, respectively. Recent octree-based approaches [9, 10] reduce expensive neural queries or incorporate 3D Gaussian splatting [11], enabling efficient rendering and high-quality surface reconstruction. Despite these advances, most neural implicit surface methods still assume that all input images are captured under a single object pose, which limits their ability to recover complete geometry when large self-occluded regions are not visible.

2.2. Neural-field registration

As illustrated in Fig. 2, our problem is related to several alignment settings. The first setting estimates relative camera poses from unposed images. Methods such as PoseDiffusion [12], RelPose [13], and Cameras as Rays [14] infer relative transformations between images or camera frames. Although these methods do not directly register reconstructed geometry, they address a closely related pose-alignment problem. The second setting is point-cloud registration, where methods such as FGR [15] and REGTR [16] align unregistered 3D point sets. The third setting is neural-field registration, which aligns neural representations reconstructed under different poses, as in nerf2nerf [17], DReg-NeRF [18], and Reg-NF [19].

NeRF-based methods can reconstruct high-quality geometry and appearance in well-observed regions, but they often struggle in view-limited areas, leading to artifacts, incomplete surfaces, and missing details. In practical object-capture settings, a single object pose rarely exposes the full surface because of self-occlusion, contact with the support plane, and acquisition constraints. A natural solution is therefore to capture the same object under multiple poses and register the resulting reconstructions to improve spatial coverage.

The method nerf2nerf [17] addresses pairwise registration of neural radiance fields by distilling a "surface field" from a pretrained NeRF. This surface field estimates the likelihood that a 3D point lies on the object surface, and registration is formulated as a robust optimization problem that aligns the surface fields of two scenes. Despite its strong performance, nerf2nerf relies on manually annotated keypoints to guide the alignment, which limits its scalability and automation.

DReg-NeRF [18] removes this manual requirement by introducing a fully automatic registration pipeline. It first encodes pretrained NeRFs into spatial voxel grids by computing multi-resolution occupancy grids from density fields. Multi-scale voxel features are then extracted using a feature pyramid network, and a transformer-based decoder predicts correspondences between feature points from different poses to estimate the relative transformation. However, DReg-NeRF still relies on iterative closest point (ICP) [20] for the final alignment, making it sensitive to initialization. As a result, it typically requires

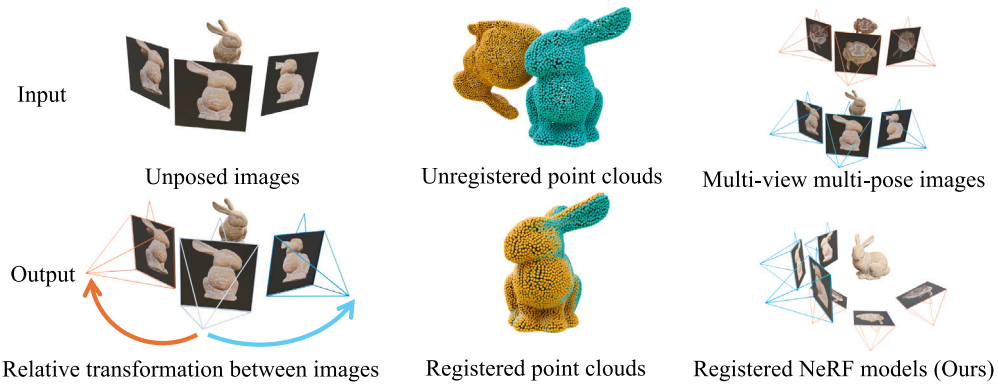

Fig. 2. Three related alignment settings. From left to right: relative camera-pose estimation from unposed images, point-cloud registration, and multi-pose neural-field registration with both geometric and photometric cues. The corresponding outputs are relative camera poses, registered point clouds, and registered pose-wise neural fields, respectively.

a reasonably accurate initial pose; reported results indicate that the method often fails to converge when the initial rotational mismatch exceeds 10°. Moreover, the quality of NeRF density fields depends strongly on view coverage during training. Under sparse or uneven view distributions, the reconstructed density fields may contain outliers or incomplete structures, leading to unreliable geometric features and degraded registration performance.

Reg-NF [19] instead uses SDFs as geometric guidance and introduces a bidirectional registration loss with multi-view surface sampling. By operating on implicit surfaces rather than density fields, Reg-NF provides stronger geometric constraints and can handle registrations involving scale differences. However, its sampling strategy depends heavily on the distribution of input views. When sampled regions are poorly observed or have little overlap across poses, the corresponding SDF values may be unreliable. As a result, sampled source points may not be transformed accurately onto the target surface, even when the estimated transformation is close to correct. This limitation becomes especially pronounced under low view coverage or severe self-occlusion.

2.3. 3DGS registration

Recently, 3D Gaussian Splatting (3DGS) [11] has emerged as an efficient alternative to NeRF. By representing scenes as explicit anisotropic 3D Gaussians and using point-based rasterization, 3DGS avoids computationally expensive volume rendering, significantly reducing training time and enabling real-time rendering. GaussReg [21] addresses registration between Gaussian fields using a coarse-to-fine alignment strategy. It first performs coarse alignment through point-cloud registration and then refines the transformation with image-guided optimization based on rendered Gaussian-splatting images. Although GaussReg is efficient for aligning point-based representations, it does not directly provide an explicit surface representation. Therefore, additional surface extraction or post-processing is required for downstream applications that depend on complete mesh geometry, such as editing, simulation, or physical analysis.

2.4. Comparison

PoseFusion targets a different setting from existing NeRF- and 3DGS-based registration methods: multi-pose, multi-view reconstruction of a single object with severe self-occlusion. Our goal is not only to align independently reconstructed neural fields, but also to fuse them into a complete SDF-based surface representation.

First, PoseFusion adopts a simple but effective oriented bounding box strategy for coarse alignment. When each pose provides sufficient view coverage, the learned geometry often captures the dominant convex structure of the object, making the OBB a useful proxy for the global object frame. However, under partial or sparse view coverage, the learned geometry may be incomplete or distorted, and the OBB alone may no longer be reliable. To improve robustness, PoseFusion augments OBB-based geometric alignment with image features, which provide complementary cues in poorly reconstructed or low-coverage regions.

Second, although PoseFusion also uses SDFs for geometric representation, it differs from Reg-NF [19] in how surface samples are selected. Rather than relying only on view-dependent surface sampling, PoseFusion uses image correspondences to guide the selection of points on the zero-level sets of the learned SDFs. This correspondence-guided strategy encourages the optimization to focus on regions that are visually matched across poses, making registration more robust when different poses observe different parts of the object.

Finally, unlike GaussReg [21], which aligns 3D Gaussian representations without directly producing explicit surface geometry, PoseFusion reconstructs an SDF-based neural surface from which a mesh can be extracted. This enables complete surface reconstruction and supports downstream tasks such as editing, simulation, and geometry-centric analysis.

3. Method

3.1. Overview

PoseFusion adopts a two-stage pipeline for reconstructing 3D objects from multi-pose captures. In the first stage, we perform coarse registration by aligning oriented bounding boxes computed from pose-wise neural implicit surfaces. This alignment is guided by SDF values sampled on the OBB surfaces together with multi-view image features. In the second stage, we establish cross-pose correspondences using image information, lift matched pixels to the zero-level sets of the learned SDFs, and refine the initial alignment with SDF-based geometric constraints. After all poses are aligned, the corresponding neural implicit surfaces can be fused into a unified reconstruction. An optional retraining step further improves fine geometric details in the final 3D model. Fig. 3 illustrates the overall pipeline, and the two stages are detailed below.

3.2. Coarse registration

The goal of coarse registration is to align all poses to a common reference pose using an oriented bounding box alignment strategy. Without loss of generality, we select the first pose as the target pose, denoted by P_t. When more than two poses are available, registration is performed pairwise by aligning each source pose to the target pose. For

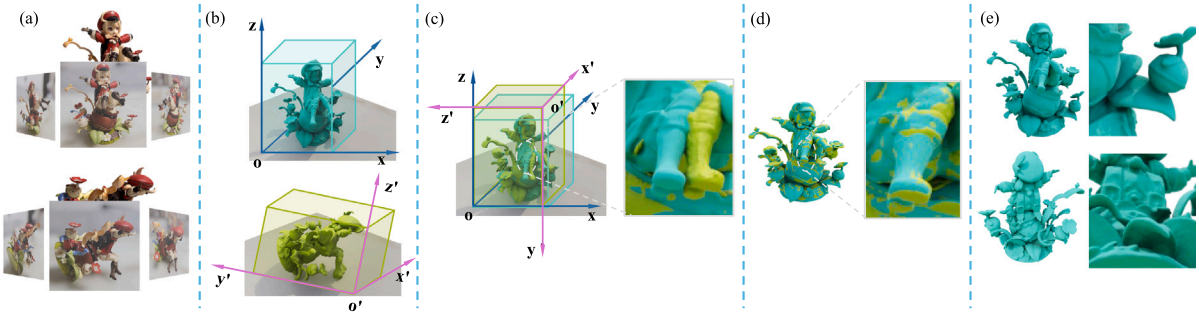

Fig. 3. Overview of the PoseFusion pipeline. PoseFusion reconstructs geometry and appearance from multi-view images captured under multiple object poses. (a) Input multi-view images from different poses. (b) Per-pose reconstructions using neural implicit surfaces. (c) Coarse alignment using oriented bounding boxes. (d) Fine alignment guided by image correspondences and SDF constraints. (e) Final fusion produces a complete 3D reconstruction with enhanced geometric details.

