EVOREG: VERSATILE AND ROBUST POINT CLOUD REGISTRATION VIA MULTI-STAGE ALIGNMENT

Anonymous authors
Paper under double-blind review

ABSTRACT

Point cloud registration methods remain specialized along two axes: most target rigid or non-rigid alignment under either supervised or self-supervised training. To address this fragmentation, we propose EvoReg, a unified multi-stage coarse-to-fine framework that covers all four (rigid/non-rigid) $\times$ (supervised/self-supervised) settings via a single architecture through staged decoupling. Gradient-free CMA-ES pre-alignment over $\mathrm{SE}(3)$ escapes local minima and provides a pose-agnostic initialization, iterative Sinkhorn soft correspondences with confidence-weighted Kabsch refinement tighten the rigid pose, a residual MLP head corrects remaining rigid error, and a conditional VAE predicts a residual non-rigid deformation field on the rigidly-aligned source; each stage inherits and solves a progressively tighter initialization. At inference, four optional training-free modules — concentrated gradient-free $\mathrm{SE}(3)$ search, point-space diffusion denoising, and global and per-point Sinkhorn test-time optimization — combine selectively to trade compute for accuracy without retraining.

Trained exclusively on ModelNet40 and evaluated under controlled and real-world protocols spanning ModelNet40, ShapeNet-13, 3DMatch, and FAUST, EvoReg achieves competitive correspondence accuracy and the best distributional alignment among tested baselines — significantly leading them on every Chamfer Distance and Earth Mover's Distance evaluation cell and realizing perfect or near-perfect Chamfer Distance Recall — across all four aforementioned settings. These results show that staged decoupling is an effective architectural principle for unifying rigid and non-rigid point-cloud registration under both pose-supervised and geometric self-supervised training, and position EvoReg as a dynamic and robust method for controlled and real-world registration. The code for this project can be found at: `https://anonymous.4open.science/r/EvoReg-0507`.

1 INTRODUCTION

Point cloud registration, the task of aligning a source point set to a target point set, is a non-convex optimization problem. Although point set registration has been studied extensively, robust alignment remains challenging in practical settings involving noise, outliers, missing data, high-dimensional point sets, and objects whose alignment cannot be fully explained by a single rigid transformation (Maiseli et al., 2017). Under large rotations, partial overlap, and noise, the loss landscape contains many local minima, and gradient-based methods cannot recover from poor initialization. This is the core structural reason why the learned registration literature has fragmented into specialized pipelines: techniques that solve the global search problem (RANSAC-based, evolutionary) sacrifice differentiability and learned representations; techniques that use learned correspondences and differentiable alignment (DCP (Wang & Solomon, 2019a), RPMNet (Yew & Lee, 2020), GeoTransformer (Qin et al., 2023)) assume a reasonable initialization and fail when it is unavailable. Existing learned approaches rarely compose both cleanly.

A second problem resulting from the coupling of global search and local refinement is that it forces specialization along two axes. The first axis is modality: most learned techniques target either rigid registration or non-rigid deformation, since the global search strategy and the deformation model are tightly coupled in each design. Techniques such as DCP (Wang & Solomon, 2019a), RPMNet (Yew & Lee, 2020), and PointNetLK (Aoki et al., 2019) achieve strong rigid alignment but cannot be extended to handle non-rigid deformation. Techniques such as NDP (Li & Harada,

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

2022) and non-rigid CPD (Myronenko & Song, 2010) handle complex articulations but are slow and assume small rigid residuals. The second axis is the supervision regime. Most techniques either require full pose supervision or adopt self-supervised objectives targeting narrower problem settings, such as partial-to-partial alignment in PRNet (Wang & Solomon, 2019b). To our knowledge, no single learned framework operates competitively across all four combinations of (rigid/non-rigid) $\times$ (supervised/self-supervised) without architectural modification.

To this end, we propose EvoReg, a multi-stage coarse-to-fine pipeline for point cloud registration that operates across (rigid/non-rigid) $\times$ (supervised/self-supervised) settings using a single architecture and a shared training objective. Our aim with EvoReg is to decouple global search and local refinement into sequential stages, wherein each stage passes a progressively tighter initialization to the next, allowing each subsequent module to solve a progressively easier subproblem. Specifically, EvoReg first performs evolutionary gradient-free CMA-ES (Hansen, 2016) search over SE(3), providing a pose-agnostic initialization that is robust to large rotations and independent of the learned stages. This alignment is iteratively refined via Sinkhorn soft correspondences and confidence-weighted Kabsch SVD, tightening the rigid pose on a residual problem. A residual rigid MLP head then absorbs small remaining errors.

To extend beyond rigid registration, a conditional VAE predicts a residual non-rigid deformation field on top of the rigid-aligned source. Each handoff from one stage to the next can be smoothed using inter-stage evolutionary refinements with progressively contracted search radii, preventing residual error from propagating forward. The full learned pipeline is end-to-end trainable.

At inference, EvoReg provides four optional training-free refinements: a concentrated evolutionary search, point-space DDPM denoising, and global and per-point Sinkhorn test-time optimization, that can be composed selectively to trade compute for accuracy without retraining. This decouples the accuracy-compute tradeoff from the training decision, giving practitioners explicit control over both phases independently.

Trained exclusively on ModelNet40, EvoReg shows particular strength when evaluated under correspondence and distributional alignment (CDA) metrics — achieving the strongest Chamfer Distance and Earth Mover's Distance among evaluated methods, with consistently perfect or near-perfect Chamfer Distance Recall, under our evaluation protocol across ModelNet40, ShapeNet-13, 3DMatch, and FAUST. Notably, EvoReg demonstrates strong CDA cross-dataset generalization to real indoor RGB-D fragments and human body articulations. Overall, EvoReg shows that staged decoupling is an effective architectural principle for unifying rigid and non-rigid point-cloud registration under both pose-supervised and geometric self-supervised training.

Contributions. We make the following contributions. (1) We show that a staged-decoupling architecture comprising evolutionary $\mathrm{SE}(3)$ search, Sinkhorn–Kabsch refinement, residual rigid correction, conditional VAE deformation, and dynamic inference-time optimization represents a viable and effective way to unify rigid and non-rigid point cloud registration under both pose-supervised and geometric self-supervised training and across datasets and evaluation protocols. (2) We introduce EvoReg, a versatile and robust registration framework of the aforementioned type that, when trained exclusively on ModelNet40, achieves competitive correspondence accuracy and the strongest distributional alignment among tested baselines, across multiple datasets (ModelNet40, ShapeNet-13, 3DMatch, and FAUST) and evaluation protocols (controlled and real-world), without requiring dataset-specific retraining. (3) We conduct a thorough component ablation study to show what drives EvoReg's performance — helping inform future work on unified point cloud registration. (4) We conduct statistical testing on EvoReg's results to show exactly where its performance is significant and where it should not be overinterpreted.

2 Related Work

Learned Rigid Registration. Learning-based rigid registration replaces hand-crafted matching with feature-driven soft correspondences and differentiable alignment, starting with PointNet's permutation-invariant encoder (Qi et al., 2017). PointNetLK unrolls a Lucas–Kanade procedure over learned features (Aoki et al., 2019); DCP attaches attention-based soft correspondences to a differentiable SVD head (Wang & Solomon, 2019a); RPMNet handles partial overlap with learned features, annealing, and Sinkhorn normalization (Yew & Lee, 2020); PRNet self-supervises keypoint

and correspondence learning for partial-to-partial pairs (Wang & Solomon, 2019b); and DeepGMR aligns latent Gaussian mixtures end-to-end (Yuan et al., 2020).

Several works in recent years addressed the partial-overlap regime more explicitly. Predator provides an overlap-attention module that routes network capacity toward co-visible regions (Huang et al., 2021). CoFiNet estimates reliable correspondences using coarse-to-fine superpoint-to-point matching weighted by predicted overlap scores (Yu et al., 2021). PointDSC forces spatial consistency between putative correspondences by using a non-local graph network to reject outliers (Bai et al., 2021). More recently, geometric transformer-based architectures have significantly elevated the state of the art: GeoTransformer represents rotation-invariant geometric structure at the superpoint level, enabling robust local-to-global registration with strong cross-dataset generalization (Qin et al., 2023). RoITr improves on this with a rotation-invariant point proxy attention mechanism that provides better performance under low overlap (Yu et al., 2023). These methods have substantially advanced rigid registration, but they do not directly model non-rigid deformation.

Learned Non-rigid registration. Learned and transport-based approaches relax these priors: Neural Deformation Pyramid represents motion as a hierarchy of coordinate MLPs for coarse-to-fine deformation (Li & Harada, 2022), and Shen et al.'s robust optimal-transport formulation relaxes correspondence estimation for registration and shape matching (Shen et al., 2021). These methods can capture complex local deformation, but they typically assume approximate rigid pre-alignment or must absorb residual pose error into the deformation field. This coupling makes non-rigid registration sensitive to large initial misalignment and motivates separating rigid alignment from deformation modeling.

Self-supervised registration. Reducing dependence on pose supervision has motivated a line of self-supervised registration techniques. PRNet (Wang & Solomon, 2019b) learns keypoint detection and correspondence estimation for partial-to-partial rigid pairs using a self-supervised cycle-consistency objective, without requiring ground-truth transformations. However, most self-supervised registration methods focus on rigid alignment, while self-supervised non-rigid registration remains less explored because dense correspondences may be ambiguous and inverse consistency is harder to define under deformation.

More Related Work can be found in section A of the supplementary material.

EvoReg addresses this fragmentation through staged decoupling. Rather than requiring one module to solve global search, local correspondence estimation, residual rigid error, and non-rigid deformation simultaneously, EvoReg separates these roles into gradient-free SE(3) initialization, Sinkhorn–Kabsch rigid refinement, residual rigid correction, and conditional deformation modeling, allowing the same architecture to support rigid and non-rigid registration under both pose-supervised and geometric self-supervised training, with optional inference-time refinements providing a runtime–accuracy tradeoff.

3 METHOD

EvoReg aligns a source point cloud to a target through a four-stage coarse-to-fine pipeline following a single design principle: staged decoupling. Each stage inherits a progressively tighter initialization from the previous, so each module solves a progressively easier subproblem. Stage 0 provides a pose-agnostic initialization via gradient-free SE(3) search; Stage 1 refines it through iterative soft correspondences and confidence-weighted Kabsch alignment; Stage 2 corrects residual rigid error via an MLP head; and Stage 3 predicts non-rigid deformation on an already-rigidly-aligned source. Inter-stage gradient-free refinements with contracting search radii smooth each transition. At inference, EvoReg provides four optional training-free refinements: a concentrated evolutionary search, point-space DDPM denoising, and global and per-point Sinkhorn test-time optimization, that can be composed selectively to trade compute for accuracy without retraining. This decouples the accuracy-compute tradeoff from the training decision, giving practitioners explicit control over both axes independently. The full architectural pipeline is illustrated in Figure 1. Additional method details not covered in this section can be found in section B of the supplementary materials.

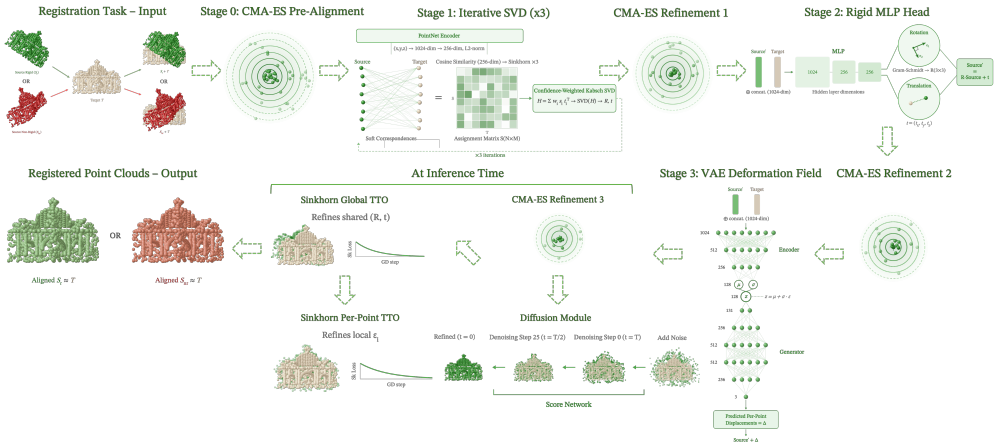


Figure 1: Overview of EvoReg's architectural pipeline

3.1 Problem Formulation

Given source and target point sets $\mathcal{X} = \{x_i\}_{i=1}^{N}, \mathcal{Y} = \{y_j\}_{j=1}^{M} \subset \mathbb{R}^3$, the goal is to generate an aligned source $\hat{\mathcal{X}}$ that matches $\mathcal{Y}$ through staged rigid and non-rigid refinement.

3.2 EvoReg's Architectural Pipeline

Stage 0: Gradient-free Pre-Alignment. Stage 0 performs gradient-free rigid pre-alignment. We estimate an initial transform (R_0, t_0) by minimizing a Chamfer-style distance between transformed source points and target points on subsampled clouds. This gradient-free stage uses CMA-ES (Hansen, 2016) over SE(3), providing a pose-agnostic initialization before any learned correspondence module is applied.

Stage 1: Iterative Soft Correspondence and Weighted SVD. Let $\mathcal{X}^{(0)} = R_0\mathcal{X} + t_0$. A shared PointNet encoder (Qi et al., 2017) maps each cloud to per-point features, projected to $\mathbb{R}^{256}$ via a two-layer MLP and L2-normalized. Target features $\{q_j\}$ are computed once and reused across all K iterations. At iteration k, a temperature-scaled cosine similarity matrix $A_{ij}^{(k)} = \langle p_i^{(k)}, q_j \rangle / \tau$ is dual-normalized via Sinkhorn (Cuturi, 2013; Yew & Lee, 2020) to yield a soft assignment matrix $S^{(k)}$. Pseudo-correspondences and confidence weights follow as $\tilde{y}_i^{(k)} = \sum_j S_{ij}^{(k)}\, y_j, c_i^{(k)} = \max_j\, S_{ij}^{(k)}$. A differentiable confidence-weighted Kabsch step estimates (R_k, t_k), applied as $\mathcal{X}^{(k)} = R_k\mathcal{X}^{(k-1)} + t_k$, with transforms composed across K=3 iterations to yield $\mathcal{X}_{\text{rigid}} = \mathcal{X}^{(K)}$. Cycle consistency is enforced by computing backward assignments S_{bwd} (swapping source and target roles) and forming $C = S_{\text{fwd}} S_{\text{bwd}}$.

Stage 2: Residual Rigid Refinement. A residual MLP head takes the concatenation of global features re-encoded from $\mathcal{X}_{\text{rigid}}$ and fixed target global features, predicting $(R_{\text{res}}, t_{\text{res}})$: $\mathcal{X}_{\text{rigid}}^{(2)} = R_{\text{res}}\, \mathcal{X}_{\text{rigid}} + t_{\text{res}}$. Rotation is parameterized in the continuous 6D representation of (Zhou et al., 2020) to avoid manifold discontinuities.

Stage 3: Conditional VAE for Non-Rigid Deformation. $\mathcal{X}_{\text{rigid}}^{(2)}$ is re-encoded and concatenated with target global features to parameterize a Gaussian latent distribution. During training, z is sampled via the reparameterization trick (Kingma & Welling, 2014); at inference, $z = \mu$ (deterministic decoding). A conditional generator produces the final output in residual form: $\hat{\mathcal{X}} = \mathcal{X}_{\text{rigid}}^{(2)} + G_\delta\left(\mathcal{X}_{\text{rigid}}^{(2)}, z\right)$. Because Stage 3 receives a rigidly pre-aligned source, the deformation network is never required to absorb large pose errors — improving accuracy and training stability.

Training Objective. The full training loss is: $\mathcal{L} = \lambda_{\text{cd}}\mathcal{L}_{\text{cd}}^{w} + \lambda_{\text{kl}}(t)\mathcal{L}_{\text{kl}} + \mathcal{L}_{\text{RT}} + \lambda_{\text{diff}}\mathcal{L}_{\text{diff}} + \lambda_{\text{corr}}\mathcal{L}_{\text{corr}} + \lambda_{\text{rigid}}\mathcal{L}_{\text{rigid_cd}} + \lambda_{\text{cyc}}\mathcal{L}_{\text{cyc}} + \lambda_{\text{inter}}\mathcal{L}_{\text{inter}} + \mathcal{L}_{\text{cent}}$. A subset of terms is active per supervision mode: $\mathcal{L}_{\text{RT}}$ combines rigid pose supervision with non-rigid regularization; when rigid ground truth is unavailable, pose-supervision terms are omitted. $\lambda_{\text{kl}}(t)$ follows a linear warm-up over the first 10,000 steps.

The primary geometric term $\mathcal{L}_{\text{cd}}^{w}$ is a centroid-distance-weighted Chamfer distance that upweights peripheral points — where misalignment is most geometrically informative — via weights $w_i = 3 + 4\,\frac{\|x_i - \bar{x}\|_2}{\max_\ell \|x_\ell - \bar{x}\|_2}$.

Auxiliary terms enforce: intermediate Chamfer supervision across Stage 1 iterates ($\mathcal{L}_{\text{inter}}$); Chamfer at the Stage 1 output ($\mathcal{L}_{\text{rigid_cd}}$); cycle consistency ($\mathcal{L}_{\text{cyc}} = \text{mean}(C - I)^2$); pointwise correspondence ($\mathcal{L}_{\text{corr}}$, when index-aligned targets are available); centroid alignment ($\mathcal{L}_{\text{cent}}$, an ℓ_2 penalty on the centroid gap between each stage output and $\mathcal{Y}$); and diffusion score matching ($\mathcal{L}_{\text{diff}}$).

Additional derivations are provided in the supplementary material in Section B.1.