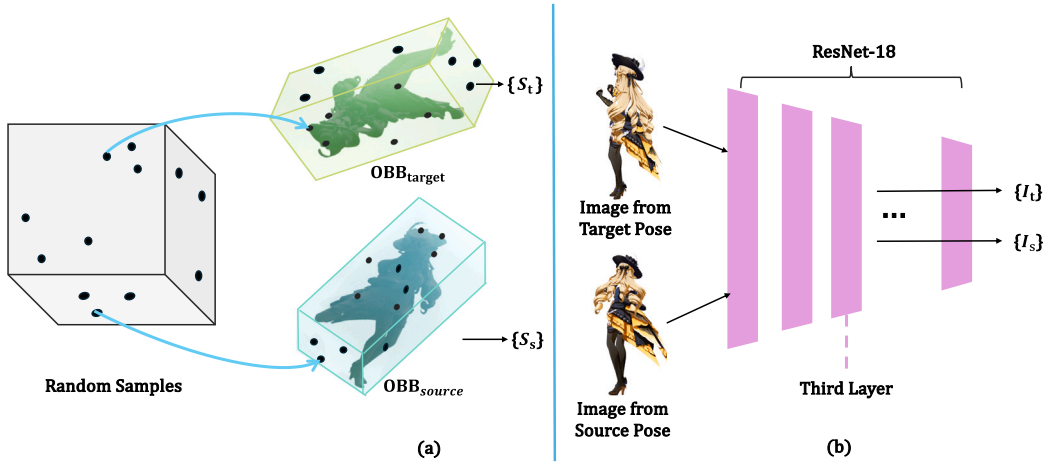

Fig. 4. SDF- and image-feature-augmented OBB alignment. (a) Sample points are randomly generated on the surface of a unit cube and mapped to the oriented bounding boxes of the source and target poses. The SDF values at the mapped points are used to compute the geometric alignment term. (b) For the selected source and target images, feature representations $\mathcal{I}_t$ and $\mathcal{I}_s$ are extracted using the third residual block, conv3, of a ResNet-18 network. These features are used to compute the image-based similarity term in the coarse-alignment objective.

clarity, we describe the two-pose case, where P_s denotes the source pose to be aligned with P_t. For each pose, we use an existing neural implicit surface method to learn an SDF from the corresponding multi-view images. We then extract a mesh from the zero-level set of the learned SDF. To estimate a coarse object coordinate frame, we apply Principal Component Analysis (PCA) [22] to the mesh and compute an OBB. Let $\mathcal{O}_s$ and $\mathcal{O}_t$ denote the OBBs of the source and target poses, respectively. Each OBB has eight vertices, yielding 24 possible vertex correspondences between $\mathcal{O}_s$ and $\mathcal{O}_t$. Each correspondence defines a candidate rigid transformation τ_i for $i = 1, \ldots, 24$. To select the best initial alignment, we augment the OBB representation with both SDF values and image features. Let S_s and S_t denote the learned signed distance fields for the source and target poses, respectively. We also incorporate image features $\mathcal{I}_s$ and $\mathcal{I}_t$ to disambiguate OBB correspondences that may be geometrically plausible but visually inconsistent.

For each candidate transformation τ_i, we randomly sample N_o points on the surfaces of the OBBs $\mathcal{O}_s$ and $\mathcal{O}_t$. The SDF values of those sample points are denoted by $\{S_s^{(i)}, S_t^{(i)}\}_{i=1}^{N_o}$. We then evaluate all 24 candidate transformations and select the optimal transformation $\tau^* \in \{\tau_1, \ldots, \tau_{24}\}$ that minimizes a combined geometric and visual objective:

$$L_{\text{coarse}} = \frac{\lambda_{\text{sdf}}}{N_o} \sum_{i=1}^{N_o} \left\| S_t^{(i)} - S_s^{(i)} \right\| + \lambda_{\text{img}} \left(1 - \frac{\phi(\mathcal{I}_t) \cdot \phi(\mathcal{I}_s)}{\|\phi(\mathcal{I}_t)\| \, \|\phi(\mathcal{I}_s)\|} \right) \tag{1}$$

where the first term measures the average distance between SDF values at corresponding sample points, normalized by the maximum absolute difference δ to ensure scale invariance. The second term evaluates the cosine distance between global image features, where $\phi(\cdot)$ denotes the image feature extractor based on ResNet-18 [23], using the output of the third residual block as shown in Fig. 4(b). The weights λ_{sdf} and λ_{img} control the relative contributions of geometric and visual terms. In our implementation, we empirically set $\lambda_{\text{sdf}} = \lambda_{\text{img}} = 0.5$.

3.3. Fine registration

The OBB-based coarse alignment provides an initialization for the fine registration stage, which further improves the alignment accuracy between poses. To address the limitations of previous sampling strategies, especially under low view overlap and large pose variation, we adopt an automatic correspondence-estimation approach using eLoFTR [24,25]. This allows us to establish robust 2D point correspondences between images captured from different poses without manual annotation. Leveraging these correspondences, we then lift the matched pixels to 3D by projecting them onto the zero-level sets of the learned SDFs, providing surface samples for SDF-guided registration.

A key challenge is selecting suitable image pairs across different poses for eLoFTR matching. Random selection often yields pairs with little or no overlap, leading to unreliable matches. Conversely, selection based solely on image features may fail for symmetric or textureless objects. We therefore select image pairs based on camera-pose similarity, which increases the likelihood that the chosen images observe overlapping object regions.

Table 1
Quantitative comparison of registration accuracy on the synthetic benchmark. We report results for two source-to-target registration pairs, (source$_1$, target) and (source$_2$, target), with each entry formatted as source$_1$/source$_2$. Metrics include rotation error **R** in degrees and translation error **T** scaled by $\times 10^2$. Lower values indicate better performance. "–" denotes failed cases, and the best results are highlighted in boldface.

Metric	Method	Bunny	Book	Nikon	Heel Slipper	Figurine A	Tank	Monster	Figurine B	Rubik's Cube	Lego
R (°, ↓)	Reg-NF	3.62/1.84	0.29/0.27	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–
	DReg-NeRF	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–
	GaussReg	13.98/–	10.21/22.27	–/–	3.27/–	–/–	–/–	–/–	25.29/–	–/–	–/–
	Ours	**0.09/0.02**	**0.11/0.18**	**0.09/0.11**	**0.12/0.07**	**0.01/0.03**	**0.14/0.09**	**0.03/0.03**	**0.04/0.06**	**0.08/0.11**	**0.05/0.20**
T($\times 10^2$, ↓)	Reg-NF	1.47/0.75	0.13/3.84	29.56/23.0	26.25/18.68	21.29/–	–/–	–/–	–/–	–/–	–/–
	DReg-NeRF	11.48/15.11	14.7/12.3	27.72/1.33	2.36/15.55	13.05/12.41	–/–	–/–	–/–	–/–	–/–
	GaussReg	2.40/26.58	1.13/15.15	7.83/–	2.73/–	30.09/–	–/–	–/28.40	11.69/–	–/–	–/–
	Ours	**0.10/0.02**	**0.06/0.03**	**0.07/0.14**	**0.03/0.02**	**0.01/0.02**	**0.13/0.10**	**0.03/0.02**	**0.03/0.06**	**0.14/0.10**	**0.07/0.18**

We first apply the coarse transformation τ^* to transform the source camera extrinsics into the coordinate frame of the target pose. We then select matching image pairs by comparing the transformed camera poses with the target camera poses:

$$\text{trace}(\mathbf{E}_1 \mathbf{E'}_2^{-1}) < \epsilon, \quad \mathbf{E'}_2 = \tau^* \mathbf{E}_2, \tag{2}$$

where $\mathbf{E}_1$ and $\mathbf{E}_2$ denote the extrinsic matrices of image sets I_1 and I_2, respectively, as illustrated in Fig. 5. The trace function quantifies the similarity of two camera poses based on their relative transformation. The threshold ϵ controls the angular closeness. This selection procedure identifies image pairs with similar viewpoints across poses, ensuring robust 2D keypoint matching.

After eLoFTR produces 2D correspondences, we lift the matched keypoints to 3D by tracing along their camera rays and snapping them to the zero-level sets of the corresponding SDFs. Using these 3D correspondences, we optimize the rigid transformation $\mathbf{T} = (\mathbf{R}, \mathbf{t})$ between poses by minimizing the following loss

$$L_{\text{fine}} = \kappa(L_{\text{sdf}}) + \lambda L_{\text{rotation}}, \tag{3}$$

where the SDF consistency term L_{sdf} and the rotation regularization L_r are defined as

$$L_{\text{sdf}} = \sum_{\mathbf{p} \in P_{\text{source}}} \left| S_{\text{target}}(\mathbf{Tp}) \right| + \sum_{\mathbf{q} \in P_{\text{target}}} \left| S_{\text{source}}(\mathbf{T}^{-1}\mathbf{q}) \right|,$$

and

$$L_{\text{rotation}} = \left\| \mathbf{R}^T \mathbf{R} - \mathbf{I} \right\|,$$

respectively. Here, P_{source} and P_{target} denote the sample points from the zero level sets of the learned SDFs for the source and target poses, respectively. The vectors $\mathbf{p}$ and $\mathbf{q}$ represent the coordinates of points from P_{source} and P_{target}. The robust loss kernel κ, adopted from [26], is used to mitigate the influence of outliers. The SDF term L_{sdf} encourages transformed source points to lie on the target zero-level set, thus aligning the two implicit surfaces. The rotation regularization term L_{rotation} encourages **R** to remain orthogonal during optimization.

3.4. Fusion

After fine registration, all poses are aligned in a common coordinate system, yielding a unified representation of the object. Since each pose is initially reconstructed independently, we can use the aligned multi-pose data to retrain a single implicit function from all available input images. Specifically, we transform the camera extrinsics of all views into the target coordinate frame and jointly train one neural implicit surface using the merged multi-view image set.

This fusion step is optional. For objects with relatively simple geometry, the fused mesh obtained directly after fine registration is often sufficient. Retraining the unified implicit function is most useful for objects with fine geometric details or severe self-occlusion, where integrating complementary observations from multiple poses can further improve reconstruction fidelity.

4. Experiments

4.1. Dataset

To evaluate our multi-pose, multi-view reconstruction framework, we construct a dataset comprising 18 3D objects, including 11 synthetic models and 7 real-world captures. The objects exhibit complex shapes and severe self-occlusions, covering both controlled virtual settings and practical acquisition scenarios. Following a consistent capture protocol, each object is recorded under three different poses, with 30 to 40 multi-view images per pose captured along an upper-hemisphere trajectory. Fig. 6 provides an overview of the dataset and the corresponding fusion results.

For the synthetic data, we use BlenderNeRF[1] together with Blender to generate high-quality RGB images and ground-truth camera parameters. For the real-world data, we capture multi-view, multi-pose images using an *iPhone*. Since our method does not estimate or optimize object scale, scale consistency across poses is required for accurate registration. We therefore calibrate camera intrinsics and extrinsics using a 1 m × 1 m ArUco marker board rather than relying solely on COLMAP [27,28]. Given the known physical dimensions of the marker board, the recovered camera parameters share a common metric scale across all poses, eliminating the unit-scale discrepancy that would otherwise arise between independently reconstructed poses. As a result, all per-pose reconstructions are represented in the same coordinate system, which is a prerequisite for accurate registration.

4.2. Implementation details and runtime performance

We implement PoseFusion using PyTorch and conduct all experiments on a single NVIDIA RTX 4090 GPU. While PoseFusion is compatible with any NeRF-based method that supports SDF learning, such as NeuS [2], VolSDF [3], Neuralangelo [29], and HF-NeuS [7], we adopt Voxurf [4] as our backbone due to its efficiency and ease of use.

For all experiments, we uniformly sample $N_o = 100{,}000$ points on the OBB surfaces of the input shapes. For zero-level set operations, we use a threshold of 1×10^{-3} to determine convergence during snapping. For feature matching, we use the pre-trained eLoFTR model without additional fine-tuning. The optimization is performed using the Adam optimizer with a learning rate of 1×10^{-4} and a total of 5000 iterations.

We report the computational cost of the Voxurf-based PoseFusion implementation. The dominant cost arises from the SDF reconstruction and fusion retraining stages, each of which takes approximately 30 min. In contrast, the registration stages are relatively lightweight: coarse registration takes 3.82 min on average, while fine registration takes 0.73 min on average. Overall, the complete pipeline requires approximately 65 min, with similar runtimes for both synthetic and real-world cases (see Table 3). During registration, the peak GPU memory usage is about 8 GB on a single NVIDIA RTX 4090, leaving sufficient headroom for deployment on a standard high-end GPU.

[1] https://github.com/maximeraafat/BlenderNeRF.

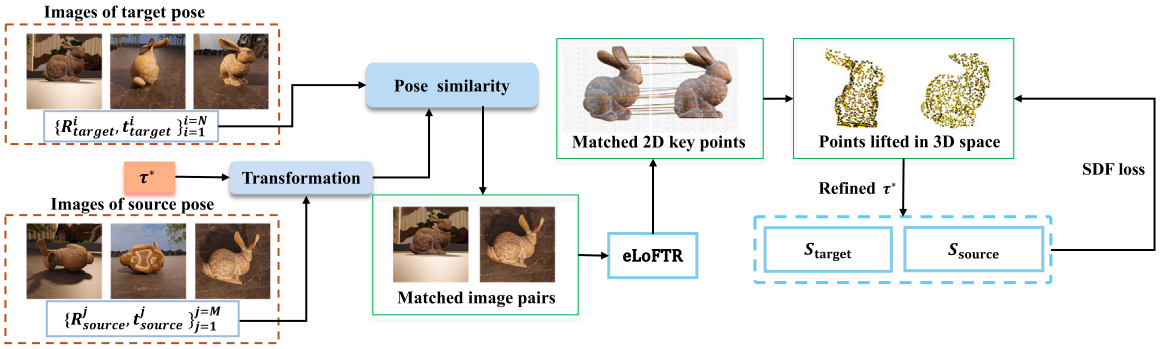

Fig. 5. Overview of the fine registration stage. This stage refines pose alignment by combining 2D keypoint correspondences extracted by eLoFTR with SDF-based geometric constraints. The coarse transformation is first applied to the source camera extrinsics, and candidate source–target image pairs are selected according to camera-pose similarity. eLoFTR then extracts 2D correspondences from the selected image pairs. These correspondences are lifted to 3D by tracing along camera rays and snapping to the zero-level set of the learned SDFs. Finally, the source and target SDFs, S_{source} and S_{target}, are used to compute consistency losses that refine the inter-pose transformation.