Supervised Mode. When rigid ground truth $(R_{\text{gt}}, t_{\text{gt}})$ is available, $\mathcal{L}_{\text{RT}}$ activates a geodesic rotation loss and an ℓ_2 translation loss applied to the composed rigid output of Stages 1 and 2: $\mathcal{L}_{\text{RT}} = \arccos\left(\frac{\text{tr}(R_{\text{pred}}^{\top} R_{\text{gt}}) - 1}{2}\right) + \|t_{\text{pred}} - t_{\text{gt}}\|_2$. When rigid ground truth is unavailable (self-supervised mode), $\mathcal{L}_{\text{RT}}$ is dropped entirely and the model is supervised solely by geometric reconstruction losses ($\mathcal{L}_{\text{cd}}^{w}$, $\mathcal{L}_{\text{inter}}$, $\mathcal{L}_{\text{rigid_cd}}$, $\mathcal{L}_{\text{cyc}}$). Non-rigid variants activate Stage 3 by including parametric deformation augmentation during pair construction and adding the KL term $\mathcal{L}_{\text{kl}}$; rigid variants omit both. The four training settings — rigid supervised, rigid self-supervised, non-rigid supervised, non-rigid self-supervised — thus share the same architecture and differ only in which loss terms and data augmentations are active.

Inference-Time Refinements. Four optional training-free modules can be composed at inference without retraining: (i) concentrated gradient-free $\text{SE}(3)$ search; (ii) point-space DDPM denoising at noise level σ; (iii) Sinkhorn global test-time optimization over the rigid pose; and (iv) Sinkhorn per-point test-time optimization. Each trades compute for accuracy independently.

4 Experiments

4.1 Experimental Setup

4.1.1 Datasets

We train on ModelNet40 (Wu et al., 2015) (n=9,843, standard train partition across 40 categories) and evaluate on four datasets: ModelNet40 (n=2,468, standard test partition), ShapeNet-13 (Chang et al., 2015) (n=8,762, 13-category subset with per-category 80/20 split), 3DMatch (Zeng et al., 2017) (n=433, standard test partition across 8 test scenes), and FAUST (Bogo et al., 2014) (n=100, full dataset spanning 10 subjects with 10 distinct poses each, train/test split is not implemented since these samples are not used during training and dataset size is small). The latter three serve as cross-dataset generalization benchmarks.

4.1.2 Baselines

Rigid. We compare EvoReg against 16 rigid registration baselines, organized by supervision regime, to match our rigid EvoReg variants. The self-supervised and optimization group comprises: PRNet (Wang & Solomon, 2019b), CPD-Rigid (Myronenko & Song, 2010), ICP (Besl & McKay, 1992b), FGR (Zhou et al., 2016), RANSAC (Fischler & Bolles, 1981), and TEASER++ (Yang et al., 2021). The supervised group comprises: DCP (Wang & Solomon, 2019a), RPMNet (Yew & Lee, 2020), PointNetLK (Aoki et al., 2019), DeepGMR (Yuan et al., 2020), iPCRNet (Sarode et al., 2019), GeoTransformer (Qin et al., 2023), RoITr (Yu et al., 2023), PointDSC (Bai et al., 2021), CoFiNet (Yu et al., 2021), and Predator (Huang et al., 2021).

DCP, RPMNet, PointNetLK, DeepGMR, iPCRNet, and PRNet use the implementations and bundled pretrained weights from the `Learning3D` (Sarode, 2020) library, and are trained on ModelNet40 and evaluated across datasets (following the same pipeline as our model). GeoTransformer is

faithfully reimplemented following the original paper and reference code, and is similarly trained on ModelNet40 and evaluated across datasets. TEASER++ is included via its official implementation and is training-free. Classical baselines use standard library implementations: CPD-Rigid (CPD-R) via `PyCPD` (Gatti & Khallaghi, 2022); ICP, FGR, and RANSAC via `Open3D` (Zhou et al., 2018), and are likewise training-free. The remaining supervised baselines: RoITr, PointDSC, CoFiNet, and Predator, are integrated through their official released code and pretrained weights, and are trained and evaluated on their native 3DMatch benchmark.

Non-Rigid. We compare EvoReg against 4 non-rigid registration baselines, organized by supervision regime, to match our non-rigid EvoReg variants. The self-supervised and optimization group comprises CPD-Non-Rigid (CPD-NR) (Myronenko & Song, 2010), BCPD (Hirose, 2021), and NDP (Li & Harada, 2022). The supervised group comprises: DefTransNet (Monji-Azad et al., 2025).

Classical baselines use standard library implementations: CPD-Non-Rigid via `PyCPD` (Gatti & Khallaghi, 2022) and BCPD via `probreg` (Kenta-Tanaka et al.); NDP is integrated from the authors' official code, with no algorithmic changes — all of these methods are training-free. DefTransNet is reimplemented following the original paper and authors' reference code, and is trained on ModelNet40 and evaluated across datasets.

Importantly, all baselines are evaluated and scored identically to EvoReg. Baseline adaptations for reimplemented methods were strictly limited, and applied only where necessary to ensure compatibility with the established pipeline and experimental fairness across baselines.

4.1.3 IMPLEMENTATION

We train all EvoReg variants exclusively on ModelNet40, using the same training pipeline, as referenced in subsection 3.2 and illustrated in Figure 1. Registration pairs are constructed by applying randomly sampled similarity transformations, and for non-rigid variants parametric non-rigid deformations, to real point clouds taken directly from ModelNet40. Specifically, the source is taken directly from each normalized point cloud, and the target is generated by applying a similarity transformation whose Euler angles (roll, pitch, yaw) are each sampled uniformly from $[-45^\circ, +45^\circ]$, translation uniformly from $[-1.0, 1.0]^3$, and an isotropic scale factor $s = \exp(\epsilon)$ with $\epsilon \sim \mathcal{N}(0, 0.03^2)$. For non-rigid variants, we additionally apply a smooth Gaussian-RBF deformation field defined by $K{=}8$ random control points, per-control displacements drawn from $\mathcal{N}(\mathbf{0}, 0.05^2\mathbf{I})$, and bandwidth $\sigma{=}0.3$.

For each shape, we sample $1{,}024$ points and normalize the underlying point cloud to the unit ball. Unless otherwise stated, we train for 200 epochs with batch size 16 using Adam, with an initial learning rate of 10^{-3}; and apply a step learning-rate schedule that decays the learning rate by a factor of 0.5 every 50 epochs. To improve robustness against partial overlap, we use point dropout occlusion augmentation by randomly removing 20% of source points with probability 0.5.

Following the loss formulation depicted in subsection 3.2, we set $\lambda_{\text{CD}}{=}3.0$, $\lambda_{\text{KL}}{=}0.01$, $\lambda_{\text{diff}}{=}1.0$ with $T{=}1{,}000$ diffusion steps, $\lambda_{\text{rigid-CD}}{=}0.5$, $\lambda_{\text{int-CD}}{=}0.5$, and $\lambda_{\text{cyc}}{=}0.1$. Moreover, the conditional VAE uses a 128-dimensional latent, and correspondence features are projected to 256 dimensions with softmax temperature $\tau{=}0.1$. Supervised variants additionally penalize the predicted rigid transform against the ground-truth pose with $\lambda_R{=}\lambda_t{=}5.0$; self-supervised variants omit this term. Non-rigid variants use a slightly lower correspondence weight ($\lambda_{\text{corr}}{=}0.10$ vs. 0.18) and stronger per-stage centroid regularization ($\lambda_{\text{cen}}{=}0.30$ vs. 0.10) to compensate for softer point-level correspondences under deformation.

For evaluation, we test all EvoReg variants on ModelNet40, ShapeNet-13, FAUST, and 3DMatch under two distinct protocols: (1) the pair-construction procedure outlined in section C.1, but with a narrower translation range: $[-0.2, 0.2]^3$, and (2) a native, real-world data procedure defined in section E.1 of the supplementary material. Moreover, our models utilize $1{,}024$ points per shape, unit-ball normalization, and a CD-Recall threshold of 0.15. Regarding our different variants, we report two test-time variants, *EvoReg-Fast* and *EvoReg-Full* per registration modality. These variants differ only in their test-time refinement, they share a single set of trained weights. EvoReg-Fast denotes the configuration with the lowest possible overhead that still performs competitively (minimal or no test-time refinement; lighter inference-time hyperparameters when refinement is applied), while EvoReg-Full achieves the better performance, albeit at a higher computational cost (a more

Table 1: Rigid evaluation results on ModelNet40. **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised									
EvoReg-self-sup-full (Ours)	**0.01298**	0.03964	**0.01644**	0.00589	0.03453	99.39	3.521	0.00405	1.014
EvoReg-selfsup-fast (Ours)	0.03627	0.04868	0.03099	0.01275	0.04482	99.47	5.532	0.00913	0.096
PRNet (Wang & Solomon, 2019b)	0.07729	0.07791	0.05053	0.02749	0.07289	95.42	7.301	0.03395	0.048
CPD (Myronenko & Song, 2010)	0.02255	**0.02832**	0.02130	**0.00483**	**0.02606**	**99.96**	**1.979**	**0.00087**	1.376
ICP (Besl & McKay, 1992b)	0.14031	0.16837	0.05130	0.05765	0.15204	79.17	17.203	0.07034	**0.006**
FGR (Zhou et al., 2016)	0.19049	0.20387	0.07388	0.06112	0.18534	69.66	26.889	0.04078	0.069
RANSAC (Fischler & Bolles, 1981)	0.16025	0.27314	0.05162	0.07743	0.25203	81.40	36.244	0.12119	0.065
Supervised									
EvoReg-sup-full (Ours)	**0.01130**	**0.04264**	**0.01616**	**0.00605**	**0.03719**	99.31	**3.690**	**0.00440**	0.381
EvoReg-sup-fast (Ours)	0.03491	0.04940	0.02967	0.01233	0.04571	**99.35**	5.405	0.00932	0.094
DCP (Wang & Solomon, 2019a)	0.20684	0.22076	0.08305	0.06112	0.20483	74.39	28.704	0.04310	0.012
RPMNet (Yew & Lee, 2020)	0.17289	0.17522	0.09382	0.05535	0.16349	82.21	20.923	0.05033	0.017
PointNetLK (Aoki et al., 2019)	0.28015	0.27927	0.14300	0.11755	0.27253	62.80	17.781	0.21736	0.036
DeepGMR (Yuan et al., 2020)	0.35843	0.55934	0.10083	0.10462	0.51530	41.82	91.500	0.05753	0.103
iPCRNet (Sarode et al., 2019)	0.45484	0.45493	0.11213	0.17166	0.42819	15.92	48.984	0.24269	**0.009**
GeoTransformer (Qin et al., 2023)	0.08717	0.07820	0.03751	0.03143	0.07253	93.48	5.718	0.04213	0.194

substantive application of test-time refinement modules; heavier inference-time hyperparameters when refinement is applied). Additional details on inference-time module usage across the two variants, including hyperparameter breakdowns, can be found in section C.2.

All training is conducted on NVIDIA A100 80GB PCIe GPUs. Rigid evaluation for protocol one is conducted on NVIDIA A100 80GB PCIe GPUs; rigid evaluation for protocol two is conducted on NVIDIA A10G 24GB PCIe GPUs. All non-rigid evaluation is conducted on NVIDIA A10G 24GB PCIe GPUs. Moreover, the entire training pipeline, across all runs, uses a fixed random seed of 42 for consistency and reproducibility of non-deterministic components; and the entire evaluation pipeline, across all runs, uses a fixed random seed of 123 for the same purposes.

4.2 Evaluation Metrics

Distance-based metrics. (1) Chamfer Distance (CD) measures bidirectional nearest-neighbor discrepancy between the registered source $\hat{\mathcal{X}}$ and the target $\mathcal{Y}$. (2) Registration Error (RE_g) follows the RMSE-style registration error used in the CPD literature. (3) Earth Mover's Distance (EMD) is approximated via a Sinkhorn-based solver for computational efficiency. (4) Sliced Wasserstein Distance (SWD) compares distributions by averaging 1D Wasserstein distances over multiple random projections. (5) Point-to-Point error (P2P) measures mean per-point error under an assumed 1-to-1 correspondence (when applicable under the evaluation protocol).

CD-Recall, pose errors, & time. (6) Chamfer Distance Recall (CD-Recall) is defined as the percentage of test pairs whose CD is below a fixed threshold; we use a threshold of 0.15 throughout our experiments. We additionally report (7) Rotation Error (RE) in degrees and (8) Translation Error (TE) as the ℓ_2 difference in translation vectors. Lastly, we report (9) the per-pair inference time (Infer.), measured as the wall-clock runtime required to register a single source–target pair, in seconds (s). This metric highlights the practical trade-off between accuracy and efficiency. When a method does not explicitly output $(\mathbf{R}, \mathbf{t})$, we recover the best-fit rigid transform from the registered point cloud via the Kabsch algorithm (Kabsch, 1976), and report RE and TE from that estimate.

4.3 Results

4.3.1 Rigid

ModelNet40. Table 1 and Figure 2 display the rigid ModelNet40 results. In the self-supervised block, EvoReg-self-sup-full achieves the best CD (0.01298) and EMD (0.01644), reducing CD by 42.4% relative to the strongest non-EvoReg baseline, CPD (0.02255). CPD remains strongest on $\mathrm{RE_g}$, SWD, P2P, CD-Recall, RE, and TE, while EvoReg-self-sup-fast preserves 99.47% CD-Recall at 0.096 s per pair. In the supervised block, EvoReg-sup-full achieves the best CD (0.01130), EMD (0.01616), SWD (0.00605), P2P (0.03719), RE (3.690°), and TE (0.00440), while EvoReg-sup-fast attains the highest CD-Recall in that block (99.35%) with 0.094 s inference.

3DMatch. Table 2 and Figure 3 report rigid 3DMatch results. EvoReg-self-sup-full is best overall on CD (0.00257), $\mathrm{RE_g}$ (0.01716), EMD (0.00757), SWD (0.00291), and P2P (0.01019), while

Table 2: Rigid evaluation results on 3DMatch. **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised									
EvoReg-self-sup-full (Ours)	**0.00257**	**0.01716**	**0.00757**	**0.00291**	**0.01019**	99.77	0.437	0.00291	0.881
EvoReg-selfsup-fast (Ours)	0.03803	0.02805	0.02948	0.01012	0.02589	**100.0**	1.525	0.01061	0.094
PRNet (Wang & Solomon, 2019b)	0.08108	0.10035	0.05007	0.02771	0.09281	93.30	11.606	0.03271	0.063
CPD (Myronenko & Song, 2010)	0.02854	0.01893	0.02181	0.00334	0.01741	**100.0**	**0.302**	**0.00100**	1.599
ICP (Besl & McKay, 1992b)	0.04691	0.04624	0.03049	0.01466	0.04219	96.77	3.076	0.01197	**0.005**
FGR (Zhou et al., 2016)	0.04814	0.04005	0.03404	0.01470	0.03648	98.85	3.107	0.01659	0.093
RANSAC (Fischler & Bolles, 1981)	0.04601	0.03721	0.03472	0.01443	0.03446	99.77	2.819	0.01748	0.054
Supervised									
EvoReg-sup-full (Ours)	**0.00320**	**0.02149**	**0.00826**	**0.00383**	**0.01428**	99.77	**0.736**	**0.00400**	0.386
EvoReg-sup-fast (Ours)	0.03772	0.02794	0.02921	0.00999	0.02581	**100.0**	1.521	0.01076	0.095
DCP (Wang & Solomon, 2019a)	0.06346	0.05214	0.04727	0.01641	0.04795	99.77	6.151	0.00744	**0.008**
RPMNet (Yew & Lee, 2020)	0.14862	0.17508	0.08696	0.04454	0.16150	88.68	22.248	0.02570	0.014
PointNetLK (Aoki et al., 2019)	0.22030	0.23866	0.11272	0.10683	0.23389	79.68	9.958	0.20245	0.035
DeepGMR (Yuan et al., 2020)	0.28799	0.49851	0.10684	0.08055	0.45550	38.80	84.394	0.03097	0.115
iPCRNet (Sarode et al., 2019)	0.30991	0.38967	0.10517	0.13313	0.36450	32.56	44.516	0.20481	0.010
GeoTransformer (Qin et al., 2023)	0.04007	0.02855	0.02928	0.01134	0.02647	99.77	1.245	0.01093	0.196

Table 3: Non-rigid evaluation results on ModelNet40. **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised									
EvoReg-selfsup-full (Ours)	**0.00509**	0.05525	**0.01664**	**0.00723**	0.04713	**100.00**	6.991	**0.03423**	0.676
EvoReg-selfsup-fast (Ours)	0.01098	0.05596	0.01886	0.00754	0.04792	**100.00**	6.991	**0.03423**	**0.253**
CPD-NR (Myronenko & Song, 2010)	0.05316	0.08051	0.04162	0.02112	0.06522	98.87	7.919	0.03480	1.390
BCPD (Hirose, 2021)	0.19331	0.27821	0.08439	0.09064	0.23573	86.14	18.300	0.04018	3.756
NDP (Li & Harada, 2022)	0.01847	**0.04844**	0.02175	0.00956	**0.04333**	**100.00**	**6.685**	0.03497	29.553
Supervised									
EvoReg-sup-full (Ours)	**0.00612**	0.05719	**0.01683**	0.00787	0.04883	99.84	6.909	0.03471	0.675
EvoReg-sup-fast (Ours)	0.01300	0.05778	0.01946	0.00833	0.04955	99.80	6.909	0.03471	0.254
DefTransNet (Monji-Azad et al., 2025)	0.01659	**0.02894**	0.02111	**0.00683**	**0.02207**	**100.00**	**3.073**	**0.03060**	**0.124**

EvoReg-sup-full is the best supervised model on RE (0.736°) and TE (0.00400). Relative to the strongest non-EvoReg supervised baseline under CD, DCP (0.06346), EvoReg-sup-full reduces CD by 95.0%, and EvoReg-self-sup-full reduces it by 96.0%. The fast EvoReg variants trade some geometric precision for perfect CD-Recall (100.0%) and low inference cost (0.094–0.095 s), while CPD remains the strongest non-EvoReg method on RE (0.302°) and TE (0.00100).

ShapeNet-13 & FAUST. The full rigid evaluation results on the ShapeNet-13 and FAUST datasets can be found in section C.3 of the supplementary materials.

4.3.2 Non-Rigid

ModelNet40. Table 3 reports the non-rigid registration results on ModelNet40. In the self-supervised regime, EvoReg-Full achieves the best CD (0.00509), EMD (0.01664), SWD (0.00723), and TE (0.03423) while reaching 100% CD-Recall, improving over CPD-NR and BCPD by an order of magnitude on CD. NDP attains slightly lower RE_g, P2P, and RE, but at roughly $50\times$ the inference cost of EvoReg-Full. In the supervised regime, EvoReg-Full retains the best CD and EMD.