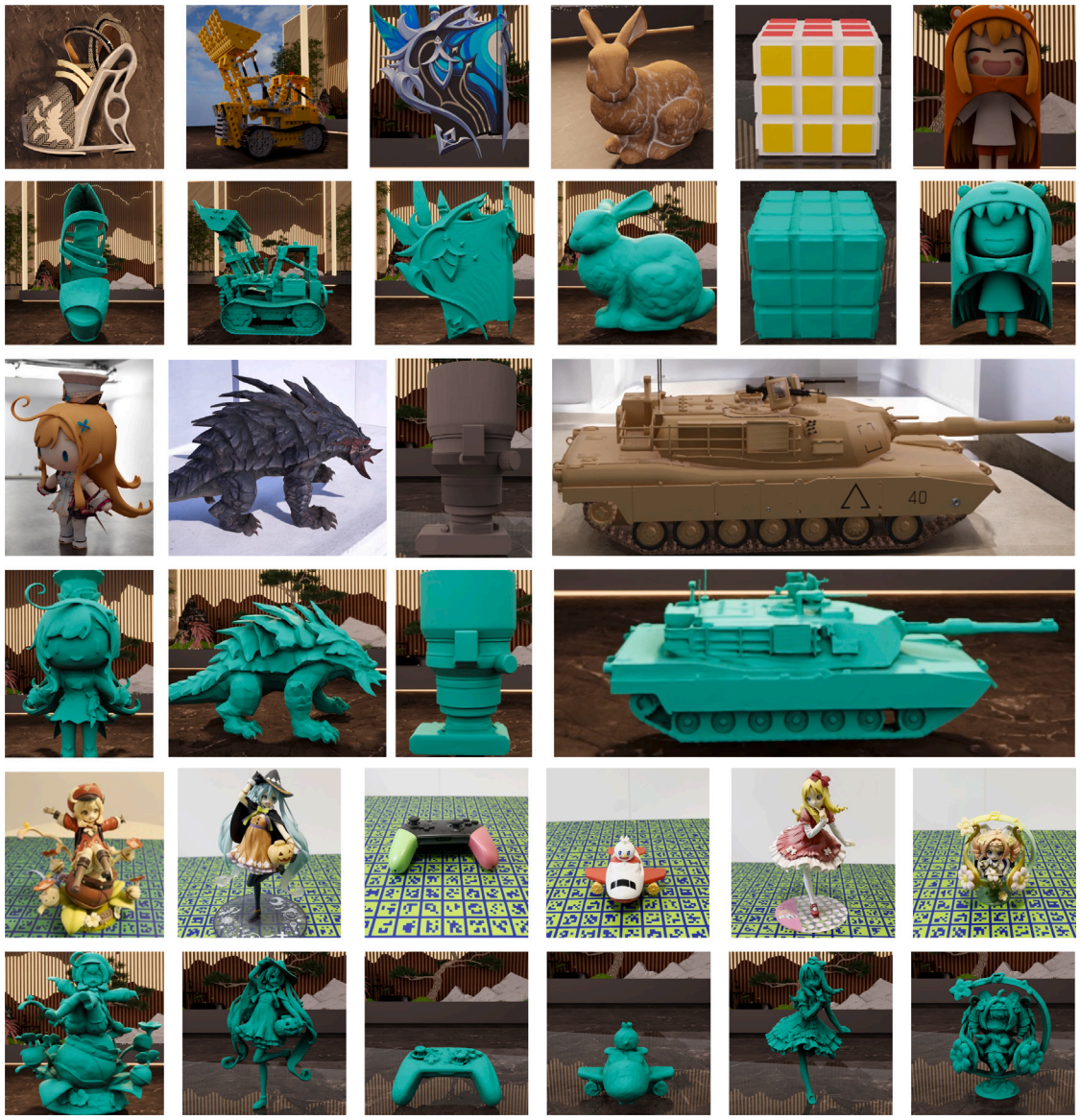

Fig. 6. Overview of the dataset and PoseFusion results. Our dataset consists of 11 synthetic models and 7 real-world captures; the real-world captures are calibrated using a 1 m × 1 m ArUco marker board. For each object, we show a representative input image and a rendered view of the reconstructed 3D surface. These results demonstrate that PoseFusion effectively aligns and fuses pose-wise neural implicit surfaces across significantly different object poses. Two additional dataset models are shown in Fig. 10.

Table 2
Ablation study of the proposed registration pipeline. We report quantitative registration results for two source-to-target pairs, (source$_1$, target) and (source$_2$, target), with each entry formatted as source$_1$/source$_2$. "–" denotes failure cases. Removing fine registration significantly reduces accuracy, while removing coarse registration causes the pipeline to fail across all evaluated objects.

Metric	Method	Bunny	Book	Nikon	Heel Slipper	Figurine A	Tank	Monster	Figurine B	Rubik's Cube	Lego
R (°, ↓)	w/o coarse reg.	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–
	w/o fine reg.	6.72/8.58	14.76/10.66	1.80/1.82	10.25/4.43	5.85/1.44	0.98/1.61	3.47/1.12	8.82/10.49	**0.04**/0.20	0.22/0.74
	RANSAC + fine reg.	0.15/2.09	0.12/–	–/–	0.13/1.66	0.07/0.25	0.16/0.23	–/0.17	0.06/0.22	–/–	–/–
	Coarse reg. + HLoc	3.29/1.47	14.12/9.99	13.99/1.94	3.49/4.58	4.70/0.90	1.43/1.20	0.49/3.45	5.88/7.99	0.09/0.15	–/–
	Full model	**0.09/0.02**	**0.11/0.18**	**0.09/0.11**	**0.12/0.07**	**0.01/0.03**	**0.14/0.09**	**0.03/0.03**	**0.04/0.06**	0.08/**0.11**	**0.05/0.20**
T($\times 10^2$) ↓	w/o coarse reg.	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–	–/–
	w/o fine reg.	1.19/3.98	8.05/5.86	1.31/1.81	3.25/3.82	4.49/1.29	6.98/8.00	2.39/1.78	4.61/12.56	0.55/0.14	4.16/2.79
	RANSAC + fine reg.	**0.01**/0.46	1.62/–	–/–	0.09/0.50	0.13/0.25	**0.11/0.07**	–/0.43	0.04/**0.05**	–/–	–/–
	Coarse reg. + HLoc	0.60/0.14	3.46/3.14	0.48/**0.13**	1.06/1.45	2.70/0.49	2.10/0.59	1.21/3.53	1.03/0.34	–/–	–/–
	Full model	0.10/**0.02**	**0.06/0.03**	**0.07**/0.14	**0.03/0.02**	**0.01/0.02**	0.13/0.10	**0.03/0.02**	**0.03**/0.06	**0.14/0.10**	**0.07/0.18**

Table 3
Running time of the coarse and fine registration stages. Time is measured in minutes on a single NVIDIA RTX 4090 GPU. We report results separately for synthetic and real-world cases, as well as the average runtime across both settings.

Category	Coarse registration	Fine registration
Synthetic	4.42	0.63
Real	3.02	0.85
Average	3.82	0.73

Table 4
Quantitative comparison under different Top-k candidate settings. Each row reports the mean rotation and translation errors over all test cases. The configuration with $k = 10$ achieves the lowest errors, indicating the best overall registration accuracy.

Top-k setting	**R** (°, ↓)	**T** ($\times 10^2$, ↓)
$k = 1$	0.9325	0.3956
$k = 3$	0.4106	0.1944
$k = 5$	0.4493	0.1933
$k =$ **10**	**0.1373**	**0.1533**

4.3. Results and comparisons

In the following, we first validate PoseFusion on synthetic multi-pose benchmarks to assess its robustness under controlled pose variations and large baselines. We then evaluate our method on real captured multi-pose sequences, comparing against state-of-the-art neural-scene registration approaches, including DReg-NeRF [18], Reg-NF [19], and GaussReg [21], in terms of both registration accuracy and reconstruction quality.

We conducted comparative experiments with two state-of-the-art neural registration methods: DReg-NeRF [18] and Reg-NF [19]. We do not compare with nerf2nerf [17] as it requires manual operation. Qualitative results are shown in Figs. 7, and 8 shows the detailed result of PoseFusion, compared to the result of the single-view reconstruction. Table 1 provides a quantitative evaluation of the synthetic dataset, further confirming the strength and accuracy of our approach. As shown in the results, our method consistently achieves the most accurate and stable performance across all test cases.

DReg-NeRF does not align the three poses due to the significant differences between them. In the initial reconstruction stage, the limited view coverage in our dataset leads to an occupancy grid with many outliers. These inaccuracies in the reconstructed geometry significantly reduce the quality of the following registration process, ultimately leading to failure.

Reg-NF performs reasonably well on simple objects, such as *Bunny* and *Book*, but struggles with challenging cases involving severe self-occlusion, such as *Heel Slipper* and *Monster*. This is because the method samples only on a certain area for each pose. When two sampling areas are hardly overlapped (which is common in models with severe self-occlusion), it fails to align the three poses correctly. GaussReg also fails in these challenging cases because it is mainly designed for registration of indoor scenes, which exhibit a large amount of planar geometries (e.g., walls and furniture) and its geometry transformer is not suitable for object level cases, failing to identify sufficient matching feature point pairs of objects.

Further analysis shows that the failure cases of Reg-NF are mainly due to its reliance on coarse registration initialization. When the initial alignment is inaccurate, the method tends to get stuck in local minima, leading to significant registration errors. This issue is clear in the final results, such as the misalignment of the camera lens and the cloak of the figurine. Despite these challenges, our method shows strength in handling objects with large pose variations, complex shapes, and severe self-occlusions, which are situations where other neural registration methods often struggle.

In addition to the registration accuracy reported above, we further evaluate the final reconstruction quality from both geometric and rendering perspectives. Table 6 reports compact geometry metrics on ten representative cases, including bidirectional Chamfer distance (CD_{bi}), bidirectional normal consistency (NC_{bi}), and F-score under multiple thresholds. The fused result achieves the best CD_{bi} and NC_{bi} on most objects and maintains strong F-score performance across the evaluated thresholds. This indicates that the proposed registration-and-fusion pipeline does not merely align pose-wise neural surfaces, but also improves the completeness and geometric consistency of the final reconstructed surface. Table 7 further compares the rendering quality of the fused result against the independently reconstructed single-pose results using PSNR and SSIM. Across all listed objects, the fusion result obtains the best PSNR and SSIM, while the strongest single-pose baseline varies from case to case. This trend confirms that the improvement comes from integrating complementary observations from multiple poses rather than relying on one favorable pose. The advantage is especially meaningful for objects with severe self-occlusion or pose-dependent visibility, where different individual poses miss different regions and the fused model benefits from the union of visible surface evidence. These additional experiments further demonstrate that PoseFusion maintains stable registration and fusion performance across both synthetic and real-world multi-pose reconstruction settings.