3DMatch. Table 4 shows the cross-dataset non-rigid registration results on the real indoor RGB-D fragments from 3DMatch. Self-supervised EvoReg-Full obtains the best CD (0.00316) and EMD (0.01416), while achieving perfect 100% CD-Recall, and running over $45\times$ faster than NDP, its closest competitor. The supervised variants follow the same pattern, with EvoReg-Full leading on CD/EMD, and achieving near-perfect CD-Recall. The performance gap of EvoReg over CPD-NR and BCPD widens substantially relative to the in-distribution ModelNet40 results, indicating that the staged decoupling generalizes more reliably to unseen real-scan geometry than classical alternatives.

ShapeNet-13 & FAUST. The full non-rigid evaluation results on the ShapeNet-13 and FAUST datasets can be found in section C.3 of the supplementary materials.

4.3.3 Native and Real-World Scans

On native 3DMatch, EvoReg-sup-full leads four geometric alignment and pointwise correspondence metrics (CD, EMD, SWD, and P2P), and ties first with DCP and EvoReg-sup-fast on RE_g and CD-Recall, respectively. EvoReg-sup-fast ranks closely behind, second place on CD, EMD, SWD, P2P, and tied first with EvoReg-sup-full on CD-Recall — running faster than RoITr, PointDSC, CoFiNet,

Table 4: Non-rigid evaluation results on 3DMatch. **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised									
EvoReg-selfsup-full (Ours)	**0.00316**	0.04730	**0.01416**	0.00899	0.03599	**100.00**	3.561	0.03476	0.680
EvoReg-selfsup-fast (Ours)	0.00989	0.04786	0.01752	0.00900	0.03702	**100.00**	3.561	0.03476	**0.258**
CPD-NR (Myronenko & Song, 2010)	0.05518	0.07842	0.04373	0.01827	0.06776	**100.00**	7.455	0.03466	1.843
BCPD (Hirose, 2021)	0.11465	0.24236	0.08577	0.04870	0.20995	99.77	15.603	0.03844	2.981
NDP (Li & Harada, 2022)	0.02246	**0.01922**	0.01930	**0.00336**	**0.01660**	**100.00**	**2.986**	**0.03030**	31.912
Supervised									
EvoReg-sup-full (Ours)	**0.00325**	0.04752	**0.01431**	0.00918	0.03619	**100.00**	3.374	0.03447	0.686
EvoReg-sup-fast (Ours)	0.01113	0.04789	0.01833	0.00925	0.03715	**100.00**	3.374	0.03447	0.257
DefTransNet (Monji-Azad et al., 2025)	0.01357	**0.02402**	0.01827	**0.00442**	**0.01572**	**100.00**	**2.640**	**0.02993**	**0.119**

and Predator. On FAUST intra-subject pairs, EvoReg-sup-full achieves the best RE, along with the best CD, EMD, P2P, and ties first with all methods on CD-Recall; in the self-supervised/optimization block, EvoReg-selfsup-full achieves the best CD, EMD, P2P, and ties first on CD-Recall with all methods, while running faster than all tested baselines: $1.6\times$ faster than CPD-NR, $3.5\times$ faster than BCPD, and $45.9\times$ faster than NDP. In both inter-subject blocks, EvoReg wins CD and EMD, and ties on CD-Recall. These results are reported in Tables E1, E2, and E3; additional protocol details and analyses are provided in Appendix E.

4.3.4 Statistical Significance Testing

To assess statistical significance across both evaluation protocols, we performed paired Wilcoxon signed-rank tests on per-sample metrics, and McNemar's test for the binary CD-Recall outcome, with Holm-Bonferroni correction at α=0.05. For each registration modality and supervision regime, the corresponding EvoReg-Full variant serves as the reference against which every baseline is compared. The full per-baseline, per-metric significance tables and discussion is provided in section C.4 of the supplementary document.

4.4 Ablation Studies

We ablate the key components and design choices of EvoReg under the training and evaluation protocols outlined in Sections 3.2 and 4.1.3, and measure the resulting changes in registration quality. The primary component ablation is summarized in Table D1 of the supplementary document; additional studies are similarly found in the supplementary document, in section (D). All ablation studies were conducted using the ModelNet40 dataset.

5 Limitations and Practical Guidance

EvoReg's design trades pose precision for correspondence and distributional alignment accuracy; on rotation and translation error it is generally significantly outperformed by methods whose design favors accurate pose recovery. Table F1 shows when EvoReg is preferable and when other methods should be used instead. Furthermore, despite the comprehensive inference-time suite increasing model accuracy, it implies significant overhead and introduces sensitivity to refinement hyperparameters; improving test-time refinement efficiency and reducing hyperparameter sensitivity, while preserving accuracy, remain open and likely fruitful directions.

6 Conclusion

Point cloud registration has long been fragmented along two axes: rigid versus non-rigid alignment, and supervised versus self-supervised training. EvoReg resolves this fragmentation through staged decoupling. Each stage inherits a progressively tighter initialization, reducing its subproblem to one that is tractable without forcing design choices that preclude the other settings. A single underlying architecture, trained on ModelNet40, achieves competitive correspondence and the strongest distributional alignment accuracy, significantly outperforming all baselines on Chamfer Distance and Earth Mover's Distance, while achieving perfect or near-perfect CD-Recall, across multiple evaluation protocols spanning ModelNet40, ShapeNet-13, 3DMatch, and FAUST, under all four training regimes. Our results demonstrate that staged decoupling can unify registration modalities, supervision regimes, and evaluation protocols, and position EvoReg as a dynamic and robust method for controlled and real-world registration.

AI USE STATEMENT

In this work, generative AI tools were used to assist with writing, editing, and formatting of the manuscript, and to assist with code implementation and debugging. AI tools were also used to support ideation and related work discovery throughout the project. Importantly, all AI-generated content was screened and verified prior to inclusion in this work. The authors take full responsibility for the final content of this work.

ETHICS STATEMENT

This work uses publicly available benchmark datasets (ModelNet40, ShapeNet-13, 3DMatch, and FAUST) and standard point-cloud registration evaluation protocols. It does not involve human-subjects experiments, collection of sensitive personal data, or applications with foreseeable direct societal harm. There are no ethical concerns associated with this work.

REPRODUCIBILITY STATEMENT

We describe EvoReg's full architectural pipeline in Section 3.2 and Appendix B, including the training objective, loss terms, and inter-stage gradient-free refinement schedule. Dataset construction, splits, and evaluation protocol are described in Section 4.1.3 and Appendix C, and the statistical significance testing procedure is detailed in Section C.4. Additional method details not covered in the main text are provided in Appendix A and Appendix B. Hardware details are documented in Section 4.1.3. The code for this project can be found at: `https://anonymous.4open.science/r/EvoReg-0507`.

REFERENCES

Yasuhiro Aoki, Hunter Goforth, Rangaprasad Arun Srivatsan, and Simon Lucey. Pointnetlk: Robust & efficient point cloud registration using pointnet. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7163–7172, 2019.

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

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

P.J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992b. doi: 10.1109/34.121791.

Federica Bogo, Javier Romero, Matthew Loper, and Michael J. Black. Faust: Dataset and evaluation for 3d mesh registration. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3794–3801, 2014. doi: 10.1109/CVPR.2014.491.

Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. URL `https://arxiv.org/abs/1512.03012`.

Haili Chui and Anand Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, 2003.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, volume 26, 2013.

Haoqiang Fan, Hao Su, and Leonidas J. Guibas. A point set generation network for 3D object reconstruction from a single image. In *CVPR*, 2017.

Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, June 1981. ISSN 0001-0782. doi: 10.1145/358669.358692. URL `https://doi.org/10.1145/358669.358692`.

Wei Gao and Russ Tedrake. Filterreg: Robust and efficient probabilistic point-set registration using gaussian filter and twist parameterization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11095–11104, 2019.

Anthony A. Gatti and Siavash Khallaghi. Pycpd: Pure numpy implementation of the coherent point drift algorithm. *Journal of Open Source Software*, 7(80):4681, 2022. doi: 10.21105/joss.04681. URL https://doi.org/10.21105/joss.04681.

Nikolaus Hansen. The CMA evolution strategy: A tutorial. *arXiv preprint arXiv:1604.00772*, 2016.

Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001. doi: 10.1162/106365601750190398.

Osamu Hirose. A bayesian formulation of coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(7):2269–2286, 2021.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020.

Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. PREDATOR: Registration of 3D point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4267–4276, 2021.

Bing Jian and Baba C. Vemuri. Robust point set registration using gaussian mixture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1633–1645, 2011.

Haobo Jiang, Mathieu Salzmann, Zheng Dang, Jin Xie, and Jian Yang. Se(3) diffusion model-based point cloud registration for robust 6d object pose estimation. *Advances in Neural Information Processing Systems*, 36:21285–21297, 2023.

W. Kabsch. A solution for the best rotation to relate two sets of vectors. *Acta Crystallographica Section A*, 32(5):922–923, Sep 1976. doi: 10.1107/S0567739476001873. URL https://doi.org/10.1107/S0567739476001873.

James Kennedy and Russell C. Eberhart. Particle swarm optimization. In *Proceedings of ICNN'95 — International Conference on Neural Networks*, volume 4, pp. 1942–1948, 1995. doi: 10.1109/ICNN.1995.488968.

Kenta-Tanaka et al. probreg. URL https://probreg.readthedocs.io/en/latest/.

Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In *ICLR*, 2014.

Yang Li and Tatsuya Harada. Non-rigid point cloud registration with neural deformation pyramid. *Advances in Neural Information Processing Systems*, 35:27757–27768, 2022.

Shitong Luo and Wei Hu. Diffusion probabilistic models for 3d point cloud generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2837–2845, 2021.

Baraka Maiseli, Yanfeng Gu, and Huijun Gao. Recent developments and trends in point set registration methods. *Journal of Visual Communication and Image Representation*, 46:95–106, 2017.

Seyedali Mirjalili, Seyed Mohammad Mirjalili, and Andrew Lewis. Grey wolf optimizer. *Advances in Engineering Software*, 69:46–61, 2014. doi: 10.1016/j.advengsoft.2013.12.007.

Sara Monji-Azad, Marvin Kinz, Siddharth Kothari, Robin Khanna, Amrei Carla Mihan, David Maennel, Claudia Scherl, and Juergen Hesser. Deftransnet: A transformer-based method for non-rigid point cloud registration in the simulation of soft tissue deformation, 2025. URL https://arxiv.org/abs/2502.06336.

Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(12):2262–2275, 2010.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, Slobodan Ilic, Dewen Hu, and Kai Xu. Geotransformer: Fast and robust point cloud registration with geometric transformer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45:9806–9821, 2023. URL https://api.semanticscholar.org/CorpusID:257646283.

Vinit Sarode. learning3d: A modern library for deep learning on 3d point clouds data. https://github.com/vinits5/learning3d, 2020.

Vinit Sarode, Xueqian Li, Hunter Goforth, Yasuhiro Aoki, Rangaprasad Arun Srivatsan, Simon Lucey, and Howie Choset. Pcrnet: Point cloud registration network using pointnet encoding. *arXiv preprint arXiv:1908.07906*, 2019.

Zhengyang Shen, Jean Feydy, Peirong Liu, Ariel H Curiale, Ruben San Jose Estepar, Raul San Jose Estepar, and Marc Niethammer. Accurate point cloud registration with robust optimal transport. *Advances in Neural Information Processing Systems*, 34:5373–5389, 2021.

Rainer Storn and Kenneth Price. Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11(4):341–359, 1997. doi: 10.1023/A:1008202821328.

Yue Wang and Justin M Solomon. Deep closest point: Learning representations for point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3523–3532, 2019a.

Yue Wang and Justin M Solomon. Prnet: Self-supervised learning for partial-to-partial registration. *Advances in Neural Information Processing Systems*, 32:8812–8824, 2019b.

Qianliang Wu, Haobo Jiang, Lei Luo, Jun Li, Yaqing Ding, Jin Xie, and Jian Yang. Diff-reg: Diffusion model in doubly stochastic matrix space for registration problem. In *European Conference on Computer Vision*, pp. 160–178. Springer, 2024.

Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1912–1920, 2015.

Heng Yang, Jingnan Shi, and Luca Carlone. Teaser: Fast and certifiable point cloud registration. *IEEE Transactions on Robotics*, 37(2):314–333, 2021. doi: 10.1109/TRO.2020.3033695.

Xin-She Yang. Firefly algorithms for multimodal optimization. In Osamu Watanabe and Thomas Zeugmann (eds.), *Stochastic Algorithms: Foundations and Applications*, volume 5792 of *Lecture Notes in Computer Science*, pp. 169–178, Berlin, Heidelberg, 2009. Springer. doi: 10.1007/978-3-642-04944-6_14.

Zi Jian Yew and Gim Hee Lee. Rpm-net: Robust point matching using learned features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11824–11833, 2020.

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

Hao Yu, Zheng Qin, Ji Hou, Mahdi Saleh, Dongsheng Li, Benjamin Busam, and Slobodan Ilic. Rotation-invariant transformer for point cloud matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5384–5393, 2023.

Wentao Yuan, Benjamin Eckart, Kihwan Kim, Varun Jampani, Dieter Fox, and Jan Kautz. Deepgmr: Learning latent gaussian mixture models for registration. In *European Conference on Computer Vision*, pp. 733–750. Springer, 2020.

Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. URL https://arxiv.org/abs/1603.08182.

Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. *Advances in Neural Information Processing Systems*, 35:10021–10039, 2022.

Linqi Zhou, Yilun Du, and Jiajun Wu. 3d shape generation and completion through point-voxel diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5826–5835, 2021.

Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Fast global registration. In *European Conference on Computer Vision*, 2016. URL https://api.semanticscholar.org/CorpusID:27362942.

Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3D: A modern library for 3D data processing. *arXiv:1801.09847*, 2018.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *CVPR*, 2020. URL https://arxiv.org/abs/1812.07035.

SUPPLEMENTARY MATERIAL

A MORE RELATED WORK

Classical Rigid Registration. Point-set registration is classically posed as the joint estimation of correspondences and a transformation that aligns one point set to another. ICP captures this idea in its simplest form by alternating between nearest-neighbor matching and rigid updates (Besl & McKay, 1992a), but its hard matching is fragile to poor initialization, noise, and partial overlap. Coherent Point Drift (CPD) softens the matching step by treating one point set as Gaussian mixture centroids that move coherently toward the other, supporting rigid registration and non-rigid registration through a regularized displacement field (Myronenko & Song, 2010); related mixture and density-based formulations explore alternative density-alignment objectives and Gaussian-filtered updates (Jian & Vemuri, 2011; Gao & Tedrake, 2019). In contrast, EvoReg keeps the correspondence-then-transform decomposition but learns soft matching and refines alignment across several stages, rather than relying on a single local optimizer.

Classical Non-rigid Registration. When source and target differ in articulation or local deformation, rigid alignment alone is insufficient, and the higher-dimensional motion space requires a structured deformation model to remain well-posed. Classical methods supply this structure: TPS-RPM couples soft assignment with thin-plate-spline warping under deterministic annealing (Chui & Rangarajan, 2003), non-rigid CPD regularizes a coherent displacement field (Myronenko & Song, 2010), and BCPD reformulates CPD in a Bayesian framework that handles rigid and non-rigid registration uniformly (Hirose, 2021).

Diffusion for 3D Registration. Diffusion offers a complementary mechanism: iterative correction via reversal of a learned noising process (Ho et al., 2020). In 3D, diffusion has been applied to point cloud generation, completion, and latent shape modeling (Luo & Hu, 2021; Zhou et al., 2021; Zeng et al., 2022). More recent work targets registration directly: SE(3) diffusion denoises rigid transformations on the manifold for 6D pose estimation (Jiang et al., 2023), and Diff-Reg diffuses over doubly stochastic matching matrices to construct correspondences (Wu et al., 2024). These methods introduce flexible iterative correction but remain specialized. SE(3) diffusion targets rigid transformations only, and Diff-Reg operates on correspondences rather than point positions. In contrast to these formulations, EvoReg applies diffusion to point coordinates rather than to transformations or matching matrices, and only as an optional inference-time refinement on top of an already-registered cloud, leaving structural alignment to the deterministic earlier stages.

Gradient-free SE(3) initialization. Coarse rigid pre-alignment is non-convex under large rotations, partial overlap, and outliers, which makes derivative-free population search a natural fit. Nature-inspired algorithms such as PSO (Kennedy & Eberhart, 1995), DE (Storn & Price, 1997), CMA-ES (Hansen & Ostermeier, 2001), GWO (Mirjalili et al., 2014), and the Firefly Algorithm (Yang, 2009) are standard candidates for continuous, gradient-free objectives. Rather than claiming any one optimizer is intrinsically best for $\mathrm{SE}(3)$, EvoReg treats gradient-free optimization as a dedicated initialization module whose output is the input to the learned correspondence stages.

B METHOD DETAILS

EvoReg predicts an aligned source $\hat{\mathcal{X}}$ via staged rigid $(R, t) \in \mathrm{SE}(3)$ and non-rigid $\delta(\cdot)$ refinement, where $R \in \mathrm{SO}(3)$ is a rotation matrix, $t \in \mathbb{R}^3$ is a translation vector, and $\delta : \mathbb{R}^3 \to \mathbb{R}^3$ is a per-point displacement field that captures non-rigid deformation.

B.1 TRAINING OBJECTIVE

The model is trained by minimizing a weighted sum of geometric, variational, and regularization terms:

$$\begin{aligned}\mathcal{L} \;=\; & \lambda_{\mathrm{cd}}\mathcal{L}_{\mathrm{cd}}^{w} + \lambda_{\mathrm{kl}}(t)\mathcal{L}_{\mathrm{kl}} + \mathcal{L}_{\mathrm{RT}} + \lambda_{\mathrm{diff}}\mathcal{L}_{\mathrm{diff}} \\ & + \lambda_{\mathrm{corr}}\mathcal{L}_{\mathrm{corr}} + \lambda_{\mathrm{rigid}}\mathcal{L}_{\mathrm{rigid_cd}} + \lambda_{\mathrm{cyc}}\mathcal{L}_{\mathrm{cyc}} \\ & + \lambda_{\mathrm{inter}}\mathcal{L}_{\mathrm{inter}} + \mathcal{L}_{\mathrm{cent}},\end{aligned} \tag{1}$$

A subset of these terms is used depending on the supervision mode. A standard linear KL warm-up is used, increasing λ_{kl} from 0 to $\lambda_{\text{kl}}^{\max}$ over the first 10,000 training steps and keeping it constant thereafter.

B.1.1 WEIGHTED CHAMFER

Building upon the standard Chamfer distance (Fan et al., 2017), a centroid-distance weighting is introduced:

$$w_i = 3 + 4\,\frac{\|x_i - \bar{x}\|_2}{\max_\ell \|x_\ell - \bar{x}\|_2}. \tag{2}$$

Weights are applied to the source-to-target term:

$$\mathcal{L}_{\text{cd}}^{w} = \frac{1}{N}\sum_{i=1}^{N} w_i \min_j \|\hat{x}_i - y_j\|_2 + \frac{1}{M}\sum_{j=1}^{M} \min_i \|y_j - \hat{x}_i\|_2. \tag{3}$$

B.1.2 RT LOSS ($\mathcal{L}_{\text{RT}}$).

Let $\tilde{x}_i = R_{\text{pred}}\, x_i + t_{\text{pred}}$ be the rigidly-transformed source point and $\delta_i = \hat{x}_i - \tilde{x}_i$ the per-point displacement produced by the non-rigid head. Let $\mathcal{N}_k(i)$ denote the k-nearest-neighbor set of $\tilde{x}_i$ within $\{\tilde{x}_\ell\}$, and $\mathcal{N}_k^{\mathcal{Y}}(\hat{x}_i)$ the k-NN of output point $\hat{x}_i$ in the target $\mathcal{Y}$.

Non-rigid regularizers (retained in both supervised and self-supervised modes):

$$\mathcal{L}_{\text{align}} = \frac{1}{Nk}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_k^{\mathcal{Y}}(\hat{x}_i)} \|\hat{x}_i - y_j\|_2^2, \quad \text{(exact: } \tfrac{1}{N}\textstyle\sum_i \|\hat{x}_i - y_i\|_2^2 \text{ when } N = M) \tag{4}$$

$$\mathcal{L}_{\text{disp}} = \frac{1}{N}\sum_{i=1}^{N} \|\delta_i\|_2^2, \tag{5}$$

$$\mathcal{L}_{\text{deform}} = \frac{1}{Nk}\sum_{i=1}^{N}\sum_{j\in\mathcal{N}_k(i)} \big\|(\hat{x}_i - \hat{x}_j) - (\tilde{x}_i - \tilde{x}_j)\big\|_2^2, \tag{6}$$

$$\mathcal{L}_{\text{lap}} = \frac{1}{N}\sum_{i=1}^{N} \left\| \delta_i - \frac{1}{k}\sum_{j\in\mathcal{N}_k(i)} \delta_j \right\|_2^2. \tag{7}$$

Supervised mode ($R_{\text{gt}}, t_{\text{gt}}$ provided): three pose-supervision terms are added,

$$\mathcal{L}_{\text{rot}} = \arccos\left(\frac{\text{tr}(R_{\text{pred}}^\top R_{\text{gt}}) - 1}{2}\right), \qquad \mathcal{L}_{\text{trans}} = \|t_{\text{pred}} - t_{\text{gt}}\|_2, \tag{8}$$

$$\mathcal{L}_{\text{rmse}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|(R_{\text{pred}} x_i + t_{\text{pred}}) - (R_{\text{gt}} x_i + t_{\text{gt}})\|_2^2}, \tag{9}$$

giving the full supervised RT loss:

$$\mathcal{L}_{\text{RT}}^{\text{sup}} = \lambda_{\text{rot}}\mathcal{L}_{\text{rot}} + \lambda_{\text{trans}}\mathcal{L}_{\text{trans}} + \lambda_{\text{rmse}}\mathcal{L}_{\text{rmse}} + \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{\text{disp}}\mathcal{L}_{\text{disp}} + \lambda_{\text{deform}}\mathcal{L}_{\text{deform}} + \lambda_{\text{lap}}\mathcal{L}_{\text{lap}}. \tag{10}$$

Self-supervised mode (no ground truth): the three pose-supervision terms are dropped,

$$\mathcal{L}_{\text{RT}}^{\text{self}} = \lambda_{\text{align}}\mathcal{L}_{\text{align}} + \lambda_{\text{disp}}\mathcal{L}_{\text{disp}} + \lambda_{\text{deform}}\mathcal{L}_{\text{deform}} + \lambda_{\text{lap}}\mathcal{L}_{\text{lap}}. \tag{11}$$

B.1.3 AUXILIARY LOSSES.

When enabled, pointwise correspondence supervision is applied as an MSE, $\mathcal{L}_{\text{corr}} = \frac{1}{3N}\sum_i \|\hat{x}_i - y_i\|_2^2$, and is used only when pointwise correspondence is meaningful (e.g., training pairs constructed by applying a known transformation to a source point cloud, where the target is index-aligned to

the source by construction), since it assumes index-aligned targets. Cycle consistency is enforced by penalizing deviations of the cycle matrix from identity using an elementwise MSE, $\mathcal{L}_{\text{cyc}} = \text{mean}\big((C - I)^2\big)$. Intermediate supervision is applied to the Stage 1 rigid iterates, where $k = 1, \ldots, K$ indexes the SVD iteration and $\mathcal{X}^{(k)}$ denotes the source after the k-th SVD step:

$$\mathcal{L}_{\text{inter}} = \sum_{k=1}^{K-1} \frac{k}{K} \, \text{CD}^w\Big(\mathcal{X}^{(k)}, \mathcal{Y}\Big), \qquad \mathcal{L}_{\text{rigid_cd}} = \text{CD}^w(\mathcal{X}_{\text{rigid}}, \mathcal{Y}). \tag{12}$$

B.1.4 Centroid alignment loss.

To further stabilize coarse-to-fine registration, we additionally introduce a centroid alignment term that encourages the globally aligned source cloud to share the same mass center as the target cloud at each refinement stage. Let $\mathcal{X}^{(s)} = \{x_i^{(s)}\}_{i=1}^N$ ($s \in \{1, 2, 3\}$, distinct from the SVD iteration index k above) denote the stage-s aligned source point cloud and let $\mathcal{Y} = \{y_j\}_{j=1}^M$ be the target point cloud. Their centroids are defined as

$$\mathbf{c}\Big(\mathcal{X}^{(s)}\Big) = \frac{1}{N} \sum_{i=1}^{N} x_i^{(s)}, \qquad \mathbf{c}(\mathcal{Y}) = \frac{1}{M} \sum_{j=1}^{M} y_j. \tag{13}$$

The centroid loss at stage s is then given by

$$\mathcal{L}_{\text{cent}}^{(s)} = \Big\| \mathbf{c}\Big(\mathcal{X}^{(s)}\Big) - \mathbf{c}(\mathcal{Y}) \Big\|_2. \tag{14}$$

Using the three-stage outputs of our model, the final centroid term is written as

$$\mathcal{L}_{\text{cent}} = \lambda_{\text{cent}}^{(1)} \mathcal{L}_{\text{cent}}^{(1)} + \lambda_{\text{cent}}^{(2)} \mathcal{L}_{\text{cent}}^{(2)} + \lambda_{\text{cent}}^{(3)} \mathcal{L}_{\text{cent}}^{(3)}. \tag{15}$$

This term acts as a lightweight global constraint complementary to pointwise geometric losses such as Chamfer distance. Whereas local matching losses mainly enforce fine geometric proximity, $\mathcal{L}_{\text{cent}}$ directly regularizes the first-order spatial moment of the predicted point set, thereby reducing large translational drift and making the intermediate refinement stages more stable. In particular, it is beneficial when local correspondences are noisy, partial, or temporarily ambiguous, since it pulls each intermediate prediction toward the correct global region before fine-scale alignment is fully established.

EvoReg supports four optional inference-time refinements that operate on the Stage 3 output $\hat{\mathcal{X}}$ — at no retraining cost — and are executed independently of each other. **CMA-ES Refinement 3** first runs the third inter-stage CMA-ES refinement module (created during training but not invoked in the forward pass) at $\pm 5^\circ / \pm 0.1$ search range, with population size 25, over 15 generations, to catch residual rigid offsets. Next, **diffusion-based refinement** denoises $\hat{\mathcal{X}}$ via a score network conditioned on $(x_t, \mathcal{Y}, t)$, and predicts $-\epsilon$ under the standard DDPM noise-prediction MSE objective (Ho et al., 2020). Finally, two distinct Sinkhorn-EMD test-time optimizations (Cuturi, 2013) sharpen the result via L2-regularized Adam updates: **Sinkhorn Global TTO** refines a single rigid correction $(\delta R, \delta t)$, while **Sinkhorn Per-Point TTO** refines each point with an additive displacement ϵ_i.

B.2 Inter-Stage CMA-ES Refinement

Two CMA-ES refinements run between the learned stages: the first after Stage 1 at $\pm 20^\circ / \pm 0.5$ and the second after Stage 2 at $\pm 10^\circ / \pm 0.2$, each with population size 25 over 15 generations. The tighter search ranges were chosen to match the increased alignment accuracy in the latter stages of training. These passes absorb residual rigid error before it propagates to the next learned stage.

C Experiment Details

C.1 Training Data Construction

During training, source points are unit-ball normalized, and the target is generated by applying a randomly sampled rigid transformation, optionally combined with a non-rigid deformation:

$$y_i = sRx_i + t + \delta_i + \eta_i, \tag{16}$$

where s is a mild log-normal scale factor and η_i is additive Gaussian noise. Furthermore, source-side occlusion is optionally applied via random dropout, with point count restored by resampling. Importantly, rigid labels $(R_{\text{gt}}, t_{\text{gt}})$ are used only when pose supervision is enabled.

C.2 EvoReg Variants

This subsection provides additional details regarding inference-time module usage across the two EvoReg variants, fast and full, as it pertains to the two established evaluation protocols.

Experiments under protocol one. Rigid EvoReg-Fast reports the raw network output without test-time refinement; Rigid EvoReg-Full applies Stage 3 CMA-ES refinement (5×15 within $\pm2^\circ/\pm0.05$), diffusion denoising (10 steps, noise 0.005), point-level Sinkhorn TTO (50 Adam steps, $\eta{=}10^{-3}$, $\lambda{=}1$), and a final Sinkhorn TTO over the predicted $(\mathbf{R}, \mathbf{t})$ recovered via Kabsch (80 steps, $\eta{=}5\times10^{-3}$, $\lambda{=}1$). Non-rigid EvoReg-Fast applies Stage 3 CMA-ES refinement (5×15 within $\pm2^\circ/\pm0.05$), diffusion denoising (20 steps, noise 0.001), and a light per-point Sinkhorn TTO (15 steps, $\eta{=}5\times10^{-3}$, $\lambda{=}1$); Non-rigid EvoReg-Full similarly applies Stage 3 CMA-ES refinement (5×15 within $\pm2^\circ/\pm0.05$), diffusion denoising (20 steps, noise 0.001), and a heavier per-point Sinkhorn TTO (80 steps, $\eta{=}5\times10^{-3}$, $\lambda{=}5$). All Sinkhorn solves use 10 unrolled iterations.

Experiments under protocol two. Every real-world scan configuration applies the Stage 3 CMA-ES refinement (5×15 within $\pm2^\circ/\pm0.05$) and a per-point Sinkhorn test-time optimization, while global (rigid pose) Sinkhorn TTO is disabled; on rigid 3DMatch a Kabsch step recovers a rigid transform from the refined points, whereas on non-rigid FAUST the deformation field is kept. On 3DMatch, both self-supervised variants use diffusion denoising (55 steps, noise 0.025) and a per-point Sinkhorn TTO ($\eta{=}3\times10^{-2}$, $\lambda{=}1$), with EvoReg-Full running 50 Sinkhorn steps and EvoReg-Fast 20; the supervised variants are identical except for diffusion denoising (50 steps, noise 0.025). On FAUST intra-subject, self-supervised EvoReg-Full applies diffusion denoising (20 steps, noise 0.001) and an 80-step Sinkhorn TTO ($\eta{=}5\times10^{-3}$, $\lambda{=}5$) while EvoReg-Fast reduces the Sinkhorn TTO to 30 steps. The supervised variants instead use diffusion denoising (200 steps, noise 0.02) with a 50-step Sinkhorn TTO ($\eta{=}5\times10^{-3}$, $\lambda{=}1$) for EvoReg-Full and diffusion denoising (100 steps, noise 0.02) with a 20-step Sinkhorn TTO for EvoReg-Fast. On FAUST inter-subject, the supervised variants use diffusion denoising (20 steps, noise 0.001) and a Sinkhorn TTO ($\eta{=}5\times10^{-3}$, $\lambda{=}1$) with 80 steps for EvoReg-Full and 30 steps for EvoReg-Fast; the self-supervised variants match this but with $\lambda{=}5$, and EvoReg-Fast additionally raises the diffusion noise to 0.003.

C.3 Additional Datasets

ShapeNet

Results and analysis. Table C1 reports the rigid ShapeNet-13 results. EvoReg-sup-full achieves the best overall CD (0.00286), while EvoReg-self-sup-full achieves the best overall EMD (0.00661), SWD (0.00331), and P2P (0.01851). Relative to the strongest non-EvoReg supervised baseline under CD, DCP (0.07289), EvoReg-sup-full reduces CD by 96.1%, and EvoReg-self-sup-full reduces it by 95.3%. CPD remains strongest on RE_g, RE, TE, and CD-Recall in the self-supervised/optimization block, while the fast EvoReg variants retain 99.84–99.85% CD-Recall at 0.093–0.095 s per pair. Table C2 reports the corresponding non-rigid ShapeNet-13 results. Self-supervised EvoReg-Full similarly leads on CD (0.00367) and EMD (0.01355), and ties with NDP on CD-Recall (100%). Notably, our fast variant, which runs approximately $125\times$ faster than the closest competing method, NDP, also beats it on CD and EMD, and ties on CD-Recall. The supervised variants mirror the ModelNet40 ranking: EvoReg-Full wins CD and EMD, and once again achieves near-perfect CD-Recall.

FAUST

Results and analysis. Table C3 reports the rigid FAUST results. EvoReg-self-sup-full achieves the best overall CD (0.00211), RE_g (0.01719), EMD (0.00876), SWD (0.00261), and P2P (0.01007), while maintaining 100.0% CD-Recall. Relative to the strongest non-EvoReg baseline under CD, CPD (0.02691), this corresponds to a 92.2% reduction in CD; EvoReg-sup-full achieves CD=0.00222, a

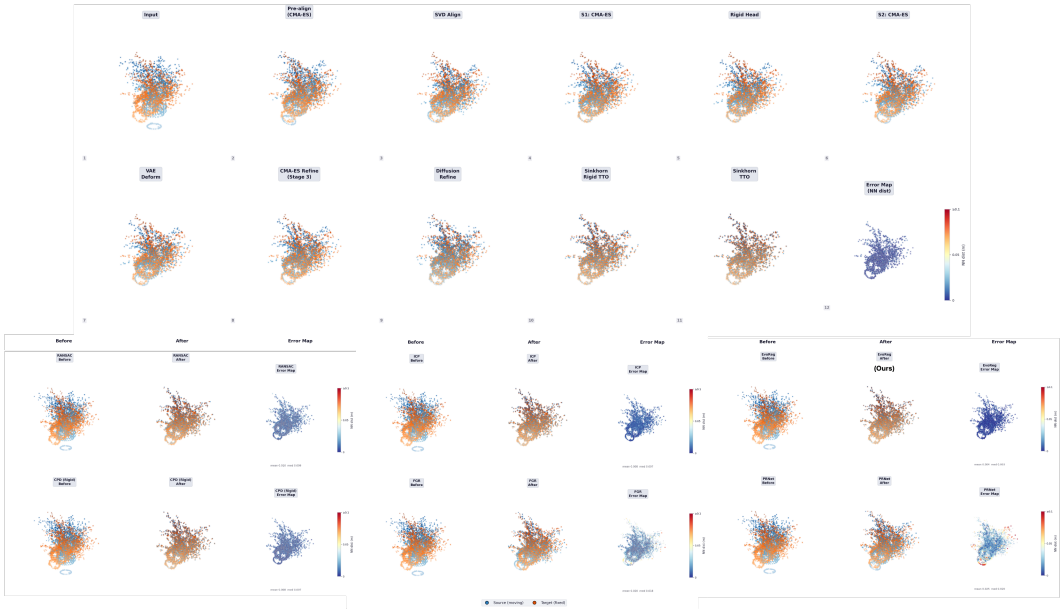


Figure 2: Top: Qualitative stage-by-stage registration example on ModelNet40 for the rigid self-supervised EvoReg variant. The moving source (blue) is progressively aligned to the fixed target (orange) through pre-alignment. Bottom: qualitative baseline comparison on the same ModelNet40 example under rigid self-supervised registration. Rows show EvoReg and representative baselines, while columns show the input pair, final alignment, and nearest-neighbor error map. EvoReg produces the cleanest overlap and the smallest residual error.