Table 5 summarizes the qualitative differences among several recent registration methods. We categorize them based on six criteria: whether manual initialization is required, the use of pose supervision, the level of registration (object or scene), the type of dataset used, the registration target (e.g., camera poses or surface geometry), and the input modality. As shown in this table, we quantitatively compare with DReg-NeRF [18], Reg-NF [19], and GaussReg [21] because these methods are the closest to our setting in terms of multi-view neural-scene registration. While RelPose [13], PoseDiffusion [12], and Cameras-as-Rays [14] focus on estimating object-level camera poses from unposed RGB inputs, REGTR [16] and FGR [15] target geometric alignment using point clouds. NeRFuser [30], VF-NeRF [31], and F2M-Reg [32]

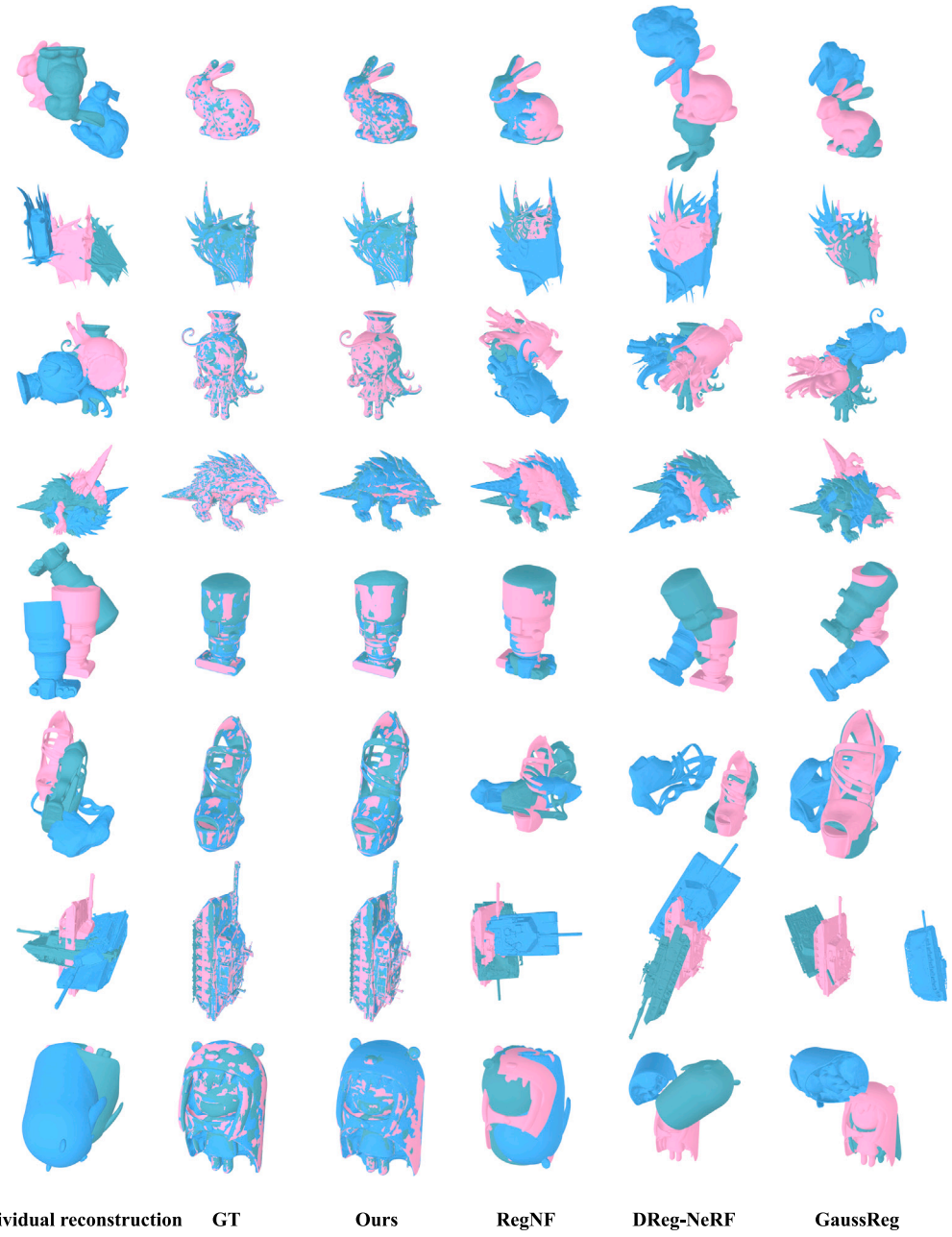

Fig. 7. Qualitative registration comparison with DReg-NeRF, RegNF, and GaussReg. Each row shows the individual pose-wise reconstructions, ground truth, our aligned result, and baseline results. Colors indicate reconstructions from different object poses. The compared objects exhibit large pose variation, complex topology, partial symmetry, and strong self-occlusion. In challenging cases such as Heel Slipper, Reg-NF can converge to an almost 180° misalignment, while DReg-NeRF fails to correctly align the poses. In contrast, PoseFusion successfully aligns all three poses and produces coherent, complete reconstructions. Since nerf2nerf requires manually annotated keypoints, it is not directly applicable to our fully automatic multi-pose setting and is omitted from this comparison. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

further push registration directly into the NeRF space by aligning implicit radiance fields or jointly optimizing camera poses from RGB(-D) sequences, while recent GS-based methods such as GaussReg [21] and RegGS [33] perform scene-level registration in the space of 3D Gaussian primitives. In contrast, PoseFusion operates at the object level on SDF-based neural surfaces and targets multi-pose fusion under large pose changes, thereby bridging traditional raw-sensor registration and neural-scene registration.

4.4. Ablation studies

To evaluate the necessity of each registration stage, we conduct ablation studies to disentangle the contributions of SDF-based coarse registration, OBB alignment, and the fine registration stage that jointly optimizes SDF consistency and image features.

Quantitative results are reported in Table 2. Removing the fine registration module (w/o fine reg.) significantly reduces accuracy across all cases, especially on the *Book* and *Nikon* objects. In contrast, removing the coarse registration stage (w/o coarse reg.) resulted in complete registration failure across all evaluated objects, as indicated by the "All failed" row in the table. We pick up one case as an example, shown in Fig. 9. This demonstrates the critical role of coarse alignment in providing a sufficiently close initialization for subsequent fine registration to succeed. We replace our coarse registration stage with RANSAC (RANSAC + fine reg.) and it fails on certain cases because

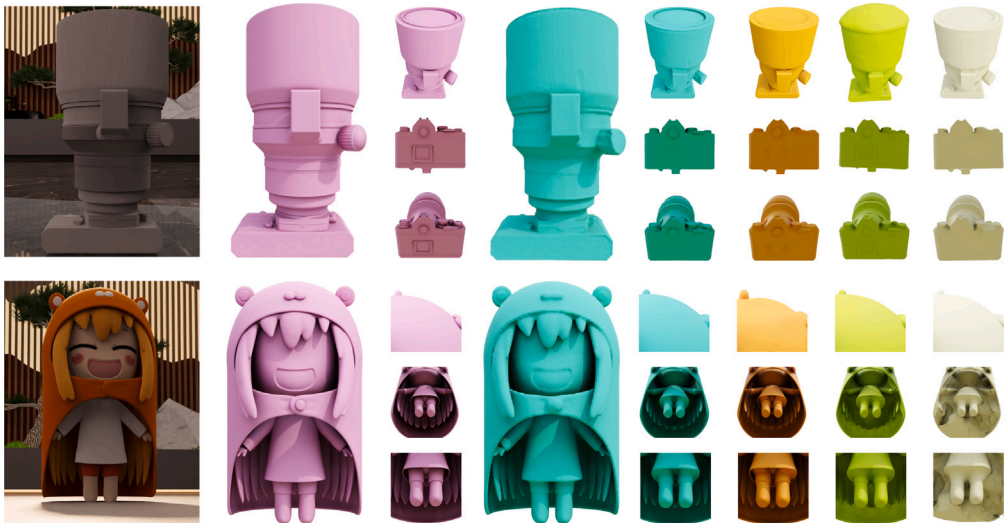

Fig. 8. Visual comparison of the final fused geometry and individual pose-wise reconstructions. Using the transformations estimated by PoseFusion, we express the camera extrinsics of all input views in the coordinate frame of the target pose. We then merge the multi-view images from all poses and jointly train a single neural implicit surface. Panel (a) shows a representative input image from the upright orientation; panel (b) shows the ground-truth mesh; panel (c) presents the final fused reconstruction; and panel (d) shows reconstructions trained from individual poses. The highlighted regions show details that are missing or distorted in individual reconstructions due to incomplete view coverage. In contrast, the fused reconstruction recovers these regions with improved geometric completeness and surface detail. Colors indicate reconstruction results from different individual poses. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 5
Qualitative comparison of recent registration methods. We compare each method by input modality, initialization requirement, supervision setting, application scenario, dataset type, and registration target. "R" and "S" indicate real captures and synthetic inputs, respectively.