Table C1: Rigid evaluation results on ShapeNet-13. **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised									
EvoReg-self-sup-full (Ours)	**0.00345**	0.02557	**0.00661**	**0.00331**	**0.01851**	99.84	2.009	0.00343	0.838
EvoReg-selfsup-fast (Ours)	0.04660	0.04301	0.03310	0.01236	0.03990	99.85	3.978	0.01393	0.093
PRNet (Wang & Solomon, 2019b)	0.09381	0.13243	0.05292	0.03386	0.12384	92.99	15.075	0.04361	0.033
CPD (Myronenko & Song, 2010)	0.03004	**0.02184**	0.02148	0.00344	0.02019	**99.99**	**0.714**	**0.00093**	1.622
ICP (Besl & McKay, 1992b)	0.04495	0.04570	0.02971	0.01253	0.04213	98.95	3.138	0.00746	**0.003**
FGR (Zhou et al., 2016)	0.05211	0.04267	0.03532	0.01501	0.03934	99.00	3.506	0.01687	0.066
RANSAC (Fischler & Bolles, 1981)	0.04777	0.03549	0.03410	0.01350	0.03283	99.95	2.778	0.01530	0.042
Supervised									
EvoReg-sup-full (Ours)	**0.00286**	**0.02761**	**0.00710**	**0.00360**	**0.02015**	**99.90**	**2.040**	**0.00349**	0.379
EvoReg-sup-fast (Ours)	0.04581	0.04230	0.03250	0.01198	0.03926	99.84	3.840	0.01384	0.095
DCP (Wang & Solomon, 2019a)	0.07289	0.06332	0.05085	0.01891	0.05796	99.10	8.080	0.00702	0.010
RPMNet (Yew & Lee, 2020)	0.13304	0.17070	0.07920	0.04268	0.15832	94.88	20.849	0.03078	0.014
PointNetLK (Aoki et al., 2019)	0.21477	0.25842	0.10609	0.10782	0.25159	81.43	14.931	0.20523	0.035
DeepGMR (Yuan et al., 2020)	0.27919	0.61683	0.09297	0.08222	0.56318	48.29	97.844	0.03084	0.109
iPCRNet (Sarode et al., 2019)	0.33670	0.43757	0.09597	0.13750	0.40475	30.70	50.803	0.18422	**0.009**
GeoTransformer (Qin et al., 2023)	0.07517	0.06260	0.03290	0.02546	0.05702	95.73	5.856	0.03440	0.195

91.8% reduction. CPD remains strongest on RE (0.125°) and TE (0.00056), while all EvoReg variants preserve perfect CD-Recall and the fast variants reduce inference time to 0.282–0.305 s. Table C4 reports the corresponding non-rigid FAUST results. In the self-supervised setting, EvoReg-selfsup-full achieves the best CD (0.00395) and EMD (0.01470) while maintaining 100.00% CD-Recall; EvoReg-selfsup-fast also outperforms NDP on CD and EMD, while preserving the same perfect CD-Recall and reducing inference time from 31.119 s to 0.267 s. In the supervised setting, EvoReg-sup-full again achieves the best CD (0.00442) and EMD (0.01468), while DefTransNet remains strongest on RE_g, SWD, P2P, RE, TE, and inference time. Overall, the FAUST results show that EvoReg transfers strongly to articulated human shapes in both rigid and non-rigid settings, with the full variants leading on the primary geometric alignment metrics and the fast variants preserving most of that accuracy at substantially lower runtime.

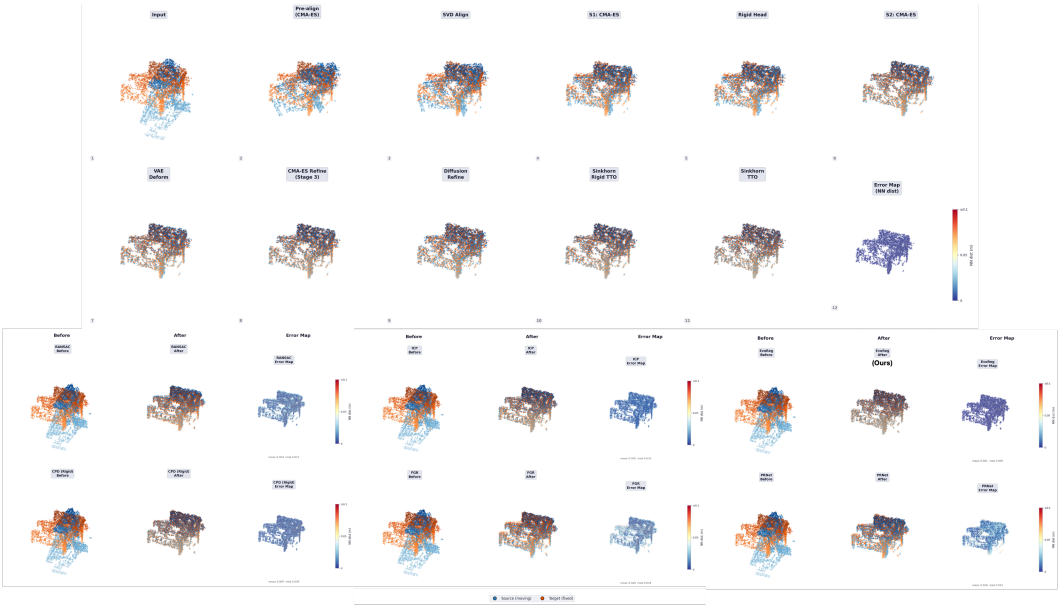


Figure 3: Top: Qualitative stage-by-stage registration example on 3DMatch for the rigid self-supervised EvoReg variant. The moving source (blue) is progressively aligned to the fixed target (orange) through pre-alignment. Bottom: qualitative baseline comparison on the same 3DMatch example under rigid self-supervised registration. Rows show EvoReg and representative baselines, while columns show the input pair, final alignment, and nearest-neighbor error map. EvoReg produces the cleanest overlap and the smallest residual error.

Table C2: Non-rigid evaluation results on ShapeNet-13. **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised									
EvoReg-selfsup-full (Ours)	**0.00367**	0.06071	**0.01355**	0.01086	0.04678	**100.00**	5.722	0.03668	0.673
EvoReg-selfsup-fast (Ours)	0.01097	0.06114	0.01733	0.01089	0.04786	**100.00**	5.722	0.03668	**0.254**
CPD-NR (Myronenko & Song, 2010)	0.08825	0.10514	0.06336	0.02457	0.09585	99.19	9.058	0.03353	2.001
BCPD (Hirose, 2021)	0.18610	0.27549	0.11392	0.07263	0.25061	94.61	18.923	0.03577	2.705
NDP (Li & Harada, 2022)	0.02312	**0.02673**	0.01853	**0.00386**	**0.02350**	**100.00**	**3.961**	**0.03060**	31.889
Supervised									
EvoReg-sup-full (Ours)	**0.00405**	0.06155	**0.01368**	0.01132	0.04750	99.94	5.561	0.03646	0.673
EvoReg-sup-fast (Ours)	0.01259	0.06177	0.01817	0.01142	0.04851	99.93	5.561	0.03646	0.253
DefTransNet (Monji-Azad et al., 2025)	0.01344	**0.03300**	0.01864	**0.00537**	**0.02100**	**100.00**	**3.223**	**0.03015**	**0.117**

C.4 Statistical Significance Testing

C.4.1 Protocol One Testing

Rigid. On ModelNet40 (Table C5), EvoReg-sup-full is unbeaten ($55/56$ wins, with the non-win being a tie) and EvoReg-selfsup-full is significantly better than every learning-based baseline; the only competitive baseline is classical CPD (Myronenko & Song, 2010), which significantly beats EvoReg-selfsup-full on six of eight metrics.

On 3DMatch (Table C6), EvoReg-sup-full again wins every comparison ($53/56$, no losses), and EvoReg-selfsup-full is only significantly outperformed by CPD (Myronenko & Song, 2010) on rotation and translation error.

On ShapeNet-13 (Table C7; n=8,762), EvoReg-sup-full wins $55/56$ comparisons with no losses; EvoReg-selfsup-full is significantly worse than CPD only on the four pose-related metrics (RE_g, CD-Recall, RE, TE).

On FAUST (Table C8; n=100), reduced statistical power yields more ties, but EvoReg-sup-full remains unbeaten ($52/56$, no losses) and EvoReg-selfsup-full's only significant losses are to CPD (Myronenko & Song, 2010) on RE and TE.

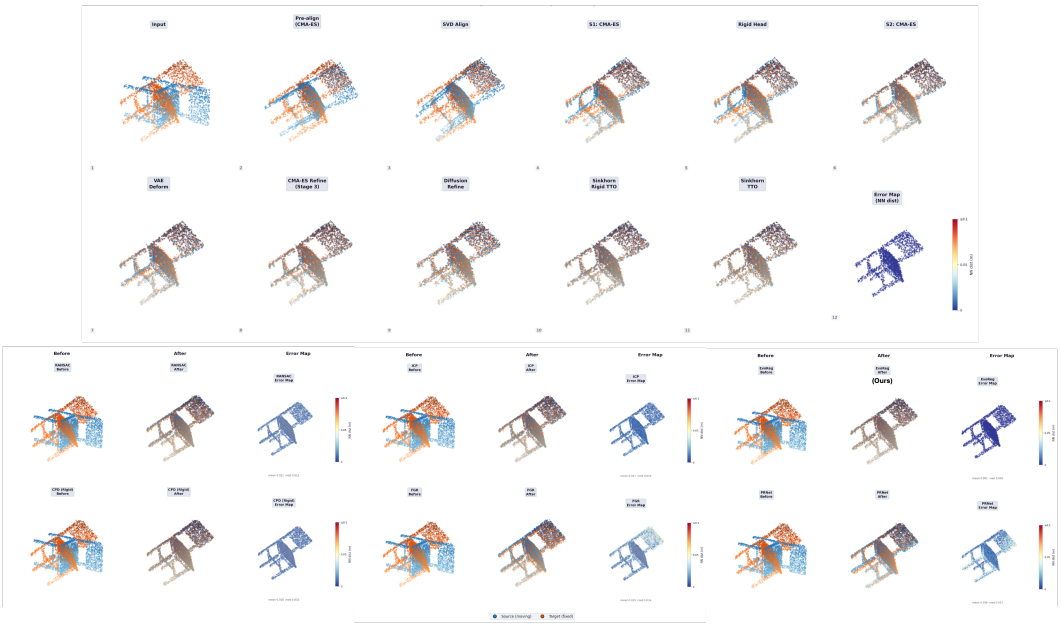


Figure 4: Top: Qualitative stage-by-stage registration example on ShapeNet for the rigid self-supervised EvoReg variant. The moving source (blue) is progressively aligned to the fixed target (orange) through pre-alignment. Bottom: qualitative baseline comparison on the same ShapeNet example under rigid self-supervised registration. Rows show EvoReg and representative baselines, while columns show the input pair, final alignment, and nearest-neighbor error map. EvoReg produces the cleanest overlap and the smallest residual error.

Table C3: Rigid evaluation results on FAUST. **Bold**: best, underline: second best. CD: Chamfer Distance, RE_g: Registration Error, EMD: Earth Mover's Distance, SWD: Sliced Wasserstein Distance, P2P: Point-to-Point, CD-Recall, RE: Rotation Error, TE: Translation Error, Time: per-pair inference time in seconds.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised									
EvoReg-self-sup-full (Ours)	**0.00211**	**0.01719**	**0.00876**	**0.00261**	**0.01007**	**100.0**	0.523	0.00179	0.824
EvoReg-selfsup-fast (Ours)	0.03678	0.02702	0.02965	0.00881	0.02502	**100.0**	3.631	0.00979	0.282
PRNet (Wang & Solomon, 2019b)	0.05093	0.04904	0.03876	0.01983	0.04591	99.00	4.114	0.02996	0.150
CPD (Myronenko & Song, 2010)	0.02691	0.01730	0.02216	0.00291	0.01594	**100.0**	**0.125**	**0.00056**	1.732
ICP (Besl & McKay, 1992b)	0.03387	0.02725	0.02594	0.00813	0.02451	**100.0**	1.115	0.00434	**0.005**
FGR (Zhou et al., 2016)	0.03297	0.02237	0.02642	0.00703	0.02053	**100.0**	1.625	0.00493	0.094
RANSAC (Fischler & Bolles, 1981)	0.03794	0.02714	0.03054	0.01000	0.02509	**100.0**	2.631	0.01177	0.037
Supervised									
EvoReg-sup-full (Ours)	**0.00222**	**0.01776**	**0.00909**	**0.00264**	**0.01074**	**100.0**	**0.227**	**0.00167**	0.397
EvoReg-sup-fast (Ours)	0.03442	0.02477	0.02837	0.00883	0.02281	**100.0**	2.098	0.00965	0.305
DCP (Wang & Solomon, 2019a)	0.05154	0.04361	0.04219	0.01387	0.03992	**100.0**	8.225	0.00731	0.089
RPMNet (Yew & Lee, 2020)	0.08904	0.09600	0.06678	0.03085	0.08884	97.00	17.601	0.02944	0.063
PointNetLK (Aoki et al., 2019)	0.19538	0.21806	0.11029	0.10439	0.21520	86.00	7.203	0.20466	0.041
DeepGMR (Yuan et al., 2020)	0.13832	0.26524	0.08676	0.04275	0.23995	90.00	75.844	0.02168	0.111
iPCRNet (Sarode et al., 2019)	0.30303	0.35518	0.10340	0.13121	0.32730	37.00	51.555	0.18628	**0.010**
GeoTransformer (Qin et al., 2023)	0.04171	0.03221	0.03483	0.01333	0.02976	**100.0**	2.253	0.01767	0.193

Non-rigid. On ModelNet40 in the non-rigid setting (Table C9), EvoReg dominates the distribution-level metrics (CD, EMD) against every baseline, but loses on per-point pose metrics: NDP (Li & Harada, 2022) significantly beats EvoReg-selfsup-full on three pose metrics, and DefTransNet beats EvoReg-sup-full on five (RE_g, SWD, P2P, RE, TE).

On 3DMatch (Table C10), the same trade-off holds: EvoReg significantly wins CD and EMD against every baseline, while NDP and DefTransNet each significantly beat the corresponding EvoReg reference on five pose-related metrics.

On ShapeNet-13's n=8,762 pairs (Table C11), nearly every comparison is significant; EvoReg wins CD and EMD universally, loses to NDP on five pose metrics (and to CPD-NR on TE), and to DefTransNet on the same five pose metrics in the supervised section.

Table C4: Quantitative results on FAUST (non-rigid). **Bold**: best, underline: second best. CD: Chamfer Distance, RE_g: Registration Error, EMD: Earth Mover's Distance, SWD: Sliced Wasserstein Distance, P2P: Point-to-Point, CD-Recall, RE: Rotation Error, TE: Translation Error, Time: per-pair inference time in seconds.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised									
EvoReg-selfsup-full (Ours)	**0.00395**	0.04620	**0.01470**	0.00678	0.03697	**100.00**	4.849	0.03215	0.696
EvoReg-selfsup-fast (Ours)	0.01026	0.04680	0.01742	0.00678	0.03787	**100.00**	4.849	0.03215	**0.267**
CPD-NR (Myronenko & Song, 2010)	0.07090	0.08300	0.05922	0.02847	0.07362	**100.00**	6.887	0.03659	1.238
BCPD (Hirose, 2021)	0.12100	0.19025	0.09705	0.05943	0.16697	**100.00**	11.494	0.04432	2.413
NDP (Li & Harada, 2022)	0.02008	**0.01709**	0.02076	**0.00314**	**0.01488**	**100.00**	**2.725**	**0.02947**	31.119
Supervised									
EvoReg-sup-full (Ours)	**0.00442**	0.04678	**0.01468**	0.00714	0.03738	**100.00**	4.500	0.03171	0.682
EvoReg-sup-fast (Ours)	0.01173	0.04727	0.01792	0.00726	0.03822	**100.00**	4.500	0.03171	0.265
DefTransNet (Monji-Azad et al., 2025)	0.01371	**0.02258**	0.02013	**0.00421**	**0.01661**	**100.00**	**2.887**	**0.02928**	**0.125**

Table C5: Statistical significance of EvoReg vs. each baseline on ModelNet40 (rigid). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with †** indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised (reference: EvoReg-selfsup-full)								
EvoReg-selfsup-fast (Ours)	***	***	***	***	***	*ns*	***	***
PRNet (Wang & Solomon, 2019b)	***	***	***	***	***	***	***	***
CPD (Myronenko & Song, 2010)	***	***** †**	***	***** †**	***** †**	***** †**	***** †**	***** †**
ICP (Besl & McKay, 1992b)	***	***	***	***	***	***	***	***
FGR (Zhou et al., 2016)	***	***	***	***	***	***	***	***
RANSAC (Fischler & Bolles, 1981)	***	***	***	***	***	***	***	***
Supervised (reference: EvoReg-sup-full)								
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	***	***
DCP (Wang & Solomon, 2019a)	***	***	***	***	***	***	***	***
RPMNet (Yew & Lee, 2020)	***	***	***	***	***	***	***	***
PointNetLK (Aoki et al., 2019)	***	***	***	***	***	***	***	***
DeepGMR (Yuan et al., 2020)	***	***	***	***	***	***	***	***
iPCRNet (Sarode et al., 2019)	***	***	***	***	***	***	***	***
GeoTransformer (Qin et al., 2023)	***	***	***	***	***	***	***	***

On FAUST (Table C12), smaller n yields more ties, but the pattern persists: EvoReg-selfsup-full beats CPD-NR and BCPD on all significant metrics, NDP beats it on four pose metrics, and DefTransNet beats EvoReg-sup-full on the four pose metrics where the test reaches significance.

C.4.2 Protocol Two Testing

On native 3DMatch (Table C13), both EvoReg references are significantly better than every baseline on all distributional-alignment metrics — CD, RE_g, EMD, SWD, P2P — and on CD-Recall ($p<0.001$ throughout), the sole exceptions being a handful of ties where a classical method is marginally stronger (CPD-R on RE_g/SWD, and CPD-R P2P is not significant). The reversal is confined to the pose metrics (RE, TE): the methods explicitly engineered for rigid-pose precision — TEASER++ and the correspondence-transformer baselines RoITr, PointDSC, CoFiNet, and Predator — are significantly better than EvoReg on RE and TE, whereas EvoReg remains significantly better than the weaker learned baselines (DCP, RPMNet, PointNetLK, DeepGMR, iPCRNet, GeoTransformer). This is exactly the tradeoff reported in Table E1: EvoReg wins geometric overlap and registration success with significance, while dedicated pose estimators win minimal rotation/translation error with significance. EvoReg-Full is also significantly better than EvoReg-Fast on the distributional metrics and pose, confirming the inference-time refinements contribute measurably; CD-Recall is not significant between them, as both saturate at 99.4%.

On FAUST intra-subject deformation (Table C14), EvoReg-Full is significantly better than every baseline on CD and EMD ($p<0.001$), the core distributional-alignment metrics. CD-Recall is not significant for any method, since every method reaches 100% recall against the 0.15 threshold, leaving no discordant pairs for McNemar. On the finer correspondence- and pose-oriented metrics (RE_g, SWD, P2P, and especially TE), the deformation-specialized baselines — CPD-NR, NDP, and the supervised

Table C6: Statistical significance of EvoReg vs. each baseline on 3DMatch (rigid). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with †** indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	***	***	*ns*	***	***
PRNet (Wang & Solomon, 2019b)	***	***	***	***	***	***	***	***
CPD (Myronenko & Song, 2010)	***	***	***	***	***	*ns*	***** †**	***** †**
ICP (Besl & McKay, 1992b)	***	***	***	***	***	**	***	***
FGR (Zhou et al., 2016)	***	***	***	***	***	*ns*	***	***
RANSAC (Fischler & Bolles, 1981)	***	***	***	***	***	*ns*	***	***
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	***	***
DCP (Wang & Solomon, 2019a)	***	***	***	***	***	*ns*	***	***
RPMNet (Yew & Lee, 2020)	***	***	***	***	***	***	***	***
PointNetLK (Aoki et al., 2019)	***	***	***	***	***	***	***	***
DeepGMR (Yuan et al., 2020)	***	***	***	***	***	***	***	***
iPCRNet (Sarode et al., 2019)	***	***	***	***	***	***	***	***
GeoTransformer (Qin et al., 2023)	***	***	***	***	***	*ns*	***	***

Table C7: Statistical significance of EvoReg vs. each baseline on ShapeNet-13 (rigid). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with †** indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	***	***	*ns*	***	***
PRNet (Wang & Solomon, 2019b)	***	***	***	***	***	***	***	***
CPD (Myronenko & Song, 2010)	***	***** †**	***	***	***	***** †**	***** †**	***** †**
ICP (Besl & McKay, 1992b)	***	***	***	***	***	***	***	***
FGR (Zhou et al., 2016)	***	***	***	***	***	***	***	***
RANSAC (Fischler & Bolles, 1981)	***	***	***	***	***	*ns*	***	***
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	***	***
DCP (Wang & Solomon, 2019a)	***	***	***	***	***	***	***	***
RPMNet (Yew & Lee, 2020)	***	***	***	***	***	***	***	***
PointNetLK (Aoki et al., 2019)	***	***	***	***	***	***	***	***
DeepGMR (Yuan et al., 2020)	***	***	***	***	***	***	***	***
iPCRNet (Sarode et al., 2019)	***	***	***	***	***	***	***	***
GeoTransformer (Qin et al., 2023)	***	***	***	***	***	***	***	***

DefTransNet — are significantly better, reflecting their optimization for dense correspondence and translation precision rather than global shape overlap; RE is not significantly different for any baseline. EvoReg's advantage on this benchmark is thus localized to global distributional alignment (CD/EMD), while the baselines retain a significant edge on translation/correspondence fidelity.

On FAUST inter-subject (body-shape) variation (Table C15), EvoReg-Full's distributional advantage widens relative to the intra-subject case: against the classical baselines CPD-NR and BCPD it is now significantly better not only on CD and EMD but also on SWD and P2P ($p<0.001$). CD-Recall is again non-significant (all methods at 100%). The correspondence-specialized methods NDP and DefTransNet are significantly better on the fine-grained metrics (RE_g, SWD, P2P) and on pose (RE, TE), consistent with their design goal of accurate dense correspondence. The pattern is the recurring one: EvoReg significantly wins the broad shape-overlap metrics, while methods built for correspondence/pose significantly win the fine pose metrics.

Across all three native benchmarks, the significance tests confirm a consistent and interpretable pattern: EvoReg-Full is significantly better than every baseline on the distributional-alignment

Table C8: Statistical significance of EvoReg vs. each baseline on FAUST (rigid). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with †** indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	***	***	*ns*	***	***
PRNet (Wang & Solomon, 2019b)	***	***	***	***	***	*ns*	***	***
CPD (Myronenko & Song, 2010)	***	*	***	**	***	*ns*	***†	***†
ICP (Besl & McKay, 1992b)	***	***	***	***	***	*ns*	*ns*	*ns*
FGR (Zhou et al., 2016)	***	***	***	***	***	*ns*	***	***
RANSAC (Fischler & Bolles, 1981)	***	***	***	***	***	*ns*	***	***
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	***	***
DCP (Wang & Solomon, 2019a)	***	***	***	***	***	*ns*	***	***
RPMNet (Yew & Lee, 2020)	***	***	***	***	***	*ns*	***	***
PointNetLK (Aoki et al., 2019)	***	***	***	***	***	***	***	***
DeepGMR (Yuan et al., 2020)	***	***	***	***	***	**	***	***
iPCRNet (Sarode et al., 2019)	***	***	***	***	***	***	***	***
GeoTransformer (Qin et al., 2023)	***	***	***	***	***	*ns*	***	***

Table C9: Statistical significance of EvoReg vs. each baseline on ModelNet40 (non-rigid). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with †** indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	***	***	*ns*	*ns*	*ns*
CPD-NR (Myronenko & Song, 2010)	***	***	***	***	***	***	*ns*	*ns*
BCPD (Hirose, 2021)	***	***	***	***	***	***	***	***
NDP (Li & Harada, 2022)	***	***†	***	**	***†	*ns*	***†	***
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	*ns*	*ns*
DefTransNet (Monji-Azad et al., 2025)	***	***†	***	***†	***†	*ns*	***†	***†

metrics (CD and EMD everywhere, frequently SWD and P2P), CD-Recall is significant only on 3DMatch (where recall is discriminative) and saturates on FAUST, and the pose metrics (RE/TE) favor whichever specialized pose- or correspondence-estimator each benchmark provides — never overturning EvoReg's distributional advantage.

D Comprehensive Ablations

D.1 Main Ablations Study

The primary component ablation study, isolating the effects of key architectural elements utilized in EvoReg, is depicted in this section.

D.2 Inference-Time Ablations

All inference-time ablations use the same two pretrained models, both trained self-supervised for 200 epochs on the full ModelNet40 training set (9,843 samples, 1,024 points per cloud) with UnitBall normalization. Both models share the following training configuration: batch size 16, learning rate 10^{-3} with step decay (factor 0.5 every 50 epochs), rotation range $\pm 45°$, translation range ± 1.0, VAE latent dimension 128, 3 SVD iterations, CMA-ES pre-alignment (50 particles, 30 iterations) with progressive inter-stage refinements ($\pm 20°/\pm 0.5 \rightarrow \pm 10°/\pm 0.2 \rightarrow \pm 5°/\pm 0.1$), diffusion score-matching loss (1,000 steps, weight 1.0), correspondence projection dimension 256, temperature

Table C10: Statistical significance of EvoReg vs. each baseline on 3DMatch (non-rigid). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with †** indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	*ns*	***	*ns*	*ns*	*ns*
CPD-NR (Myronenko & Song, 2010)	***	*ns*	***	***	***	*ns*	***	*ns*
BCPD (Hirose, 2021)	***	***	***	***	***	*ns*	***	***
NDP (Li & Harada, 2022)	***	***†	***	***†	***†	*ns*	***†	***†
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	*ns*	*ns*
DefTransNet (Monji-Azad et al., 2025)	***	***†	***	***†	***†	*ns*	***†	***†

Table C11: Statistical significance of EvoReg vs. each baseline on ShapeNet-13 (non-rigid). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with †** indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	***	***	*ns*	*ns*	*ns*
CPD-NR (Myronenko & Song, 2010)	***	***	***	***	***	***	***	***†
BCPD (Hirose, 2021)	***	***	***	***	***	***	***	*ns*
NDP (Li & Harada, 2022)	***	***†	***	***†	***†	*ns*	***†	***†
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	*ns*	*ns*
DefTransNet (Monji-Azad et al., 2025)	***	***†	***	***†	***†	*ns*	***†	***†

0.1, loss weights: $\lambda_{\text{chamfer}} = 1.5$, $\lambda_{\text{rigid_chamfer}} = 0.5$, $\lambda_{\text{intermediate_chamfer}} = 0.5$, $\lambda_{\text{correspondence}} = 0.1$, $\lambda_{\text{cycle}} = 0.1$, $\lambda_{\text{KL}} = 0.01$, and dropout occlusion augmentation (ratio 0.2, probability 0.5). The two models differ only in: (1) the non-rigid model includes RBF-based non-rigid deformations in the training data pipeline, and (2) centroid loss weights $\lambda_c = 0.1$ (rigid) vs. $\lambda_c = 0.3$ (non-rigid). Evaluation is performed on the full ModelNet40 test set (2,468 samples) with rotation range $\pm 45°$ and translation range ± 0.2.

Table D4: Preliminary diffusion refinement noise sweep on ModelNet40 (200 epochs, 50 refinement steps, CD-Recall threshold = 0.15). σ=0.001 yields the best result.

Noise σ	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓
No diffusion	0.0582	0.0662	0.0465	0.0210	0.0609	98.9	7.10	0.0231
0.001 (default non-rigid)	0.0236	0.0644	0.0346	0.0155	0.0573	100.0	7.06	0.0228
0.005 (default rigid)	0.0259	0.1027	0.0422	0.0179	0.0903	100.0	7.03	0.0230
0.01	0.0370	0.1325	0.0519	0.0221	0.1167	100.0	7.32	0.0227
0.02	0.0496	0.1738	0.0642	0.0271	0.1516	100.0	7.12	0.0226
0.05	0.0783	0.3042	0.0885	0.0483	0.2614	100.0	7.11	0.0228
0.1	0.1342	0.4961	0.1059	0.0932	0.4347	99.8	7.11	0.0228

Diffusion Module Noise Level. At inference time, the diffusion refinement module (Stage 4) applies controlled noise followed by iterative denoising to refine the registration output. We do a preliminary sweep of the noise level σ using the full 200-epoch model with 50 refinement steps; results are shown in Appendix Table D4. We later complete a final noise level sweep for both rigid and non-rigid tasks; these results are shown in Tables D5 and D6, respectively. For rigid registration, a noise level of 0.005 achieves desirable and balanced results, while for non-rigid registration a noise level of 0.001 achieves a similar outcome.

Table C12: Statistical significance of EvoReg vs. each baseline on FAUST (non-rigid). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with** † indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	*ns*	***	*ns*	*ns*	*ns*
CPD-NR (Myronenko & Song, 2010)	***	***	***	***	***	*ns*	*	**
BCPD (Hirose, 2021)	***	***	***	***	***	*ns*	***	***
NDP (Li & Harada, 2022)	***	***†	***	***†	***†	*ns*	***†	*ns*
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	*ns*	*ns*
DefTransNet (Monji-Azad et al., 2025)	***	***†	***	***†	***†	*ns*	***†	*ns*

Table C13: Statistical significance of EvoReg vs. each baseline on 3DMatch (rigid, native pairs). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $*** \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with** † indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	***	***	*ns*	*†	***†
PRNet (Wang & Solomon, 2019b)	***	***	***	***	***	***	***†	***
CPD-R (Myronenko & Song, 2010)	***	**†	***	*†	*ns*	***	***	***
ICP (Besl & McKay, 1992b)	***	***	***	***	***	***	**†	*ns*
FGR (Zhou et al., 2016)	***	***	***	***	***	***	***	***
RANSAC (Fischler & Bolles, 1981)	***	***	***	***	***	***	**	***†
TEASER++ (Yang et al., 2021)	***	***	***	***	***	***	***†	***†
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***	***	*ns*	**†	***†
DCP (Wang & Solomon, 2019a)	***	*ns*	***	***	***	***	***	***
RPMNet (Yew & Lee, 2020)	***	*ns*	***	***	***	***	***	***
PointNetLK (Aoki et al., 2019)	***	***	***	***	***	***	***	***
DeepGMR (Yuan et al., 2020)	***	***	***	***	***	***	***	***
iPCRNet (Sarode et al., 2019)	***	***	***	***	***	***	***	***
GeoTransformer (Qin et al., 2023)	***	***	***	***	***	***	***	***
RoITr (Yu et al., 2023)	***	***	***	***	***	***	***†	***†
PointDSC (Bai et al., 2021)	***	***	***	***	***	***	***†	***†
CoFiNet (Yu et al., 2021)	***	***	***	***	***	***	***†	***†
Predator (Huang et al., 2021)	***	***	***	***	***	***	***†	***†

Table D5: Final diffusion noise level (σ) ablation on rigid ModelNet40 (self-supervised, λ_c=0.1). **Bold**: best, underline: second best. A noise level of 0.005 achieves a very balanced result.

Noise σ	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
No diffusion	0.0347	0.0472	0.0296	0.0125	0.0436	99.4	5.11	0.0096	**0.132**
0.0005	0.0206	**0.0467**	**0.0246**	0.0112	**0.0415**	99.5	5.11	0.0096	0.256
0.001	0.0153	0.0553	0.0248	**0.0101**	0.0492	99.5	5.13	0.0092	0.261
0.002	0.0151	0.0691	0.0284	0.0125	0.0608	99.4	4.93	0.0090	0.266
0.003	0.0155	0.0792	0.0301	0.0139	0.0694	99.4	5.23	0.0095	0.267
0.004	0.0151	0.0841	0.0310	0.0141	0.0735	99.3	4.90	0.0092	0.264
0.005 (default)	**0.0148**	0.0875	0.0317	0.0141	0.0761	99.5	**4.76**	**0.0088**	0.264
0.01	0.0156	0.1141	0.0356	0.0161	0.0984	99.6	5.05	0.0096	0.265
0.02	0.0190	0.1579	0.0446	0.0209	0.1335	**100.0**	4.96	0.0091	0.266
0.03	0.0242	0.2055	0.0536	0.0280	0.1721	**100.0**	5.25	0.0093	0.266
0.04	0.0287	0.2503	0.0598	0.0358	0.2089	**100.0**	5.25	0.0093	0.270
0.05	0.0396	0.2947	0.0664	0.0439	0.2464	**100.0**	5.20	0.0089	0.265
0.075	0.0653	0.3993	0.0761	0.0678	0.3392	**100.0**	5.20	0.0089	0.265
0.1	0.1076	0.4956	0.0843	0.0951	0.4298	92.5	5.20	0.0089	0.266

Table C14: Statistical significance of EvoReg vs. each baseline on FAUST intra-subject (non-rigid, native pairs). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $* * * \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with** † indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	*ns*	***	***	*ns*	*ns*	*ns*	*ns*
CPD-NR (Myronenko & Song, 2010)	***	***†	***	*ns*	*ns*	*ns*	*ns*	***†
BCPD (Hirose, 2021)	***	*ns*	***	*ns*	***	*ns*	*ns*	***†
NDP (Li & Harada, 2022)	***	*ns*	***	*†	*ns*	*ns*	*ns*	***†
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***†	*	*ns*	***†	*ns*	*ns*	*ns*
DefTransNet (Monji-Azad et al., 2025)	***	***†	**	***†	***†	*ns*	*ns*	***†

Table C15: Statistical significance of EvoReg vs. each baseline on FAUST inter-subject (non-rigid, native pairs). Each cell summarizes a paired Wilcoxon signed-rank test (McNemar for CD-Recall), Holm-Bonferroni-corrected within each metric. Stars denote significance level: $* \; p<0.05$, $** \; p<0.01$, $* * * \; p<0.001$. Plain stars indicate the EvoReg reference is significantly *better*; **bold stars with** † indicate the reference is significantly *worse*; *ns* indicates no significant difference.

Baseline	CD	RE_g	EMD	SWD	P2P	CD-Recall	RE (°)	TE
Self-Supervised	(reference: EvoReg-selfsup-full)							
EvoReg-selfsup-fast (Ours)	***	***	***	***	***	*ns*	*ns*	*ns*
CPD-NR (Myronenko & Song, 2010)	***	*ns*	***	***	***	*ns*	***†	***†
BCPD (Hirose, 2021)	***	***	***	***	***	*ns*	***†	**†
NDP (Li & Harada, 2022)	***	***†	***	***†	***†	*ns*	***†	***†
Supervised	(reference: EvoReg-sup-full)							
EvoReg-sup-fast (Ours)	***	***	***	***†	***	*ns*	*ns*	*ns*
DefTransNet (Monji-Azad et al., 2025)	***	***†	***	***†	***†	*ns*	***†	***†

Table D6: Diffusion noise level (σ) ablation on non-rigid ModelNet40 (self-supervised, λ_c=0.3). **Bold**: best, underline: second best. A noise level of 0.001 achieves a very balanced result.

Noise σ	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
No diffusion	0.0461	0.0526	0.0390	0.0175	0.0473	99.8	**6.28**	**0.0338**	**0.182**
0.0005	0.0219	**0.0477**	0.0292	0.0139	**0.0409**	99.7	**6.28**	**0.0338**	0.263
0.001 (default)	0.0158	0.0548	**0.0288**	0.0134	0.0475	99.5	6.38	0.0339	0.260
0.002	0.0134	0.0685	0.0290	**0.0133**	0.0595	99.6	6.45	0.0341	0.264
0.003	0.0135	0.0776	0.0300	0.0137	0.0673	99.3	6.62	**0.0338**	0.263
0.004	**0.0130**	0.0820	0.0307	0.0138	0.0707	99.4	6.47	0.0341	0.199
0.005	0.0136	0.0888	0.0315	0.0143	0.0765	99.4	6.47	0.0343	0.200
0.01	0.0171	0.1140	0.0367	0.0165	0.0976	99.6	6.60	0.0345	0.200
0.02	0.0240	0.1588	0.0473	0.0221	0.1340	99.9	6.53	0.0342	0.199
0.03	0.0317	0.2047	0.0574	0.0291	0.1716	**100.0**	6.55	0.0340	0.200
0.04	0.0389	0.2504	0.0649	0.0373	0.2098	**100.0**	6.55	0.0340	0.202
0.05	0.0520	0.2955	0.0722	0.0465	0.2487	**100.0**	6.46	0.0340	0.205
0.075	0.0843	0.4025	0.0828	0.0722	0.3446	99.9	6.46	0.0340	0.202
0.1	0.1307	0.5021	0.0896	0.1014	0.4379	50.8	6.46	0.0340	0.199

CMA-ES Refinement Three - Population Size. We sweep the number of CMA-ES particles in the post-Stage 3 evolutionary refinement CMA-ES_3 across 8 values (tables D7, D8): $\{5, 10, 15, 25, 35, 50, 75, 100\}$, plus a baseline without CMA-ES_3. The number of CMA-ES iterations is fixed at 15 with a search range of $\pm 5°/\pm 0.1$. All runs include diffusion refinement at $\sigma = 0.001$ with 50 denoising steps.

Table D1: Rigid and non-rigid component ablations on ModelNet40; the corresponding EvoReg self-supervised model is used as the basis model for both modalities. Each row removes one component from the full EvoReg pipeline. **Bold**: best; underline: second best.