Method	Input	Initialization	Registration	Type	Target	Dataset
RelPose [13]	Single-pose multi-view RGB images	Auto	Supervised	Object	Camera poses	R & S
PoseDiffusion [12]	Single-pose multi-view RGB images	Auto	Supervised	Object	Camera poses	R & S
NeRFuser [30]	Neural fields	Auto	Unsupervised	Scene	Neural fields	R & S
VF-NeRF [31]	Neural fields	Auto	Unsupervised	Scene	Neural fields	R & S
Cameras as Rays [14]	Single-pose multi-view RGB images	Auto	Supervised	Object	Camera poses	R & S
REGTR [16]	Point clouds	Auto	Supervised	Scene	Point clouds	R & S
FGR [15]	Point clouds	Auto	Unsupervised	Scene	Point clouds	R & S
F2M-Reg [32]	Single-pose multi-view RGB-D images	Auto	Unsupervised	Scene	Camera poses	R & S
RegGS [33]	Single-pose multi-view RGB images	Auto	Unsupervised	Scene	3D Gaussians	R & S
GaussReg [21]	Multi-pose multi-view RGB images	Auto	Supervised	Scene	3D Gaussians	R & S
nerf2nerf [17]	Multi-pose multi-view RGB images	Manual	Unsupervised	Object	Neural fields	R & S
DReg-NeRF [18]	Multi-pose multi-view RGB images	Auto	Supervised	Object	Neural fields	S
Reg-NF [19]	Multi-pose multi-view RGB images	Auto	Unsupervised	Object	Neural fields	S
PoseFusion (ours)	Multi-pose multi-view RGB images	Auto	Unsupervised	Object	Neural fields	R & S

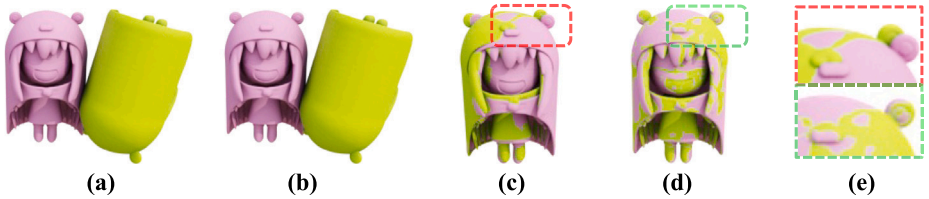

Fig. 9. Ablation study of the coarse and fine registration stages. (a) Individual pose-wise reconstructions before registration. (b) Fine registration without coarse initialization fails to converge. (c) Coarse registration alone roughly aligns the poses but leaves noticeable residual misalignment. (d) The full PoseFusion model accurately registers the objects. (e) Close-up views comparing the coarse-only and full-model results, highlighting the improvement brought by fine registration.

of lacking image feature information. We ablate the fine registration by replacing our eLoFTR-based correspondence lifting with Hloc (SuperPoint + SuperGlue) while keeping a loss-based optimization(Coarse reg + HLoc). Matched pixels are lifted to 3D via differentiable sphere tracing on our SDF; we then minimize a weighted robust reprojection loss optionally combined with zero-level SDF consistency and depth visibility terms. This keeps the optimization in the same family as ours, isolating the effect of the feature/matcher choice rather than switching to a closed-form pipeline.

We further analyze the effect of the Top-k matched image pairs used as input to eLoFTR in the fine registration stage. As shown in Table 4, registration accuracy improves rapidly with k and is best at $k = 10$, which we adopt as the default; smaller budgets ($k \in \{1, 3, 5\}$) leave too few valid 2D correspondences after zero-level-set lifting and

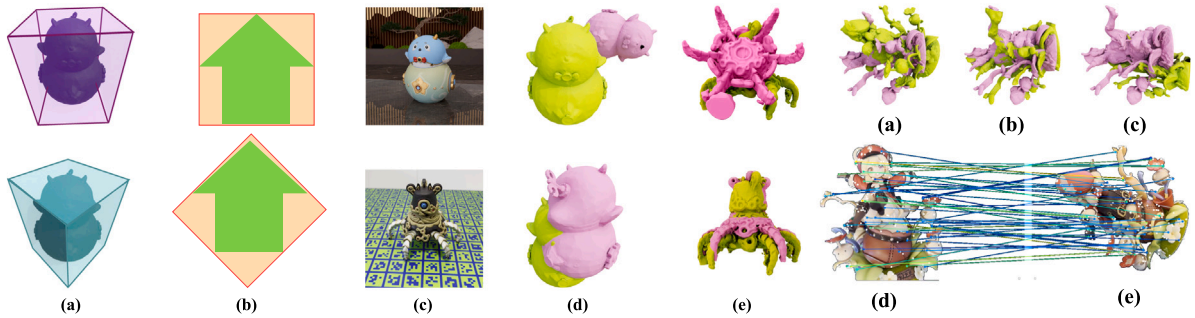

Fig. 10. Failure cases. Left sub-figure shows our method fails due to ambiguous object geometry with multiple plausible OBBs. Column (a): The object can be enclosed by two spatially distinct OBBs. Column (b): 2D illustration of this ambiguity. Column (c): Input images of failure cases, including one real scan and one synthetic object. Column (d & e): Even after coarse registration, large rotational misalignment (over 30 degrees) remains between the two poses. Right sub-figure shows the failure case caused by low view overlap. (a) draws the individual reconstruction results before registration. (b) shows the coarse registration output. (c) illustrates the failure of the fine registration stage. (d) and (e) highlight incorrect correspondences predicted by eLoFTR due to the lack of overlapping regions.

Table 6
Quantitative evaluation of reconstructed mesh quality on ten representative objects. We report bidirectional Chamfer distance CD_{bi}, shown in $\times 10^{-3}$, bidirectional normal consistency NC_{bi}, and F-score at thresholds 0.005, 0.01, and 0.02. For each object and metric, the best result among **Fusion**, **Pose-1**, **Pose-2**, and **Pose-3** is highlighted in boldface, and the second-best result is underlined. The fused results consistently outperform those obtained from individual poses.

Setting	Metric	Bunny	Book	Nikon	Heel Slipper	Figurine A	Tank	Monster	Figurine B	Rubik's Cube	Lego
Fusion	CD_{bi} ($\times 10^{-3}$, ↓)	**0.438**	**5.035**	**4.376**	6.088	**2.882**	**8.356**	**1.468**	11.131	**3.546**	**8.775**
	NC_{bi} (↑)	**0.9921**	**0.8872**	**0.8944**	**0.9037**	**0.9281**	**0.7944**	**0.9255**	**0.8797**	**0.9541**	**0.6559**
	$F_{0.005}$ (↑)	**0.9998**	**0.7988**	**0.7594**	**0.7029**	**0.8715**	**0.6545**	**0.9336**	0.5733	**0.7302**	**0.4345**
	$F_{0.01}$ (↑)	**1.0000**	**0.8420**	**0.8115**	**0.8334**	**0.9166**	**0.7889**	**0.9670**	**0.7636**	**0.9136**	**0.7014**
	$F_{0.02}$ (↑)	**1.0000**	**0.8827**	**0.9414**	0.9223	**0.9559**	**0.8967**	**0.9923**	0.8192	**1.0000**	**0.8752**
Pose-1	CD_{bi} ($\times 10^{-3}$, ↓)	1.077	5.937	5.521	6.988	3.177	11.736	2.390	10.631	7.583	11.033
	NC_{bi} (↑)	0.9872	0.8633	0.8684	0.8759	0.9242	0.7687	0.9092	0.8683	0.9446	0.6415
	$F_{0.005}$ (↑)	0.9805	0.7025	0.6802	0.6550	0.8490	0.5708	0.8717	0.5495	0.6380	0.3818
	$F_{0.01}$ (↑)	0.9976	0.8176	0.7678	0.7937	0.9035	0.7055	0.9340	0.7398	0.8271	0.6217
	$F_{0.02}$ (↑)	0.9995	0.8787	0.9223	0.9009	0.9541	0.8332	0.9769	0.8088	0.9218	0.8256
Pose-2	CD_{bi} ($\times 10^{-3}$, ↓)	3.688	7.027	4.919	**5.746**	3.597	18.776	2.423	14.085	8.388	11.774
	NC_{bi} (↑)	0.9474	0.8560	0.8862	0.8976	0.9136	0.7481	0.9058	0.8684	0.9436	0.6402
	$F_{0.005}$ (↑)	0.8891	0.7330	0.7061	0.6892	0.8264	0.5103	0.8717	0.4948	0.6046	0.3918
	$F_{0.01}$ (↑)	0.9076	0.7865	0.7910	0.8319	0.8815	0.6150	0.9337	0.7051	0.7946	0.6172
	$F_{0.02}$ (↑)	0.9374	0.8423	0.9383	**0.9225**	0.9444	0.7134	0.9799	0.7561	0.8985	0.8131
Pose-3	CD_{bi} ($\times 10^{-3}$, ↓)	2.867	6.383	9.070	5.918	3.139	10.619	2.583	**9.898**	7.390	10.612
	NC_{bi} (↑)	0.9608	0.8628	0.8451	0.8997	0.9211	0.7749	0.9019	0.8727	0.9406	0.6357
	$F_{0.005}$ (↑)	0.9196	0.6871	0.6328	0.6660	0.8494	0.5871	0.8537	**0.5758**	0.6499	0.4006
	$F_{0.01}$ (↑)	0.9399	0.8177	0.6958	0.8038	0.9066	0.7254	0.9222	0.7593	0.8427	0.6499
	$F_{0.02}$ (↑)	0.9573	0.8804	0.8388	0.9141	0.9553	0.8438	0.9779	**0.8287**	0.9228	0.8416

produce noticeably larger rotation and translation errors. Finally, we examine the optional retraining stage of the fusion module. Table 8 compares the full pipeline (*Fusion*, with retraining) against *Fusion (no retraining)*, a purely explicit alternative in which the three registered per-pose meshes are concatenated into a single multi-component mesh, with vertices and any resulting degenerate triangles merged at a tolerance of 5×10^{-5} of the object's bounding-box diagonal. Both variants substantially outperform the best single-pose reconstruction on CD_{bi}, NC_{bi}, and F-scores, and retraining further improves over direct merging on most objects