	Phase	EvoReg Variant	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
Rigid	Training	EvoReg Self-Sup. (Full)	0.01298	0.03964	0.01644	0.00589	0.03453	99.39	3.521	0.00405	1.014
		w/o CMA-ES Pre-alignment (Stage 0)	0.0214	0.0764	0.0175	0.0143	0.0655	96.6	10.5397	0.0144	0.3004
		w/o Iterative SVD (Stage 1)	0.1227	0.7988	0.0325	0.1156	0.7224	65.2	157.0645	0.1432	0.3004
		w/o Rigid Refinement (Stage 2)	0.0110	0.0416	0.0162	0.0059	0.0360	99.5	3.5650	0.0042	0.3774
		w/o VAE Deformation Field (Stage 3)	0.0129	0.0404	0.0165	0.0060	0.0352	99.5	3.76	0.0044	1.029
		w/o Inter-Stage CMA-ES	0.0117	0.0465	0.0168	0.0065	0.0400	99.4	4.3328	0.0043	0.2527
		w/o Occlusion	0.0109	0.0429	0.0161	0.0059	0.0373	99.4	3.7671	0.0042	0.3799
		w Supervision	0.01130	0.04264	0.01616	0.00605	0.03719	99.31	3.690	0.00440	0.381
	Inference	w/o Inf. CMA-ES Refinement	0.0128	0.0405	0.0165	**0.0058**	0.0354	99.5	3.63	**0.0040**	0.996
		w/o Diffusion Refinement	0.0127	0.0400	**0.0164**	0.0059	0.0350	99.5	**3.53**	0.0043	1.005
		w/o Sinkhorn Global TTO	**0.0077**	0.0687	0.0180	0.0068	0.0585	**99.6**	3.87	0.0053	**0.424**
		w/o Sinkhorn Per-Point TTO	0.0338	0.0446	0.0285	0.0118	0.0412	99.4	4.09	0.0047	0.748
		w/o Kabsch Alignment	0.0130	**0.0396**	0.0164	0.0059	**0.0345**	99.4	4.08	0.0047	1.023
Non-Rigid	Training	EvoReg Self-Sup. (Full)	0.00509	0.05525	0.01664	0.00723	0.04713	100.00	6.991	0.03423	0.676
		w/o CMA-ES Pre-alignment (Stage 0)	0.0063	0.0577	0.0167	0.0078	0.0491	99.8	7.66	0.0348	0.609
		w/o Iterative SVD (Stage 1)	0.1769	0.7572	0.0410	0.1314	0.6868	55.8	130.4262	0.1697	1.0473
		w/o Rigid Refinement (Stage 2)	0.0056	0.0500	0.0167	0.0070	0.0418	99.9	6.26	0.0341	0.612
		w/o VAE Deformation Field (Stage 3)	0.0063	0.0507	0.0169	0.0071	0.0424	99.8	6.26	0.0341	0.376
		w/o Inter-Stage CMA-ES	0.0081	0.0579	0.0176	0.0076	0.0479	99.2	8.5485	0.0303	0.5534
		w/o Occlusion	0.0061	0.0493	0.0165	0.0068	0.0412	99.8	5.8989	0.0341	0.4906
		w Supervision	0.00612	0.05719	0.01683	0.00787	0.04883	99.84	6.909	0.03471	0.675
	Inference	w/o Inf. CMA-ES Refinement	0.0052	0.0525	0.0168	0.0071	0.0441	100.0	6.54	0.0335	0.492
		w/o Diffusion Refinement	0.0101	0.0410	0.0167	0.0070	0.0342	99.8	6.48	0.0339	0.496
		w/o Sinkhorn Per-Point TTO	0.0158	0.0534	0.0297	0.0134	0.0465	100.0	6.18	0.0338	0.119

Table D2: Preliminary NIA and evolutionary comparison on ModelNet40 (200 epochs, self-supervised + diffusion, CD-Recall threshold = 0.15). DE and CMA-ES are the two best performing algorithms.

Algorithm	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓
PSO (Kennedy & Eberhart, 1995)	0.0521	0.0627	0.0409	0.0189	0.0586	98.4	6.82	0.0203
DE (Storn & Price, 1997)	0.0380	0.0508	0.0332	0.0135	0.0475	99.8	5.16	0.0163
CMA-ES (Hansen & Ostermeier, 2001)	0.0520	0.0581	0.0424	0.0186	0.0542	99.0	6.43	0.0196
GWO (Mirjalili et al., 2014)	0.0604	0.0711	0.0468	0.0217	0.0666	98.4	8.28	0.0236
Firefly (Yang, 2009)	0.1506	0.1972	0.0791	0.0472	0.1794	83.8	32.79	0.0167
PSO+DE (Kennedy & Eberhart, 1995; Storn & Price, 1997)	0.0574	0.0682	0.0447	0.0209	0.0641	98.3	7.46	0.0236
PSO+CMA-ES (Kennedy & Eberhart, 1995; Hansen & Ostermeier, 2001)	0.0554	0.0656	0.0430	0.0203	0.0612	98.3	7.41	0.0244
DE+CMA-ES (Storn & Price, 1997; Hansen & Ostermeier, 2001)	0.0470	0.0649	0.0374	0.0163	0.0606	98.5	8.24	0.0167

Table D7: CMA-ES_3 particle count ablation on rigid ModelNet40 (self-supervised, λ_c=0.1, σ=0.001). **Bold**: best, underline: second best. Once CMA-ES_3 is enabled, particle count has no effect on quality: every count from 5 to 100 produces identical metrics, indicating that 5 particles are enough to saturate the search at this stage of the registration process.

Particles	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
No CMA-ES_3	0.0154	**0.0550**	0.0249	0.0102	**0.0488**	99.3	**5.01**	**0.0091**	**0.230**
5	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.264
10	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.266
15	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.262
25	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.263
35	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.262
50	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.264
75	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.265
100	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.265

Table D8: CMA-ES_3 particle count ablation on non-rigid ModelNet40 (self-supervised, λ_c=0.3, σ=0.001). **Bold**: best, underline: second best. Once CMA-ES_3 is enabled, particle count has no effect on quality: every count from 5 to 100 produces identical metrics, indicating that 5 particles are enough to saturate the search at this stage of the registration process.

Particles	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
No CMA-ES_3	0.0165	0.0563	0.0292	0.0137	0.0489	**99.5**	6.50	0.0340	**0.227**
5	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.264
10	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.262
15	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.262
25	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.259
35	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.259
50	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.261
75	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.260
100	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.260

CMA-ES Refinement Three - Search Range. We sweep the rotation and translation search ranges of the post-Stage 3 evolutionary refinement CMA-ES_3 across 5 paired configurations (tables D9, D10): $\pm 2°/\pm 0.05$, $\pm 5°/\pm 0.1$, $\pm 10°/\pm 0.2$, $\pm 15°/\pm 0.3$, $\pm 20°/\pm 0.5$, plus a baseline

Table D3: Final evolutionary comparison on ModelNet40 (200 epochs, self-supervised with diffusion, n=2000 samples, CD-Recall threshold = 0.15). Head-to-head between the two top performers identified in Tab. D2. **Bold**: per-metric winner. CMA-ES is identified as the better algorithmic choice for our framework.

Algorithm	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓
CMA-ES (Hansen & Ostermeier, 2001)	0.0188	0.0916	**0.0344**	**0.0143**	0.0803	**100.0**	**4.87**	**0.0089**
DE (Storn & Price, 1997)	**0.0164**	0.0910	0.0361	0.0169	0.0794	**100.0**	5.82	0.0168

without CMA-ES_3. The number of particles is fixed at 25 with 15 iterations. All runs include diffusion refinement at $\sigma = 0.001$ with 50 denoising steps.

Table D9: CMA-ES_3 search range ablation on rigid ModelNet40 (self-supervised, λ_c=0.1, σ=0.001). **Bold**: best, underline: second best. Once CMA-ES_3 is enabled, the search range is essentially inert: every tested range from $\pm 2°/\pm 0.05$ to $\pm 20°/\pm 0.5$ produces identical metrics on every measure, indicating that the optimum lies well within the narrowest tested neighborhood.

Range	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
No CMA-ES_3	0.0154	**0.0550**	0.0249	0.0102	**0.0488**	99.3	**5.01**	**0.0091**	**0.226**
$\pm 2°/\pm 0.05$	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.256
$\pm 5°/\pm 0.1$	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.257
$\pm 10°/\pm 0.2$	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.258
$\pm 15°/\pm 0.3$	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.258
$\pm 20°/\pm 0.5$	**0.0153**	0.0553	**0.0248**	**0.0101**	0.0492	**99.5**	5.13	0.0092	0.264

Table D10: CMA-ES_3 search range ablation on non-rigid ModelNet40 (self-supervised, λ_c=0.3, σ=0.001). **Bold**: best, underline: second best. Once CMA-ES_3 is enabled, the search range is essentially inert: every tested range from $\pm 2°/\pm 0.05$ to $\pm 20°/\pm 0.5$ produces identical metrics on every measure, indicating that the optimum lies well within the narrowest tested neighborhood.

Range	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
No CMA-ES_3	0.0165	0.0563	0.0292	0.0137	0.0489	**99.5**	6.50	0.0340	**0.228**
$\pm 2°/\pm 0.05$	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.261
$\pm 5°/\pm 0.1$	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.266
$\pm 10°/\pm 0.2$	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.266
$\pm 15°/\pm 0.3$	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.262
$\pm 20°/\pm 0.5$	**0.0158**	**0.0548**	**0.0288**	**0.0134**	**0.0475**	**99.5**	**6.38**	**0.0339**	0.266

Diffusion Refinement - Denoising Steps. We sweep the number of DDPM denoising steps in the inference-time diffusion refinement across 11 values: $5, 10, 20, 35, 50, 65, 80, 100, 120, 150, 200$, plus a baseline without diffusion refinement (see tables D11, D12). The noise level is fixed at $\sigma = 0.001$. All runs include CMA-ES_3 refinement with default settings (25 particles, 15 iterations, $\pm 5°/\pm 0.1$).

Table D11: Diffusion denoising steps ablation on rigid ModelNet40 (self-supervised, λ_c=0.1, σ=0.001). **Bold**: best, underline: second best. Past a certain point, diffusion denoising steps only change the results marginally — once the trajectory has effectively converged, additional steps essentially refine an already-clean signal.

Steps	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
No diffusion	0.0347	**0.0472**	0.0296	0.0125	**0.0436**	99.4	**5.11**	0.0096	**0.184**
5	0.0179	0.0550	0.0263	0.0101	0.0496	99.6	5.13	**0.0092**	0.190
10	0.0154	0.0549	0.0249	**0.0099**	0.0490	**99.7**	5.13	**0.0092**	0.196
20	**0.0153**	0.0550	**0.0248**	0.0100	0.0491	**99.7**	5.13	**0.0092**	0.213
35	**0.0153**	0.0552	**0.0248**	0.0100	0.0491	99.6	5.13	**0.0092**	0.237
50	**0.0153**	0.0553	**0.0248**	0.0101	0.0492	99.5	5.13	**0.0092**	0.258
65	**0.0153**	0.0553	**0.0248**	0.0101	0.0492	99.5	5.13	**0.0092**	0.285
80	**0.0153**	0.0554	**0.0248**	0.0102	0.0493	99.5	5.13	**0.0092**	0.310
100	**0.0153**	0.0556	**0.0248**	0.0103	0.0493	99.4	5.13	**0.0092**	0.338
120	**0.0153**	0.0557	**0.0248**	0.0103	0.0494	99.4	5.13	**0.0092**	0.373
150	0.0154	0.0558	**0.0248**	0.0104	0.0494	99.2	5.13	**0.0092**	0.421
200	0.0154	0.0560	**0.0248**	0.0105	0.0495	99.1	5.13	**0.0092**	0.411

Table D12: Diffusion denoising steps ablation on non-rigid ModelNet40 (self-supervised, λ_c=0.3, σ=0.001). **Bold**: best, underline: second best. Past a certain point, diffusion denoising steps only change the results marginally — once the trajectory has effectively converged, additional steps essentially refine an already-clean signal.

Steps	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
No diffusion	0.0461	**0.0526**	0.0390	0.0175	**0.0473**	**99.8**	**6.28**	**0.0338**	**0.129**
5	0.0247	0.0559	0.0334	0.0144	0.0499	**99.8**	6.38	0.0339	0.135
10	0.0172	0.0547	0.0299	0.0135	0.0478	**99.8**	6.38	0.0339	0.141
20	**0.0158**	0.0546	0.0289	**0.0132**	**0.0473**	**99.8**	6.38	0.0339	0.157
35	**0.0158**	0.0547	0.0288	0.0133	0.0474	99.6	6.38	0.0339	0.174
50	**0.0158**	0.0548	0.0288	0.0134	0.0475	99.5	6.38	0.0339	0.194
65	0.0159	0.0550	0.0288	0.0134	0.0475	99.4	6.38	0.0339	0.210
80	0.0160	0.0551	0.0288	0.0135	0.0476	99.4	6.38	0.0339	0.230
100	0.0161	0.0553	**0.0287**	0.0136	0.0477	99.3	6.38	0.0339	0.257
120	0.0161	0.0555	**0.0287**	0.0137	0.0478	99.2	6.38	0.0339	0.286
150	0.0162	0.0557	**0.0287**	0.0138	0.0478	99.1	6.38	0.0339	0.325
200	0.0163	0.0559	**0.0287**	0.0140	0.0479	99.1	6.38	0.0339	0.393

D.3 Training Ablations

All training ablations are conducted on ModelNet40 with 500 training samples (100 validation) for 100 epochs, using 1,024 points per cloud with UnitBall normalization. Unless otherwise stated, all training hyperparameters match those of the pretrained models described above, with two exceptions: the reduced sample and epoch count and a proportionally adjusted learning rate schedule (step decay every 25 epochs instead of 50). Each trained model is evaluated on the full ModelNet40 test set (2,468 samples) with rotation range $\pm 45°$, translation range ± 0.2, CD-Recall threshold 0.15, CMA-ES_3 refinement (25 particles, 15 iterations, $\pm 5°/\pm 0.1$), and diffusion refinement ($\sigma = 0.001$, 50 denoising steps). As with the pretrained models, rigid and non-rigid variants differ only in the inclusion of RBF deformations and the centroid loss weight ($\lambda_c = 0.1$ vs. $\lambda_c = 0.3$).

D.3.1 Main Losses

Chamfer Weight. We sweep the primary Chamfer distance loss weight λ_{chamfer} across 7 values: $0.2, 0.5, 1.0, 1.5, 2.0, 3.0, 5.0$ (see tables D13, D14). All other loss weights are held fixed at their default values. The current best configuration uses $\lambda_{\text{chamfer}} = 1.5$.

Table D13: Chamfer weight (λ_{CD}) training sweep — rigid model (self-supervised, 100 epochs, 500 samples). Evaluated on ModelNet40 (n=2468, $\pm 45°$, ± 0.2 translation) with CMA-ES S3 and diffusion refinement.

λ_{CD}	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
0.2	0.0347	0.0688	0.0404	0.0211	0.0616	**100.0**	5.22	0.0091	0.262
0.5	0.0487	0.0732	0.0470	0.0238	0.0669	**100.0**	5.53	0.0095	0.260
1.0	**0.0213**	**0.0574**	**0.0285**	**0.0119**	**0.0513**	**100.0**	5.66	0.0092	0.259
1.5	0.1180	1.0093	0.0721	0.1318	0.9504	99.8	175.95	0.1745	0.263
2.0	0.0388	0.0649	0.0398	0.0191	0.0596	**100.0**	**5.16**	0.0091	**0.195**
3.0	0.0347	0.0631	0.0375	0.0178	0.0572	**100.0**	5.22	**0.0089**	0.197
5.0	0.0269	0.0584	0.0316	0.0139	0.0527	**100.0**	5.17	0.0091	0.197

Table D14: Chamfer weight (λ_{CD}) training sweep — non-rigid model (self-supervised, 100 epochs, 500 samples). Evaluated on ModelNet40 (n=2468, $\pm 45°$, ± 0.2 translation) with CMA-ES S3 and diffusion refinement.

λ_{CD}	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Time (s) ↓
0.2	0.3880	0.4337	0.0987	0.2092	0.4009	62.8	175.78	0.1794	0.262
0.5	0.3113	0.4308	0.1063	0.1772	0.3953	66.9	175.51	0.1809	0.259
1.0	0.1642	1.0006	0.0881	0.1368	0.9421	95.2	174.37	0.1777	0.263
1.5	0.1525	1.0057	0.0831	0.1365	0.9491	98.6	175.69	0.1788	0.194
2.0	0.1442	1.0020	0.0801	0.1348	0.9463	98.9	175.52	0.1803	**0.193**
3.0	0.1212	1.0487	0.0718	0.1336	0.9870	99.8	175.61	0.1821	0.194
5.0	**0.0371**	**0.0652**	**0.0412**	**0.0206**	**0.0583**	**100.0**	**6.34**	**0.0342**	0.198

D.3.2 Choice of Gradient-Free Algorithm

Stage 0 performs gradient-free pre-alignment over $\mathrm{SE}(3)$ using a nature-inspired or evolutionary algorithm. We compare PSO, DE, CMA-ES, GWO, Firefly, and three hybrids under identical settings (Appendix Table D2); DE and CMA-ES perform the best. We further compare these two algorithms on a subsequent version of our model, the results are shown in Table D3; CMA-ES outperforms DE on most metrics and, importantly, on TE and RE — difficult metrics to find improvement on. We thus choose CMA-ES as our gradient-free refinement algorithm.

E Real-World Registration - Experiments and Analysis

Beyond the experiments reported in the main text and Appendices A–D, we conducted three further sets of experiments to more directly probe: (i) whether EvoReg's advantages persist under native, real-world registration protocols; (ii) how EvoReg compares against a wider set of strong recent rigid-registration baselines; and (iii) how consistently EvoReg's advantage on distributional/alignment metrics trades off against pose-precision metrics, across baselines and datasets. No EvoReg variant was retrained for any experiment in this section, only the evaluation protocol and the set of tested baselines changed; we use the same four trained models (rigid/non-rigid $\times$ supervised/self-supervised) described in Section 4.1.3. That is, we continue to test cross-dataset generalization and, in this case, cross-protocol generalization as well.