4.5. Failure cases

Although our pipeline is robust for most object registrations, it may still fail in certain challenging scenarios. We summarize two representative failure cases as follows. The first case is multiple plausible bounding boxes. Registration may also fail when the object has multiple plausible OBBs with significantly different spatial orientations. As shown in the left part of Fig. 10(a)–(b), the object can be enclosed by two valid OBBs located in distinct spatial regions, leading to failure in the coarse registration stage. (d)–(e) illustrate this ambiguity in 2D for better visualization, while (c) and (d) show the initial state and the coarse registration result. We observe that although the object centers roughly align after coarse registration, the relative rotation between poses remains significantly large (exceeding 120°).

The other case is insufficient view coverage When the coverage between two object captures is too low, registration becomes infeasible. In such cases, matched image pairs may exhibit large viewpoint differences, resulting in a lack of reliable correspondences. Consequently, feature matching via local descriptors (e.g., eLoFTR) often leads to incorrect matches. As shown in the right part of Fig. 10, pose 1 captures the front side of the object, while pose 2 captures the back side, leaving almost no overlapping regions. Nevertheless, eLoFTR erroneously predicts matches between unrelated regions. It is important to note that large pose differences are acceptable, but at least one part of the object must appear in both poses (e.g., the head of a figurine) for successful registration.

5. Conclusion

We presented PoseFusion, a two-stage pipeline for registering and fusing multiple neural implicit surfaces of the same object captured under different poses. In the first stage, PoseFusion uses oriented bounding boxes as geometric proxies and incorporates both SDF samples and

Table 7
Rendering-quality comparison on representative cases. The fused results consistently outperform those obtained from individual poses, indicating that joint fusion improves rendering quality. For each object and metric, the best result among **Fusion**, **Pose-1**, **Pose-2**, and **Pose-3** is shown in boldface, and the second-best is underlined.

(a) Synthetic data.

Setting	Metric	Bunny	Book	Nikon	Heel Slipper	Figurine A	Tank	Monster	Figurine B	Rubik's Cube	Lego
Fusion	PSNR ↑	**28.4440**	**32.1437**	**33.9464**	**26.9313**	**30.5287**	**29.7910**	**31.7125**	**31.5415**	**34.4311**	**30.9647**
	SSIM ↑	**0.9382**	**0.9690**	**0.9587**	**0.9291**	**0.9642**	**0.9395**	**0.9354**	**0.9577**	**0.9703**	**0.9385**
Pose-1	PSNR ↑	28.3997	21.9916	24.9302	21.0697	24.1770	21.2619	24.0936	25.6936	23.9110	25.1429
	SSIM ↑	0.9244	0.8751	0.8898	0.8209	0.9179	0.8314	0.8473	0.9069	0.8749	0.8848
Pose-2	PSNR ↑	27.8412	24.4882	23.3294	21.8896	25.3104	20.6126	23.8719	26.0259	21.2495	22.8247
	SSIM ↑	0.9235	0.9099	0.8813	0.8701	0.9302	0.8159	0.8572	0.9249	0.8496	0.8660
Pose-3	PSNR ↑	27.8621	23.3804	24.7849	21.2658	22.5920	21.8648	23.5052	26.0529	25.7846	23.6938
	SSIM ↑	0.9125	0.8910	0.8669	0.8376	0.8972	0.8526	0.8435	0.9147	0.9145	0.8662

(b) Real world data.

Setting	Metric	Figurine C	Figurine D	Figurine E	Figurine F	Joystick	Airplane
Fusion	PSNR ↑	**25.8292**	**28.5216**	**29.8639**	**28.7642**	**24.0179**	**29.1024**
	SSIM ↑	**0.9254**	**0.9485**	**0.9609**	**0.9518**	**0.8724**	**0.9561**
Pose-1	PSNR ↑	21.5096	22.7843	22.1113	22.8398	21.9449	15.9110
	SSIM ↑	0.8725	0.9052	0.9092	0.9085	0.8382	0.8859
Pose-2	PSNR ↑	20.6875	23.4906	21.6551	23.3337	21.7443	24.8296
	SSIM ↑	0.8633	0.8952	0.8908	0.9154	0.8476	0.9293
Pose-3	PSNR ↑	21.8500	22.7877	24.0003	21.9944	22.0086	25.2358
	SSIM ↑	0.8717	0.8875	0.9103	0.9033	0.8390	0.9266

Table 8
Ablation study of the fusion retraining stage. "Direct fusion" concatenates the registered pose-wise surface observations without retraining, while "usion retraining" learns one unified implicit surface from all aligned multi-pose images. We report CD_{bi} in $\times 10^{-3}$, NC_{bi}, and F-score at multiple thresholds. Lower values are better for CD_{bi}, while higher values are better for NC_{bi} and F-scores. All entries have $P_f^{\text{valid}} = 1.0000$.

	Bunny	Book	Nikon	Heel Slipper	Figurine A	Tank	Monster	Figurine B	Rubik's Cube	Lego	Avg.
CD_{bi} ↓ (*Direct fusion*)	1.856	6.014	5.341	**4.677**	**2.858**	9.273	11.235	**8.071**	5.371	**8.412**	6.311
CD_{bi} ↓ (*Fusion retraining*)	**0.438**	**5.035**	**4.376**	6.088	2.882	**8.356**	**1.468**	11.131	**3.546**	8.775	**5.204**
NC_{bi} ↑ (*Direct fusion*)	0.9755	0.8645	0.8782	**0.9054**	0.9256	0.7836	0.8395	0.8750	0.9403	0.6492	0.8637
NC_{bi} ↑ (*Fusion retraining*)	**0.9921**	**0.8872**	**0.8944**	0.9037	**0.9281**	**0.7944**	**0.9255**	**0.8797**	**0.9541**	**0.6559**	**0.8816**
$F_{0.005}$ ↑ (*Direct fusion*)	0.9564	0.7592	0.7352	**0.7470**	**0.8765**	**0.6849**	0.7532	**0.6388**	0.7122	**0.5107**	0.7374
$F_{0.005}$ ↑ (*Fusion retraining*)	**0.9998**	**0.7988**	**0.7594**	0.7029	0.8715	0.6545	**0.9336**	0.5733	**0.7302**	0.4345	**0.7458**
$F_{0.01}$ ↑ (*Direct fusion*)	0.9667	0.8221	0.7973	**0.8611**	**0.9172**	**0.7969**	0.8108	**0.7880**	0.8883	**0.7461**	0.8395
$F_{0.01}$ ↑ (*Fusion retraining*)	**1.0000**	**0.8420**	**0.8115**	0.8334	0.9166	0.7889	**0.9670**	0.7636	**0.9136**	0.7014	**0.8538**
$F_{0.02}$ ↑ (*Direct fusion*)	0.9759	0.8723	0.9281	**0.9354**	0.9557	0.8837	0.8625	**0.8505**	0.9555	**0.8960**	0.9116
$F_{0.02}$ ↑ (*Fusion retraining*)	**1.0000**	**0.8827**	**0.9414**	0.9223	**0.9559**	**0.8967**	**0.9923**	0.8192	**1.0000**	0.8752	**0.9286**
$F_{0.05}$ ↑ (*Direct fusion*)	0.9900	0.9999	0.9839	**0.9965**	**0.9988**	0.9550	0.9238	**0.9675**	0.9849	**0.9835**	0.9784
$F_{0.05}$ ↑ (*Fusion retraining*)	**1.0000**	**1.0000**	**0.9916**	0.9736	0.9977	**0.9661**	**1.0000**	0.9126	**1.0000**	0.9814	**0.9823**

image features to estimate a coarse yet reliable initial alignment. This initialization reduces the risk of converging to poor local minima, especially under large rotational differences between poses. In the second stage, PoseFusion refines the alignment using image correspondences lifted to the zero-level sets of the learned SDFs, yielding more accurate final registration. Experiments on both synthetic and real-world objects demonstrate that PoseFusion effectively fuses partial pose-wise reconstructions into coherent and complete surface models.

Limitations. The current implementation does not explicitly handle photometric inconsistencies such as shadows, specular highlights, or view-dependent shading during pose fusion. Even under stable lighting, these effects can cause the same surface region to exhibit different appearances across poses, leading to pose-dependent reconstruction differences. This remains a common challenge for both NeRF and 3DGS-based methods. Recent work addresses this issue by modeling BRDFs, view-dependent appearance, or relighting [34–36]. Integrating such techniques into PoseFusion could improve photometric consistency and visual realism, which we plan to explore in future work.

CRediT authorship contribution statement

Guanli Hou: Methodology, Implementation, Experiments, Formal Analysis, Visualization, Writing – Original Draft. **Yuanmu Xu:** Methodology, Implementation, Experiments, Formal Analysis, Visualization, Writing – Original Draft. **Tenglong Ren:** Experiments, Visualization. **Jiangbei Hu:** Writing – review & editing. **Fei Hou:** Writing – review & editing. **Peng Song:** Writing – review & editing. **Ying He:** Methodology, Writing – review & editing, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This project was partially supported by the Ministry of Education, Singapore, under its Academic Research Fund Grant (RT19/22) and the AcRF Tier 2 Grant (MOET2EP20222-0008).

Data availability

Research Link Provided

Dataset of PoseFusion (Original data) (PoseFusion)

References

[1] Mildenhall B, Srinivasan PP, Tancik M, Barron JT, Ramamoorthi R, Ng R. Nerf: Representing scenes as neural radiance fields for view synthesis. Commun ACM 2021;65(1):99–106.

[2] Wang P, Liu L, Liu Y, Theobalt C, Komura T, Wang W. NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. Adv Neural Inf Process Syst 2021;34:27171–83.

[3] Yariv L, Gu J, Kasten Y, Lipman Y. Volume rendering of neural implicit surfaces. In: Proceedings of the 35th international conference on neural information processing systems. NIPS '21, Curran Associates Inc.; 2021.

[4] Wu T, Wang J, Pan X, Xu X, Theobalt C, Liu Z, Lin D. Voxurf: Voxel-based efficient and accurate neural surface reconstruction. In: International conference on learning representations. ICLR, 2023.

[5] Wang Y, Han Q, Habermann M, Daniilidis K, Theobalt C, Liu L. NeuS2: Fast learning of neural implicit surfaces for multi-view reconstruction. In: 2023 IEEE/CVF international conference on computer vision. ICCV, 2023, p. 3272–83.

[6] Fu Q, Xu Q, Ong YS, Tao W. Geo-neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. Adv Neural Inf Process Syst 2022;35:3403–16.

[7] Wang Y, Skorokhodov I, Wonka P. Hf-neus: Improved surface reconstruction using high-frequency details. Adv Neural Inf Process Syst 2022;35:1966–78.

[8] Long X, Lin C, Wang P, Komura T, Wang W. Sparseneus: Fast generalizable neural surface reconstruction from sparse views. In: European conference on computer vision. Springer; 2022, p. 210–27.

[9] Li J, Zhang L, Hu J, Zhang Z, Sun H, Song G, He Y. Real-time volume rendering with octree-based implicit surface representation. Comput Aided Geom Design 2024;111:102322.

[10] Li J, Wen Z, Zhang L, Hu J, Hou F, Zhang Z, He Y. GS-Octree: Octree-based 3D Gaussian splatting for robust object-level 3D reconstruction under strong lighting. Comput Graph Forum 2024;43(7):e15206.

[11] Kerbl B, Kopanas G, Leimkühler T, Drettakis G, et al. 3d Gaussian splatting for real-time radiance field rendering. ACM Trans Graph 2023;42(4). 139.

[12] Wang J, Rupprecht C, Novotny D. Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In: Proceedings of the IEEE/CVF international conference on computer vision. 2023, p. 9773–83.

[13] Zhang JY, Ramanan D, Tulsiani S. Relpose: Predicting probabilistic relative rotation for single objects in the wild. In: European conference on computer vision. Springer; 2022, p. 592–611.

[14] Zhang J, Lin A, Kumar M, Yang T-H, Ramanan D, Tulsiani S. Cameras as rays: Pose estimation via ray diffusion. In: International conference on learning representations. vol. 2024, 2024, p. 23345–66.

[15] Zhou Q-Y, Park J, Koltun V. Fast global registration. In: European conference on computer vision. Springer; 2016, p. 766–82.

[16] Yew ZJ, Lee GH. Regtr: End-to-end point cloud correspondences with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 6677–86.

[17] Goli L, Rebain D, Sabour S, Garg A, Tagliasacchi A. nerf2nerf: Pairwise registration of neural radiance fields. In: 2023 IEEE international conference on robotics and automation. ICRA, 2023, p. 9354–61.

[18] Chen Y, Lee GH. DReg-NeRF: Deep registration for neural radiance fields. In: 2023 IEEE/CVF international conference on computer vision. ICCV, 2023, p. 22646–56.

[19] Hausler S, Hall D, Mahendren S, Moghadam P. Reg-NF: Efficient registration of implicit surfaces within neural fields. In: 2024 IEEE international conference on robotics and automation. ICRA, 2024, p. 15409–15.

[20] Besl PJ, McKay ND. A method for registration of 3-D shapes. IEEE Trans Pattern Anal Mach Intell 1992;14(2):239–56.

[21] Chang J, Xu Y, Li Y, Chen Y, Feng W, Han X. Gaussreg: Fast 3d registration with Gaussian splatting. In: European conference on computer vision. 2024, p. 407–23.

[22] Abdi H, Williams LJ. Principal component analysis. Wiley Interdiscip Rev: Comput Stat 2010;2(4):433–59.

[23] He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 770–8.

[24] Wang Y, He X, Peng S, Tan D, Zhou X. Efficient LoFTR: Semi-dense local feature matching with sparse-like speed. In: 2024 IEEE/CVF conference on computer vision and pattern recognition. CVPR, 2024, p. 21666–75.

[25] Sun J, Shen Z, Wang Y, Bao H, Zhou X. LoFTR: Detector-free local feature matching with transformers. In: 2021 IEEE/CVF conference on computer vision and pattern recognition. CVPR, 2021, p. 8918–27.

[26] Barron JT. A general and adaptive robust loss function. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 4331–9.

[27] Schonberger JL, Frahm J-M. Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. 2016, p. 4104–13.

[28] Schönberger JL, Zheng E, Frahm J-M, Pollefeys M. Pixelwise view selection for unstructured multi-view stereo. In: European conference on computer vision. Springer; 2016, p. 501–18.

[29] Li Z, Müller T, Evans A, Taylor RH, Unberath M, Liu M-Y, Lin C-H. Neuralangelo: High-fidelity neural surface reconstruction. In: 2023 IEEE/CVF conference on computer vision and pattern recognition. CVPR, 2023, p. 8456–65.

[30] Fang J, Lin S, Vasiljevic I, Guizilini V, Ambrus R, Gaidon A, Shakhnarovich G, Walter MR. NeRFuser: Large-scale scene representation by NeRF fusion. 2023, arXiv:2305.13307.

[31] Segre L, Avidan S. Vf-nerf: Viewshed fields for rigid nerf registration. In: European conference on computer vision. Springer; 2024, p. 164–81.

[32] Yu Z, Qin Z, Tang Y, Wang Y, Yi R, Zhu C, Xu K. F2M-Reg: Unsupervised RGB-D point cloud registration with frame-to-model optimization. 2024, arXiv preprint arXiv:2405.00507.

[33] Cheng C, Hu Y, Yu S, Zhao B, Wang Z, Wang H. Reggs: Unposed sparse views Gaussian splatting with 3dgs registration. In: Proceedings of the IEEE/CVF international conference on computer vision. 2025, p. 8100–9.

[34] Qiao Y-L, Gao A, Xu Y, Feng Y, Huang J-B, Lin MC. Dynamic mesh-aware radiance fields. In: Proceedings of the IEEE/CVF international conference on computer vision. 2023, p. 385–96.

[35] Nimier-David M, Dong Z, Jakob W, Kaplanyan A. Material and lighting reconstruction for complex indoor scenes with texture-space differentiable rendering. In: Eurographics symposium on rendering - DL-only track. The Eurographics Association; 2021.

[36] Gao J, Gu C, Lin Y, Li Z, Zhu H, Cao X, Zhang L, Yao Y. Relightable 3d Gaussians: Realistic point cloud relighting with brdf decomposition and ray tracing. In: European conference on computer vision. Springer; 2024, p. 73–89.