E.1 Native (Real) Registration Pairs: 3DMatch and FAUST

Elsewhere in this paper, evaluation pairs are constructed by applying a controlled similarity transformation (and, for non-rigid settings, a controlled RBF deformation) to normalized point clouds drawn from each dataset, following Appendix C.1. This construction is useful for controlled, apples-to-apples comparison, but it does not by itself establish that EvoReg transfers to native, real-world scan-to-scan registration, where correspondences and overlap arise from real sensing. To test this directly, we also evaluate EvoReg under three native, real-world protocols:

- **3DMatch, official benchmark pairs.** Real RGB-D reconstruction fragment pairs with native partial overlap, evaluated under the standard 3DMatch rigid-registration protocol, rather than our defined pair-construction pipeline for rigid shapes.
- **FAUST, native pose-correspondence pairs.** Two evaluations, split into an *intra-subject* protocol (a subject's base pose against each of that subject's other poses, isolating pose deformation) and an *inter-subject* protocol (the same pose across different subjects, isolating body-shape variation), rather than following our deformation-based pair-construction pipeline for non-rigid shapes.

Importantly, EvoReg is not retrained for any of these tasks; all native protocols use EvoReg's predefined rigid (for 3DMatch) and non-rigid (for FAUST) set of weights (as defined in Section 4.1.3).

3DMatch. Table E1 reports results on the official 3DMatch benchmark. Against the strongest supervised baseline evaluated on native pairs, GeoTransformer, EvoReg-sup-full cuts Chamfer Distance by 86% (0.265 to 0.037) and raises CD-Recall from 57.2% to 99.4%. Against four additional strong rigid baselines evaluated under this same native protocol (Predator, CoFiNet, PointDSC, and RoITr; see Appendix E.2), EvoReg's CD-Recall (99.4%) is 2.4–2.5$\times$ theirs (39.2–41.2%) and its EMD is roughly half (0.025 vs. 0.047): EvoReg registers native fragments to acceptable overlap far more often. Those four baselines keep the rotation/translation error (RE/TE) edge (e.g., Predator's RE is 5.7° vs. EvoReg's 36.9°), reflecting their pose-precision-oriented design rather than a failure of EvoReg's alignment (Appendix E.3).

FAUST. Tables E2 and E3 report the native intra- and inter-subject FAUST results. EvoReg wins CD and EMD in all four native blocks (intra/inter subject $\times$ supervised/self-supervised). The margin is largest in the supervised blocks, where EvoReg's CD (0.0079) is 2.9$\times$ lower than the strongest supervised baseline, DefTransNet (0.0233), and EvoReg even wins on rotation error there (15.0° vs. $\geq$16.4° elsewhere). NDP and DefTransNet remain stronger on RE_g, SWD, P2P, and TE, and DefTransNet is also the fastest method tested; NDP, by contrast, runs at $\approx$32 s/pair versus EvoReg's sub-second inference. All methods, including EvoReg, reach 100% CD-Recall on native FAUST

Table E1: 3DMatch (rigid, native pairs) — official 3DMatch benchmark. **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised/Optimization-Based									
EvoReg-selfsup-full (Ours)	**0.0377**	0.7836	**0.0252**	0.1015	**0.7105**	**99.4**	35.651	0.4102	0.5356
EvoReg-selfsup-fast (Ours)	0.0437	0.7837	0.0278	0.1017	0.7107	**99.4**	35.647	0.4102	0.3413
PRNet (Wang & Solomon, 2019b)	0.2595	0.7987	0.1006	0.1186	0.7298	63.8	31.810	0.4340	0.0360
CPD-R (Myronenko & Song, 2010)	0.1967	**0.7831**	0.0970	**0.0960**	0.7127	78.2	41.206	0.4216	3.1142
ICP (Besl & McKay, 1992b)	0.2715	0.8352	0.0789	0.1602	0.7626	74.9	34.769	0.4374	**0.0123**
FGR (Zhou et al., 2016)	0.3127	0.8038	0.0955	0.1353	0.7380	35.8	51.830	0.4218	0.0533
RANSAC (Fischler & Bolles, 1981)	0.2653	0.8535	0.0683	0.1895	0.7743	53.9	48.447	0.3041	0.2032
TEASER++ (Yang et al., 2021)	0.3136	0.9121	0.0518	0.2474	0.8292	39.4	**21.575**	**0.1683**	2.2452
Supervised									
EvoReg-sup-full (Ours)	**0.0370**	**0.7844**	**0.0254**	**0.1008**	**0.7114**	**99.4**	36.941	0.4071	0.5296
EvoReg-sup-fast (Ours)	0.0431	0.7846	0.0280	0.1010	0.7116	**99.4**	36.935	0.4071	0.3377
DCP (Wang & Solomon, 2019a)	0.3578	**0.7844**	0.1058	0.1146	0.7248	16.5	102.026	0.4482	**0.0098**
RPMNet (Yew & Lee, 2020)	0.2979	0.7852	0.1068	0.1034	0.7212	45.7	45.898	0.4454	0.0164
PointNetLK (Aoki et al., 2019)	0.5472	0.9487	0.0799	0.2733	0.8816	13.9	53.897	0.6685	0.0382
DeepGMR (Yuan et al., 2020)	0.3690	0.7880	0.1031	0.1239	0.7297	15.7	112.543	0.4585	0.1145
iPCRNet (Sarode et al., 2019)	0.4480	0.8711	0.0911	0.2121	0.8059	13.8	52.445	0.5880	0.0107
GeoTransformer (Qin et al., 2023)	0.2651	0.8010	0.0949	0.1217	0.7342	57.2	49.170	0.4464	0.2232
RoITr (Yu et al., 2023)	0.3069	0.9228	0.0470	0.2558	0.8380	40.7	6.242	**0.0464**	0.5216
PointDSC (Bai et al., 2021)	0.3290	0.9134	0.0531	0.2472	0.8313	39.2	28.695	0.2268	0.4011
CoFiNet (Yu et al., 2021)	0.3069	0.9222	0.0472	0.2554	0.8375	40.7	6.946	0.0499	0.48
Predator (Huang et al., 2021)	0.2975	0.9160	0.0474	0.2505	0.8314	41.2	**5.723**	0.0598	2.04

Table E2: FAUST, native pairs, intra-subject (base pose vs. each subject's other poses; isolates pose deformation). **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised/Optimization-Based									
EvoReg-selfsup-full (Ours)	**0.01696**	0.24879	**0.03062**	0.05252	**0.18577**	**100.00**	18.818	0.07496	0.708
EvoReg-selfsup-fast (Ours)	0.02388	0.24856	0.03349	0.05299	0.18579	**100.00**	18.818	0.07496	**0.375**
CPD-NR (Myronenko & Song, 2010)	0.09718	**0.21785**	0.08182	0.04879	0.19599	**100.00**	19.121	0.03205	1.157
BCPD (Hirose, 2021)	0.11972	0.25150	0.09492	0.05512	0.22137	**100.00**	**16.404**	**0.02334**	2.491
NDP (Li & Harada, 2022)	0.02699	0.25145	0.04243	**0.04782**	0.18872	**100.00**	25.242	0.06178	32.527
Supervised									
EvoReg-sup-full (Ours)	**0.00789**	0.35283	**0.03640**	0.05897	0.29326	**100.00**	**15.009**	0.07813	0.537
EvoReg-sup-fast (Ours)	0.01456	0.31975	0.03779	0.05595	0.26352	**100.00**	17.020	0.07473	0.336
DefTransNet (Monji-Azad et al., 2025)	0.02326	**0.19463**	0.04242	**0.03382**	**0.15018**	**100.00**	19.272	**0.03819**	**0.127**

pairs, since FAUST's native deformations are mild relative to our 0.15 CD-Recall threshold; the harder, more discriminating test is 3DMatch's real partial overlap, where EvoReg's margin over baselines is largest.

Taken together, EvoReg's competitive results hold across datasets, supervision regimes, and registration modalities under this more realistic, native-pair protocol, using the exact same architecture and trained weights evaluated elsewhere in this paper – supporting that its cross-dataset generalization (Section 3.2) is not an artifact of controlled pair construction.

E.2 Additional Rigid Registration Baselines

Table E1 adds five rigid-registration baselines not present in the main comparisons of Section 4.1.2, evaluated under the native 3DMatch protocol of Appendix E.1: **Predator** (Huang et al., 2021), **CoFiNet** (Yu et al., 2021), and **PointDSC** (Bai et al., 2021), which target the partial-overlap regime explicitly; **RoITr** (Yu et al., 2023), a rotation-invariant transformer architecture; and **TEASER++** (Yang et al., 2021), a certifiably robust global registration method. All five are run using their released code or standard implementations and scored with the same metric suite (Section 4.2) as every other method in this paper. The learned baselines (Predator, CoFiNet, PointDSC, RoITr) use their pretrained 3DMatch weights on native fragments, and their predicted transforms are transported into our common evaluation frame for scoring; TEASER++, being training-free, is run directly on the same normalized, sampled pairs. None of the five are retrained.

Against these baselines on native 3DMatch pairs, EvoReg-selfsup-full/-sup-full trails on rotation and translation error (narrowest against PointDSC, widest against RoITr and Predator; Table E1) but leads by a wide margin on distributional alignment and task success. Per Table E1, the four new supervised-block baselines trail EvoReg-sup-full by 58–60 CD-Recall points and ≈2× on EMD, while beating it 1.3–6.5× on RE and 1.8–8.8× on TE. Against TEASER++ specifically, EvoReg-selfsup-full wins alignment and correspondence 6/6 (CD-Recall 99.4% vs. 39.4%) while running 4.2× faster (0.54 s vs. 2.25 s per pair), even though TEASER++ wins pose error outright (RE 21.6° vs. EvoReg's 35.7°;

Table E3: FAUST, native pairs, inter-subject (same pose across different subjects; isolates body-shape variation). **Bold**: best, underline: second best.

Method	CD ↓	RE_g ↓	EMD ↓	SWD ↓	P2P ↓	CD-Recall (%) ↑	RE (°) ↓	TE ↓	Infer. (s) ↓
Self-Supervised/Optimization-Based									
EvoReg-selfsup-full (Ours)	**0.00577**	0.06079	**0.01876**	0.01222	0.04642	**100.00**	2.947	0.02056	0.683
EvoReg-selfsup-fast (Ours)	0.00803	0.07727	0.02114	0.01297	0.06276	**100.00**	3.111	0.02052	**0.360**
CPD-NR (Myronenko & Song, 2010)	0.06826	0.06201	0.05758	0.02393	0.05651	**100.00**	1.465	0.01242	1.113
BCPD (Hirose, 2021)	0.10992	0.13011	0.08769	0.04623	0.11728	**100.00**	2.168	0.01655	2.630
NDP (Li & Harada, 2022)	0.02092	**0.02518**	0.02355	**0.00534**	**0.01969**	**100.00**	**0.906**	**0.00430**	32.679
Supervised									
EvoReg-sup-full (Ours)	**0.00659**	0.06242	**0.01873**	0.01247	0.04765	**100.00**	3.212	0.02048	0.690
EvoReg-sup-fast (Ours)	0.01125	0.06383	0.02007	0.01247	0.04886	**100.00**	3.212	0.02048	0.362
DefTransNet (Monji-Azad et al., 2025)	0.01568	**0.03778**	0.02505	**0.00798**	**0.02607**	**100.00**	**0.912**	**0.00547**	**0.123**

Table E4: Metric-win and speed comparison for EvoReg-Full against CPD-R (Myronenko & Song, 2010) (rigid), GeoTransformer (Qin et al., 2023) (rigid), and CPD-NR (Myronenko & Song, 2010) (non-rigid), shown as three panels: **(left)** alignment/correspondence metrics won (CD, RE_g, EMD, SWD, P2P, CD-Recall; out of 6); **(middle)** pose-error metrics won (RE, TE; out of 2); **(right)** per-pair inference time in seconds, EvoReg-Full / baseline (**bold** = faster), with the *Total* row instead reporting the count of experiments where EvoReg-Full was faster. A tie (including a 100% CD-Recall tie) counts as a non-win rather than a loss; no EvoReg variant is retrained or re-evaluated for this analysis. Row abbreviations: MN40 = ModelNet40, SN13 = ShapeNet-13, 3DM = 3DMatch, (nat.) = native pairs (Appendix E.1); GeoT. = GeoTransformer.

Alignment wins (/6)

	CPD-R	GeoT.	CPD-NR
MN40	2/6	6/6	6/6
SN13	4/6	6/6	6/6
3DM	5/6	5/6	5/6
FAUST	5/6	5/6	5/6
3DM (nat.)	4/6	6/6	–
F-intra (nat.)	–	–	3/6
F-inter (nat.)	–	–	5/6
Total	20/30	28/30	30/36

Pose-error wins (/2)

	CPD-R	GeoT.	CPD-NR
MN40	0/2	2/2	2/2
SN13	0/2	2/2	1/2
3DM	0/2	2/2	1/2
FAUST	0/2	2/2	2/2
3DM (nat.)	2/2	2/2	–
F-intra (nat.)	–	–	1/2
F-inter (nat.)	–	–	0/2
Total	2/10	10/10	7/12

Speed, EvoReg/baseline (s)

	CPD-R	GeoT.	CPD-NR
MN40	**1.01**/1.38	0.38/**0.19**	**0.68**/1.39
SN13	**0.84**/1.62	0.38/**0.20**	**0.67**/2.00
3DM	**0.88**/1.60	0.39/**0.20**	**0.68**/1.84
FAUST	**0.82**/1.73	0.40/**0.19**	**0.70**/1.24
3DM (nat.)	**0.54**/3.11	0.53/**0.22**	–
F-intra (nat.)	–	–	**0.71**/1.16
F-inter (nat.)	–	–	**0.68**/1.11
Total	5/5 faster	0/5 faster	6/6 faster

TE 0.168 vs. 0.410). This is the same precise, non-uniform tradeoff quantified across all baselines in Appendix E.3.

E.3 Accuracy vs. Pose-Precision Tradeoff: A Metric-Win Analysis

Sections 4.3 and E.1 show that EvoReg's advantage concentrates on distributional shape-overlap and task-success metrics (CD, EMD, CD-Recall), while methods explicitly designed for pose recovery are sometimes stronger on rotation/translation error. Rather than describe this tradeoff qualitatively, we quantify it directly against the three baselines most competitive with EvoReg on pose precision: **CPD-R** (rigid), **CPD-NR** (non-rigid), and a paper-faithful **GeoTransformer**. For every dataset and protocol evaluated in this paper, including the native-pair experiments of Appendix E.1, we count how many metrics the corresponding EvoReg-Full variant wins, split into two families: *alignment and correspondence* (CD, RE_g, EMD, SWD, P2P, CD-Recall – 6 metrics) and *pose error* (RE, TE – 2 metrics). A tie (including a 100% CD-Recall tie) counts as a non-win rather than a loss, and no EvoReg variant is retrained or re-evaluated for this analysis; it only aggregates results already reported in this paper and in Appendices E.1–E.2.

Reading. On alignment and correspondence, EvoReg leads every matchup overall (78/96 total). CD and EMD specifically are won in all 16 experiment–baseline comparisons conducted. Of the 11 comparisons where EvoReg does not sweep all six alignment metrics, seven are explained by a CD-Recall tie, where EvoReg matches rather than loses; the remaining four – three against CPD-R, one against CPD-NR – reflect genuine baseline advantages rather than ties. On pose error, EvoReg sweeps GeoTransformer (10/10) and wins the majority against CPD-NR (7/12); CPD-R wins the large majority of pose-error comparisons against EvoReg (8/10), losing only on the native 3DMatch pairs, where EvoReg reverses the pattern (2/2). On speed, EvoReg-Full is faster than both CPD variants in every experiment and slower than GeoTransformer in every experiment.

Overall, this analysis supports a precise scope for EvoReg's claims: EvoReg is the stronger choice when the downstream objective is geometric overlap and registration success (distributional alignment, CD-Recall), while CPD-R remains preferable when the objective is minimal-error rigid pose recovery,

Table F1: Registration method usage guidance.

Requirements	Preferred method	Why	Real-life use case
EvoReg preferred			
Robust shape/surface overlap + high registration success	EvoReg	Best CD/EMD in every setting; perfect or near-perfect CD-Recall	Multi-scan 3D reconstruction / scan stitching; coverage-driven scene modeling
Overlap *and* solid point-level correspondence in the rigid regime	EvoReg	Competitive-to-leading here — wins P2P and most RE_g on native 3DMatch	Rigid real-scan alignment (RGB-D fragments) needing coverage *and* usable point matches
One model for rigid *and* non-rigid (or regime unknown)	EvoReg	Single architecture spans all four (rigid/non-rigid)×(sup/self-sup) settings	Pipelines mixing rigid parts and deformable objects (cloth, organs, bodies)
No ground-truth poses / labels	EvoReg (self-sup)	Geometric self-supervision; no pose labels required	Registering raw real-world scans without annotation
Large or unknown initial misalignment	EvoReg	Gradient-free CMA-ES $\mathrm{SE}(3)$ pre-alignment escapes local minima	Loop closure / relocalization; coarse pre-alignment
Cross-domain / limited training data	EvoReg	ModelNet40-only training generalizes to real scans	Deploying to a new sensor or dataset without retraining
Real-time / on-device with good overlap	EvoReg-Fast	Retains overlap quality at a fraction of the compute	AR/VR alignment; mobile / edge 3D scanning
Specialized baseline preferred			
Precise rigid 6-DoF pose (minimal RE/TE)	RoITr (Yu et al., 2023), CoFiNet (Yu et al., 2021), Predator (Huang et al., 2021), TEASER++ (Yang et al., 2021)	Correspondence- and certifiability-driven pose precision	SLAM and camera pose, robot manipulation, metrology, sensor calibration
Low-overlap real fragment pairs, robust global registration	Predator (Huang et al., 2021), TEASER++ (Yang et al., 2021)	Overlap-attention / certifiably-robust estimation	Aligning partial LiDAR or RGB-D fragments
Dense *non-rigid* correspondence / precise deformation	NDP (Li & Harada, 2022), Def-TransNet (Monji-Azad et al., 2025)	Purpose-built for per-point correspondence in the deformable regime	Medical organ/tissue registration; character rigging; body tracking

and specialized architectures such as GeoTransformer, Predator, CoFiNet, PointDSC, and RoITr remain preferable when dense, low-error correspondence or pose precision is the primary requirement.

F Registration Method Usage Guidance

EvoReg is strongest in distributional alignment, followed by correspondence accuracy; it trades off pose-precision as a result. EvoReg is, therefore, best used in cases that align with its strengths, while other methods are preferred when strict pose alignment is the principal requirement (Table F1).